Technische Universität München
TUM School of Engineering and Design

Data openness and efficient methods to address data insufficiency in mobility analyses and simulation calibration

Vishal Mahajan

Vollständiger Abdruck der von der TUM School of Engineering and Design der Technischen Universität München zur Erlangung eines

Doktors der Ingenieurwissenschaften (Dr.-Ing.)

genehmigten Dissertation.

Vorsitz:

Prof. Dr.-Ing. Rolf Moeckel

Prüfer*innen der Dissertation:

1. Prof. Dr. Constantinos Antoniou
2. Assoc. Prof. Dr. Nikolas Geroliminis
3. Prof. Dr. Francisco Camara Pereira

Die Dissertation wurde am 19.07.2023 bei der Technischen Universität München eingereicht und durch die TUM School of Engineering and Design am 30.11.2023 angenommen.

Dedicated to this Universe
to whom we are no strangers

Abstract

Data are indispensable for transport modeling and analyses, providing insights into travel patterns and behavior. Researchers and practitioners often face challenges due to insufficient and low-quality data. Advancements in sensing and communication technologies have led to the development of non-conventional and emerging data sources. These data are increasingly being used in transport research. There is no doubt that with the new kinds of data, we can explore new opportunities in transport research to address the limitations of traditional data. However, the fact that these data are generated from a wide variety of sources that have their own shortcomings leads to new challenges. For example, these data may be inaccessible or insufficient for a given task. Further, data from open sources tend to come without any guaranteed quality, thus leading to a further burden on the data consumers for their validation. Therefore, we face a situation where despite seeming data abundance, we face challenges when it comes to using them for mobility analyses and transport modeling.

In this dissertation, we aim to address this paradox of data insufficiency in view of the diverse sources and varied data availability. We provide a conceptual and methodological framework to classify the diverse data sources based on their openness and then use them to tackle data insufficiency in different contexts. We conceptualize and demonstrate three sets of approaches to bridge the gap due to the lack of usable data. First, we use data from emerging sources [drone videography and Point of Interest (POI) busyness data from mobile crowd sensing], address their quality, and apply them to novel use cases. Second, we address the gap between the information from conventional and non-conventional data sources. This is demonstrated by applying a transfer learning-based indirect traffic estimation framework to estimate the sparse traffic flow data from relatively abundant traffic speed data. Third, we propose a methodological framework to address system underdeterminedness due to the limited availability of conventional data. We demonstrate that in the context of calibration of large-scale traffic simulations, simple heuristics, and machine learning-based techniques can help to obtain precise estimates. Through these selective transport analyses with different data sources, we further state-of-the-art research in addressing data insufficiency in different contexts. This dissertation contributes on theoretical, methodological, and practical levels to motivate researchers and practitioners for data-efficient transport analyses and modeling.

Zusammenfassung

Daten sind für die Verkehrsmodellierung und -analyse unverzichtbar, da sie Einblicke in Reisemuster und -verhalten bieten. Forscher und Praktiker stehen oft vor Herausforderungen, die auf unzureichende und qualitativ schlechte Daten zurückzuführen sind. Fortschritte in der Erfassungs- und Kommunikationstechnologie haben zur Entwicklung unkonventioneller und neuer Datenquellen geführt. Diese Daten werden zunehmend in der Verkehrsforschung genutzt. Es besteht kein Zweifel daran, dass wir mit den neuen Arten von Daten neue Möglichkeiten in der Verkehrsforschung erkunden können, um die Beschränkungen der traditionellen Daten zu überwinden. Die Tatsache, dass diese Daten aus einer Vielzahl von Quellen stammen, die ihre eigenen Unzulänglichkeiten haben, führt jedoch zu neuen Herausforderungen. So können diese Daten beispielsweise unzugänglich oder für eine bestimmte Aufgabe unzureichend sein. Darüber hinaus haben diese Daten aus offenen Quellen in der Regel keine garantierte Qualität, was zu einer weiteren Belastung der Datenkonsumenten bei der Validierung der Daten führt. Daher stehen wir vor der Situation, dass wir trotz des scheinbaren Überflusses an Daten vor Herausforderungen stehen, wenn es darum geht, diese für Mobilitätsanalysen und Verkehrsmodellierung zu nutzen.

In dieser Dissertation wollen wir dieses Paradoxon der unzureichenden Datenverfügbarkeit in Anbetracht der verschiedenen Quellen und der unterschiedlichen Verfügbarkeit von Daten angehen. Wir bieten einen konzeptionellen und methodischen Rahmen, um die verschiedenen Datenquellen auf der Grundlage ihrer Offenheit zu klassifizieren und sie dann zu nutzen, um Datenmängel in verschiedenen Kontexten zu beheben. Wir konzipieren und demonstrieren drei verschiedene Ansätze, um die Lücke zu schließen, die durch den Mangel an verwertbaren Daten entsteht. Erstens verwenden wir Daten aus neu entstehenden Quellen [Drohnenvideografie und POI Busyness-Daten aus mobilem Crowd Sensing], untersuchen deren Qualität und wenden sie auf neuartige Anwendungsfälle an. Zweitens gehen wir auf die Lücke zwischen den Informationen aus konventionellen und nicht-konventionellen Datenquellen ein. Dies wird durch die Anwendung eines auf Transfer-Lernen basierenden Rahmens für die indirekte Schätzung des Verkehrsaufkommens demonstriert, um die spärlichen Verkehrsflussdaten aus relativ reichhaltigen Verkehrsgeschwindigkeitsdaten zu schätzen. Drittens schlagen wir einen methodischen Rahmen vor, um der Unterbestimmtheit des Systems aufgrund der begrenzten Verfügbarkeit konventioneller Daten zu begegnen. Wir zeigen, dass bei der Kalibrierung von groß angelegten Verkehrssimulationen einfache Heuristiken und auf maschinellem Lernen basierende Techniken helfen können, präzise Schätzungen zu erhalten. Durch diese gezielten Experimente, die Verkehrsanalyseanwendungen mit unterschiedlichen Datenquellen einbeziehen, bringen wir den Stand der Forschung weiter voran. Diese Dissertation trägt auf theoretischer, methodischer und praktischer Ebene dazu bei, Forscher und Praktiker

zu motivieren, die verfügbaren Daten und die jüngsten Innovationen im Bereich des maschinellen Lernens für Verkehrsmodellierungs- und Data-Mining-Anwendungen zu nutzen.

Acknowledgements

I sincerely thank Prof. Constantinos Antoniou for his unwavering and overall support throughout this dissertation. His frequent and valuable feedback was crucial to completing this work, and his willingness to understand my interests and shape my doctoral journey made the process all the more enjoyable. I am also deeply grateful to Prof. Guido Cantelmo for mentoring and supervising me throughout my doctoral research. His feedback at every step of the dissertation helped me improve my research, and I appreciate his continued guidance. I would also like to extend my heartfelt thanks to Prof. Rolf Moeckel for his valuable contributions and feedback during the TraMPA project and to Dr. Nico Kuehnel and Dr. Carlos Llorca for collaborating and working on the project. Furthermore, I thank Prof. Nikolas Geroliminis and Dr. Emmanouil Barmpounakis for their collaboration and guidance in my research on the pNEUMA dataset.

I am grateful to Ma and Papa for their unwavering love and support throughout my research journey. Their relentless patience makes me more humble and grateful to have them with me in every walk of life. I am lucky to have an amazing brother who makes me laugh, supports me, and helps me see a fresh perspective in easy and difficult times. Additionally, I am thankful to my current and former colleagues at TSE for making it an exceptional place for research. Special thanks to Raoul, Moeid, Mohamed, Qinglong, Cheng, Mohammad, Rakib, Christelle, Roja, Santa, César, Filippos, Ramandeep, Hao, and Arunava. Their support at the workplace, engaging in thought-provoking and intercultural lunch talks, and occasional mountain hikes made the experience at TSE enjoyable. I also thank Margit for being highly supportive and managing the day-to-day administrative tasks at TSE. I am indebted to my friends, far and near: Aitan, Lucas, Abhishek, Nadin, Jai, and Sourabh, for making my stay in Munich delightful and bringing smiles along the way.

I also thank Deutsche Forschungsgemeinschaft (DFG) for financial support. Many thanks to the anonymous reviewers during the peer-review process, who provided valuable feedback on the research articles leading to this dissertation. I also want to thank countless open-source developers and contributors working on SUMO, Pytorch, OpenStreetMaps, and many more awesome tools, who dedicate their time and effort to making science and research more open. Finally, since most of this research was done during COVID-19, I humbly thank all the essential service workers who helped our society overcome the pandemic.

Contents

List of Figures

List of Tables

Acronyms

ACC	Adaptive Cruise Control.
AFC	Automatic Fare Collection.
API	Application Programming Interface.
AV	Autonomous Vehicle.
AVL	Automatic Vehicle Location.
CDR	Call Detail Record.
COVID-19	Corona Virus Disease 2019.
DFG	Deutsche Forschungsgemeinschaft.
DODE	Dynamic Origin-destination Demand Estimation.
DTA	Dynamic Traffic Assignment.
FCD	Floating Car Data.
GB	Gradient Boosting.
GBFS	General Bikeshare Feed Specification.
GBM	Gradient Boosting Machines.
GBR	Gradient Boosting Regression.
GF	Gaussian Filter.
GLS	Generalized Least Squares.
GMNS	General Modelling Network Specification.
GNSS	Global Navigation Satellite System.
GOF	Goodness of Fit.
GPS	Global Positioning System.
GSM	Geographical Social Media.
GTFS	General Transit Feed Specification.
highD	Highway Drone.
LBSN	Location-Based Social Network.
LSTM	Long Short Term Memory.
MAC	Media Access Control.
MAPE	Mean Absolute Percentage Error.
MCCV	Monte Carlo cross-validation.

MFD	Macroscopic Fundamental Diagram.
MOP	Measures of Performance.
MPND	Mobile Phone Network Data.
MSE	Mean Squared Error.
NFD	Network Fundamental Diagram.
NGSIM	Next Generation Simulation.
NPD	Non-Public Data.
OBD	On-board Diagnostics.
OCD	Open Community Data.
OD	Origin-Destination.
ODP	Open Data Portal.
OGD	Open Government Data.
OLS	Ordinary Least Squares.
OPD	Open Private Data.
OSM	OpenStreetMaps.
PCA	Principal Component Analysis.
PD	Public Data.
PeMS	Performance Measurement System.
POI	Point of Interest.
PRQ	Primary Research Question.
PSI	Public Sector Information.
PVD	Probe Vehicle Data.
RELP	Recursively Ensembled Low-pass filter.
RLM	Robust Linear Model.
RMSE	Root Mean Squared Error.
RNN	Recurrent Neural Network.
RO	Research Objective.
RQ	Research Question.
SA	Stochastic Approximation.
SG	Savitzky-Golay.
SGD	Stochastic Gradient Descent.
SHAP	SHapley Additive exPlanations.
SMAPE	Symmetric Mean Absolute Percentage Error.
SO	Simulation Optimization.
SPA	Stochastic Parameter Averaging.
SPSA	Simultaneous Perturbation Stochastic Approximation.
SRQ	Secondary Research Question.
SUMO	Simulation of Urban Mobility.

SWA	Stochastic Weight Averaging.
SWOT	Strengths, Weaknesses, Opportunities, and Threats.
TAZ	Traffic Analysis Zone.
UAS	Unmanned Aerial Systems.
VGI	Volunteered Geographic Information.
WAPE	Weighted Average Percentage Error.
XGBoost	Extreme Gradient Boosting.

Part I

Introduction and conceptual understanding

1 Introduction

Contents

1.1 Motivation

Transport systems play a key role in supporting economic growth and development (Canning & Fay, 1993). They play a critical role in everyday life by providing mobility to people and goods. Transport models are the foundation of planning transport and traffic systems. Apart from conventional transport models (travel demand and supply modeling), transport data mining and data science are also increasingly popular areas of research to uncover travel behavior and mobility patterns (C. Chen et al., 2016). These areas are grounded in applying scientific methods to transport data to extract valuable knowledge from them. For instance, in data science approaches (Martínez-Plumed et al., 2021), statistical and data-driven models are used to explore, explain, and forecast behavior and processes in transport systems (Vlahogianni et al., 2004). Depending on the detail and context, these models can capture components of travel demand, mobility behavior, and transport supply. These data-driven models provide insights into travel behavior and mobility patterns and augment our knowledge to build better transport models further.

Data are enablers for analysis and modeling as well as efficient and sustainable design and operation of the transport systems. The Merriam-Webster dictionary defines data as "factual information used as a basis for reasoning, discussion, and calculation" (Merriam-Webster, 2020). The input and validation data for travel demand and supply models, such as trip- and activity-based models, depend on the modeling task. Conventional (or traditional) data sources commonly include household (survey) data, socio-demographic data, land-use information, and transportation network data (Castiglione et al., 2014). Further, suppose modeling deals with a specific phenomenon [such as Electric Vehicles' (EVs) adoption or transport emissions]. In that case, additional data (such as the present share of EVs and emission data) are needed. Further details and methodological steps are required depending on the analysis and modeling requirements (Castiglione et al., 2014). Traditionally, transport models require vast amounts of data to represent travel demand and transport supply. **Conventional data** collection methods, including household travel surveys, loop detectors, and census, tend to cost more and take longer (Willumsen, 2021). In addition, conventional data from relevant authorities may be restricted or lack usability with the fast-changing landscape of open-source transport modeling formats and tools. Hence, exclusive reliance on conventional data often limits researchers and practitioners in their modeling pursuit.

Data from non-conventional sources help to overcome some of these limitations (Willumsen, 2021). **Non-conventional data** sources, such as mobile phones, social media, and public transport smart cards, can be collected and have influenced and evolved how we conduct mobility analyses and travel forecasting. For example, cellular data is a rich source of origin-destination flows (Caceres et al., 2007) and OpenStreetMaps is a common source to extract and develop road transport network models (Ziemke et al., 2019). Similarly, for other non-conventional data, existing studies have demonstrated their varied applications in transport modeling in different contexts (Mahajan et al., 2022). Recent advances in sensing and communication technologies have made collecting

new types of non-conventional data possible, also referred to as **emerging data**. We use the term "emerging data" to refer to data types emerging from relatively newer sources, such as mobile crowdsensing social media and drone videography (Harrison et al., 2020). By definition, emerging data are a subset of non-conventional data. An example of the emerging data collection method is mobile crowdsensing, where smartphones are used to collect the data over a large scale (Liu et al., 2016), and then the data is transmitted to the central repository via the internet. Another example is drone videography. Drones can record the agents (people or vehicles) from an aerial point-of-view from which agents' naturalistic driving behavior can be extracted for potential applications in traffic research. Non-conventional data allows us to have more information and uncover new dynamics, thanks to the higher frequency, coverage, and spatial-temporal and contextual details (Harrison et al., 2020; Torre-Bastida et al., 2018). This means that with these new types of data, it could be possible to address the limitations of conventional data. For instance, conventional archived data does not allow modelers to analyze real-time mobility patterns in response to special events or interventions. However, streaming data from emerging data sources could be helpful in such applications for informed decision-making.

Despite the flux of studies demonstrating the application of a wide variety of data in research (Mahajan et al., 2022), the availability of these data for practice faces a major challenge and has a wide variability and lack of harmonization across regions (Máchová & Lněnička, 2017). For instance, mobile phone data are promising in terms of their applications (Willumsen, 2021), but they are proprietary, privacy sensitive, and not free of charge. Even if some of the data are available to the user free of charge, it does not necessarily mean that they can be used (Mahajan et al., 2022). The usability of the data depends on many factors, such as data formats and data quality (Máchová et al., 2018; R. Y. Wang & Strong, 1996). In the case of priced data, data providers are incentivized to ensure adequate data quality. However, this is not necessarily true in the case of public data sources, such as open data. In these cases, the burden for data pre-processing, cleaning, and validation is mostly with the data consumer since, generally, openness compliance licenses exclude such guarantees or liability (Creative Commons, 2023). Data pre-processing can be expensive in terms of time and cost. Due to a lack of resources, data users might not be motivated to use the available data if the pre-processing costs are substantial. Even if specific data are available (and usable) for a location or a region, the same may not hold true for another location (Barrington-Leigh & Millard-Ball, 2017). Further, in many cases, the available data are insufficient for reliable modeling or analysis (ITF, 2021).

Apart from newer data collection methods, advances in computing hardware and open source software have made the state-of-the-art machine learning models accessible to a broader research community (Langenkamp & Yue, 2022). In transport research too, this is evident from the steep rise in the number of publications using big data and machine learning algorithms for a variety of applications (Kaffash et al., 2021). For instance, machine learning models are increasingly used for prediction and forecasting tasks in traffic and mobility behavior research (Tizghadam et al., 2019). Machine learning is "a branch of artificial intelligence concerned with the construction of programs that learn from experience" (*A Dictionary of Computing*, 2008) or in other words, it is a process of

training models or machines by learning from the data, as opposed to the set of explicit manual instructions. Corollary to the famous "law of the hammer" quote by Abraham Maslow (Wikipedia, 2023) is that with the popularity of machine learning "hammer", not every problem should be treated as a nail. Instead, novel use cases for solving existing challenges with conventional and non-conventional data need to be identified. These use cases could pertain to the processing and analyzing of conventional and new data to unlock their full potential (Anda et al., 2017).

1.2 Problem definition and dissertation objectives

Transport systems are evolving unprecedentedly in terms of changes to demand and supply components, patterns, and their interactions (Hoppe et al., 2014). This motivates researchers and practitioners to advance their models, for instance, by developing large-scale models and simulations to increase spatial coverage and account for complex interactions or use real-time predictive frameworks for fast, responsive traffic and transport management. Designing and developing detailed and advanced spatial-temporal models is challenging because such models are data-hungry and need more and better data (ITF, 2021; Willumsen, 2021). The open data movement has undoubtedly helped improve access to more data (M. Janssen et al., 2012), but not all open data are usable. Further, the data's usability depends on the application's context and goal. Lack of usable data or **data insufficiency** or scarcity is a challenge in many research areas (Alzubaidi et al., 2023). In transport research and modeling, data insufficiency affects different stages of transport data mining, model development, calibration, and validation. For instance, traffic flow/ volume data from loop detectors (data source) are commonly used to calibrate and validate traffic simulation models (application). In large-scale traffic simulation models, the observed data from sensors covering only a small part of the network is insufficient for the "unique" and reliable estimation of these parameters (Gupta, 2005). Data insufficiency can span spatial, temporal, and contextual dimensions for a given data source and application, e.g., the loop detector data are generally spatially sparse. Traditional archived data are inadequate for dynamic operational applications, such as during special planned or unplanned events when the transport systems can behave unusually. For these cases, access to dynamic or real-time data is required. Thus, transport modeling or analysis cannot be done without access to sufficient good-quality and suitable data. Only when good quality data is available can modelers and data users create value out of the data.

Apart from having sufficient and suitable data, models also need to evolve. Milne and Watling (2019) reviewed implications of big data in the context of transport planning and listed "big challenges for big data". One of the main challenges is "re-specifying analytical and predictive modeling approaches in response to the modified data landscape and the new insights it facilitates" (Milne & Watling, 2019). The need for new modeling approaches arises because the emerging data sources may not have been designed specifically for transport planning and management, in contrast to the traditional data (household or origin-destination surveys). Data-driven or machine learning models are

increasingly used because of their better predictive performance. For example, a study by Vlahogianni et al. (2004) has noted the advantages of these machine learning models, such as accurate results, successful modeling of complex spatial and temporal relationships, and the ability to model non-linear relationships. Considering the issues surrounding data quality and insufficiency, it is not sufficient to apply data-driven models but to meticulously design models while addressing the data hungriness of the typical machine learning (Adadi, 2021).

Nowadays, many traditional and emerging data are open-sourced or "somewhat" publicly available (Mahajan, Cantelmo, et al., 2023) and thus offer the potential for use in transport modeling or transport data mining. The utility of emerging data depends on whether they can replace, complement, or supplement traditional data sources in solving the current challenges in transport analyses and modeling (Mahajan, Cantelmo, et al., 2023). The spatial or temporal overlap among data from different sources can help to augment information to capture real-world phenomena (Milne & Watling, 2019) and better understand travel behavior and model traffic dynamics. In view of the above, this dissertation aims to use publicly available data to address the challenges of data insufficiency or limited observability in large-scale analysis of mobility behavior and traffic systems. Data insufficiency depends on the context of the data being used and the goal of the analysis, and thus there could be a lack of data on varying levels. Incorporating publicly available data for research and practice can be viewed as a two-step process.

1. The first step is to identify the appropriate data sources and enable access to these data. It is important to understand what makes the data open and how to classify the prominent transport datasets according to their openness systematically. Based on this classification system, we can identify and collect publicly available data from conventional and non-conventional data sources.

2. In the second step, data are processed and applied for transport-related research, such as data analysis, predictive modeling, and model calibration. While conventional data sources are well documented and studied, emerging data collection methods, such as mobile crowd sensing or drone videography, are still in the early phases of their application in transport research. Thus, the analysis of conventional or non-conventional data will focus on the aspects relevant to each data. On the one hand, for conventional data sources, it is interesting to see how to scale these data for unobserved parts, i.e., addressing their sparsity. On the other hand, for emerging data sources, data processing to remove errors and their novel use cases are interesting avenues for research. Therefore, this step can be further subdivided into four perspectives:

   a) Gathering and processing data from emerging data sources

   b) Developing methods using non-conventional data for new use cases in analysis and modeling

   c) Developing machine learning models that utilize data from available non-conventional sources to address sparsity of conventional data

d) Developing new methods for efficient use of data from conventional sources

Though the above perspectives can be non-exhaustive, these provide a conceptual direction to tackle data insufficiency by using available data and novel methods. To demonstrate the above perspectives in the context of transport and mobility-related analysis, we list down the following Primary Research Questions (PRQs) as a focus of this dissertation:

How to address data insufficiency in transport analysis and modeling by

PRQ(1): understanding the public availability of transport data systematically?

PRQ(2): enhancing the usability of emerging data?

PRQ(3): applying available emerging data for novel use cases?

PRQ(4): using machine learning methods to tackle limitations of conventional data ?

To operationalize the research in this dissertation, we have identified and listed the following Secondary Research Questions (SRQs). The background behind these questions will be provided and discussed later (Chapter 2) in the dissertation. The SRQs can be broadly divided along three dimensions:

1. PRQ(1) is further divided into following SRQs on reviewing transport data openness:
   - **SRQ(1)**: What are the main attributes to classify data based on their public availability or openness?
   - **SRQ(2)**: Which categories of the proposed typology do the prominent non-conventional data used for transport analyses belong to?
   - **SRQ(3)**: What are the common applications of these data for transport modeling?
   - **SRQ(4)**: What are the strengths and weaknesses of these data in terms of their applications and availability?
2. PRQ(2) and PRQ(3) lead to following SRQs on creating value from emerging data:
   - **SRQ(5)**: How to develop a scalable methodology to treat noise and anomalies in emerging data?
   - **SRQ(6)**: How to apply publicly available crowdsensing information for changes in spatial-temporal demand patterns during special events?
3. PRQ(4) is further divided into following SRQs on development of data-efficient methods:
   - **SRQ(7)**: How to use machine learning to automate the calibration of large-scale traffic simulations?

SRQ(8): How to use ensembling to obtain precise traffic simulation calibration parameters?

SRQ(9): How to use transfer learning to address the sparsity of dynamic link flows at the network level?

SRQ(10): What are the conditions for the successful transfer of pre-trained models?

The above SRQs lead to the framing of the following set of Research Objectives (ROs):

RO(1): Develop a systematic typology to scope the data landscape and classify the non-conventional data according to openness.

RO(2): Investigate the applications of prominent non-conventional data sources in transport modeling research.

RO(3): Develop and evaluate a scalable method for improving the usability and quality of publicly available data from emerging sources.

RO(4): Identify and apply publicly available opportunistic data for novel use cases and demonstrate their application.

RO(5): Develop and evaluate data-efficient methods to tackle challenges due to insufficient conventional data for traffic prediction and calibration.

1.3 Dissertation contributions

This doctoral dissertation contains the author's research towards identifying new data sources, their novel application for mobility analyses, and data-efficient methods to address the data quality and data scarcity for traffic behavior and transport model calibration. Below is a summary of the dissertation's theoretical, methodological, and practical contributions. The mapping between the research questions and corresponding contributions is illustrated in Table 1.1.

1. Theoretical and methodological contributions
   a) Data openness typology.
   b) Review and SWOT analysis of non-conventional data sources.
   c) Scalable methodology for treating errors in emerging data.
   d) Opportunistic application of emerging data for analyzing mobility patterns during special events.
   e) Efficient and automated framework for large-scale traffic simulation calibration.
   f) Transfer learning-based methodology to address the gap between conventional and non-conventional data.
2. Practical contributions

a) Open source codes for treatment of noise and anomaly in vehicle trajectory data from drone videography.

b) Shared source code for fast-asynchronous data collection of live POI busyness data.

c) Open source platform to automate the calibration of large-scale demand (origin-destination) and supply parameters.

d) Open source traffic data curated from diverse sources for two cities.

1.4 Dissertation design and structure

The overall outline of the dissertation is shown in Figure 1.1. First, the foundation of this dissertation is established by reviewing the state-of-the-art literature on topics (transport data openness, errors in emerging data, emerging data for special events, methods for large-scale traffic simulation calibration, and indirect flow estimation) motivated by the PRQs set out in the current Chapter. This review helps to define the SRQs and ROs. To fulfill the objectives of this dissertation, five studies are conducted to review and demonstrate data openness, novel emerging data applications, and data-efficient traffic prediction and simulation calibration methods. The contents of these five studies are derived from the author's publications during his doctoral research (Mahajan, Barmpounakis, et al., 2023; Mahajan et al., 2021, in review; Mahajan, Cantelmo, et al., 2023; Mahajan et al., 2022). The implementation (Figure 1.1) of the research can be divided into three dimensions: data openness, emerging data, and new methods. First, we conceptually investigate data openness and develop a framework to deal systematically with transport data. The second and third dimensions involve data-driven modeling, i.e., the pipeline of steps such as goal definition, data collection, data preparation, modeling or simulation, and evaluation (Chapman et al., 2000). This process is iterative and is incrementally improved based on the feedback, e.g., from the evaluation step. Specifically, in the second dimension, the main focus is on improving or showing the potential of emerging data. Here we develop methodological frameworks to process emerging data and apply them for transport analyses. Whereas in the third dimension, the focus is on developing methods to exploit the conventional and non-conventional data. Here, we develop new and efficient methods using established data to deal with challenges in traffic simulation calibration and forecasting.

This dissertation is structured into four parts with eight chapters (Figure 1.2) as follows:

- The Part I contains the following three chapters, which provide the current state of the literature on the topics covered in this dissertation and provide a theoretical framework for data openness.
  - In **Chapter 1** (current Chapter), we introduce the motivation and problem statement, research questions, objectives, contributions, and the structure of this dissertation.

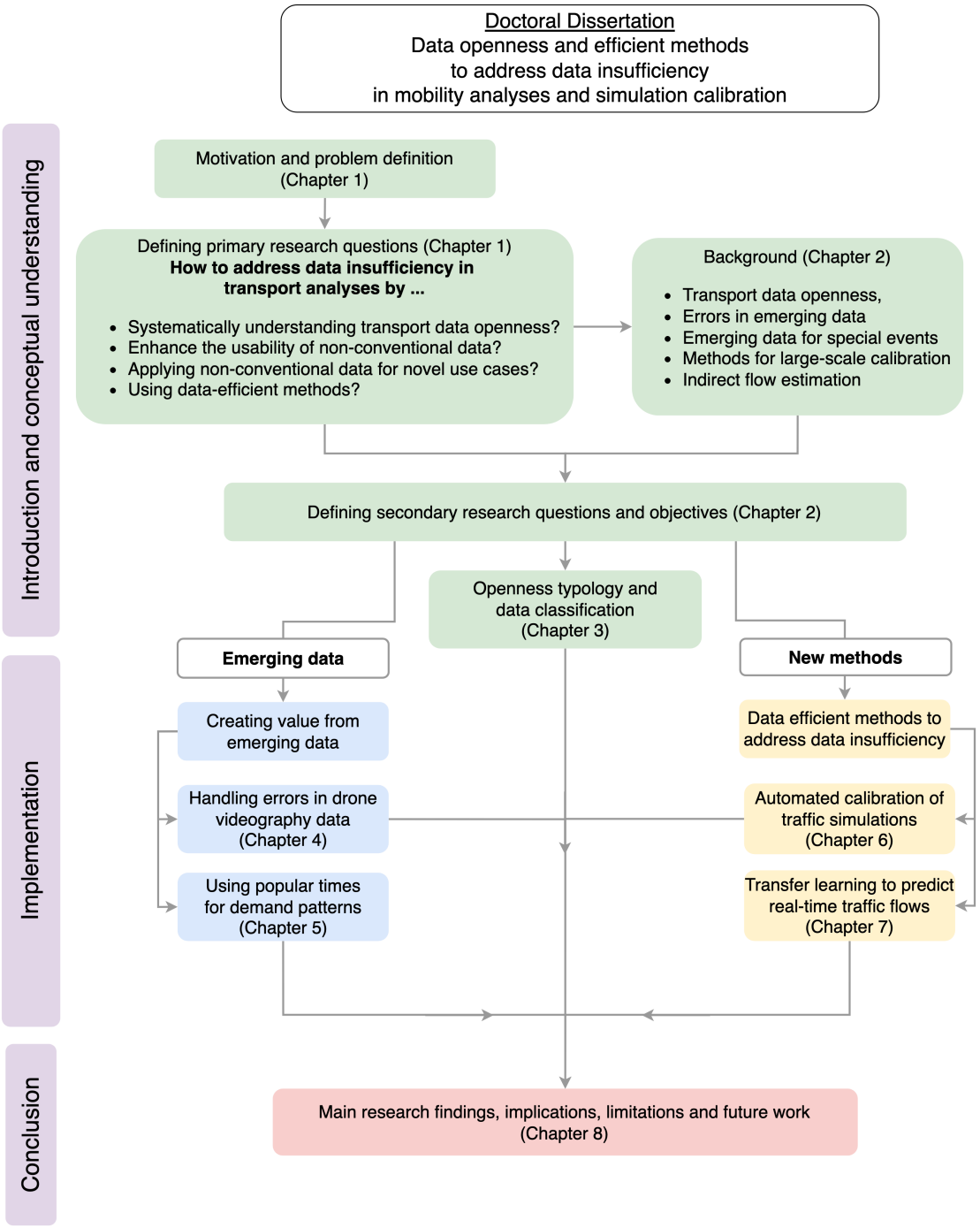


Figure 1.1: Dissertation outline

- In **Chapter 2**, we provide the background of the research topics and establishes the research context and practical importance of this research. We also provides the basis for SRQs and ROs.
- In **Chapter 3**, we introduce the data openness typology applied to prominent non-conventional data used in transport research. We review the applications of non-conventional data in transport modeling to identify potential data and opportunities.

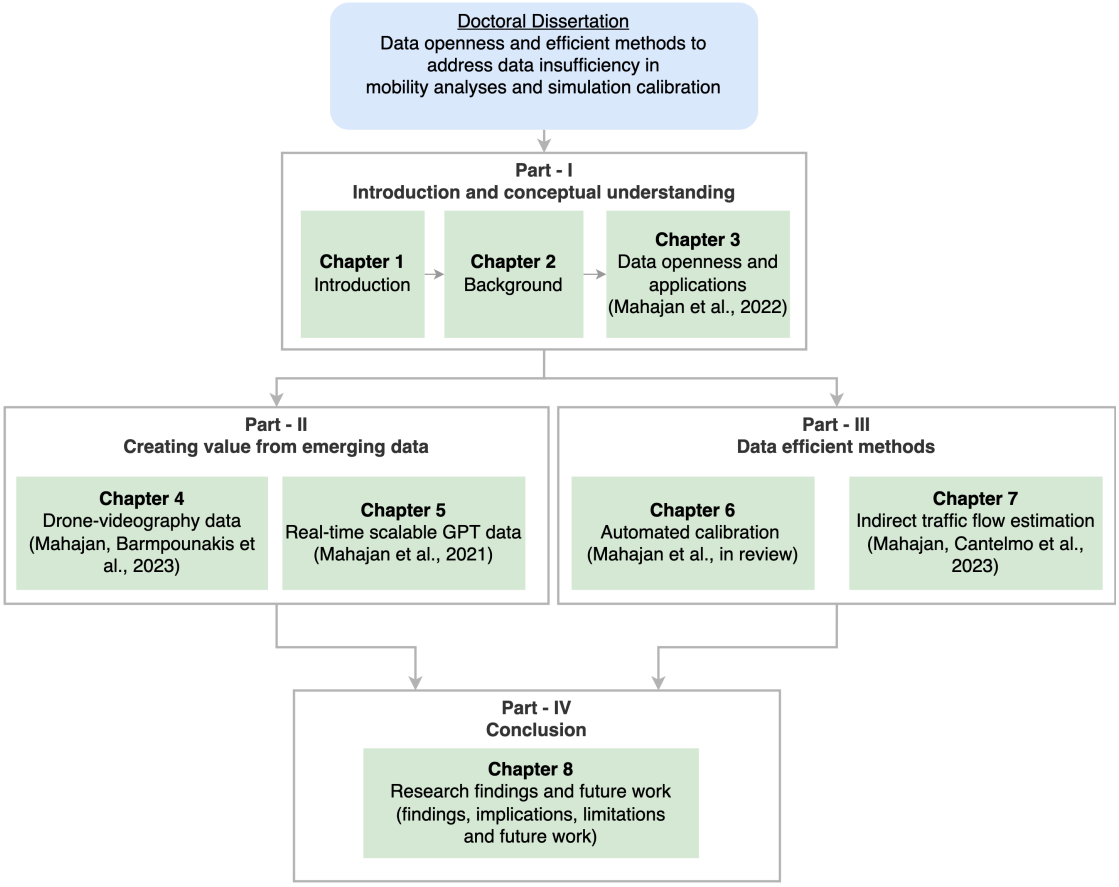


Figure 1.2: Dissertation structure

- The Part II contains two chapters, which are focused on creating value from the emerging data sources by improving their quality and demonstrating novel applications:
  - In **Chapter 4**, we consider the processing of errors in vehicle trajectory data from drone videography. We propose an efficient machine learning method to treat the noise and anomalies. The proposed method focuses on automating the anomaly detection process because of the diversity and complexity of the data to replace the manual specification of heuristics.

Table 1.1: Mapping of the research questions, objectives, chapters and publications

Primary questions	Secondary questions	Objectives	Contributing chapter(s)	Publications
PRQ(1)	SRQ(1), SRQ(2)	RO(1)	Chapter 3	Mahajan et al. (2022)
PRQ(1)	SRQ(3), SRQ(4)	RO(2)	Chapter 3	Mahajan et al. (2022)
PRQ(2)	SRQ(5)	RO(3)	Chapter 4	Mahajan, Barmpounakis, et al. (2023)
PRQ(3)	SRQ(6)	RO(4)	Chapter 5	Mahajan et al. (2021)
PRQ(4)	SRQ(7), SRQ(8), SRQ(9), SRQ(10)	RO(5)	Chapter 6, Chapter 7	Mahajan et al. (in review); Mahajan, Cantelmo, et al. (2023)

  - In **Chapter 5**, we demonstrate the use of unique crowd-sensing data collected from sources in the public domain and their application to analyze mobility patterns.

- The Part III contains the following two chapters, which are focused on the development of data-efficient methods using conventional and non-conventional data to tackle data insufficiency:
  - In **Chapter 6**, we present a methodological framework to automate large-scale demand and supply calibration of traffic simulations and address bias-variance in the calibration estimates.
  - In **Chapter 7**, we address the insufficiency of conventional data with the help of auxiliary non-conventional data sources from the public domain and transfer learning.

- The Part IV contains the following chapter and concludes the dissertation:
  - In **Chapter 8**, we discuss the research findings and their broad theoretical and practical implications, summarize the overall limitations, and provide an outlook on future directions.

Finally, an overview of the mapping between the PRQs, SRQs, ROs, chapters, and publications is provided in Table 1.1.

2 Background

Contents

This chapter incorporates select elements sourced from the author's following works:

- Mahajan, V., Kuehnel, N., Intzevidou, A., Cantelmo, G., Moeckel, R., & Antoniou, C. (2022). Data to the people: a review of public and proprietary data for transport models. Transport Reviews, 42(4), 415–440. doi:10.1080/01441647.2021.1977414
- Mahajan, V., Cantelmo, G., Rothfeld, R., Antoniou, C. (2023). Predicting network flows from speeds using open data and transfer learning. IET Intell. Transp. Syst. 17, 804– 824. doi:10.1049/itr2.12305
- Mahajan, V., Cantelmo, G., & Antoniou, C. (2021). Explaining demand patterns during COVID-19 using opportunistic data: a case study of the city of Munich. European Transport Research Review, 13(1), 26. doi:10.1186/s12544-021-00485-3
- Mahajan, V., Barmpounakis, E., Alam, M. R., Geroliminis, N., & Antoniou, C. (2023). Treating Noise and Anomalies in Vehicle Trajectories from an Experiment with a Swarm of Drones. IEEE Transactions on Intelligent Transportation Systems. doi: 10.1109/TITS.2023.3268712
- Mahajan, V., Cantelmo, G., & Antoniou, C. (in review). One-shot heuristic and ensembling for automated calibration of large-scale traffic simulations. In Review. Retrieved from https://mediatum.ub.tum.de/doc/1701188/document.pdf

2.1 Introduction

In this chapter, we identify the PRQs and SRQs by reviewing the literature and latest developments in the field. This chapter identifies specific data and challenges in transport modeling to demonstrate the development and application of methods to tackle this dissertation's problem statement or PRQs. Using this discussion, we identify and list specific SRQs and ROs.

This chapter is structured as follows: the following section provides the background on data openness and applications, focusing on transport modeling and analysis. The third section discusses the literature on data errors in trajectory data and why it is crucial to develop methods to process the data from emerging sources such as drone videography and improve the data quality. The fourth section highlights the need for other emerging data (crowdsensed busyness trends) for opportunistic applications during special events or interventions. The above sections lay the groundwork for research involving emerging data sources in this dissertation.

The following two sections are focused on reviewing the studies to identify the need to develop efficient methods. The fifth section discusses the need for developing efficient model calibration methods using traditional data. The sixth subsection identifies the research gaps due to the lack of conventional traffic data and the need to estimate them from auxiliary and non-conventional data sources indirectly. Finally, we summarize the ROs of this dissertation and link them with the corresponding Research Questions (RQs)

2.2 Data openness and emerging data

2.2.1 Data flow: from sources to users

The spread of mobile phones, affordable sensors, and the internet and innovations in communication technologies have created a data-generating ecosystem and led to an explosion of data available for transport analyses. This data revolution has prompted public and private organizations to release their data in part or entirely to the public as a free or paid product or service, with or without restrictions. Simultaneously, the advances in computing and telecommunication technologies have encouraged users to explore innovative use cases of the available data. Public transport schedule data [through the use of the General Transit Feed Specification (GTFS)], for example, are used to provide real-time public transport information through smartphone applications[1].

Before introducing the concept of Public Data, it is vital to understand a few key terms related to the data landscape with reference to transport data. This and the following paragraphs are primarily based on the report "Enabling Access to and Sharing of Data" (OECD, 2019). Generally, data are produced from personal or non-personal sources. Data from a personal source contain information that can be used to identify the data subjects. In such a scenario, the data will be referred to as personal data. The personal data source can be smartphones, social media accounts, or onboard vehicle sensors, and non-personal

[1] `https://developers.google.com/transit/gtfs-realtime`

data sources can be inductive loop detectors or weather monitoring stations. Specialized de-identification techniques such as anonymization, unlinking, or aggregation transform personal data into non-personal data. Two such examples are traffic speed datasets (by TomTom[2] and Uber[3]) or public transport flow data (from smart cards), where the personally identifiable information is removed, and data from numerous personal subjects are aggregated. Another distinction to note here is that the mode of personal data collection is primarily of two types: volunteered and observed (OECD, 2019). For the former, a person or an individual can either actively or passively, but consciously, contribute to the data collection, even if they are using a service, such as participating in a household survey or crowdsourcing data. For the latter, the data are captured or observed passively, as in mobile devices with enabled Global Navigation Satellite System (GNSS), such as Global Positioning System (GPS). Here, the primary motivation for an individual is always to use a service instead of offering the data. Generally, the individual is required to give a one-time consent, after which the data collection occurs passively unless the consent is revoked. Organizations collect the data, perform data processing (cleaning, curation, analysis) and create different value-added data products. This new information is called derived or inferred data (OECD, 2019).

Data is primarily owned by either the public or the private sector. The ownership is governed by who was involved in the data generation and production stages. In addition to these entities, individual(s) or household(s) might also have some ownership rights in the case of personal data, depending on the prevailing laws and contractual rights. Public and private organizations incur expenses for data collection, production, and operation. Private-sector and most public-sector data are initially proprietary (OECD, 2019). Communities consisting of individuals with common goals can also act as data collectors via crowdsourcing and share the data amongst themselves or with the public, e.g., OpenStreetMaps (OSM)[4]. An organization decides if it is suitable (minimum privacy and commercial risks), easy (marginal sharing costs), and beneficial (reciprocity, tangible and intangible benefits) to share their data publicly. Some data, such as individuals' GNSS mobility traces or ride-hailing ridership, might be sensitive and cannot be released without anonymization. Data with no or limited risks can be shared with partner organizations, clients, communities, or the general public.

2.2.2 Proprietary, public and open data

The public and proprietary data are differentiated in Figure 2.1. Informally, the term "Public data[5]" refers to publicly available, free data with or without usage restrictions. In this chapter, we formally define public data as a superset of open data, inspired by Kerle (2018) and Wynne-Jones (2019). When data are accessible, allowed to be used for any

[2] `https://www.tomtom.com/products/historical-traffic-stats/`

[3] `https://movement.uber.com/`

[4] `www.openstreetmap.org/`

[5] We could not find an official definition of "public data" in two popular dictionaries, namely Oxford and Merriam Webster, although there are references in the grey literature (Kerle, 2018; Wynne-Jones, 2019)

purpose, and redistributed free of charge with almost no restrictions, they can be termed Open data (The World Bank, 2019).

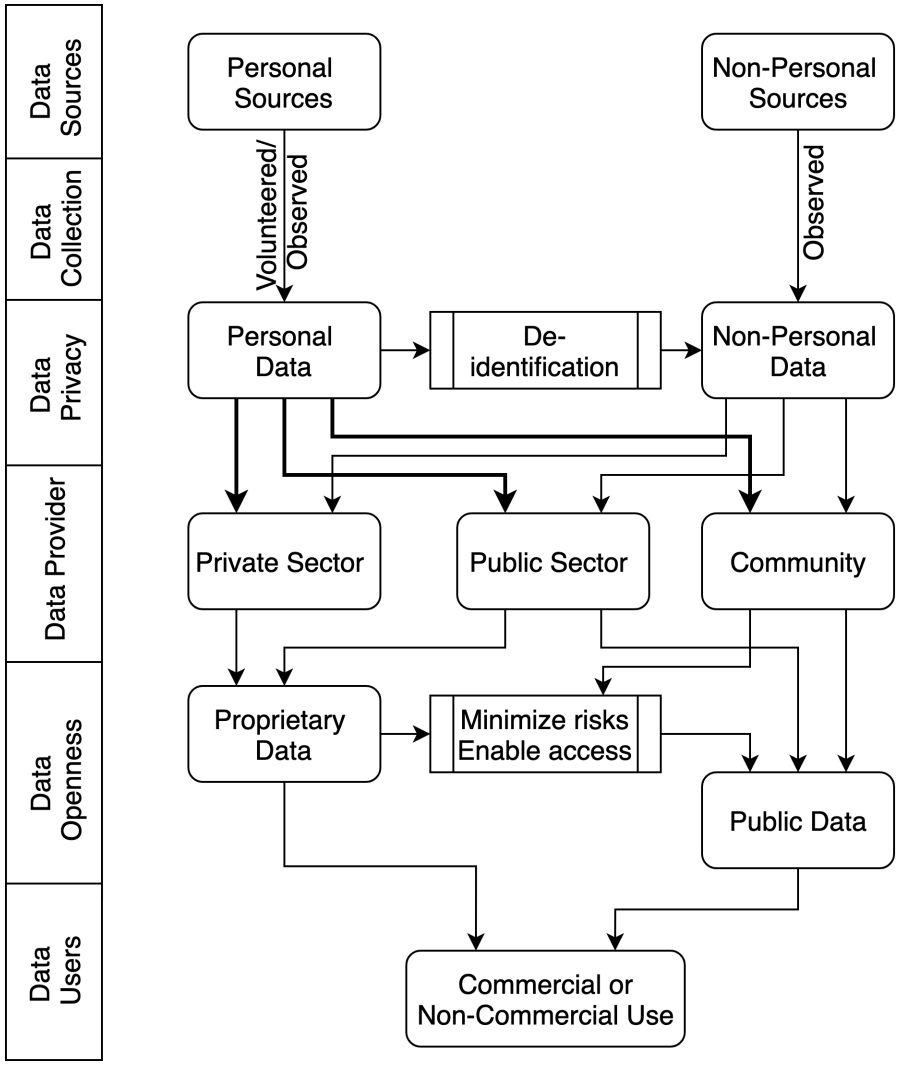


Figure 2.1: Overview of the production and operational flow of the data, partly inspired by OECD (2019).

In this paper, the term public data refers to data that are accessible and free of cost. Unlike open data, public data can be restricted in their usage (e.g., non-commercial licensing) and shareability. Consequently, while open data are always "public," public data are not necessarily "open". Furthermore, public data are not the same as Public Sector Information (PSI), where the latter denotes data emerging from government institutions. During the past few years, the data revolution has played a definitive role in creating public awareness and participation in using public data. The number of published articles shows that research using public data has gained momentum in the last

15 years (Figure 2.2). This rise in public or open data research was strengthened by policy initiatives introduced in 2009/2010 to increase access to government data. The open data revolution received a significant push by Obama's Open Government Directive in 2009 (U.S. Government, 2009) to increase transparency in the Executive branch. This step was complemented by other initiatives, such as the OGP (2011) initiative, the amendment to the EU's PSI Directive in 2013 (European Commission, 2013) or the G8 Open Data Charter in 2013 (Welle Donker & van Loenen, 2017). These and many other initiatives in different parts of the world continue to advance the formalization of open data's legal and technical aspects (K. Janssen, 2011).

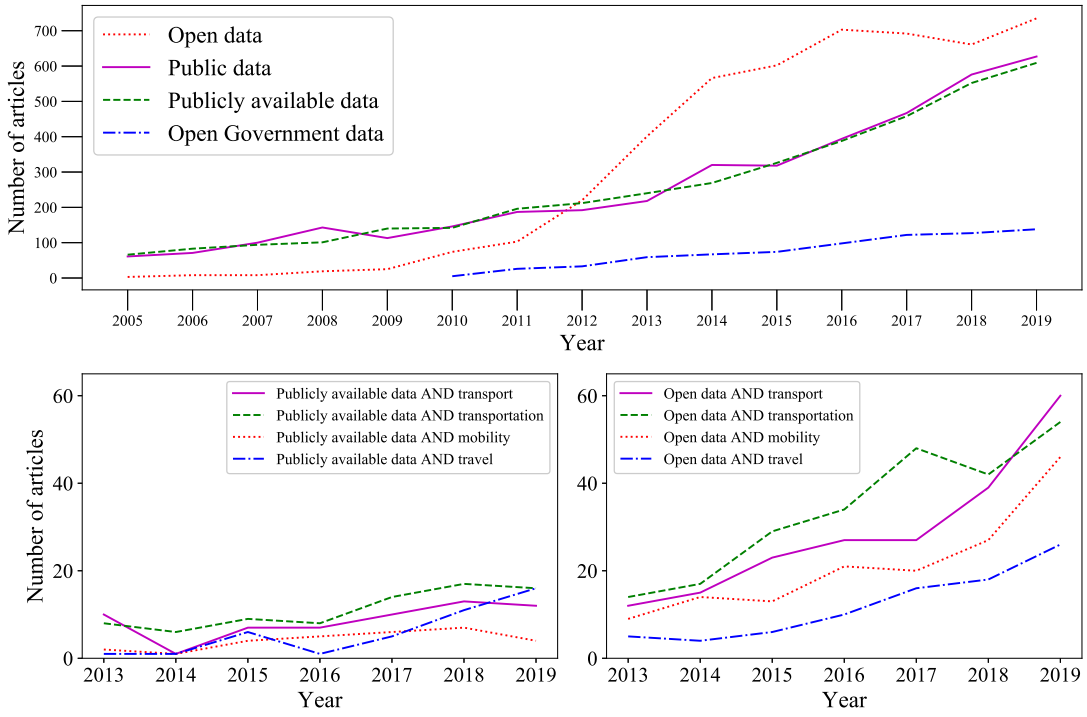


Figure 2.2: The trend of articles in SCOPUS published from 2005 to 2019 with the keywords: (top) "public data", "publicly available data", "open data", and "open government data"; (bottom) combinations of "publicly available data" (left) and "open data" (right) with transport domain keywords. The analysis was done in early 2020, so the articles up to 2019 were analyzed.

The trend of the studies reflects that the term "open data" is more prevalent in the scientific literature than "public data" (Figure 2.2). The same trend is also observed in studies related to the transport domain. However, using the terms "open data" and "public data" interchangeably can be misleading. For instance, European Data Portal (2018) refers to the Uber Movement dataset as private open data, even though Uber (a private company) does not allow its data for commercial purposes. Thus, these specific data do not entirely fulfill the open data principles stated by the Sunlight

Foundation (2010). There are several reasons why private organizations and even public sector institutions are unwilling to share their data. Protecting user privacy and business interests are two commonly cited reasons for this reluctance (K. Janssen, 2011). Some other (non-exhaustive) reasons include bureaucratic laxity, lack of political support, fear of public criticism (especially for public sector organizations), lack of skills, infrastructure, and demand for the data (Young & Verhulst, 2016). However, private (or even public) organizations may permit limited data usage for research, non-commercial, and commercial purposes with a "razor and blades" model for creating vendor lock-in (Welle Donker & van Loenen, 2016). For example, Twitter's Application Programming Interface (API) and Google's routing APIs allow limited free usage. In the transport domain, researchers, academics, and even policymakers can profit from such data by using it for modeling and data mining applications (Chaniotakis & Antoniou, 2015; Y. Cui et al., 2018; Llorca et al., 2018). Therefore, public data often are valuable for data users even if they come with certain restrictions.

Data-based value creation can be accelerated by both the providers and the users. On the one hand, the supply or availability of data (by a provider) in the public domain will result in more proof of concepts and applications for those specific datasets, e.g., geotagged text from Twitter. On the other hand, demand for some actionable data for humanitarian and societal causes can push organizations to innovate, collaborate and share the data (Google Mobility Reports[6], Facebook's Data for Good[7]). Public data also holds the potential to support research and validation. However, in some cases, sharing data might be a challenge for researchers. Childs et al. (2014) find that researchers face pressure from funding organizations to open the research data, which may be impossible due to professional ethical and methodological concerns. Further, the different scientific disciplines differ in their needs and use of the data (Arzberger et al., 2004).

2.2.3 Prominent non-conventional and emerging data in transport

The current data age (innovation and improvements in information, communication, and computing) allows using passively collected, big and crowdsourced data in transport modeling. For example, data from mobile devices, social media, and Automatic Fare Collection (AFC) sources are often labeled as Big data, allowing researchers to analyze their role, benefits, and challenges due to their "Big" nature. Milne and Watling (2019) studied the implications of big data for transport systems planning and highlighted future challenges. Welch and Widita (2019) reviewed big data applications in public transport under different categories, including user behavior and demand. On a similar note, Zannat and Choudhury (2019) analyzed the role of big data in public transport planning by focusing on the three types of data, namely smart card data, mobile phone data, and GNSS/ Automatic Vehicle Location (AVL) data. Prominent non-conventional and emerging data include:

[6] `https://www.google.com/covid19/mobility/`
[7] `https://dataforgood.fb.com`

1. Social media has grown dramatically over the last decade. Over 4 billion people worldwide used social media in 2022 (Statista, 2023). Social media applications are popular for social networking (Facebook, LinkedIn), microblogging (Twitter, Sina-Weibo), location discovery (Foursquare, Google Places), media sharing (Instagram, Flickr), as well as rating and reviewing (Yelp, Trip Advisor). Social Media data can be featured alongside the geographic location captured through mobile devices (smartphones and wearables). These geotagged social media data are sometimes referred to as Geographical Social Media (GSM) services or Location-Based Social Network (LBSN). Chaniotakis et al. (2016) and Rashidi et al. (2017) reviewed the potential of social media data for travel behavior modeling.

2. Mobile phones act as ubiquitous sensors and generate large amounts of location data of basically two types: mobile phone network data (H. Huang et al., 2019) and sensor data (Prelipcean et al., 2017; Zannat & Choudhury, 2019). The network data is generated when the user makes/ receives a call or SMS, accesses the internet, and during network-related events such as location updates (H. Huang et al., 2019). The smartphone sensor data, consisting primarily of GPS and motion sensors, are collected through mobile applications. Both the network and sensor data have applications in travel behavior modeling (Gadziński, 2018; Rojas et al., 2016).

3. Crowdsensed information, for example, consists of large datasets built with the help of a large group of people. In Mobile Crowdsensing, individuals "collectively share data and extract information" with the help of a sensing device (like smartphones) towards a common goal (Liu et al., 2016), such as identifying spatial-temporal patterns of a phenomenon. Crowdsensed data from mobile phones and social media platforms (Chaniotakis et al., 2016; Efthymiou & Antoniou, 2012), such as Twitter data, can, for example, help to study highly dynamic and disruptive events (Bagrow et al., 2011; Chaniotakis et al., 2017).

4. Traffic data collection is transforming, too. Antoniou et al. (2011) proposed a classification based on data collection functionalities of the sensor, i.e., point sensor, point-point sensor, and area-wide sensors. AVL is a computer-based system to collect and transmit information about the vehicle's actual location (Strong & Wolenetz, 2005). AVL data can be collected primarily by following three methods:

    a) On-board sensors: GPS provides information about a user's or vehicle's location, time, and velocity at any moment, based on signal exchange with a system of more than 20 satellites. Vehicles equipped with onboard sensors participate in transmitting their location data using GPS receivers. These data are called probe vehicles or Floating Car Data (FCD) (Westerman, 1995). Besides navigation devices, smartphones carried in private cars and commercial and transit fleets are also used to transmit GPS location data by, e.g., Google Maps, INRIX, Waze, or TomTom. As these data are collected with the help of several devices on the road, they are also referred to as crowdsourced traffic or AVL data (Travers, 2010).

b) Static ground-based scanners: WIFI/ Bluetooth scanners can be an alternative to conventional fixed signposts, street cameras, or loop detectors for traffic data collection. Bluetooth is a short-distance communication protocol used by mobile phones and vehicles. A Bluetooth inquiry device searches for nearby Bluetooth devices and two devices connect if they operate at the same frequency (Bhaskar & Chung, 2013). The use of Media Access Control (MAC)[8] data from the WIFI signals also follows a similar principle.

c) Mobile (moving) scanners like drones are relatively new candidates for traffic data collection. A few pilot studies have recently demonstrated their application in collecting rich traffic data (Barmpounakis & Geroliminis, 2020). In recent years, advances such as fast microprocessors, efficient storage, and wireless communication technologies have allowed the use of drones for many civil applications (González-Jorge et al., 2017). Aerial footage from Unmanned Aerial Systems (UAS) or more commonly known as "drones", is one of the newest methods for collecting traffic data and has notable advantages such as observation of naturalistic driving behavior and detailed driving trajectory (Barmpounakis et al., 2016; Pham et al., 2020).

5. AFC systems are popular among transit agencies, especially in closed transit systems. These systems use contact or contactless smart cards for efficient fare collection and help control station access. Smart cards can store and process passenger data, such as personal information, trip data (boarding and/ or alighting time and location, frequency of use), and fare transactions (Pelletier et al., 2011). These data are known as AFC data, and public transit planning and modeling have benefited from them (Faroqi et al., 2018; Pelletier et al., 2011).

6. Volunteered Geographic Information (VGI) belongs to the context of big data and represents crowdsourced georeferenced data that are recorded voluntarily by a large user community. VGI emerged during the first decade of the 21st century and is mostly driven by communities, such as OSM (OpenStreetMap Contributors, 2018). As data are crowdsourced, they are usually available free of charge and, therefore, are open.

Certain datasets have gained prominence due to their standardization. Google developed GTFS for an online public transport trip planner in Portland, Oregon. Since then, it has been applied to many regions worldwide and was established as the de-facto standard for sharing public transport schedules. Similarly, the General Bikeshare Feed Specification (GBFS), an open data standard for bike and scooter sharing systems, was developed under the North American Bikeshare Association (2015). It aims to provide real-time information about bike-sharing systems' current status and availability.

[8] Unique identifier assigned to a device by the device manufacturer for communication within a network

2.2.4 Research gaps

The potential of a few emerging data, such as social media and mobile phone data, has been demonstrated in previous transport modeling studies (Milne & Watling, 2019; Rashidi et al., 2017; Zannat & Choudhury, 2019). Nevertheless, there is a need to systematically define and discuss public data for transport modeling to clarify the topic. This could benefit mobility data providers and users to understand the non-conventional data for transport modeling in terms of their applications and availability in one place. This could also help initiate a conversation and efforts to make the high-potential transport data a priority for increased access and sharing. To the best of our knowledge, most review studies on transport-related datasets mentioned above do not focus on the openness or public availability aspects, which are crucial from data users' viewpoint. For a comprehensive overview, it is essential to concurrently analyze these data's applications, specifically in transport modeling. Because of the above, we see an opportunity to address the following SRQs:

SRQ(1): What are the main attributes to classify data based on their public availability or openness?

SRQ(2): Which categories of the proposed typology do the prominent non-conventional data used for transport analyses belong to?

SRQ(3): What are the common applications of these data for transport modeling?

SRQ(4): What are the strengths and weaknesses of these data in terms of their applications and availability?

2.3 Errors in traffic data from emerging sources

2.3.1 Data errors

Researchers and practitioners need real-world traffic data to study traffic behavior and implement traffic management strategies. Traffic data collection can be primarily classified into three types, point measurements (loop detectors), point-to-point (FCD and Bluetooth scanners) or edge measurements, and area-wide measurements (Antoniou et al., 2011). Sensor measurements often come with errors, and thus errors are prevalent in traffic data too. Noise and anomalies (outliers) are two common types of errors (Teh et al., 2020). These errors deviate the measured signal from its desirable value, and therefore the data require processing before use. The desired value is the best possible representation of the *true underlying signal* that can be measured (O'Haver, 2022). The desired value may differ from the absolute true value of a signal that may or may not be possible to measure. Processing time-series or sequence data is classified into three main tasks: filtering, smoothing, and prediction. Kalman (1960) formalized the distinction between filtering and smoothing. Suppose the observed time sequence $y(t_0), \ldots, y(t_n)$ of length $n+1$ from which we need to estimate the unobservable or desired value of the true signal at $t = t_i$, where t_i is the time of interest. If $t_i < t_n$, it is data smoothing,

whereas if $t_i = t_n$, it is data filtering, and if $t_i > t_n$, then it is a prediction task (Kalman, 1960). In the following paragraphs, we briefly introduce the topics of noise and anomaly, followed by a discussion of a few studies on vehicle trajectory datasets from the traffic research domain.

2.3.1.1 Noise

Noise is the unwanted component of the signal which is not relevant to a specific task. Removal or treatment of noise is a general prerequisite for data usage. The source of noise in the data can be the measuring device or sensor and its surroundings during the data collection, e.g., in drone videography, noise can be introduced due to the vibrations of the camera apparatus. This could be characterized by the presence of a periodic high-frequency signal superimposed on the desired value. Data processing methods can also introduce noise in the data, e.g., extraction of trajectories from even a stabilized video could be noisy depending on the algorithms and tools used. For time-series data, smoothing refers to a broad array of methods to remove the noise from the data. This is commonly done by allowing only the low frequency of the signal to pass while attenuating the high-frequency component (Holloway, 1958). The moving average filter is one of the most common low-pass filters, where the current estimate is the rolling average of neighboring values.

Savitzky-Golay (SG) filter is another example of a low-pass filter. SG filter is an efficient method of data smoothing using local least-square polynomial approximation (Schafer, 2011). Polynomial fitting on a sub-sequence of length 2M+1 (where M is a positive integer) and then evaluation of the polynomial's output at the central point is equivalent to the convolution of the sub-sequence with a fixed set of integers (impulse response) (Savitzky & Golay, 1964). The output samples (y) obtained from the discrete convolution of fixed weights (h) with the input sample (x) is shown in equation 2.1 (Schafer, 2011):

$$y[n] = \sum_{m=-M}^{M} h[m] \cdot x[n-m] = \sum_{m=n-M}^{n+M} h[n-m] \cdot x[m] \tag{2.1}$$

For a uniformly spaced sequence, the weights are computed only once based on the length of the sub-sequence (window size 2M+1) and the polynomial degree. This is beneficial for the sensor data because the sensor data are generally generated at a fixed frequency and equally spaced. Common weights for every convolution operation on the sub-sequence make it highly efficient in speed and memory. The output of the SG filter is suitable for subsequent application of the anomaly detection algorithm due to the high signal-noise ratio. For a polynomial of degree 0 or 1, SG filter is equivalent to a moving average filter (Savitzky & Golay, 1964).

Gaussian Filter (GF) is another popular filter, especially in image processing. As the name suggests, the input sample is convoluted with a Gaussian kernel (Equation 2.2) to get the smoothed estimates:

$$N(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2}{2\sigma^2}} \tag{2.2}$$

Where σ is the standard deviation. The kernel is truncated symmetrically beyond a specified number of σ. The output of the GF is weighted more towards the central values of the input samples due to the characteristic shape of the Gaussian kernel. This makes the GF a gentler smoothing filter than a moving average filter (Fisher et al., 1997). We want to point out that the methodical steps in noise filters, such as numerical approximation or truncation, could introduce or exacerbate errors in processed data (Rafati Fard et al., 2017). It is a trade-off between introducing these errors and removing the data's noise. Filters are justified if their output is closer to the desired value than the input data.

2.3.1.2 Anomalies

The term "anomaly" refers to a behavior different from the usual or representative behavior of the system (Chandola et al., 2009). When the primary objective is to recover the representative signal, anomalies are detected, removed, or replaced. In contrast to noise, anomalies are not always unwanted. Anomaly detection in time-series data is an active research topic. It is also the primary focus in various fields, such as finance, network security, and health (Chandola et al., 2009). Time-series data can consist of either a single-point anomaly or an anomalous sub-sequence. In high-frequency sensor data, sequence anomalies can be prominent as a single disturbance in the signal can span over multiple points. For instance, unrealistic high transient values or peaks can characterize anomalies in the data. Chandola et al. (2009) categorized the anomaly detection techniques under classification, clustering, nearest-neighbor, statistical, information-theoretic, and spectral-based methods. Anomaly detection can be seen as a supervised learning task, but this is practically constrained due to the often unavailability of ground-truth labels. This is why unsupervised techniques hold significant potential for anomaly detection. These methods aim to find the best separation between the usual and anomalous data points/ sequences based on the specified parameters (distance, density, and probability). For instance, Eskin (2000) used a machine learning model to learn the probability distribution over the data and then applied a statistical test to detect the anomalies.

2.3.2 Challenges in traffic data collection from drones

In a study by Barmpounakis and Geroliminis (2020), Authors describe the challenges associated with collecting drone video data for an extended area. An essential set pertains before and during the drones' flight, which must be planned and accurately determined because of weather, battery backup, video quality, regulatory approvals, and technical expertise. However, another set of essential challenges pertains to post-processing the video recording after the drone flight. The researchers use state-of-the-art computer vision algorithms to detect and track vehicles and extract vehicle trajectories from the raw videos. Since the errors in the position of the vehicle are in the order of 20 cm or less

Barmpounakis and Geroliminis (2020), trajectory extraction follows quite accurately the position of the vehicle. Nevertheless, vehicles are not point objects but cover significant space; minor errors in the position can accumulate when speed or acceleration variables are calculated. Furthermore, the urban driving environment is more complex than highways due to the increased heterogeneity of vehicle classes, traffic signals, congestion, intersections, parking vehicles, and occlusion due to high-rise buildings and trees. All these factors could introduce noise and anomalies or outliers in the acceleration profiles of the extracted trajectories from the drone videos, and thus, the data may require additional treatment. There is an opportunity to address this issue by analyzing the trajectories and treating the noise and anomalies to obtain the desired speed and acceleration data. These data contain many naturalistic trajectories, and thus filtering the anomalies and smoothing the noise could accelerate subsequent research attempts. For researchers to fully take advantage of such a detailed and large dataset, it is first necessary to find appropriate techniques to detect these cases and filter them efficiently.

2.3.3 Noise and anomalies in vehicle trajectories

It is essential to clarify what is considered usual or representative behavior of the system viz-a-viz "abnormal" or "unusual" within the scope of this research. This sub-section deals with the naturalistic trajectory data, which is of great interest to the researchers as it provides in-situ driving behavior. In a recent study (Venthuruthiyil & Chunchu, 2020), the authors provide a review of the trajectory smoothing/ filtering techniques to process trajectory data from diverse data sources such as ground-based camera videos, drone videos, and instrumented vehicles. Since this section focuses on anomaly detection, we list the range of acceleration and deceleration noted in some previous studies (Figure 2.3). We point out that the drone videography data are still in the early phases, and thus there are limited studies on processing such data. To compensate for this, we consider a few prominent studies from other data sources dealing with vehicle trajectories. We also identify the context (driving environment, vehicle types, data collection, and processing methods) and data sources in these studies, and find that the acceleration values have varying ranges. Thus the context of the study should be considered prior to its acceptability.

Bokare and Maurya (2017) analyzed the acceleration and deceleration behavior of different vehicles using GPS data. They noted that acceleration rates (for all vehicles except trucks) increase from minimum to maximum at initial speeds and then decrease with speed. Sangster et al. (2013) used naturalistic data ("100-Car study") from different sources such as GPS, On-board Diagnostics (OBD) and accelerometer box for studying the car-following behavior. Their study found that lags in GPS data can result in oscillations in calculated speed. They identified the outliers in the speed and acceleration time series by checking the observed data against the anticipated physical limitations and replaced outliers with the interpolated data. As a result, they transformed the maximum instantaneous acceleration (negative sign for deceleration) ranging [-303.6 m/s^2, 303.0 m/s^2] in the raw data to [-8.9 m/s^2, 9.0 m/s^2] in the smoothed data. Punzo et al. (2011) used jerk values (derivative of acceleration) to identify the infeasible accelerations in Next

Generation Simulation (NGSIM) dataset, which pertains to highway context. Further on, Montanino and Punzo (2015) reconstructed trajectories from the NGSIM dataset using a series of steps such as outlier and noise removal and local reconstruction. They adopted a threshold of 8 m/s^2 and 5 m/s^2 for deceleration and acceleration, respectively, for outlier detection, and after reconstruction, longitudinal accelerations range in [-5 m/s^2, 3 m/s^2]. They mention that vehicle mistracking is a likely source of error when extracting the trajectories from the video recordings. Coifman and Li (2017) addressed the vehicle mistracking by manually re-extracting the trajectories with better quality from the NGSIM videos while reporting that the errors in the data cannot be corrected through cleaning or interpolation.

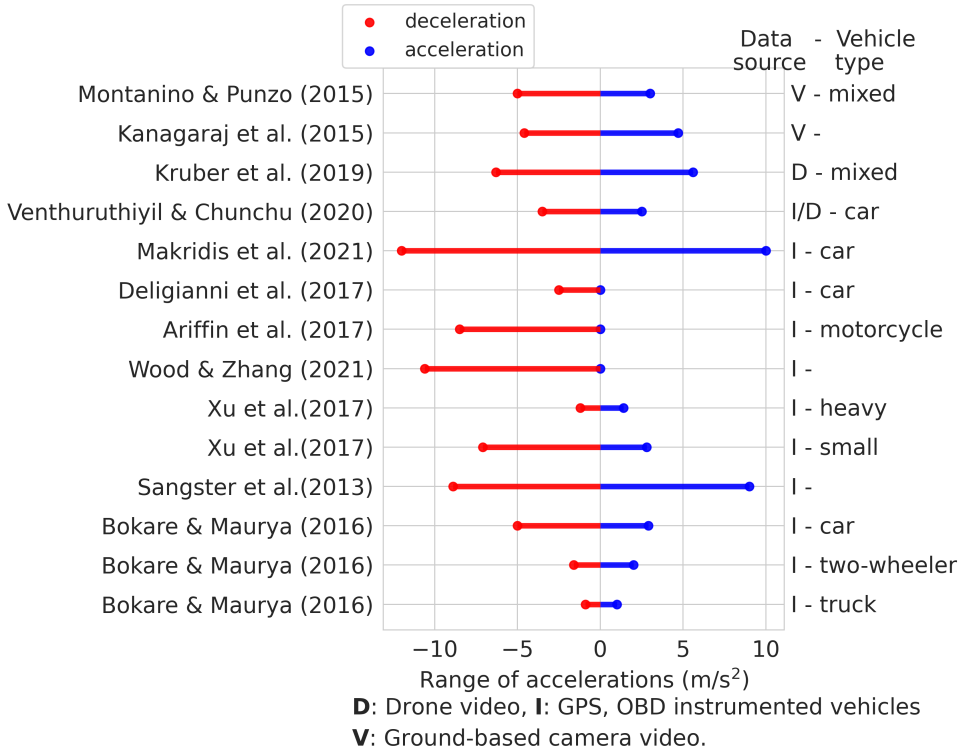


Figure 2.3: Acceleration and deceleration ranges in the selected studies (Ariffin et al., 2017; Bokare & Maurya, 2017; Deligianni et al., 2017; Kanagaraj et al., 2015; Kruber et al., 2019; Makridis et al., 2021; Montanino & Punzo, 2015; Sangster et al., 2013; Venthuruthiyil & Chunchu, 2020; Wood & Zhang, 2021; Xu et al., 2017) including those using naturalistic trajectory data.

Analysis by Kruber et al. (2019) on a newer dataset [Highway Drone (highD)] on German freeways found the longitudinal acceleration in the range [-6.3 m/s^2, 5.6 m/s^2]. Xu et al. (2017) collected longitudinal acceleration data using motion sensors on a two-lane mountain highway. They applied data filtering and peak detection algorithms to remove noise and determine the maximum accelerations. According to their findings, acceleration

values ranged between [-7.1 m/s^2, 2.8 m/s^2] and [-1.2 m/s^2, 1.4 m/s^2] for small and heavy vehicles, respectively. In the urban driving context, Kanagaraj et al. (2015) extracted trajectories from a video recording on a section of the road. They smoothed the data using locally weighted regression and obtained longitudinal accelerations in [-4.6 m/s^2, 4.7 m/s^2]. OpenACC is a recently released dataset related to car-following experiments. In this study, Makridis et al. (2021) used U-Blox M8 devices that were equipped with motion sensors and GNSS receivers. They post-processed speed and acceleration values using the piece-wise cubic polynomial to compensate for the noise levels in the raw data. The range of accelerations is about [-12 m/s^2, 10 m/s^2] considering human and Adaptive Cruise Control (ACC) drivers. Rafati Fard et al. (2017) used wavelet transform and wavelet-based filter to process the outliers and noise in the NGSIM data. Their method detects outliers based on the local properties of the data and thus is an improvement over globally defined thresholds. Venthuruthiyil and Chunchu (2018) reconstructed an error-prone trajectory from video data using locally weighted polynomial regression. In their recent work, Venthuruthiyil and Chunchu (2020) processed drone video data by retrieving missing data and then smoothing them. Before smoothing, they removed the outliers using a median filter. A median filter is a statistical filter wherein the window size and threshold are specified to detect outliers. Afterward, they processed the data using the combination Recursively Ensembled Low-pass filter (RELP) and adaptive tri-cubic kernel.

2.3.4 Research gaps

From the above discussion, we find that studies for highway driving are more prevalent than urban driving. The range of practical or possible acceleration/ deceleration can depend on many factors, such as desired speed, driving context (intersection, highway, ramp), vehicle class and type, driving style, surrounding vehicles, and data sources. One of the challenges of drone videography is that the errors in the data are not consistent as it could be a result of extrinsic (wind burst, object occlusion) and intrinsic (image processing, object tracking (Coifman & Li, 2017)) causes, e.g., the reasons for outliers in the drone data could be i) a sudden wind burst that can move the drone, ii) tracked vehicles with reduced visibility (minor roads, occlusion due to buildings or trees), iii) vehicles being tracked in the edges or not well-calibrated areas of the video. Although computer vision algorithms have advanced massively during recent years, it has been recognized in previous studies that trajectory data from drone videography have a heavy-tailed data distribution due to outliers and needs special treatment (Barmpounakis & Geroliminis, 2020; Venthuruthiyil & Chunchu, 2020).

Compared to noise treatment, outlier detection is a relatively challenging task to identify systematic issues during the data collection process, thus requiring specialized treatment. Accelerations with unrealistic peaks characterize outliers. They are generally removed in the trajectory datasets using a pre-defined threshold (Montanino & Punzo, 2013) or a statistical filter (such as the median filter used in (Venthuruthiyil & Chunchu, 2020)) on the speed or acceleration series. Such thresholds are manually defined by domain experts with care so that possible driving observations are not classified as outliers

(false positives) (Montanino & Punzo, 2015). Simple heuristics such as global thresholds are also trajectory invariant and cannot account for complex scenarios. An all-embracing and insufficiently flexible filter would not be suitable, as there is always the caveat of over-smoothing or false positives. It overlooks crucial information, especially when it comes to lane-changing maneuvers or aggressive driving behavior like harsh acceleration or harsh deceleration (Barmpounakis et al., 2020; Mahajan et al., 2020; Vlahogianni & Barmpounakis, 2017). Outlier detection is challenging, given the heterogeneity of traffic and driving behaviors.

We want to point out the drawbacks of simple and popular outlier detection methods, namely z-score or modified z-score algorithms. Using domain expertise to label the anomalies, one needs to specify the window size (over which statistical measure such as mean or median is estimated) and threshold distance (such as the number of standard deviations). There is a trade-off between false positives and false negatives depending on the window size and threshold distance, which emphasizes fine-tuning these parameters. The use of mean or median statistics can be biased in urban traffic when the vehicle is stationary at the intersection and thus needs tuning for each vehicle.

Research by Rafati Fard et al. (2017) is based on local detection of outliers using wavelet transforms. The use of wavelet transform has its challenges, such as the selection of the mother wavelet (Rafati Fard et al., 2017). Large and diverse datasets, such as the pNEUMA dataset from drones, demand complex heuristics for anomaly detection, and their manual specification is impractical. The above aspects emphasize selecting a scalable methodology for a large dataset, which exploits data-driven or machine learning models and replaces complex heuristics. In view of the above discussion, we aim to address the following SRQs:

SRQ(5): How to develop a scalable methodology to treat noise and anomalies in vehicle trajectories from drone videography data?

2.4 Opportunistic data from emerging sources for mobility analysis

2.4.1 Human mobility during special events

Human mobility is a complex phenomenon, the manifestation of people undertaking different daily activities to satisfy their needs and wants. The performance of such activities depends on various population and environmental factors. The interaction of land-use and transport systems in activity generation has also been widely researched and modeled (Acheampong & Silva, 2015). Places well connected with transport systems and dense neighborhoods will generate more trips than less connected and sparse neighborhoods. The interaction between land use and transport and individual constraints is captured by the concept of accessibility (Geurs & van Wee, 2004; Hansen, 1959). Apart from the above "structural" factors, planned or unplanned special events (Dunn, 2007), weather conditions (Cantelmo et al., 2020; Sabir, 2010) can also influence where, when, and how people move in the short term. Cities strive to plan, design, and operate their

transportation systems based on the forecasted demand derived from the activities due to these factors.

In case of disruptive and highly dynamic events, such as natural or human-made hazards, people tend to adapt their short-term (Yabe et al., 2019) and long-term mobility behavior (Gray & Mueller, 2012; Yamamura et al., 2014) to the prevailing conditions. For example, travelers might be reluctant to enter the underground metro after an earthquake or go near the sea coast in case of a cyclone or tsunami warning. COVID-19, one of the most severe pandemics in the last 100 years, affected almost the entire world in unprecedented ways. To control COVID-19 transmission, guidelines such as social distancing, masks, and movement restrictions were recommended or enforced. In response to these measures, people not only reduce their mobility (Pullano et al., 2020), but also adapt their travel patterns to limit their exposure by avoiding places with many cases (Brinkman & Mangum, 2020).

2.4.2 Crowdsensing data for mobility behavior analysis

The study of human activity and travel behavior is traditionally (and commonly) based on the data from Stated and Revealed preference surveys. Alternate data sources can play a crucial role, especially during situations like Corona Virus Disease 2019 (COVID-19), when responses or policies have to be adopted faster, whereas surveys take some time in planning and execution. Emerging sources of data (Antoniou et al., 2011), such as social media (Rashidi et al., 2017) or mobile phones (Järv et al., 2014), have pushed the use of data-centric approaches to study activity patterns. Mobile devices or sensors with wireless communication or the internet can help understand when, where, and if people are crowding. Some of these data are available on the internet. They could be exploited to create the first line of defense against this pandemic and to develop policies to mitigate its impact on transport systems and local businesses. During COVID-19, Mobile phone data emerged as a potential source to understand and respond to the pandemic, as it provides a large spatial-temporal information (Grantz et al., 2020). A study using mobile phone data found that the lockdown in France caused a 65% reduction in the performed trips, especially work-related trips during peak hours and long trips (Pullano et al., 2020). Researchers in the US and China also applied the mobile phone data to establish that social distancing and decreased mobility (due to restrictions or lockdowns) are positively correlated to the reduced growth in COVID-19 cases (Badr et al., 2020; Fang et al., 2020).

The importance of public data sources cannot be overstated in the case of special events. During COVID-19, several organizations came forward by making some of their data publicly available to help governments and citizens understand the changes in activity patterns and travel behavior. Some of the prominent examples are COVID-19 Community Mobility Reports (Google, 2020a) and Apple Mobility Reports (Apple, 2020). Using these data, we obtain activity and mobility trends for Munich (in Bavaria, Germany), which provide information about the overall changes in activity and travel mode patterns in a region, respectively (Figure 2.4). In this Figure, the overall activity and mobility trends confirm some expected behavioral patterns during COVID-19, such as a decline in transit mode use, a drop in retail and workplace-related visits, and increased stays

at residences. The grocery-related visits can be seen to recover from the initial drop in visits gradually.

One exciting and seemingly potential data set is the crowdsensed check-in rate or busyness at the POIs. The crowdsensed check-in rate is the representation (in absolute or relative terms) of the number of people or customers visiting a specific establishment at a given time, thus showing its busyness. These data are primarily collected from smartphone applications, in which the user's location history is enabled, such as geotagged data or LBSNs. Geotagged tweets (Chaniotakis et al., 2017), Foursquare check-ins (D'Silva et al., 2018), and popularity trends (Capponi et al., 2019; Timokhin et al., 2020) are some examples of such data that capture the spatial-temporal evolution of the demand and have already shown their utility in previous studies.

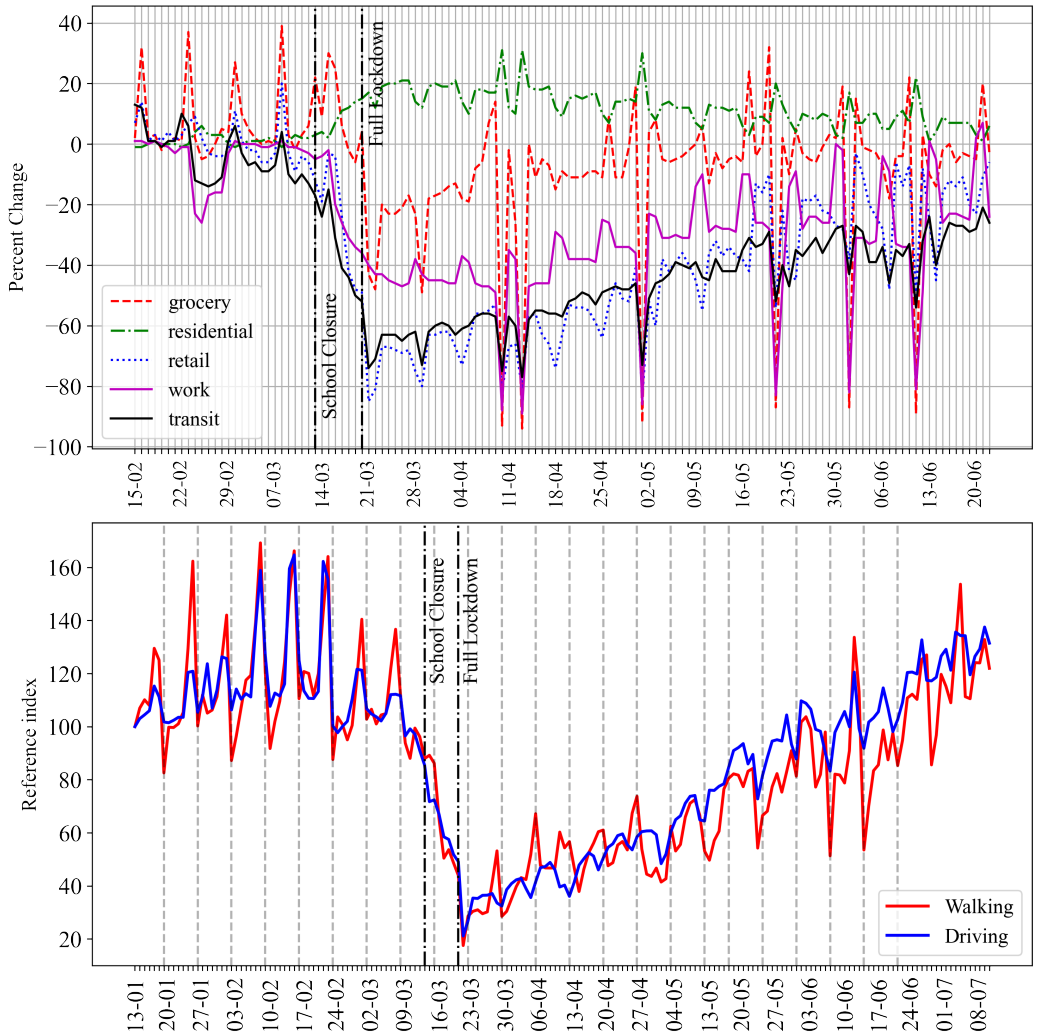


Figure 2.4: (Top) Activity patterns in Bavaria and (Bottom) travel mode patterns in Munich, data sources: (Apple, 2020; Google, 2020a)

2.4.3 Research gaps

The above examples illustrate the potential of passively collected and publicly available data for informed policy decisions during special events, such as COVID-19. Apart from the data source, the level of data aggregation also decides its usability. The aggregated datasets (at the city or county level) do not provide detailed information at finer geographical scales such as the POI level and thus have limited applications. On the other hand, disaggregate datasets (at finer geographical scales) could provide richer information for understanding heterogeneity across POIs and demographics (Roy & Kar, 2020). Thus, disaggregated data allows for analysis at a local level for understanding the mechanisms between activity patterns and environmental factors, e.g., at the level of a shop or a transit stop. Therefore, crowdsensed data in disaggregate form could be potentially useful for analyzing the spatial-temporal changes in demand patterns during special events and leads to the following SRQ:

SRQ(6): How to apply publicly available crowdsensing information for changes in spatial-temporal demand patterns during special events?

2.5 Efficient methods for traffic calibration

2.5.1 Traffic simulation calibration

A transportation system comprises different parts and their interactions, resulting in travel demand and supply of transport services (Cascetta, 2001). Researchers and practitioners develop transport models to study the effects of an ongoing or new phenomenon on the transport system, e.g., the effect of new technology or a policy change on - how, when, from/ to where people move, and their resultant social, economic, and environmental impacts. While analytical transport models exist, their outputs are often inaccurate (as they fail to fully capture the complex dynamic interactions in a transport network). Additionally, the computational burden of these models is high when the goal is to simulate large-scale scenarios. Dynamic Traffic Assignment (DTA) simulation can represent the short-term traffic flow variations and behavioral choices in a large-scale network (Ben-Akiva et al., 2012). Therefore, traffic simulation models are increasingly preferred in modeling applications.

Calibration of model parameters is a key requirement before the models are applied for analysis and forecasting, as inaccurate parameters translate into unreliable simulation outputs. Calibration is the process of finding the simulation model's parameters so that the difference between the simulated behavior (counts, travel time, speed) and observed behavior is minimized. Calibration is formulated as an optimization problem to minimize the value of the objective function subject to constraints. Thus, calibration of traffic simulation models depends on three main factors, namely calibration method [objective or fitness function and its formulation, calibration approach, optimization algorithms, the Goodness of Fit (GOF) criteria], simulation model (assignment method, level of detail) and data [Measures of Performance (MOP), data sources, aggregation, coverage]

(Antoniou et al., 2016). The calibration of DTA models is an active field of research with applications such as demand calibration and real-time traffic management.

2.5.2 Calibration approaches

Omrani and Kattan (2012) reviewed DTA model calibration, focusing on the calibration parameters and calibration approach. The calibration parameters belong to two categories: demand and supply. Demand model parameters pertain to trip generation, destination, departure time, mode, and pre-trip route choices. OD estimation is a specific case of demand calibration where time-dependent OD matrices are calibrated. On the other hand, supply model parameters pertain to during-trip route choice, link and junction performance functions, traffic flow models, and driving behavior models such as lane-changing and car-following. Depending on the granularity of the models, such as macroscopic, mesoscopic, and microscopic simulations, the nature of these parameters can change.

Researchers have proposed various methods exploiting the interaction of demand-supply parameters, data, models, and problem structure. Regarding the interaction of parameters, earlier demand models were calibrated, considering other supply parameters as constant and vice-versa. These approaches were followed by sequential (or iterative) calibration (Toledo et al., 2014), where supply calibration is followed by demand calibration in a loop. These approaches, however, failed to capture the intrinsic interaction between demand-supply (Toledo et al., 2014). In contrast, simultaneous calibration of all supply and demand parameters is reported to provide the most efficient estimates (Toledo et al., 2014), although at the cost of additional complexity. Another important distinction is between the *offline* and *online* calibration procedures depending on the planning or operational application, respectively. The former calibrates the model parameters given a set of historical observations. After this initial calibration, these parameters can be updated based on the real-time or streaming data for prevailing traffic conditions in an online calibration (Antoniou et al., 2005; Balakrishna et al., 2007).

As for the optimization algorithms, global search methods, EA (Evolutionary Algorithms) (T. Ma & Abdulhai, 2002), are reported to give good quality solutions. The global search methods are relatively less popular on large-scale networks, presumably because they become time-consuming and computationally expensive for large problems. The success of global algorithms depends on the properties of the model and might not scale very well on large networks. Only a few studies have used the algorithms' distribution and parallelization to improve the efficiency of these algorithms and demonstrated their application on medium-sized networks (Omrani & Kattan, 2018). On the other hand, researchers use local search heuristics, such as Simultaneous Perturbation Stochastic Approximation (SPSA) (Spall, 1998a), which are efficient in terms of time and computation. Black-box optimization using approximated gradients is widely used to calibrate the Origin-Destination (OD) matrices. Large-scale calibration is a highly under-deterministic problem with multiple possible solutions. Therefore, local search approaches need enhancements, domain knowledge, and sensitivity analysis to obtain the desired solution.

2.5.3 SPSA-based approaches for demand calibration

Demand calibration or estimation/ updating of the OD demand matrices using traffic counts is a well-studied problem for transport modelers. In fact, OD demand estimation is a special case of demand calibration where the traffic flow/ counts are used to estimate the OD matrix (Cascetta et al., 1993). When multiple time-dependent OD matrices are to be calibrated, the problem is also referred to as Dynamic Origin-destination Demand Estimation (DODE) (Cantelmo et al., 2018). Researchers have further tried incorporating domain knowledge to improve the performance of SPSA for demand calibration. Some of the successful applications of local heuristics are Weighted-SPSA (W-SPSA) (Antoniou et al., 2015; Lu et al., 2015), cluster-SPSA (c-SPSA) (Tympakianaki et al., 2015), adaptive-SPSA (Cantelmo, Cipriani, et al., 2014). Djukic et al. (2012) applied Principal Component Analysis (PCA) to tackle the high dimensionality of the calibrations to capture the input variation with fewer parameters. Subsequently, the potential of dimensionality reduction was demonstrated in PC-Generalized Least Squares (GLS) (Prakash et al., 2017), and PC-SPSA (Qurashi et al., 2022, 2020). Another approach is to assume a prior distribution (quasi-dynamic assumption) of the data to artificially reduce the number of variables (Cascetta et al., 2013), or to divide the problem into sub-task (Cantelmo, Viti, et al., 2014). Using meta-models to provide more domain knowledge in black-box optimization helps converge faster. For example, Osorio (2019) approximated the network model using an analytical representation and embedded it as a meta-model within the Simulation Optimization (SO) algorithm. This approach gave promising results for large-scale networks. In another recent study by Ho et al. (2023), authors used modified gradients in SPSA and proposed a differentiable Meta-model assisted SPSA (MSPSA) to speed up the convergence of the SPSA.

2.5.4 Averaging to handle parameter variance

Traffic simulators are stochastic systems, which implies that the simulation outputs and gradient approximations based on these outputs are stochastic too. Thus, different types of averaging are used to address this stochasticity. For instance, to address the variance in the simulation outputs (e.g., due to randomness in flow propagation, and route choice), averaging multiple simulations is done during each function evaluation. Random search choice algorithms, such as SPSA, lead to additional stochasticity because the random choice is made in a selection of perturbation vectors during the gradient approximation step, which induces randomness in the search process. To address the randomness in gradient approximation, Spall (1998b) recommended that for each gradient approximation, an average of a few gradient evaluations in every single iteration should be used. We term this technique as "gradient replications" to differentiate it from "gradient averaging" wherein gradients across current and past iterations are averaged. On this note, Kostic et al. (2017) tested gradient replications and gradient averaging with the SPSA for demand calibration. They found that gradient replications provide better convergence, whereas gradient averaging does not provide meaningful benefit, which supposedly could be due to a highly uneven and complex loss surface. However,

in a general context, such averaging is beneficial when the curvature of the objective function starts to flatten along a dimension, e.g., as in the case of the canal or a valley. In such situations, gradient descent-based optimization methods can be very slow in convergence. In these cases, *Momentum* can help to tackle the slow convergence (Ruder, 2016). Momentum tweaks the gradient descent by providing a short-term memory and taking the weighted average of the gradients from the past runs. References to gradient smoothing across iterations for SPSA can be found in literature (Spall, 1998b; Spall & Cristion, 1994).

Instead of gradients, averaging parameters or iterates (also called weights in machine learning) across iterations is another popular idea. Spall (2003) mentions that the innovation of the seminal work of Stochastic approximation method by Robbins and Monro (1951) is to do a "form of averaging across iterations". This was followed by maintaining the running average of the iterates in the case of stochastic optimization algorithms (Polyak & Juditsky, 1992; Ruppert, 1988) for better convergence. For iterate averaging to perform better than individual estimates, it is important that the majority of the individual estimates land within the local neighborhood of the true or desired estimate. Otherwise, averaging will lead to poorer estimates (Spall, 2003). Different modifications of iterate averaging are also applied in the case of Stochastic Gradient Descent (SGD) based algorithms in machine learning, where the running average of the weights of the neural network helps to smooth the trajectory of the SGD. For instance, Izmailov et al. (2018) proposed Stochastic Weight Averaging (SWA) where an average of the points/ iterates traversed by SGD with cyclical or constant learning rate is used. SWA finds much flatter solutions than SGD, leads to higher test accuracy, and improves the generalization ability of the neural networks.

Another averaging-related method is based on the ensemble concept. An ensemble of models means combining the decisions/ predictions of a set of individual models to provide a better prediction. Dietterich (2000) pointed out that in case of insufficient data, there can be many possible solutions to a problem, as in the case of an OD estimation problem. An ensemble of models can help to average the individual model "votes" and help to obtain optimal predictions. Further, in machine learning, many models use local search to optimize the objective function and can often get stuck in local optima. Therefore, an ensemble made by running multiple models with different initializations can provide better results. Bagging (short for Bootstrap Aggregating) is a common ensemble method (Breiman, 1996). Bagging predictor (Breiman, 1996) is a technique in machine learning, where multiple models are trained on subsets of the training data (bootstrapped datasets) and then the final prediction is the average of the predictions of these trained models. Bagging is effective if individual models have higher variance. Bagging is popular in machine learning to reduce the variance of the models. Breiman (1996) found that for unstable procedures, bagging works well and "can push a good but unstable procedure a significant step towards optimality". Bagging is useful if the individual models have high variance since the variance of the averaged model is reduced. One can draw parallels between bagging and iterate averaging since both involve using individual iterates/ models to obtain better estimates.

2.5.5 Research gaps

For a large-scale simulation scenario, calibration suffers from the "curse of dimensionality"(Cascetta et al., 2013; Djukic et al., 2012), because the size of the OD matrix is large and thus the number of parameters. This means parameter calibration becomes increasingly difficult with the increase in the number of parameters or OD pairs. For calibration and validation (Buisson et al., 2014) of the transport models, researchers and practitioners need MOP. The commonly used MOP is traffic flow data or link volumes. It is well known that N independent equations are needed to find the unique solution of the system of linear equations with N unknowns. In transport demand calibration, the number of unknowns (OD demand pairs) exceeds the number of equations (observed data). Classically, in the case of a linear system of equations, such a system is referred to as an "indeterminate" system, meaning that the number of equations is less than the number of unknowns. The availability of fewer equations compared to the number of unknowns leads to an under-determined system. This means, that with the available information, the system parameters cannot be uniquely determined.

Further, the higher the level of error (bias and noise) in a priori OD estimates, it will be challenging to obtain the desired solution. This is further compounded by the stochasticity, such as from the gradient approximation or optimization heuristics, vehicle routing in a simulation model. When the number of unknowns equals the number of equations, multiple solutions can still occur due to the nonlinear nature of traffic, not always captured by conventional traffic data (Frederix et al., 2013). The fact that there are multiple solutions might also make the algorithm prone to getting trapped in undesired local optima instead of converging to the desired local optima. To reduce the chance of undesired local optima, extensive analysis is needed to check the reliability and robustness of the solutions. All these practical challenges can lead to increased time complexity and computational burden. Moreover, if the calibration approach is not carefully designed, the calibrated OD parameters might be far from the desired solution.

The next set of challenges pertains to tuning the parameters of the optimization algorithm. In the case of gradient-based optimization, the learning rate decides the convergence rate. The algorithm can be very slow if the learning rate is too small. In contrast, if the learning rate is large, the algorithm can jump beyond the optimum and oscillate or land in an unsuitable local optimum (too far from the starting iterate), leading to high variance. Large learning rate values can also lead to high values in the OD matrix, leading to simulation overload, slow down, and even more time to tune the parameters of the optimization algorithm. In the literature, for instance, SPSA gain coefficients, i.e., step-size (a) and perturbation vector (c), are predominantly manually selected after some sensitivity analysis. Spall (1998a) suggested that if the parameters to be optimized vary greatly in magnitude, scaling should be applied to the gain coefficients. Such scaling was applied to step-size coefficients of SPSA by Tympakianaki et al. (2018). However, even after scaling, finding the optimal values of gain coefficients requires conducting sensitivity analysis, and this requires expensive function evaluations. Thus, it takes a considerable time to select the optimum parameters. The costly function evaluations limit the application of automatic parameter tuning methods such as Bayesian optimization

to OD demand estimation. Although Bayesian optimization works better than random sampling, the former's application will also be slowed due to time-consuming simulations. Thus, we conclude that there is no existing systematic approach for automated tuning of the calibration algorithm's parameters in the context of OD demand estimation.

The above challenges motivate us to apply enhancements to the existing demand (OD estimation) and supply calibration framework and propose an end-to-end methodology to find optimal calibrated estimates while keeping the computational burden in check. Applying the above-mentioned ensembling techniques with state-of-the-art calibration algorithms could be beneficial for OD estimation by reducing the estimates' variance. To the best of our knowledge, we could not find the application of ensembling for OD estimation/ demand calibration. Because of the above, we list the following research questions:

SRQ(7): How to use machine learning to automate the calibration of large-scale traffic simulations?

SRQ(8): How to use ensembling to obtain precise traffic simulation calibration parameters?

2.6 Indirect flow estimation to tackle sparsity and insufficiency of traffic flow data

2.6.1 Traffic forecasting

Traffic forecasting is a prevalent task in traffic research with applications in traffic management. Traffic forecasting is formulated as time series forecasting to predict the values of a target given input covariates. Forecasting models, where the input and target variables are the same yet time-lagged, are called autoregressive models since the future values of the variables are predicted using their past values. Time series forecasting can be formulated or extended as non-linear time series forecasting, multi-time step forecasting (predicting a sequence of future values instead of a single value), and multivariate forecasting (input features consist of multiple time series or scalar variables). A generalized formulation for autoregressive multivariate and multi-step forecasting is given below:

$$\left[W, (X^{t-kf}, \ldots, X^{t-f}, X^{t}), (Y^{t-kf}, \ldots Y^{t-f}, Y^{t})\right] \rightarrow \left[Y^{t+f}, Y^{t+2f}, \ldots, Y^{t+(p-1)f}, Y^{t+pf}\right] \tag{2.3}$$

Where W and X are fixed and dynamic input features, respectively. Y is the target variable. f is the fixed time interval or the granularity of the data; k is the lookback length, i.e., how much past data the model has access to make a prediction; p is the prediction or future horizon length; t denotes the current time instant.

In traffic forecasting, common features are traffic variables such as traffic flow, link speed, trip travel time, traffic density, occupancy, and congestion (Vlahogianni et al.,

2004). Traffic state is characterized by the three main variables, i.e., flow (or volume), speed, and density. Regarding targets, the main task in traffic prediction or forecasting is to predict traffic flow or speed. In this dissertation, we use traffic "flow" for the volume or number of vehicles passing a road section over time, whereas "speed" is the link speed which could be space-mean speed or time-mean speed. While researchers have used different predictors or features in their models, one common feature is using a time-lagged target variable as one predictor since most time-series forecasting models use an autoregressive model formulation. Other features are representations/ metrics derived from trajectory data, covariates (weather or time of the day), spatial-temporal maps, or videos (S. Wang et al., 2019). Commonly, studies on short-term traffic forecasting deal with forecasting horizons in the range of a few minutes to a few hours (Vlahogianni et al., 2014). The following paragraphs provide representative details on models and data used in traffic forecasting. For a more thorough discussion of traffic forecasting, we refer the reader to reviews by Vlahogianni et al. (2004, 2014).

Data and models are the main pillars supporting traffic forecasting research. Traffic data come in many forms, such as point data, point-point data, or area-wide data (Lopes et al., 2010). The form depends on the method of data collection and its source. Fixed ground-based sensors, such as induction loop detectors or street-side cameras, detect vehicles on or across a specific location. They can only provide localized observations such as flow, density, or spot speed. Traffic data from onboard devices, such as smartphones or navigation systems, primarily use GNSS receivers to record the location of vehicles during their trip and, thus, provide mobility metrics (location, speed, travel time) for trip legs. This data, when gathered from a larger fleet of vehicles, is also known as AVL data or FCD, or Probe Vehicle Data (PVD). New drone-based data collection methods can provide observations over an area or part of the network since they have a wide field of view (Barmpounakis & Geroliminis, 2020). A dataset collected by the California Transportation Agencies (CalTrans), known as CalTrans Performance Measurement System (PeMS) is one of the most popular datasets for traffic prediction. This data is point-type data [from inductive loops, side-fire radar, and magnetometers (California Department of Transportation, 2020)] containing traffic volume, occupancy, and speed. Other popular datasets are the Beijing point and trajectory datasets (Tedjopurnomo et al., 2020).

Researchers have developed a wide range of models for traffic forecasting ranging from so-called white-box models (statistical models such as simple moving average or autoregressive regression) to black-box ones (deep learning models such as feedforward or recurrent graph neural networks) (Vlahogianni et al., 2014). The current state of the art shows that deep learning has outpaced the traditional time-series forecasting models such as Autoregressive Integrated Moving Average (ARIMA), as evidenced by recent studies (T. Ma et al., 2020; Polson & Sokolov, 2017). This development has resulted in many innovative traffic forecasting model architectures using cross-domain concepts such as convolutional neural networks and recurrent networks from computer vision and Natural Language Processing (NLP). Recurrent Neural Network (RNN) are special neural networks with a chain-like structure capable of learning time dependencies. Long Short Term Memory (LSTM) networks (Hochreiter, 1991) and Gated Recurrent Units (GRU)

are specialized to learn long-term dependencies using a similar chain-like structure with modified units. LSTMs and GRUs have been successfully applied in various tasks such as language translation and image captioning. In traffic forecasting, too, LSTMs have been used for extreme event forecasting (Laptev et al., 2017) or network-wide traffic speed prediction (Z. Cui et al., 2020). Lara-Benítez et al. (2021) conducted an experimental review of multiple deep learning models for time-series forecasting, including traffic datasets. They found that LSTM and CNN are the best models, the LSTM models obtaining the most accurate results. It is relevant to point out that their analysis did not cover relatively new models such as Graph Neural Networks (GNN), transformers, or models with attention mechanisms. RNN-based architectures struggle to learn long-term dependence, and thus, attention mechanisms were applied by T. Wu et al. (2018) to address this shortcoming. The attention mechanisms can identify and select information in the input relevant to a specific task, even if it is a long-term dependency. The attention layer assigns weights to specific input sequence regions relevant to the prediction task.

GNNs have recently gained popularity due to their ability to handle non-euclidean data. GNN models can also handle topological correlations between entities in the data using node and edge features in graphs. Further, GNN has been applied on Spatio-temporal datasets, e.g., by stacking time-dependent graph snapshots leading to architectures such as Graph RNN (GRNN) (X. Wang et al., 2018), Diffusion Convolutional RNN (DC-RNN) (Li et al., 2017), Temporal Graph Convolutional Networks (T-GCN) (Zhao et al., 2020), consisting of Graph Convolutional Networks (GCN) to handle spatial features and either of the RNN, GRU or LSTM units to handle the temporal features. Buroni et al. (2021) applied a multi-task learning strategy with GCNs to predict flow and speed and tested their method on different types of roads. Despite the benefits of GNNs, Zhao et al. (2020) found that graph networks struggle to predict peaks because of averaging effects.

2.6.2 Traffic state estimation

Traffic state estimation is closely related to traffic forecasting with some notable differences. The process of inferring traffic state variables using partially observed information is known as traffic state estimation (Seo et al., 2017). In the comprehensive review by Seo et al. (2017), authors classified traffic state estimation methods into three categories: model-driven, data-driven, and streaming-data-driven, based on preliminary information and input data. When the target or the predicted variable is traffic flow, it is called traffic flow estimation. Since practitioners and researchers use different traffic data for flow estimation, it can also be classified into direct and indirect methods based on the applied data collection methodology. Direct methods refer to counting vehicles on the road using manual or automatic techniques (magnetic loop detectors, gantry cameras, or drone videography). Direct methods use physical, visual (street cameras, drone videography, or satellite imagery), acoustic (Lefebvre et al., 2017), or other signals (Bluetooth or cellular) to detect the presence of a vehicle. Indirect methods try to estimate the flow using exogenously-correlated data. As shown by Aslam et al. (2012), indirect methods use analytical and data-driven models for mapping predictor variables to the flow variables. A generalized multivariate and multi-step formulation for indirect state estimation is

given in Equation 2.4. A clear distinction from Equation 2.3 is that the time-lagged target variable is not available as a predictor in the indirect traffic state estimation. Thus, for a given domain and data, the indirect state estimation is more challenging than traffic forecasting due to less information in its predictors.

$$\left[W, (X^{t-kf}, \ldots, X^{t-f}, X^{t})\right] \rightarrow \left[Y^{t+f}, Y^{t+2f}, \ldots, Y^{t+(p-1)f}, Y^{t+pf}\right] \tag{2.4}$$

In this paragraph, we review some representative studies on dynamic and indirect flow estimation. The traffic fundamental diagram is one of the most popular and well-established models relating traffic state variables (flow, speed, and density or occupancy). Different fundamental diagrams, such as Greenshield's fundamental diagram (Greenshields et al., 1935), correlate traffic flow with speed (Kühne, 2008). Therefore, readily available speed data (together with other covariates) can be used to predict traffic flows if the fundamental relationship is manifested. One drawback of the fundamental diagram is that it lacks a time-dependent representation of traffic state variables and, thus, requires suitable model extensions for a dynamic representation. To handle this, Neumann et al. (2013) used Bayesian networks to model the time dependencies and predict traffic flows (from six hundred detectors) from speed data for the city of Berlin. Kumarage et al. (2018) used K-nearest neighbor regression with spatial-temporal attributes to predict flow at fewer locations. Pun et al. (2019) used topological and geometric features for traffic flow estimation. Gkountouna et al. (2020) used data from thirty-six sensor locations for developing bi-level flow estimation models. The novelty in their method was the use of principal component analysis and clustering to identify road segment archetypes in the first level. This information is used in the second-level regression model. Rinaldi and Viti (2020) used a Kalman filtering framework for flow estimation. Zhang et al. (2020) used a geometric matrix completion model for network-wide traffic flow estimation, using real-world (twenty-four road segments) and synthetic datasets. Recently, Abdelraouf et al. (2022) used speed and volume features from PVD as an input to recurrent GCN to predict the traffic state parameters. In contrast to other indirect estimation works, using the flow from PVD as an additional feature provides direct information for accurate prediction but also makes the model dependent on such data. Among the above, only a few studies (Abdelraouf et al., 2022; Neumann et al., 2013) consider data from more than a hundred detectors, whereas the other studies use relatively minor datasets. Further, Neumann et al. (2013) note that their model performance was not accurate enough for freeways or roads with higher speed limits. Traffic flow estimation is complicated by static or dynamic changes in link characteristics such as speed limit, the number of lanes, data collection, data quality (presence of noise and anomalies), and data processing (smoothing, aggregation). Further, the methodological steps are pivotal when processing raw FCD to obtain link speed data (Zhu et al., 2009). These factors can distort and induce scatter in the fundamental diagram, thus rendering indirect traffic flow estimation quite challenging.

2.6.3 Transfer learning

Machine learning models, also in the case of traffic forecasting and traffic state estimation models, are developed assuming that the training data and test data "must be in the same feature space and have the same distribution" (S. J. Pan & Yang, 2010). However, this does not generally hold true when applying models to the new study area. The data distributions from the two study areas can differ even if the same set of features is developed for the data from two locations. This will have an impact on the model's performance. This challenge is tackled using transfer learning, which is improving the learning of a new task (target task) using the knowledge from an already learned related task (source task) (Torrey & Shavlik, 2010).

Transfer learning is defined formally in terms of domain and task in the survey paper by S. J. Pan and Yang (2010). A domain $\mathcal{D}$ consists of a feature space $\mathcal{X}$ ($X = \{x_1, \ldots, x_n\} \in \mathcal{X}$) and a marginal probability distribution $P(X)$ (S. J. Pan & Yang, 2010). A task consists of a label space $\mathcal{Y}$ and an objective predictive function $f(\cdot)$ (denoted by $\mathcal{T} = \{\mathcal{Y}, f(\cdot)\}$) (S. J. Pan & Yang, 2010). The objective predictive function is not observed but can be learned from the training data. $f(x)$ can be written as $P(y \mid x)$. After the model is learned on the source domain ($\mathcal{D}_S$) for a source task ($\mathcal{T}_S$), the aim is to transfer the learned knowledge for learning the target predictive function $f_T(\cdot)$ for solving the target task ($\mathcal{T}_T$) in the target domain ($\mathcal{D}_T$) (S. J. Pan & Yang, 2010), where $\mathcal{D}_S \neq \mathcal{D}_T$, or $\mathcal{T}_S \neq \mathcal{T}_T$. For instance, two cities, even with the same road links, could display different levels and patterns of traffic flow, depending on local traffic conditions.

S. J. Pan and Yang (2010) noted four transfer learning algorithms based on how the knowledge is transferred from the source task to the target task. These four types are instance-based, feature-based, model-based, and relation-based algorithms. In deep learning, using pre-trained models for secondary tasks is essentially the same as model-based transfer learning (S. J. Pan & Yang, 2010). In model-based transfer learning, it is assumed that the model's parameters learned from the source domain will be helpful for the target task in the target domain. These parameters and hyperparameters are fine-tuned using the limited training data from the target domain.

For short-term travel-time prediction, Luan et al. (2018) showed that link-to-link transfer of their model is possible but emphasized further research into factors that affect transferability. Li et al. (2021) and Mallick et al. (2021) used transfer learning techniques in short-term traffic prediction using source and target links consisting of links from different locations. Li et al. (2021) found that transfer learning can provide more accurate predictions when the source and target links have consistent data patterns. When abundant labeled data is available, training a model from scratch without any pre-trained model makes sense. Mallick et al. (2021) proposed the Transfer Learning-DCRNN model using a graph-partitioning-based transfer learning approach for short-term traffic forecasting, which outperformed other models. Using source knowledge through transfer learning can help to reduce dependence on large datasets and improve the existing models. Investigating when or at what levels of data transfer learning outperforms the new models

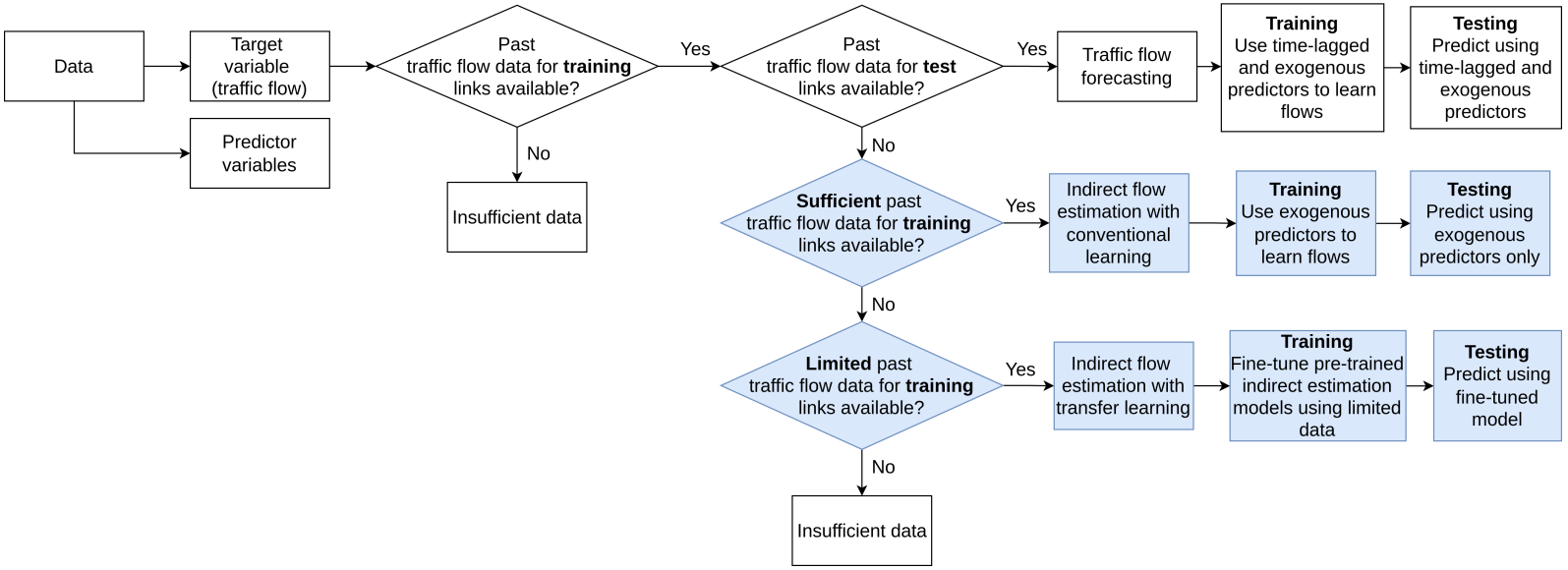


Figure 2.5: Flow chart showing scenarios for traffic flow forecasting, indirect flow estimation, and transfer learning, depending on data availability.

trained from scratch is a promising research direction. Transfer learning for indirect traffic flow estimation is still not explored.

2.6.4 Research gaps

One of the main features of traffic flow forecasting methods is that time-lagged flow or speed data are used as an input in autoregressive formulations (T. Ma et al., 2020; Polson & Sokolov, 2017). For instance, past/ time-lagged values of a signal (flow or speed) or variable are used to predict a future variable (speed or flow, respectively). However, in some cases, such flow data is wholly or partially unavailable, i.e., spatially sparse or comes with large temporal lag, thus, posing challenges for deploying traffic prediction models and their applications in dynamic or real-time settings. Moreover, using benchmark datasets, such as the Caltrans PeMS (California Department of Transportation, 2020), does not portray the data availability challenges, varying from region to region. Hence, it could inhibit the practical deployment of traffic forecasting models.

Another issue worth highlighting is the ease of collection and data availability. Speed and flow (volume) data tell us different aspects of the traffic state. Flow tells us about the load on the link or the number of vehicles passing through a specific road and has applications in highway and pavement design, highway-side advertising, and commercial or real-estate investments. This work uses the term "link" to imply road segments. Speed depicts the link's congestion, travel time, or time delay. Speed does not directly represent the number of vehicles on the road. Instead, speed data is used to provide travel time estimates or the level of service on the road. Traffic flow data are more challenging to collect than link speed data since the latter can be approximated from a sample of vehicles (Aslam et al., 2012; Neumann et al., 2013). Speed data are derived by aggregating the traces from a fleet of cars, also known as PVD or FCD, or mobile phones via the GNSS receivers. This is one of the reasons network-wide traffic speed data are more prevalent for more cities than traffic flow data. Many companies (TomTom, 2021; Uber Movement, 2020a) collect data from vehicle fleets or smartphones, and some make it publicly available to a certain extent (Mahajan et al., 2022). For instance, Uber, a global ride-hailing company, shared such data during 2016-2020 for many cities worldwide under non-commercial use licenses (Uber Movement, 2020a). The data is available for download in bulk. In addition, navigation companies such as TomTom provide link speed data with limited free API calls followed by paid usage.

On the other hand, collecting traffic flow or count data requires dedicated hardware or collection methods which incur time and cost burdens. Specifically, traffic flow data is primarily collected via magnetic loop detectors installed on the streets. Such data collection infrastructure comes with high installation costs, is not scalable, and is common in many cities. As a result, traffic flow data are not available for many cities worldwide and are scarce even for cities in developed countries. Due to the collection and coverage asymmetry between traffic flow data and traffic speed data, there arises an opportunity to use the speed data to infer traffic flows. However, the possible solutions exclude using autoregressive models since time-lagged flow is assumed to be unknown and needs to be indirectly estimated. Therefore, this is a problem of indirect flow estimation. Here, we

see the potential to use publicly available data to derive flow data from samples to larger parts of networks.

With the help of the flow chart (Figure 2.5), we identify the research gaps when there is no historical/ real-time flow data for the links we want to make predictions. We divide the links into two sets: training links and test links. If traffic flow data is available for both sets, then traffic forecasting can be used, where time-lagged traffic flow is used as one of the predictors. Suppose past data for test links are unavailable. In that case, the indirect traffic flow estimation approach using conventional learning is applicable, where only exogenous predictors are used to learn the traffic flow mapping so that during the model testing, we do not need time-lagged flow values as predictors. In the previous case, if sufficient flow data for training links are unavailable, flow estimation is done using transfer learning. Here, the pre-trained model is fine-tuned using limited data. Finally, if insufficient flow data for training are unavailable, which means we only have predictors but insufficient labels, then it is a case of insufficient data.

From the above discussion, few research gaps are evident. Firstly, unlike traffic forecasting, deep learning models are not prevalent in dynamic indirect traffic state estimation. This is true even though both approaches have inherent similarities among input and target features. Apart from prediction, quantifying the uncertainty in flow estimation is also vital. A model which outputs a range of predictions can help us to judge its preciseness. Lastly, the transferability of the indirect state estimation models using real data is still unexplored. Because of the above, we see an opportunity to address the following SRQs:

SRQ(9): How to use transfer learning to address the sparsity of dynamic link flows at the network level?

SRQ(10): What are the conditions for the successful transfer of pre-trained models?

The former research question is motivated to address data scarcity within a study area or city. The latter research question addresses data scarcity across cities. It helps to scale the flow of information from a few links to the whole network, provided the characteristics of the test data (link characteristics, spatial-temporal conditions) are similar to the training data.

2.7 Research Objectives

Based on the above review of the existing studies and identification of research gaps, the ROs following from the above SRQs are listed below:

RO(1): Develop a systematic typology to scope the data landscape and classify the non-conventional data according to openness. [SRQ(1), SRQ(2)]

RO(2): Investigate the applications of prominent non-conventional data sources in transport modeling research. [SRQ(3), SRQ(4)]

RO(3): Develop and evaluate a scalable method for improving the usability and quality of publicly available data from emerging sources. [SRQ(5)]

RO(4): Identify and apply publicly available opportunistic data for novel use cases and demonstrate their application. [SRQ(6)]

RO(5): Develop and evaluate data-efficient methods to tackle challenges due to insufficient conventional data for traffic prediction and calibration. [SRQ(7), SRQ(8), SRQ(9), SRQ(10)]

3 Data openness and scoping for transport analysis and modeling

Contents

The content of this chapter has been presented in the following work:

Mahajan, V., Kuehnel, N., Intzevidou, A., Cantelmo, G., Moeckel, R., & Antoniou, C. (2022). Data to the people: a review of public and proprietary data for transport models. Transport Reviews, 42(4), 415–440. doi:10.1080/01441647.2021.1977414

3.1 Introduction

In this chapter, we revisit the emerging transport data and classify them according to their public availability or openness. The chapter is structured as follows: The following section lists the chapter's contributions, followed by a section introducing the study's methodology. The following section presents a data classification typology based on openness attributes. This typology is applied to classify these data into appropriate categories. The next section reviews data applications in transport modeling, followed by a section presenting a Strengths, Weaknesses, Opportunities, and Threats (SWOT) analysis to get an overview of trends by focusing on application and availability aspects together.

3.2 Research contributions

The contributions of this chapter are listed as under, with the relevant RQs in parentheses:

- We compile the relevant attributes for data openness from the literature and define a classification typology based on the data's public availability. This typology is applied to classify the prominent non-conventional data in transport modeling. [SRQ(1),SRQ(2)]
- We also review the modeling and analysis applications of non-conventional data from mobile phones, social media, Global Navigation Satellite System (GNSS), Bluetooth, smart cards, Volunteered Geographic Information (VGI), and standardized datasets such as General Transit Feed Specification (GTFS). We analyze the benefits and future challenges viz-à-viz public availability. [SRQ(3), SRQ(4)]
- An extended version of the codes, based on the original implementation by Narayanan and Antoniou (2022), to obtain data on scientific articles are shared on GitHub[1].

3.3 Methodology

We collected articles from the scientific database (SCOPUS[2]) by specifying a combination of keywords representing domain area and focus area (Table 3.1). The search query is formulated as DOMAIN WORD + AND + FOCUS WORD for articles in the English language between 1990-2020.

The domain keywords are somewhat generic (and not just specific to transport modeling) because we feel that transport modeling is a vast field and keywords should be more inclusive. We process the fetched articles by removing duplicates. The journal keywords (Table 3.1) are used to filter the items by checking if the respective journal's title contains

[1] `https://github.com/vishalmhjn/scopus_caller`

[2] The data was downloaded using the Scopus API between January 1 and 31, 2020 via `http://api.elsevier.com` and `http://www.scopus.com`

Table 3.1: Keywords for collecting scientific articles from SCOPUS

Domain words	Focus words	Journal words
transport, travel, transportation, transit, mobility, traffic, trip	open data, public data, big data, social media, crowd sensing, location-based social media, Twitter, Foursquare, Google, opportunistic data, passively collected data, cell phone, smartphone, mobile phone, smart card, automatic fare collection, AFC, Bluetooth, automatic vehicle location, AVL, floating car, global positioning system, GPS, traffic count, taxi, bike sharing, ride sharing, car sharing, open street maps, OSM, volunteered geographic information, VGI	traffic, transportation, intelligent transport systems, ITS, transport, urban planning, spatial, transit, mobility, urban, cities, land, land-use, sensors, location, sustainability, geography, railway, smart, civil

one of these words. Since the journal keywords are relevant to the transportation field and help reduce the number of articles. Even then, we expected many papers due to the large number of domain and focus keyword combinations (e.g., transport AND open data). Specifically, approx. Fourteen thousand articles were found to satisfy the keywords and filtering criteria. Finally, we filter the articles for the prominent data/ sources, namely, mobile phones, social media, Automatic Vehicle Location (AVL), Bluetooth, GTFS, General Bikeshare Feed Specification (GBFS), Global Positioning System (GPS), smart card and Volunteered Geographic Information (VGI) only, referred to as data of interest. This was done by checking if the paper's title includes the name of such data. Even though this selection might miss some data, we feel that we can cover most emerging data/ sources and thus ensure the representativeness of the data application review. Finally, there were a few "wild card" articles that the author(s) came across during analysis (through the bibliography of reviewed articles, social media, and previous experience) and were included in the review, primarily due to their significance and if they added additional information to the data-application context. Finally, this exercise resulted in 315 articles. For brevity, we usually only include one or unique scientific reference to an application-data pair since the purpose was to provide evidence that certain data have been used for a specific use case.

3.4 Openness typology

The Open Knowledge Foundation[3] defines open data as "any content information or data that people are free to use, re-use and redistribute without any legal, technological or social restriction." They mention the key openness features as availability, access,

[3] `http://opendefinition.org/`

reuse, redistribution, and universal participation. We use the above definition and the concentric shell model by Backx (2003), further used by van Loenen and Grothe (2014), to compile the most important attributes and check if the data are known, legally attainable, accessible, affordable, usable, and distributable. These attributes are defined below:

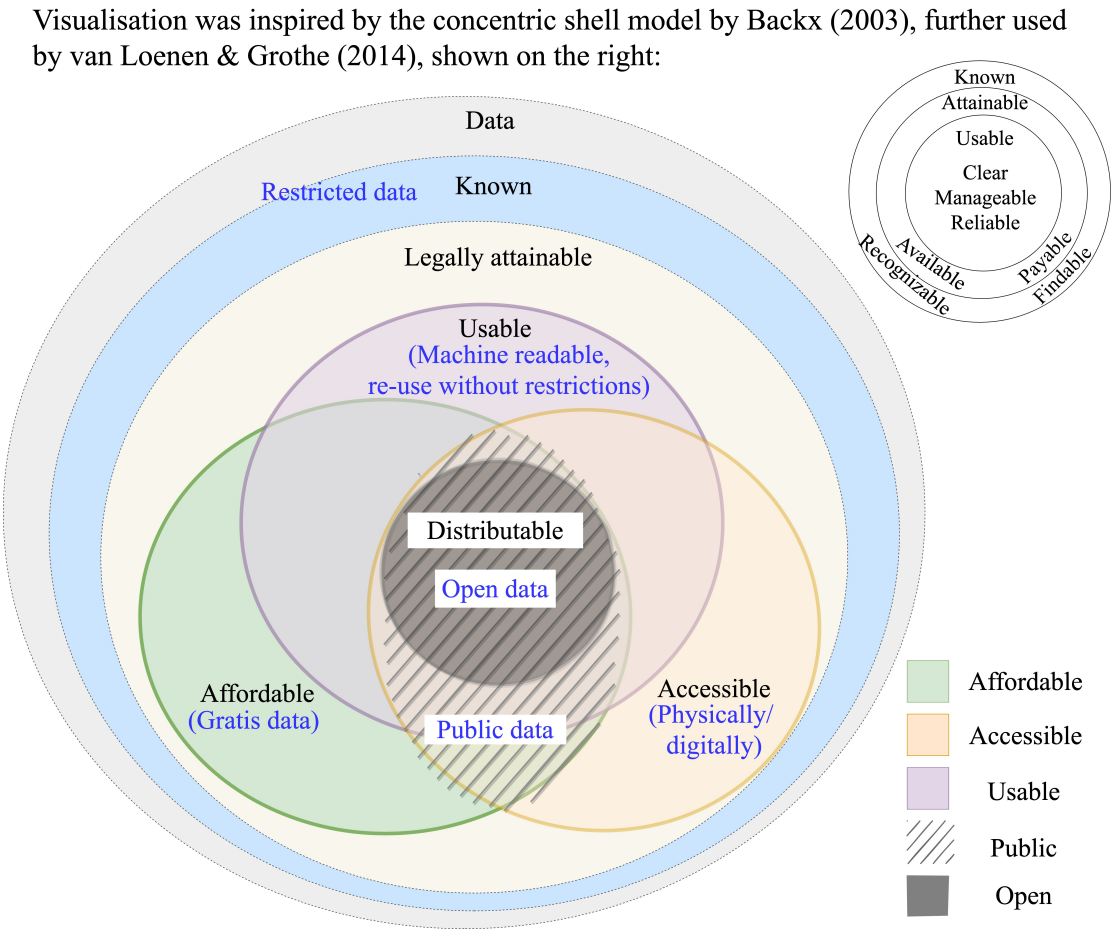


Figure 3.1: Public availability/ openness attributes (Extending the concentric shell model by Backx (2003), English translation by van Loenen and Grothe (2014), in the top right corner).

- Known: Data are findable (van Loenen & Grothe, 2014), or at least their existence can be confirmed with the help of common tools, such as Web search engines, catalogs, or Freedom of Information requests. Highly restricted data (Government or commercial secrets), undocumented data, or unfindable data are unknown and thus are totally out of public reach.

- Legally attainable: When the data are not restricted by way of statutory enactments[4], they can be classified as legally attainable. The governing legislation aims to mitigate such risks if the data contain sensitive information, such as personal data, defense, and trade secrets. Unless the related risks are mitigated, these kinds of data (at least in raw form) cannot be legally obtained and are beyond the public domain. Further, it is pointed out that physical/ digital access to legally attainable data is not always guaranteed. The data owner could refuse to share the data due to bureaucratic/ enforcement laxity, fear of criticism, competition, etc.

- Accessible: We use accessibility to refer to the physical aspect of attainability, according to van Loenen and Grothe (2014). We include both the physical mode (via post) for records in soft/ hard format and digital modes [Application Programming Interface (API), bulk download facilities] of access for cloud or local computer databases. Universally accessible data implies that the data are publicly accessible, irrespective of the cost and usage restrictions, e.g., an API that is publicly accessible, which may/ may not be priced.

- Affordable: This is akin to financial attainability and part of the second shell of Backx's model (van Loenen & Grothe, 2014). Data that are available free[5] of charge (i.e., gratis) are universally affordable. The data provider bears the cost[6] from other revenue sources, such as the organization's general annual budget in case of open government data (Welle Donker & van Loenen, 2016). Despite the ongoing emphasis on open data, the commercialization of proprietary data is growing [OECD, 2015 as cited in OECD (2019)]. However, if the user costs of the data remain small, it can also be affordable at large. This concept is similar to public transport pricing, which is commonly not free but below operating costs to improve equity. Commercial datasets are considered unaffordable in this research.

- Usable: Usability is a multi-faceted character that could refer to the ease of use, quality of the data, and end-use restrictions. Ease of use increases with machine readability and their compatibility with open-source tools (Braunschweig et al., 2012). Structured datasets offer high usability, whereas processing unstructured data (like textual data, PDFs, and scanned documents) is more cumbersome. Data quality attributes such as data context (in terms of meta-data), completeness, timeliness, and consistency affect the data usability depending on the use case. Re-use of data implies data usage by someone other than the original user for a different purpose (Pasquetto et al., 2017). Specific licenses such as CC-BY-NC restrict the application of the datasets only for non-commercial purposes.

[4] Examples of such Statutory rules are the General Data Protection Regulation in the EU, the State Secrecy Law in Japan, the Defense Secrets Act in the USA or the Trade Secrets Act in Germany.

[5] In this paper, "free" implies gratis or free-of-charge datasets, where users don't need to pay any fees for using the data. Another interpretation of "free" is "free as in the freedom of speech" or libre, which gives the user freedom to modify, adapt, and even distribute the data (Suber, 2008).

[6] Data costs can correspond to different stages, such as production, curation, analytics, publication, marketing, etc. Thus, the data owner or provider can decide to cover these costs in part or full, from diverse revenue streams including a budget, licensing, etc.

- Distributable: This refers to the right to re-publish or share the data in an original or modified version with a third party, without any or minor restrictions[7]. This implies that the data come with a suitable license that allows redistribution. The extent of distribution freedom depends on the specific licenses, e.g., distribution in the adapted or original format. Examples of open data conformant licenses are Creative Commons (CC0, CC-BY-4.0, CC-BY-SA-4.0), Open Data Commons, and Open Database License ODbL1.0. A review of licensing frameworks is given by Mockus and Palmirani (2015).

The above typology is summarised in Figure 3.1. Open data should satisfy all the above parameters, whereas public or publicly available data are not always usable or redistributable.

3.5 Data classification

We propose a classification scheme (Table 3.2) based on whether the legally attainable data discussed in the previous section (Figure 3.1) are universally affordable, accessible, usable, and distributable. Legally attainable data could be either Public Data (PD) or Non-Public Data (NPD) and are classified into four main categories: (i) Commercial/ Proprietary data (NPD-1), (ii) Inaccessible data (NPD-2), (iii) Gratis and accessible data with restricted use (PD-1), and (iv) Open data (PD-2). As the commercial or inaccessible data (NPD-1 and NPD-2) are not within public reach, they are non-public data. On the other hand, gratis and accessible data (PD-1 and PD-2) are referred to as public data.

- Commercial data (NPD-1) are priced data mainly from private companies, such as mobile phone data, social media data (e.g., premium API from Twitter[8] and Foursquare[9]), personal car AVL data. In some cases, Government data may also be priced, e.g., premium GTFS[10] data. There might be exceptions where the priced data are shared for free, particularly with researchers or policymakers. Still, the data are not affordable at large, i.e., universally. Data intermediaries also play a crucial role by sourcing data from multiple providers and providing processed derived or inferred information as a premium service (OECD, 2019).

- Inaccessible data (NPD-2) includes data owned by transport operators, such as smart card data or detector-based AVL data. Some transport operators are willing to share these data (on specific requests/ academic research). Still, they cannot be assumed to be generally accessible as long as these data are not within reach of the public. When such data providers share these data, they are commonly uploaded as open data on their website or open data portal (PD-2).

[7] As per the `www.opendefinition.org`, requirements of attribution and share-alike conform with the Open data definition, and thus do not count towards restricting usage or distribution of the data.
[8] `https://developer.twitter.com/en/pricing`
[9] `https://developer.foursquare.com/places`
[10] `https://gtfs.de/en/services/`

Table 3.2: Data Classification

Data Type	Data Provider	Openness attribute					Data Category	Examples
		Legally attainable	Accessible	Affordable	Usable	Distributable		
Mobile Phone Network Data (MPND)	Telecom operator/ Data intermediaries (SaaS[a])	✓	✓	-	✓[b]	-	NPD-1	OD matrices derived by data intermediaries are offered as a premium service
Social Media	Social media platforms	✓	✓	✓[c]	✓[b]	✓[b]	NPD-1/PD-1	Premium access/ Free access
Smart card	Transit operator	✓	-	✓[c]	✓[b,d]	-	NPD-2	Shared selectively for research purposes only.
Bluetooth	Traffic operators	✓	-/ ✓	✓	✓[b,d]	✓[b]	NPD-2/ PD	Aggregated information such as flow, travel time are shared
GNSS -derived AVL	Navigation service providers, OEMs, Commercial fleets	✓	✓	-	✓[b]	-	NPD-1	Vehicle level information is seldom shared publicly
		✓	✓	✓[c]	✓[d]	✓[b]	PD-1	Aggregated traffic data as premium or free service
GTFS	Transit operator	✓	✓	✓[c]	✓[d]	✓[b]	PD-2	Stop locations and schedules, sometimes real-time data too
GBFS	Shared mobility provider	✓	✓	✓	✓[d]	✓	PD-2	Bike-share data
VGI	Crowdsourced	✓	✓	✓	✓[d]	✓	PD-2	OpenStreetMap

[a] Software as a Service [b] Depends on the terms and conditions/ license of data (re-) use and sharing.
[c] Could be offered as a free or a premium product/ service. [d] Data may or may not be usable depending on the data format and end-user requirements.

- Gratis and accessible data with restricted use (PD-1): Examples include free-of-charge (gratis) data from private companies that come with specific licenses, such as Creative Commons Non-Commercial (CC-BY NC). Examples are UBER Movement data and social media data (gratis) such as Twitter API. In many cases, such as Google Directions API, aggregate information derived from personal data is shared in the public domain to mitigate privacy risks and maintain an advantage among competitors.

- Open data (PD-2): This segment is subdivided into 3 data ownership types, namely the private sector, public sector, or community, depending on who is responsible for the data collection and provision (Figure 2.1).
  - Open Government Data (OGD) refers to the open data produced and collected by public organizations. Mobility or transport datasets are listed under a separate category on most Open Data Portals (ODPs). Open government data are thematically rich and cover a wide range of technical and non-technical areas (Charalabidis et al., 2016). The EU's ITS Directive[11] aimed for optimal use of the road, traffic, and travel data. Mobility is one of the six themes targeted for high-quality datasets in the EU's Open data directive 2019 (European Commission, 2020). However, despite the OGD's progress in recent years, much data shared by the government lacks usability and clear guidelines/ licenses for the distribution of data (Mockus & Palmirani, 2015).
  - Open Private Data (OPD) are still at an early stage. Private companies have varied terms of conditions regarding data release and usage. Some companies value data sharing (Welle Donker & van Loenen, 2016). For example, many bike-share companies share real-time bike feeds using the open data standard GBFS.
  - Open Community Data (OCD) refers to crowdsourced open datasets/ databases neither owned by the government nor the private sector, such as OpenStreetMap. Research data, such as complete transport models, have been made openly available by researchers (Ziemke et al., 2019). While the input data do not necessarily have to be open, the post-processed scenario data can be used by other users.

Not every dataset might fit perfectly into one category. Social media data, for example, could be priced or gratis. Detector count data tend to be inaccessible (NPD-2), but in some cases (e.g., the city of Paris), these data are open (PD-2). The above classification typology is a fair attempt to segment data logically.

3.6 Review of data applications

Figure 3.2 shows the transport modeling applications of different data types and their availability category, discussed below in more detail.

[11] https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX%3A32010L0040

3.6.1 Mobile Phone Network Data (MPND)

Mobile Phone Network Data (MPND) can be event-driven or network-driven. Event-driven MPND are generated when a mobile user actively interacts with the device, such as making/ receiving a call or SMS (H. Huang et al., 2019). On the other hand, network-driven mobile phone data are generated even passively and thus are much denser compared to event-driven data (H. Huang et al., 2019).

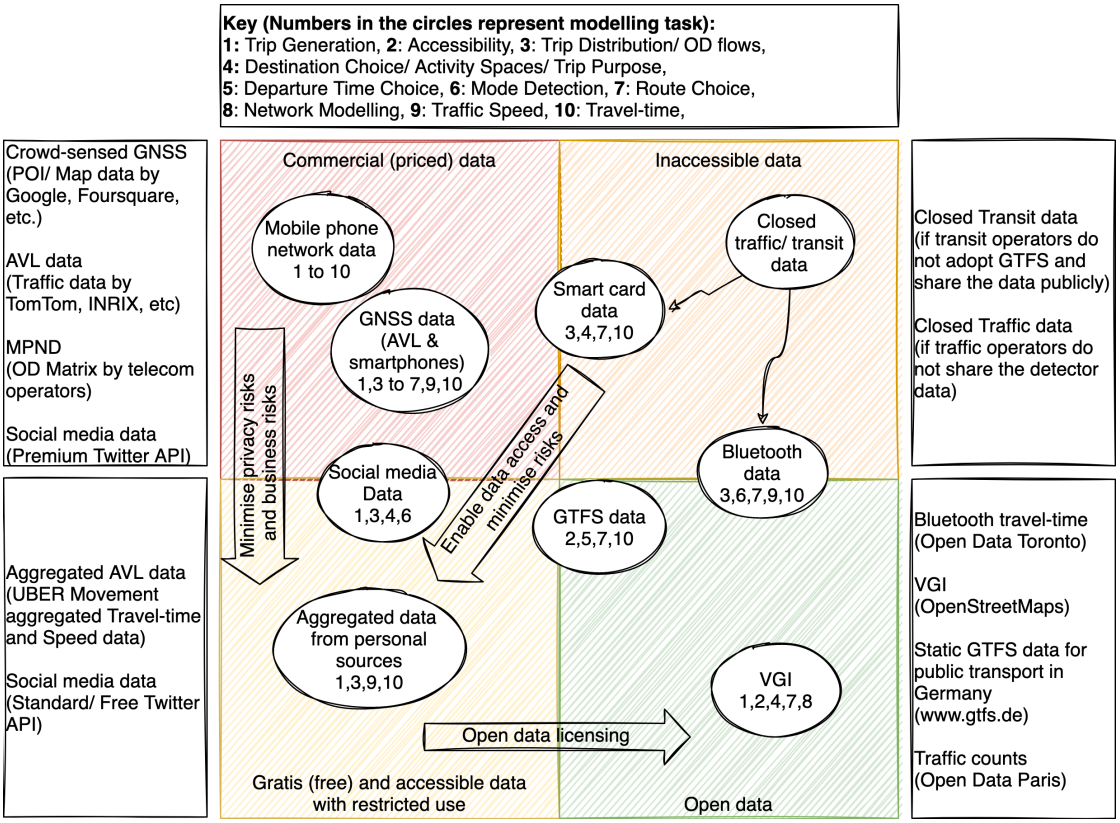


Figure 3.2: Public availability and applications of the prominent datasets used in transport modeling.

Event-driven data such as Call Detail Record (CDR) contain caller ID, timestamp, latitude, longitude, duration of the call or other activity, and receiver's ID (Rojas et al., 2016). Earlier studies showed the feasibility of MPND for Origin-Destination (OD) estimation using a mobile network simulator (Caceres et al., 2007). These data are a convenient alternative to conventional methods (roadside interviews and household travel surveys) for estimating OD matrices (Bonnel et al., 2018; Tolouei et al., 2017). Accurate user trajectories can be obtained from network-driven data and applied for route choice modeling (Schlaich, 2010). Travel mode can be detected from mobile phone data using rule-based or machine learning models based on travel-time/ speed distribution (Rojas et al., 2016; H. Wang et al., 2010). If MPND are collected over a longer duration, they

can be a source for activity location analysis (Järv et al., 2014). Mobile phone data were also used for accessibility modeling (Guo et al., 2019) and land-use detection (Furno et al., 2017). A few studies have combined MPND with other datasets, such as GNSS data for departure time choice (Bwambale et al., 2019), and Household Travel Surveys for activity location analysis (C. Chen et al., 2014). Despite their large sample sizes, MPND might not cover specific sections of the population that use mobile phones less frequently, such as children and older people (Tolouei et al., 2017). Furthermore, MPND are owned by telecommunication companies and are not publicly available due to privacy and commercial reasons. The sociodemographic attributes are generally not available in MPND. For many modeling tasks with MPND, more traditional observed data (e.g., traffic counts, travel surveys) are still required for validation and scaling the models to the full population. Lastly, inferring trajectories from MPND, such as CDRs, is challenging due to discontinuity issues and data noise.

3.6.2 Smart card data

Smart card data are suitable for OD matrix estimation, as the journey's start and/ or end is recorded when a passenger enters or exits a public transport station (Barry et al., 2002). The large volume of OD pairs is also useful for route choice modeling using other attributes such as waiting time, in-vehicle travel time, headway, or the potential number of transfers, which can be directly or indirectly inferred from smart card data (Jánošíkova et al., 2014). The network-wide scale of smart card data is advantageous for calibrating and validating the public transport assignment models (Tavassoli et al., 2017). Smart card data could be useful for destination choice estimation, at least to identify the alighting station (Trépanier et al., 2007). As smart card data lack information on journey purpose, researchers need travel survey data and other geographical data to infer the trip purpose (Bagchi & White, 2005). Although smart card data have clear benefits, they are not a panacea for public transport modeling. Smart card data are generally not universally accessible since public transport companies might be restricted or unwilling to share the data due to privacy, commercial, or other reasons. Even if the data are accessible, the data might not be fully representative of public transport behavior, since some public transport users do not own or regularly use a smart card. Inferring the OD trips is difficult if the smart card is not used at the alighting stop. Sociodemographic attributes are absent in the smart card data. Smart card data are generally used with GTFS data to associate mobility patterns with the public transport network and schedules. Thus, smart card data depend on GTFS data to realize their full potential in public transport planning and operation studies.

3.6.3 Global Navigation Satellite System (GNSS) Data

Global Navigation Satellite Systems, such as GPS, GLONASS, BeiDou, and Galileo, have been explored to gather data from personal devices/ vehicles that complete or replace household survey data. Research has shown that using GNSS data to confirm user diaries leads to more accurate trip information, overcoming user biases and miscalculations

(Kelly et al., 2013). These data can then give a clearer insight into the travelers' behavior (Grengs et al., 2008) and help decode the user choices regarding travel frequency (Stopher et al., 2007), travel mode (Feng & Timmermans, 2013) and trip routes (Papinski et al., 2009), as well as infer their trip purpose and estimate non-vehicle travel (Wolf et al., 2003). Similar applications exist for taxis regarding trip patterns and congestion (Tang et al., 2018) and selecting the optimal commercial vehicle fleet size (Yang et al., 2019). Other taxi applications that use GNSS include analyzing route choice (Duan & Wei, 2014) and land use classification (G. Pan et al., 2013). In addition, the spatiotemporal context in the GPS data offers valuable information on transport network performance (Sandim et al., 2016). GNSS data were also used to understand bicyclists' route choices, considering the surrounding environment and infrastructure (Broach et al., 2012). A limitation of the GNSS data is the inaccuracy due to delays in signal acquisition (cold starts), data loss, and errors stemming from obstacles, such as high-rise buildings. Besides, GNSS data can be biased when it mostly stems from specific vehicle fleets (e.g. taxis, staff cars), leading to results that could be misinterpreted when making inferences about general traffic conditions and behavior.

3.6.4 Bluetooth data

The most popular application of Bluetooth data is travel time estimation. Bhaskar and Chung (2013) have reviewed the technical aspects of the Bluetooth data collection. Bluetooth data are a proxy for license plate recognition match for travel time estimation (Hainen et al., 2011) because Bluetooth scanners can identify the vehicles based on the device's Media Access Control (MAC) address. Vehicle detection at multiple routes in the network can help travel time estimation and trajectory extraction (Bhaskar et al., 2015) and construct the Bluetooth origin-destination matrices (Barceló et al., 2010). Data from Bluetooth detectors has been applied for trip behavior classification (Crawford et al., 2018), route choice modeling (Hainen et al., 2011), and mode detection (Bathaee et al., 2018). Bluetooth data are also used for modeling active modes of transport, i.e., bike travel time and walking (Malinovskiy et al., 2012; Ryeng et al., 2016). Some case studies have confirmed that travel time data from Bluetooth or WIFI sensors are very similar to actual data (Ryeng et al., 2016). To collect Bluetooth data, scanning hardware needs to be installed at different places in the network, which may be cost-intensive and requires permissions from authorities and safeguarding privacy concerns. The trade-off between location ambiguity and the Bluetooth antenna's penetration rate (coverage) should be considered when collecting and processing Bluetooth data (Araghi et al., 2015).

3.6.5 Social media data

Various social media data have been used to extract variables for travel behavior analysis: trip purpose, destination choice, mode detection, and activity duration (Rashidi et al., 2017). Social media data can provide insights into travel behavior at a disaggregated level (at the level of an individual unit such as a user or Point of Interest (POI)) in real-time. Twitter data are a potential candidate for estimating the trip purpose or

activities (Chaniotakis et al., 2017). Combined with the point-of-interest data, they can be used to forecast the next activity besides the current activity (Y. Cui et al., 2018). Twitter and other social media data have been used to study different aspects of longitudinal travel behavior, such as destination choice (Y. Chen et al., 2018; Llorca et al., 2018; Zhang et al., 2017) and mode choice (Maghrebi et al., 2016). When combined with census and land-use data, Twitter data can help estimate OD demand matrices with adequate accuracy (Osorio-Arjona & García-Palomares, 2019). Geotagged Twitter, Flickr, and Weibo data can provide contextual information for predicting passenger flows (Ni et al., 2017) or a proxy for recreational/ leisure travel (Hamstead et al., 2018). Social media data were successfully used to describe mobility patterns, miscellaneous spatial-temporal analysis, sentiment analysis, traffic information extraction, and incident detection, among others, at the aggregate or disaggregate level. A significant proportion of Social media data, such as from Twitter, is not geotagged (Chaniotakis & Antoniou, 2015), which limits their application or requires extended data collection periods. Social media data suffers from representativeness issues, e.g., Twitter data is biased towards high-income groups and leisure activities (Chaniotakis et al., 2016; X. Wu et al., 2017). Textual data from social media applications is unstructured, noisy, ambiguous, short, and needs significant pre-processing (Grant-Muller et al., 2015). Social media data lacks guaranteed long-term availability and suffers from reliability and usability issues due to its private ownership and evolving privacy issues. If social media companies decide not to share any data, the impact on transport modeling research could be substantial, e.g., TripAdvisor prohibits using their data for any data analysis and academic research[12]. Free social media data usually come with restrictions, such as API call limits or the non-availability of historical data. These issues could cause reluctance among cities or policymakers to shift to social media data for transport modeling.

3.6.6 Volunteered geographic information

VGI have been used to estimate and map populations and jobs in a given area. Travel demand models usually require representing the actual population, including home and job locations in the study area. Traditionally, census data are used to represent the population. Bast et al. (2015) developed an approach to estimate population numbers solely based on OpenStreetMaps (OSM) data at an individual building resolution. Bakillah et al. (2014) presented a framework that disaggregates aggregated population data down to individual buildings using buildings and point-of-interest from OSM. Bienzeisler et al. (2020) used a data fusion approach to estimate job locations based on company data and building data from OSM. A similar use case to estimate traffic volumes and disruptions instead of the population was described by Camargo et al. (2020). Another use case for VGI is the classification of land use, which can be used to allocate jobs and households. Arsanjani et al. (2013) used OSM data to classify land use for the city of Vienna. On the supply side, transport models work with an abstract representation of the transport infrastructure using network graphs. VGI providers such as OSM were initially designed

[12] https://developer-tripadvisor.com/content-api/request-api-access/

to map roads and allow navigation with accurate road and public transport networks. OSM has become a standard data source for networks in transport simulations, such as SUMO or MATSim (Ziemke et al., 2019). Other transport-related applications of VGI include accessibility calculations based on network and point-of-interest data (Lantseva & Ivanov, 2016), traffic light information extraction (Rieck et al., 2015), environmental exposure analysis (Kuehnel et al., 2019), and bike ridership analysis (Duran-Rodas et al., 2019). VGI can be used in many applications and is available in most parts of the world. However, the lack of strict quality control and sometimes lax mapping or representation standards can lead to inconsistent data (Senaratne et al., 2017). Also, the level of detail and completeness differs by area and largely depends on the active community. Therefore, the quality may vary substantially in different parts of the world.

3.6.7 Standardised transport data

Public standards have been defined to simplify data exchange for some commonly used data in the transportation field. A well-known de facto standard is the GTFS, which represents the public transport supply and can be used to calculate public transport travel times. GTFS has become a frequently used standard to model public transport supply (Bienzeisler et al., 2020; Ziemke et al., 2019). GTFS data were also used to study public transport accessibilities (Owen & Levinson, 2017). Unfortunately, GTFS data are not available everywhere, mostly focusing on developed countries. While GTFS works well for regular public transport with a fixed schedule, it cannot represent demand-responsive transport types, such as mini-buses or ride-hailing. GTFS data are not always made accessible to researchers by the service provider. Routing requests through Google Maps can be used in such cases, though the number of free requests per day is limited. Similarly, GBFS is an open standard to provide real-time information about the current status of bike-sharing/ other micro-mobility systems and their availability. Thus, GBFS can play a potential role in shared mobility data by bringing the fragmented information from hundreds of bike-share and micro-mobility platforms under a common standard. DATEX II[13] is another example of a common language used for sharing road traffic data (such as vehicle flow, roadworks, parking, and traffic measures) between traffic control, management centers, and service providers in the EU. In some instances, these data are also available to the public, such as a live feed for the parking situation in Norfolk County, UK[14], or for road traffic counters in Switzerland[15]. The Zephyr Foundation and various stakeholders have introduced data standards used by the transport modeling community. For example, the OMX[16] open matrix format was developed in 2013 and allows transport modelers to share and read different models' matrices. More recently, Zephyr promoted the General Modelling Network Specification (GMNS), an open format for network data explicitly designed for transport models (Smith et al., 2020). The idea

[13] https://www.datex2.eu/

[14] https://www.data.gov.uk/dataset/b6e83001-fb1e-43e8-9ef1-a522b226160a/norfolk-county-council-live-car-park-data

[15] https://www.opentransportdata.swiss/de/rt-road-traffic-counters/

[16] https://github.com/osPlanning/omx

is that models should share a common standard for input and output data. Similar to the emergence of public transport datasets after the emergence of GTFS, this could lead to more publicly available network models in the future.

3.7 SWOT analysis

We present the SWOT analysis (Table 3.3) for the data discussed in the above section. SWOT helps us synthesize the discussion on the data by bringing together aspects that influence the applications and data availability. Spatial-temporal and contextual (travel mode, population sample) coverage, aggregation level, data frequency, and historical data availability are factors that play a role in determining their application. These factors are directly or indirectly determined by the data providers, who are responsible for protecting the user's privacy and propriety interests.

3.8 Summary

MPND have extensive spatial-temporal coverage, but these data are privately owned and publicly unavailable. Social media data offer location data with contextual information, which is unique but suffers from sample bias favoring the young and high-income population and leisure activities (Chaniotakis & Antoniou, 2015). Further evolving social media platforms and privacy issues increase uncertainty in the availability of these data in the future. Due to privacy or commercial interest issues concerning disaggregated data from mobile phones, smart cards, and social media, data owners (private or public) often reluctantly share these data or restrict and limit its availability. It is also crucial for data providers to process raw data before sharing to mitigate any privacy concerns. For example, mobile network data or AVL data need to be anonymized or aggregated so that the individual users/ patterns cannot be identified. While such intermediate steps are necessary, they commonly result in losing some information in the resulting data.

Open standards like GTFS have helped increase the usability and interoperability of public transport data. Similarly, GBFS is a relatively new step towards sharing data from new mobility forms, such as bike-sharing. Crowdsourced VGI bridges the gap of missing spatial information by providing an alternate source of large datasets, but their quality and depth depend on the involved community's participation. Successes on the open data front have been generated due to the collaboration of data consumers and data providers. These developments have had a positive cascade effect by giving birth to new tools and innovations based on these datasets.

This chapter provided a broad overview and conceptual understanding of the openness and applications of non-conventional data in transport research and also marked the end of Part I of this dissertation.

Table 3.3: SWOT Analysis

Data	Strengths	Weaknesses	Opportunities	Threats
Mobile phone network data[a]	Spatial-temporal coverage over all modes, large sample size	Needs ground truth for scaling factors, representativeness issues, missing sociodemographic attributes	applications in demand, supply and traffic modeling	Strict data anonymization for privacy protection; proprietary and commercial nature leads to sharing averseness
Smart card data[b]	Disaggregate trip and fare data, spatial-temporal trends over a long duration	Only for closed transit systems, sample representation, missing sociodemographic attributes and context	Transit modeling, the impact of disruptions on travel behavior	Data not universally accessible, often requires complementary data, such as GTFS or AVL
GNSS data and AVL[c]	Primary source of movement and traffic data	GNSS data suffer from signal loss and errors, detector-AVL has limited observability, lack sociodemographic attributes	Primary or secondary data (validation) in modeling tasks	GNSS-AVL primarily controlled by private companies, privacy-sensitive. Dedicated hardware and costs for detector-AVL
Social media data[d] (Free version)	Contextual, disaggregate and geotagged information	Sample bias, textual data requires processing, majority of data not free	Trip destination, purpose and activity space analysis	Evolving privacy issues affect data availability, commercialization and major control by private companies
VGI[e]	High spatial coverage, a rich environment of tools and programs	Varying quality by region, lack of data validation	Land use and trip attractions for travel demand models, transport networks	a Large amount of data can cloud completeness issues, limit reliability. Depends on continuing participation of contributors
GTFS[f]	Standardised format	Only works for regular, schedule-based transit, fragmented/ aggregated feeds, validation before use	Detailed spatial-temporal modeling of public transport	Risk of being used only internally in companies without providing data to public
GBFS[g]	Standardised format, Real-time information	Historical data not available, only station information, missing trip attributes	Behavioural analysis of shared and emerging micro-mobility services	Scalability depends on the participation of private or government service providers

[a] Caceres et al. (2007); Rojas et al. (2016); Tolouei et al. (2017) [b] Bagchi and White (2005); Faroqi et al. (2018); Pelletier et al. (2011) [c] Gadziński (2018); Sandim et al. (2016) [d] Chaniotakis and Antoniou (2015); Grant-Muller et al. (2015); Rashidi et al. (2017) [e] Bast et al. (2015); Senaratne et al. (2017) [f] Fransen et al. (2015); Kickhöfer et al. (2016); Kujala et al. (2018) [g] North American Bikeshare Association (2015)

Part II

Creating value from emerging data

4 Treating noise and anomalies in drone videography data

Contents

The content of this chapter has been presented in the following work:

Mahajan, V., Barmpounakis, E., Alam, M. R., Geroliminis, N., & Antoniou, C. (2023). Treating Noise and Anomalies in Vehicle Trajectories from an Experiment with a Swarm of Drones. IEEE Transactions on Intelligent Transportation Systems. doi: 10.1109/TITS.2023.3268712

4.1 Introduction

In this chapter, we identify the problematic cases and process the noise and anomalies in the traffic acceleration data from drone videography. While high-frequency noise can be addressed using available techniques, detecting anomalies is tricky since it is an unsupervised task. The outlier detection method, which accounts for the local properties of the data and could encompass variations among vehicle class, driving behavior, and anomalies, would be helpful and a step toward extending the state-of-the-art. The 2D tracking has been used in the Autonomous Vehicle (AV) motion planning-related literature (Claussmann et al., 2020) and could reduce the errors in the drone videography dataset, we aim to apply fast processing to trajectory data (derived from drone videography) to remove errors, improving the quality of the data for traffic-oriented purposes. As a result, to fully take advantage of such a detailed and massive dataset, it is necessary to find appropriate techniques to detect the outliers (unrealistic transient peaks) and filter them efficiently. We see an opportunity to propose an anomaly detection method to detect the relevant anomalies, i.e., implausible accelerations. We present our methodology in the next section, considering the large dataset, high variability in the driving attributes and context, and minimum fine-tuning and speed.

The remainder of the chapter is structured as follows: the following section lists the chapter contributions, followed by the methodology of the study, followed by a section on data description and illustration of some of the problematic cases, followed by a section on data analysis, followed by the results of this research, and finally, we summarize the chapter.

4.2 Research Contributions

The contributions of this chapter are as follows:

- We provide a detailed analysis of the pNEUMA dataset and point out the errors in acceleration and speed time series that need processing.
- We develop a scalable methodology to treat noise and unrealistic peak anomalies in the trajectory data from drone videography using Extreme Gradient Boosting (XGBoost) and smoothing filters. The XGBoost model with adaptive regularization creates an anomaly mask for each trajectory. Our methodology reduces the burden of manually fine-tuning the anomaly detection model when applying it to a large dataset. [SRQ(5)]
- The codes developed for this proposed methodology are shared on GitHub[1].

[1] `https://github.com/vishalmhjn/pneuma_treatment`

4.3 Methodology

Before removing errors, we analyze the occurrences of excess values of accelerations in the trajectory data from drone videography. Spatially and temporally near vehicles can highly correlate due to traffic flow or car-following behavior. Spatially apart but temporally near vehicles can also show correlated errors due to global events such as wind disturbance to the drone. However, such errors can also occur on account of image processing or data processing. Spatially near but temporally far vehicles can show similar anomalies if passing through the same street obstructed from drone view at different times (Kim et al., 2023). However, if the trajectories of vehicles are spatially and temporally apart, we expect little error correlation among them.

Our methodology for treating noise and anomalies is shown in Figure 4.1. We treat vehicles one by one so that the noise and anomalies are identified flexibly depending on the vehicle's attributes.

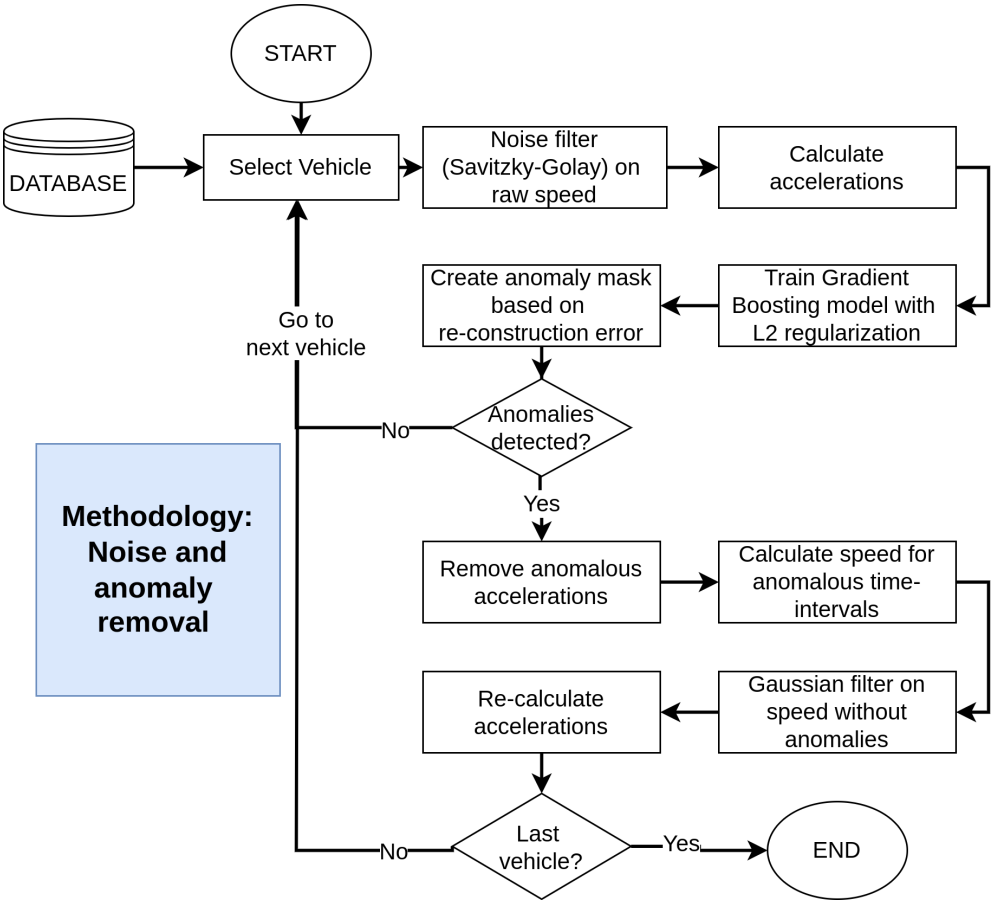


Figure 4.1: Methodology flow chart. ©2023 IEEE.

Let us denote the raw speed data by s_t^i for the i^{th} vehicle at time t. We use the SG filter to remove the noise in the speed time series from the raw data to obtain output v_t^i.

We find that the output of the Savitzky-Golay (SG) filter will be biased because the filter is applied to data containing anomalies. However, this is only an intermediate step, and we will address this specific problem subsequently. Smoothed output (from the previous step) is a better choice for evaluating acceleration from the speed time-series (Equation 4.1) than the raw data, as the gradient of noisy data could fluctuate and give even more unrealistic values of the accelerations.

$$a_t^i = \begin{cases} \frac{v_{t+\delta t}^i - v_t^i}{\delta t}, & \text{if } t = 1 \\ \frac{v_{t+\delta t}^i - v_{t-\delta t}^i}{2\delta t}, & \text{if } 1 < t < n \\ \frac{v_t^i - v_{t-\delta t}^i}{\delta t}, & \text{if } t = n \end{cases} \tag{4.1}$$

Where v is the smoothed speed output of the SG filter, δt is the granularity of the data.

We aim to detect and process the unreasonable values of the acceleration for the anomaly detection task. This is akin to peak detection in a time series, where the peaks represent anomalous behavior. Our work also makes a similar assumption as in the Eskin (2000), where the proportion of representative/ usual data is significantly larger than the anomalous data. This assumption is verifiable by plotting the density plots of data distribution and checking what portion of the data lies within the usual range. Next, we fit a regularized machine learning model to reconstruct the acceleration time series. Regularization is a popular method to analyze noisy data (Stickel, 2010). The use of reconstruction error to classify anomalies is demonstrated in previous studies (Japkowicz et al., 1995; Sakurada & Yairi, 2014) e.g., Sakurada and Yairi (2014) used an autoencoder (a deep learning model) with a regularized objective function for this task. Instead of a deep learning model, we select XGBoost model (T. Chen & Guestrin, 2016) for this purpose. XGBoost is based on the concept of Gradient Boosting Machines (GBM), but with certain algorithmic and software enhancements. We select this model because boosting models are generally considered "off-the-shelf classifiers" (Hastie et al., 2001), and thus need lesser feature preprocessing and parameter tuning than other machine learning models such as neural networks. Two basic tunable parameters for a gradient boosting model are the number of iterations (or number of estimators) and the size of each of the constituent trees (number of leaves in the tree) (Hastie et al., 2001). Boosting trees are computationally feasible on even large datasets since small trees are used as weak learners (depth of a tree varying between 4 to 8). In Boosting, observations with high residuals generally receive ever-increasing influence with each iteration (Hastie et al., 2001). Increasing the number of iterations and size of the tree will result in over-fitting. Thus, it is important to stop training the model before it starts to overfit the data. Another way to prevent over-fitting is by using a regularization (similar to Ridge regression) to shrink the contribution of each tree. XGBoost (T. Chen & Guestrin, 2016) uses the following regularized loss or objective function:

$$\mathcal{L}^{(k)} = \sum_{t=1}^{T} l\left(a_t, \hat{a}_t^{(k-1)} + f_k\left(\mathbf{x}_i\right)\right) + \Omega\left(f_k\right) \tag{4.2}$$

where, l is a differential convex loss function that measures the difference between the target a_t and prediction $\hat{a}_t^{(k-1)}$ acceleration, each f_k corresponds to an independent k^{th} tree with structure q and leaf weights w, T is the number of frames (input samples) in the series, and

$$\Omega(f) = \gamma K + \frac{1}{2}\lambda \|w\|^2, \tag{4.3}$$

K is the number of leaves in the tree, γ is the parameter that penalizes large trees, and λ is the regularization parameter that penalizes the high values of w. The use of regularization controls the over-fitting so that the models are not sensitive to outliers. We refer the reader to the paper by T. Chen and Guestrin (2016) for more details on the XGBoost model.

Ideally, the model should mirror (or fit) the non-anomalous segments except the anomalous segments because we want to preserve all the information in the raw data except the anomalies. To achieve this, we provide the input features consisting of three features: a) smoothed speed series, b) lateral accelerations, and c) acceleration from (Equation 4.1). The target variable for the model is again the same acceleration as the input feature since the aim of the model is to reconstruct the acceleration time series. Therefore, the input features will tend to be correlated. We adopt L2 regularization (Tikhonov's regularization) to prevent over-fitting. The value of the regularization parameter (λ) is adapted for each trajectory, as given by:

$$\lambda^i = b \cdot max(|a^i|)^n, \tag{4.4}$$

b is a constant, $max(a)$ is the maximum acceleration value observed for a specific trajectory, and n is a positive real number. The rationale for using an adaptive λ is that the vehicle trajectory data could be diverse from different drivers, vehicles, and contexts. Thus, it makes more sense to define an outlier within the context of each trajectory. Therefore, we hypothesize that a single value of λ does not provide this flexibility. λ is directly proportional to the absolute maximum acceleration in the input data since we want the regularization to be highly effective for the unreasonably high values of the acceleration. High values of b and n cause high penalization of the anomalies, limiting the acceleration values range. In this paper, we use different sets of parameter combinations (b and n) to conduct the sensitivity of the regularization for anomaly detection. It is also possible that different combinations of b and n lead to similar results for specific maximum acceleration values.

We select a sufficiently large fixed value of the number of iterations (say M) and then constrain the model with the L2 regularization term to prevent the model from fitting the outliers. This is motivated by the fact that the regularization term can prevent the model from fitting the extreme points by imposing a high cost. The output of the boosting model is the reconstructed acceleration profile. The reconstructed series should almost replicate the input series for a representative (part of) time series. The reconstructed series will act as a mask to filter the anomalies for the problematic time series. Therefore, the reconstructed series is called an "anomaly mask". We define a tolerance level (τ) and

check if the difference between the input series and reconstructed series exceeds that level to label the anomalous sections (0: representative, 1: anomaly) (4.5). A smaller value of τ will make the model more conservative, i.e., more data will be labeled as anomalous and vice-versa.

$$label_t^i = \begin{cases} 1, & \text{if } |a_t^i - \hat{a^i}_t| \geq \tau \\ 0, & \text{otherwise} \end{cases} \tag{4.5}$$

Where $label_t$ is the anomaly label for the t^{th} frame instance of a trajectory, and $\hat{a}_t$ is the reconstructed output. In our approach, the XGBoost model and the L2 regularization replace the statistical measures (mean or median) and distance metric (number of standard deviations) to do better than the existing methods without manually adapting the parameters for each trajectory.

Further, if the simultaneous anomalies in the trajectory are detected within a gap of f frames, we assign the complete sub-sequence as anomalous for subsequent correction. This completes the anomaly detection or labeling task. If no anomalies are detected in the previous step, processing for the current trajectory ends, and the next vehicle is selected. Thus, only nominal smoothing via the SG filter is applied to remove the noise in the absence of an anomaly. After the anomalies are labeled, we need to recalculate the speed ignoring the anomalous accelerations, to obtain the refined or reconstructed speeds. We use the constant acceleration model for speed estimation using (Equation 4.6). The constant acceleration model is reasonable given the high frame frequency in the trajectory data. Thus acceleration is only considered constant during the one-time step, e.g., 0.04 seconds for data recorded at 25 Hz. This step ensures that the speed and acceleration data are internally consistent, inspired by Montanino and Punzo (2015).

$$\hat{v^i}_t = \hat{v^i}_{t-1} + \hat{a^i}_t \cdot \Delta t \tag{4.6}$$

We replace the speed values for the anomalous points or segments with the reconstructed speed values based on the following:

$$v_t^{i,o} = \begin{cases} \hat{v^i}_t, & \text{if } label_t = 1 \\ s^i{}_t, & \text{otherwise} \end{cases} \tag{4.7}$$

The above speed time series is treated with the low-pass (Gaussian) filter to recover "unbiased" smoothed speeds. Therefore, this speed series is final since the smoothing of the raw data is done after removing anomalies. Finally, we compute the acceleration values (Equation 4.1) using these speed values. We repeat this process for all the vehicles in the sample. Due to low errors in the position of the tracked vehicle in the pNEUMA dataset, we do not reconstruct or adjust the positions using the processed speed vector. Due to this, we are prone to losing the internal consistency between position and speed (Coifman & Li, 2017), and this is a subject of a future study. Finally, we analyze the distribution of acceleration in the detected outliers and the rest of the data for all the vehicles in the sample.

Our method only relies on the feasible range of acceleration for validating the results. It is relevant to point out that we do not conduct jerk value analysis, as done in previous studies by Montanino and Punzo (2015); Punzo et al. (2011). This is partially mitigated in the final step by applying the Gaussian filter on the speed series after removing anomalies, eliminating the sharp edges in the acceleration profile.

4.4 Data collection

In October 2018, the pNEUMA experiment was conducted in Athens, Greece, aiming to record traffic streams over an urban setting using a swarm of ten drones (Barmpounakis & Geroliminis, 2020). The pNEUMA experiment aimed to revolutionize how drones as an emerging technology could reshape our understanding of traffic congestion mechanisms. Specifically, the scope was to understand better how congestion forms and propagates in congested multi-modal urban environments through massive data from aerial footage, emphasizing disturbances generated by interactions among different types of vehicles. For the specific experiment, the morning peak (8:00 a.m. to 10:30 a.m.) was recorded for each working day of the week. For improved safety and cooperation, the swarm would take off from the two take-offs/ landing areas (H1 and H2 in Figure 4.2) at the start of the experiment, and each drone would go to its area of responsibility. When all drones were at their hovering point, the recording of the traffic stream would start simultaneously, and when the battery ran low, they would return to the landing point. Considering that drones could hover for up to 25 minutes, including take-off, routing, and landing times, it was decided that each session would take place every 30 minutes for better coordination and standardization of the experiment. This setup allows 15 to 20 minutes of continuous monitoring of traffic. During the temporal blind spots, trajectories were not recorded and were not related between sessions.

The analyzed study area includes low, medium, and high-volume arterials, around 100 busy intersections (signalized or not), more than 60 bus stops, and nearly half a million trajectories. This massive dataset contains trajectories of every vehicle present in the study area, calibrated in the WGS-84 system every 0.04 seconds, as this is the maximum frequency allowed by the video's frame rate. The average ground sampling distance is calculated to be 16.5 cm/px. Except for the features that can be produced using the position information (speed, acceleration, and distance traveled), each vehicle type is available (car, taxi, motorcycle, bus, heavy and medium vehicle). We refer the reader to Barmpounakis and Geroliminis (2020) for more details on the design of the experiment and to Kim et al. (2023) for the recently released drone imagery. This dataset is also shared with the research community[2].

4.5 Data Analysis

For this paper, the dataset corresponding to all drones' recordings from the last day of the experiment (10:00 a.m. - 10:30 a.m.) is selected. These data contain trajectories of

[2]These data are downloadable from `https://open-traffic.epfl.ch`

Figure 4.2: The study area of pNEUMA experiment. Source: (Barmpounakis & Geroliminis, 2020). ©2023 IEEE.

about 10,500 vehicles with a vehicle's position, speed, and acceleration at 25 Hz. First, we visualize the acceleration samples of all vehicles (Figure 4.3). For motorcycles, median acceleration (0.62 m/s^2) is slightly higher than the other vehicles. However, the range of acceleration for motorcycles is the largest. To check the extreme values, we also visualize the maximum acceleration and deceleration according to the vehicle type (Figure 4.3) and find noticeable differences. The median maximum acceleration or deceleration of motorcycles, taxis, and buses is greater than those of cars and medium vehicles. In contrast, heavy vehicles show the lowest median values. This can be partially explained by the different acceleration capabilities (motorcycles vs. heavy vehicles) and driving behavior (taxis vs. private cars). Specifically, motorcycles, due to their limited width and advanced maneuverability, when compared to other vehicles, make their tracking more challenging (Barmpounakis et al., 2016).

In Figure 4.4, we illustrate the maximum accelerations of each vehicle at the location they occur. Excessive accelerations (red areas of the heatmap) appear mainly in the southeast (captured by drone numbers 1, 2, 3, and 4), whereas such instances are uncommon in the north. These can be explained by the limitations of drone videography, such as intersections due to bad lighting (shade, low contrast), roads on the edge of the recorded video (due to video distortion), other tracking issues, and data post-processing. While during the pNEUMA experiment, the videos were stabilized, the current work did not test for evidence of residual camera motion. However, we did not see any evidence

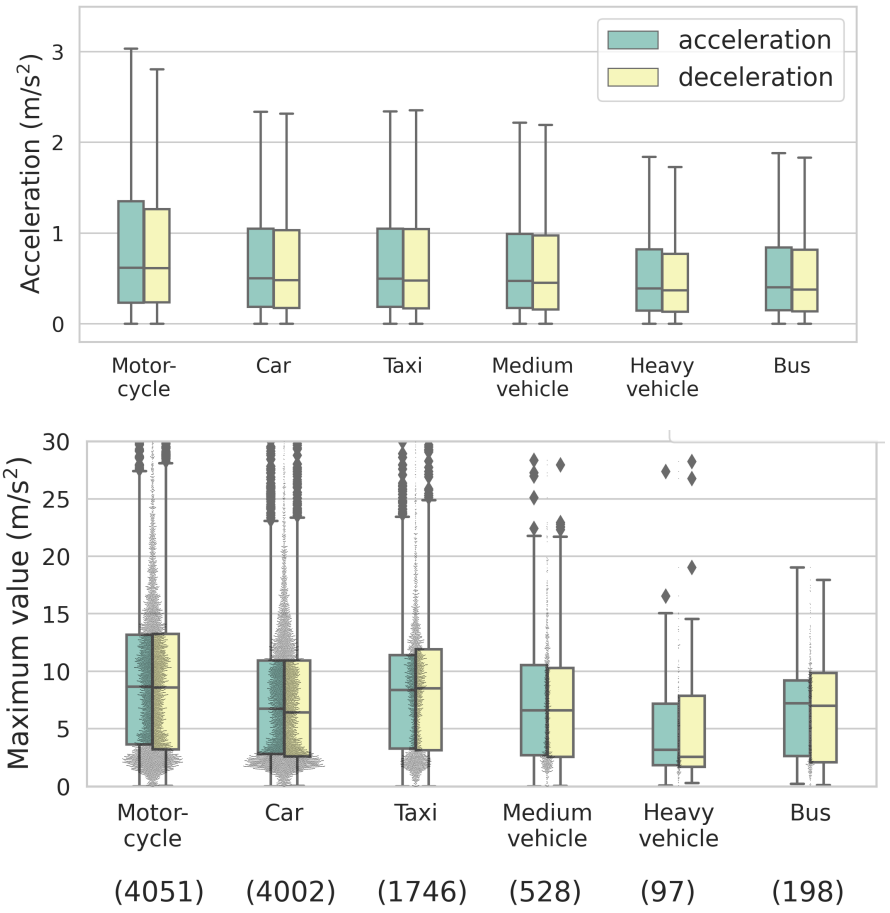


Figure 4.3: Distribution considering the (top) all accelerations of all vehicles, and (bottom) maximum acceleration and deceleration for all vehicles before treatment. The number of unique vehicles for each category is mentioned in parentheses.

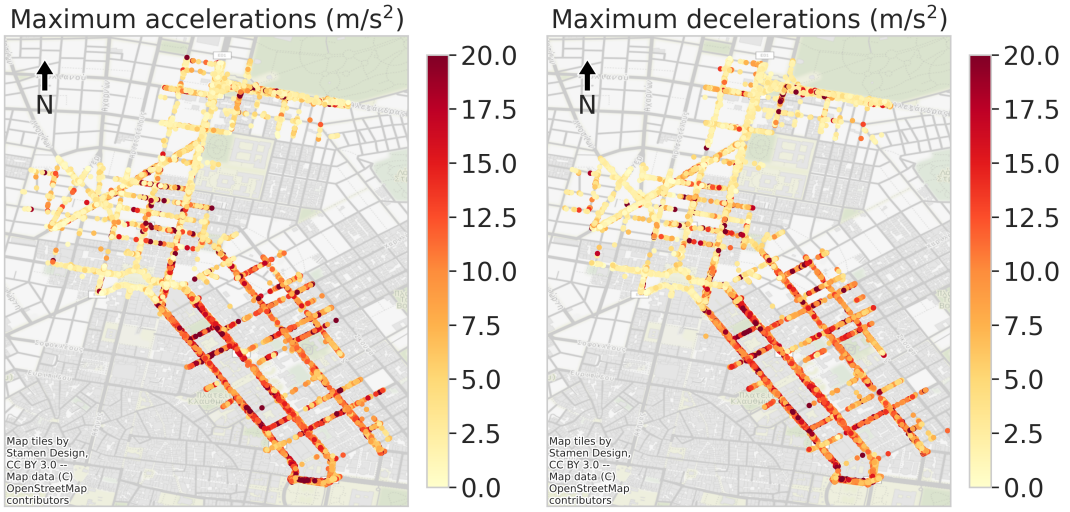
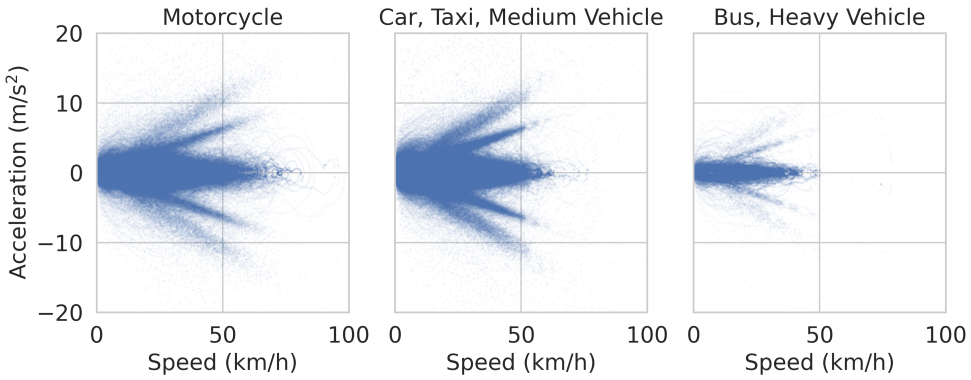


Figure 4.4: Heatmap showing location-wise maximum acceleration and maximum deceleration for each vehicle. ©2023 IEEE.

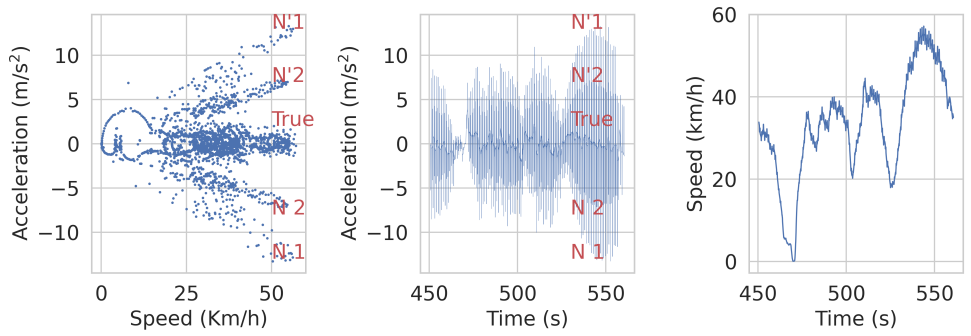
suggesting the presence of such non-vehicular motion during the experiment, processing of the videos, or data analysis efforts.

In the pretreated data, the speed-acceleration plot (Figure 4.5) shows four "flanks" (two each for acceleration and deceleration). We discover these flanks are due to the noise in the speed (and acceleration) vector. The slope of the flanks has a unit of frequency (s^{-1}) and is about constant, which points to the presence of two components of noise with fixed frequencies in the dataset. On further investigation, we find that this periodic noise is not present in the individual drone recordings but occurred while merging the datasets e.g., drone 2 and drone 3. Thus, it is not related to the nature of the experiment or the CV algorithm but to the specific dataset. We calculate the number of vehicles for which the magnitude of acceleration and deceleration exceeds a cut-off limit (5 m/s^2, 10 m/s^2, 15 m/s^2, 20 m/s^2) over time. In Figure 4.6, we expect and see that excess acceleration occurrence reduces with the increase in the cut-off limit. At a cut-off of $5m/s^2$, we see occurrences corresponding to the previously mentioned two components of high-frequency noise. Further, these occurrences are also sinusoidal over about 90 seconds. At a cut-off of 10 m/s^2, only one of the two high-frequency noises (about 1 Hz) is noticeable, meaning that the amplitude of the other noise is lesser than 10 m/s^2. Here too, the sinusoidal nature of occurrences is even more noticeable at a period of about 90 seconds. This systematic periodicity points to green waves on the arterial roads covered by drones 2 and 3, during which new traffic enters and leaves these arterial roads.

At the same time, this behavior excludes the effect of wind blows, which tend to be random. No high-frequency and sinusoidal occurrences are noticeable at a cut-off of 15 m/s^2 and 20 m/s^2, which means that very high acceleration values, such as more than 20 m/s^2, are scattered. Further spatial analysis reveals that noise and sinusoidal behavior



(a) Pretreated data showing the presence of four flanks (for a sample of 1000 vehicles)



(b) Example with two noise components (N1 and N2)

Figure 4.5: Acceleration-speed plots. ©2023 IEEE.

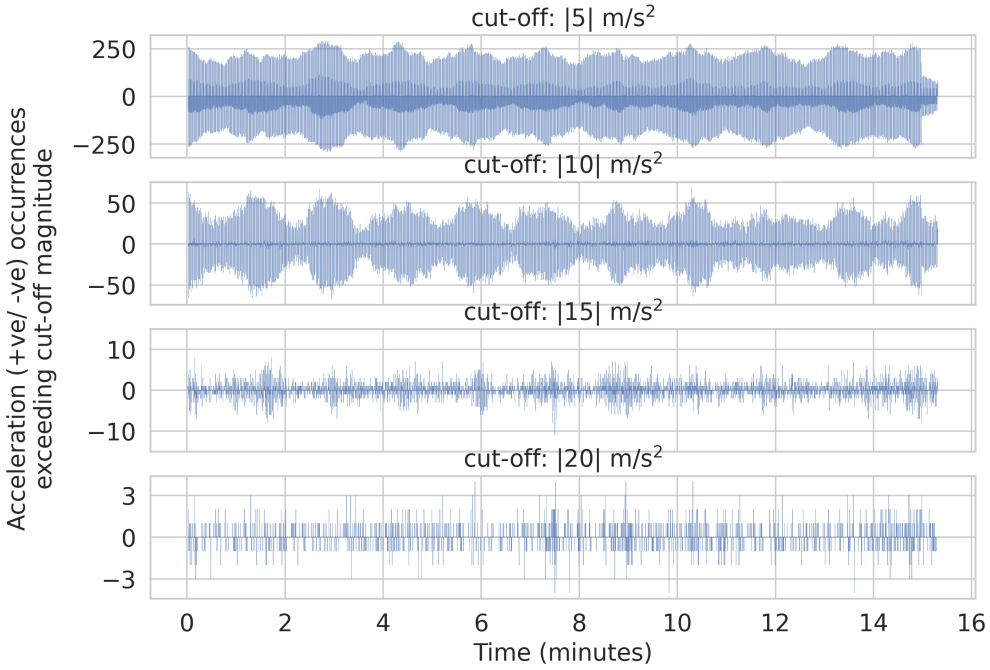


Figure 4.6: Occurrences (positive for longitudinal acceleration and negative for longitudinal deceleration) exceeding the cut-off limit. ©2023 IEEE.

of anomalous instances is prevalent on drone recordings 1, 2, 3, and 4. This is shown in Figure 4.7. The correlation between vehicle influx at high speeds (green wave) and excessive acceleration is noticeable in the bottom part of the map. Accurate vehicle tracking can be challenging for the object tracker during this sudden acceleration and deceleration behavior, thus resulting in such periodicity of anomalous instances. For terminology, we conclude that periodic excessive acceleration values are attributed to noise, whereas random (transient) peaks are anomalies.

In Figure 4.8, the challenge of detecting the actual noise becomes even trickier due to the inconsistency of its occurrence. For example, vehicle trajectory id 758 (Figure 4.8) does not need treatment; the same does not apply to vehicle id 1490, 493, or 146. The longitudinal acceleration is noisy for the whole trajectory and shows unrealistic values (around t= 40 s for vehicle 146). Additionally, for vehicle id 1780, it is seen that from time t_1= 0 s to t_2= 150 s, the noise is negligible. In contrast, specific treatment is necessary for the trajectory beyond t_2= 150 s due to unrealistic longitudinal acceleration values. In Figure 4.8, we labeled the data according to the drone and thus found that noise is primarily contributed by drones 1, 2, 3, and 4. When we visualize the longitudinal acceleration for these vehicles over a few seconds (Figure 4.9), we find that noise is synchronized for vehicles 146, 493, and 1490, which complements previous findings (Figure 4.6 and Figure 4.7) about temporal synchronization of noise and specific locations/ drones contributing to the noise. Thus, to recover the desired data, periodic noise and unrealistic transient values should be eliminated.

Figure 4.7: Heatmap showing location-wise speeds and accelerations (exceeding 6 m/s^2) at frames 45 seconds apart to highlight the correlation between them and the periodicity of excess accelerations at about 90 seconds. Similar findings were found for deceleration. ©2023 IEEE.

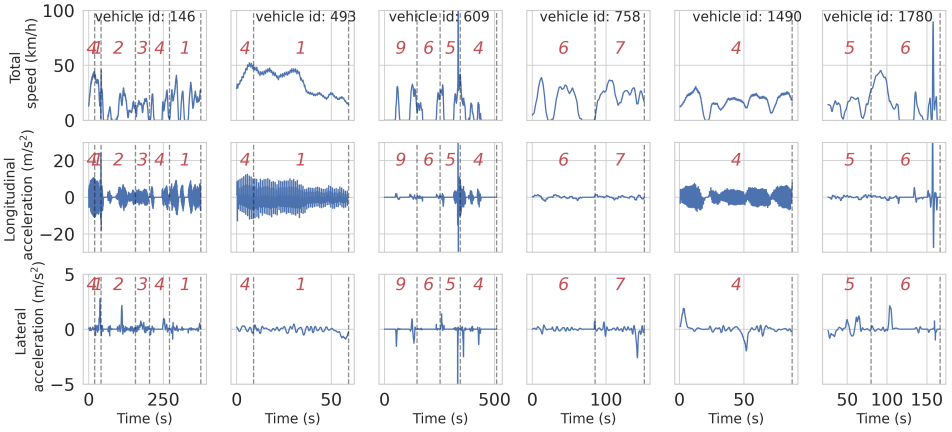


Figure 4.8: Speed, longitudinal acceleration, and lateral acceleration plots of six vehicles showing different characteristics. The number within vertical dashed lines indicates the id of the drone which recorded the corresponding trajectory segment. ©2023 IEEE.

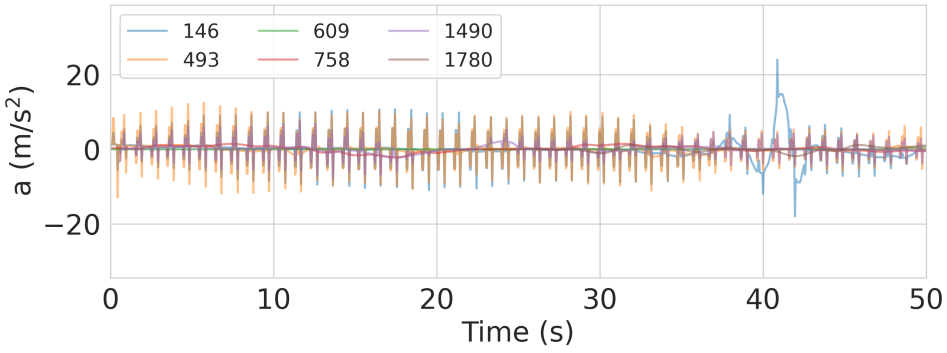


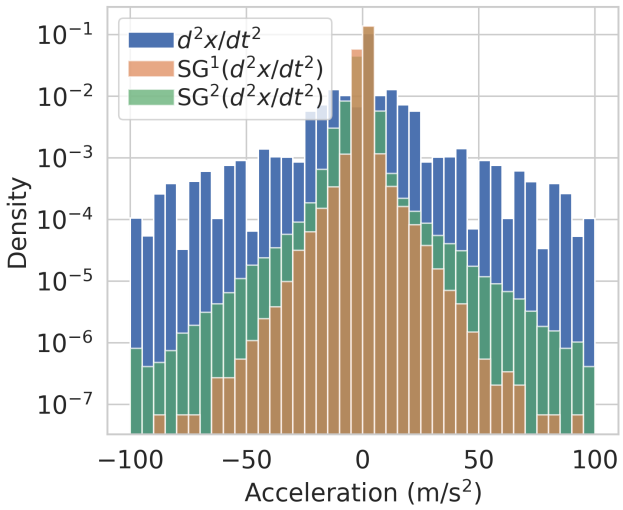
Figure 4.9: Temporal synchronization of noise in the longitudinal acceleration of selected vehicles. ©2023 IEEE.

4.6 Results

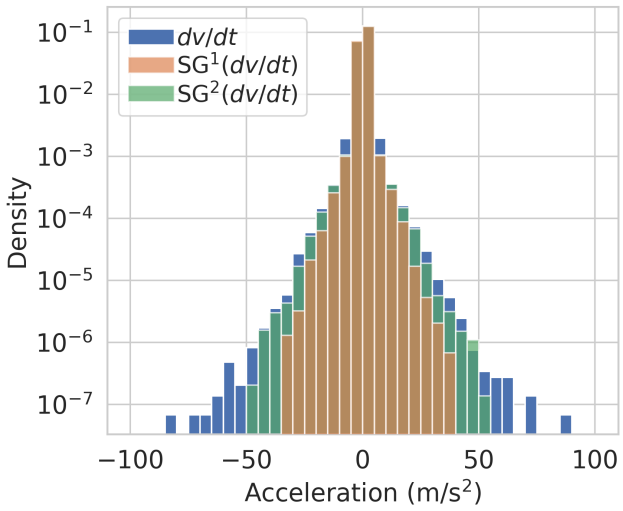
The acceleration calculated from the second derivative of the position is noisy. It has unreasonably high values, possibly due to accumulated errors, as seen by the incidence of values up to ± 100 m/s^2 in Figure 4.10a. The histograms in Figure 4.10 are plotted on the log scale so that heavy-tailed data are visualized easily. Such large values are also expected given the spatial resolution and high sampling frequency (25 Hz). On the other hand, the first derivative of the speed results in less extreme values, shown in Figure 4.10b. This could be because the application of some smoothing filter preprocesses the speed attribute in the pNEUMA data. Thus the processed speed series is a better candidate for calculating acceleration, as noted by Bokare and Maurya (2017). Indeed we find that the first derivative of the position with a moving average filter of a 1-second window (25 frames) has a somewhat similar distribution as the speed attribute given in the data. Therefore, we rely on the speed attribute for our model and take its first derivative to calculate the acceleration further.

The acceleration values range up to ± 75 m/s^2, emphasizing further data processing. We find that there is still high-frequency noise in the data, and it needs treatment. The SG filter is best suited for this task, also evident in Figure 4.10b. The SG filter of the polynomial order one is denoted by SG^1 (same as the moving average filter) and performs better than the polynomial of order two (SG^2). The acceleration distribution is also improved by substantially reducing its heavy-tailedness on the aggregate scale. In Figure 4.10b, the SG^1 filter removes the high-frequency noise and recovers the noise-free signal. Moreover, it also reduces the extreme values since each point is a weighted average of its neighborhood. Despite this, the anomalies remain, and thus the need for anomaly detection.

We use the XGBoost implementation (T. Chen & Guestrin, 2016) in Python for our study. We set the number of iterations or estimators for the XGBoost model as 300 and use different types of the regularization parameter, i.e., fixed or adaptive (Equation 4.4). Figure 4.11 shows the effect of regularization. We find that L2-regularization prevents

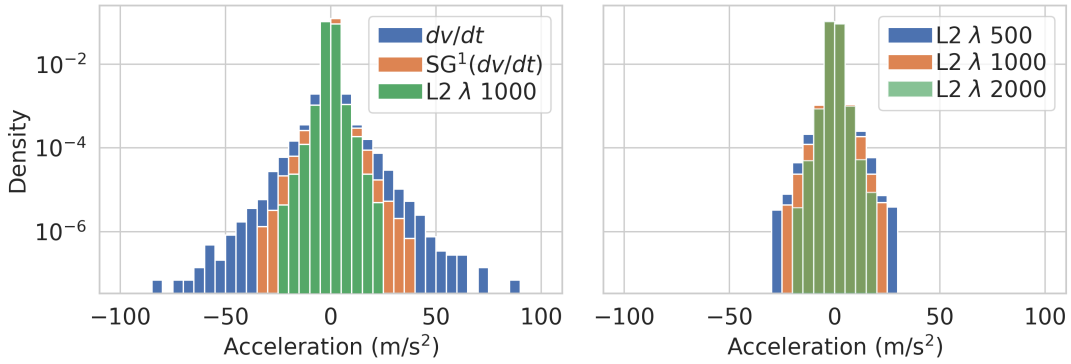


(a) Using position coordinates

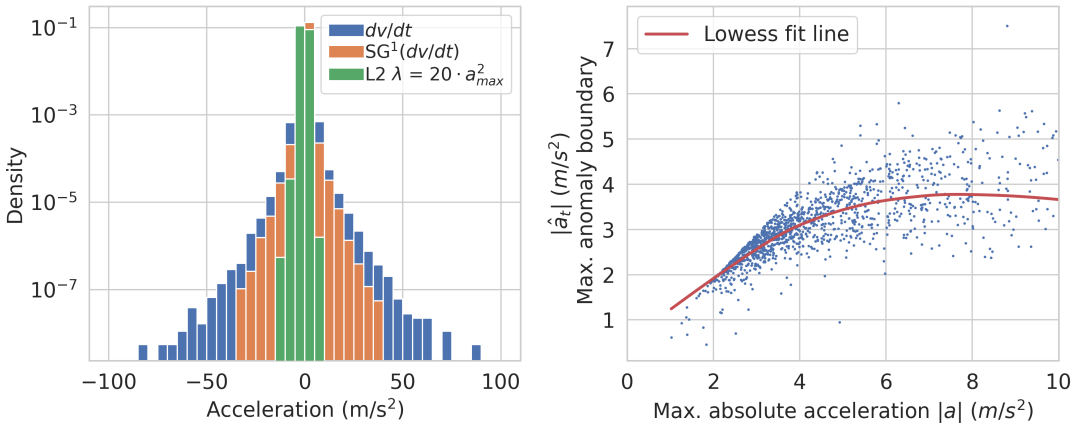


(b) Using processed speed series

Figure 4.10: Savitzky-Golay filter to remove noise from acceleration series. The y-axis is plotted on the log scale for better visualization of distribution tails.



(a) (left) Fixed λ and (right) effect of increasing λ



(b) (left) Adaptive λ with $b = 20, n = 2$ and (right) anomaly decision boundary

Figure 4.11: Effect of regularization parameter λ. ©2023 IEEE.

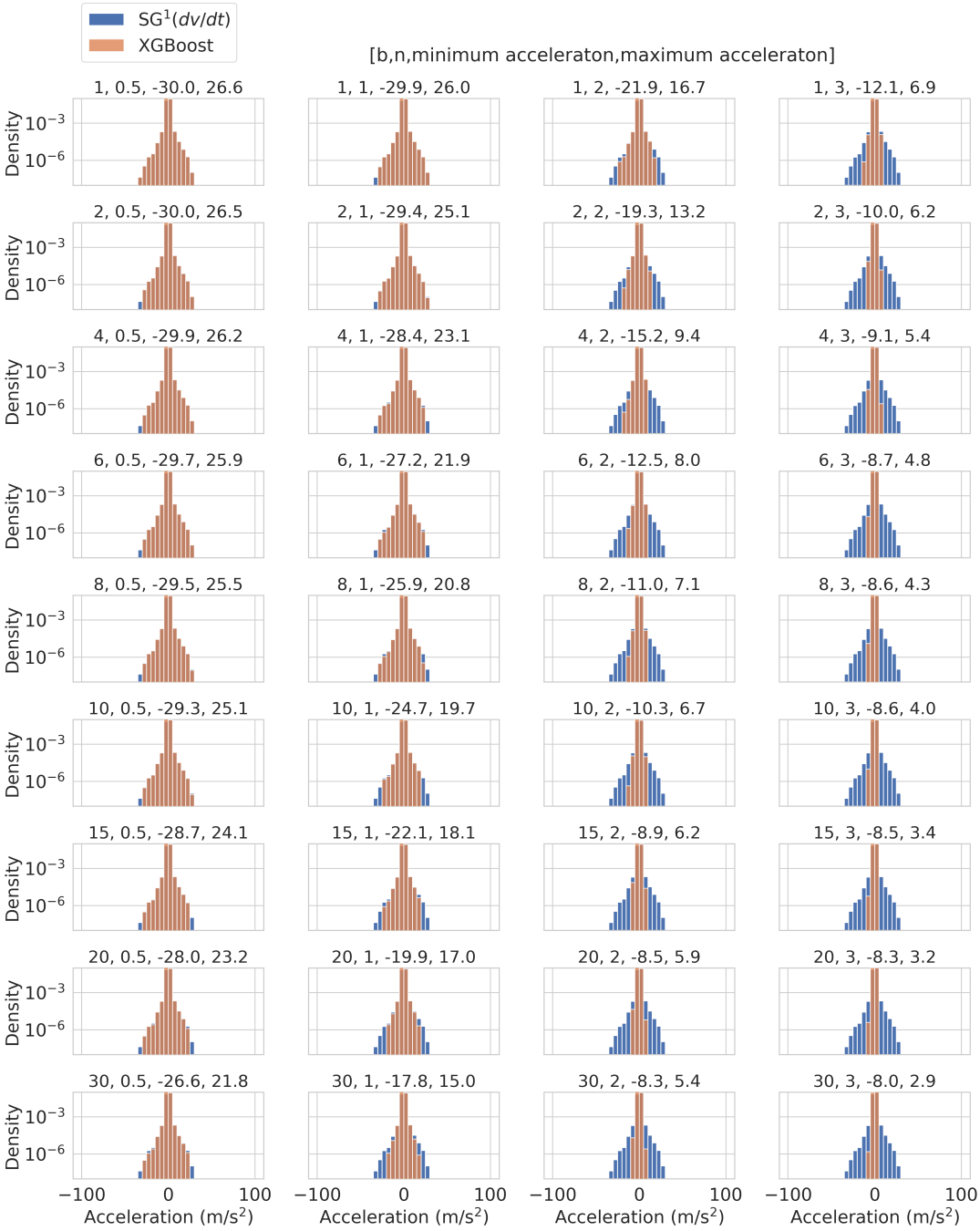


Figure 4.12: Sensitivity analysis of the parameters b and n. ©2023 IEEE.

the reconstructed profile from achieving extreme values. A high λ squeezes the range of the acceleration values, as the histogram for $\lambda = 2000$ is narrower than that for $\lambda = 500$. After these preliminary trials, we use the adaptive λ (with $b = 20$ and $n = 2$), which performs better than the fixed λ in constraining the range of reconstructed accelerations.

We analyze the characteristics of the anomaly decision boundary in Figure 4.11. The decision boundary is correlated with the maximum acceleration simply due to the formulation of the L2 regularization in Equation 4.4. The Lowess fit (Locally Weighted Scatterplot Smoothing) shows a linear trend between (post-processed) maximum acceleration and anomaly decision boundary at small values. At high values, the decision boundary is asymptomatic at around 4 m/s^2, but the range of values goes up to 7 m/s^2. Thus the anomaly detection threshold is not fixed and depends on the trained model. We see this as an advantage of our approach compared to the fixed threshold, which does not adjust to vehicle-specific kinematics.

The sensitivity analysis is done on the regularization parameters used in the boosting model (b and n). We adopt $b \in \{1, 2, 4, 6, 8, 10, 15, 20, 30\}$ and $n \in \{0.5, 1, 2, 3\}$ for a grid-based evaluation based on the combination of these parameters. We use the values of other parameters: τ and f as 0.1 and 10, respectively. The results are shown in Figure 4.12. For low values ($b = 1$, $n = 0.5$), there is hardly any reduction in the anomalies in the reconstructed trajectories. In contrast, for the high values ($b = 30$, $n = 3$), the reconstructed trajectories are heavily biased due to false positives. It is also observed that the proposed method is more sensitive to n and less sensitive to b. In between ($15 \leq b \leq 20$, $n = 2$), the parameter values are found to be optimal for our task. These findings also show that different combinations of b and n can lead to similar results for specific maximum acceleration values.

After removing anomalies, we use the Gaussian filter for smoothing the data, i.e., the filter is applied to the original data without anomalies. We find that window sizes between 12 and 25 frames provide an acceptable range of processed accelerations. Thus, we do not recommend a single best value but a range of values for anomaly detection and smoothing, which can provide a practical solution. This is obvious because our criteria for acceptance are based on the range of final accelerations. However, it is an essential conclusion since multiple optimal parameter combinations help approximately recover the desired signal from the raw data.

We demonstrate the complete methodology in Figure 4.13, showing the de-noising, anomaly detection and removal, retrieval of consistent speeds, and accelerations. Here window sizes for both the SG and Gaussian filters are set as 25 frames. In the third and fourth sub-figure in Figure 4.13 for each vehicle, applying a low-pass filter without removing anomalies will result in biased profiles due to extreme values. We also show in Figure 4.14 cases where the trajectories have either noise or anomalies or none of both. In the case of vehicle id 493 and 505, reconstruction of speed and acceleration is skipped since no anomaly is detected. This also shows our method treats data so as not to cause significant over-smoothing when the data are without noise or anomalies. Thus the final output in both cases is the result of the smoothing only. The post-processed maximum acceleration and deceleration for all types of vehicles (Figure 4.15) are found to be within the reasonable range because their values for most of the sample vehicles' trajectories

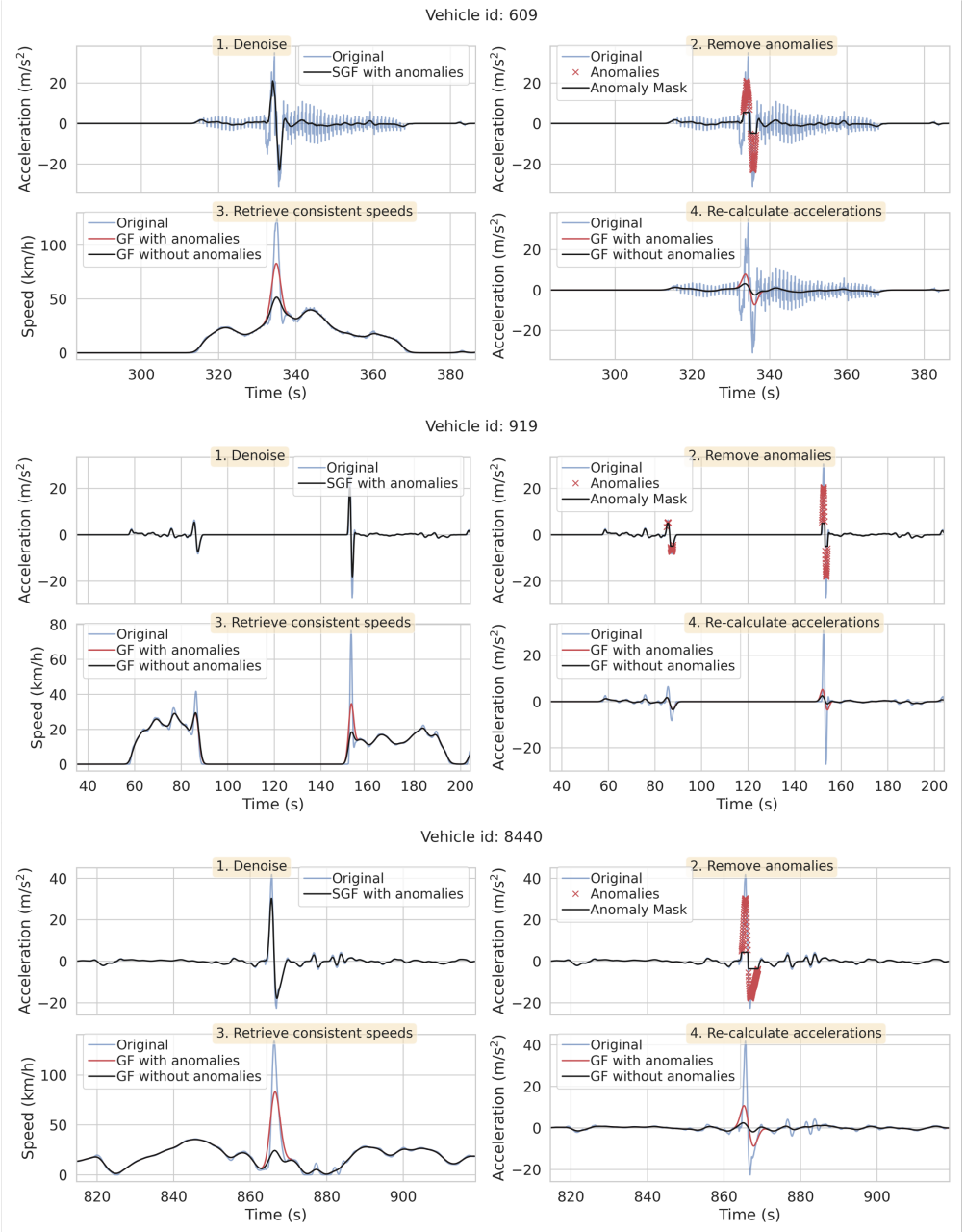


Figure 4.13: Individual steps in the treatment of noise and anomalies for three example vehicles. ©2023 IEEE.

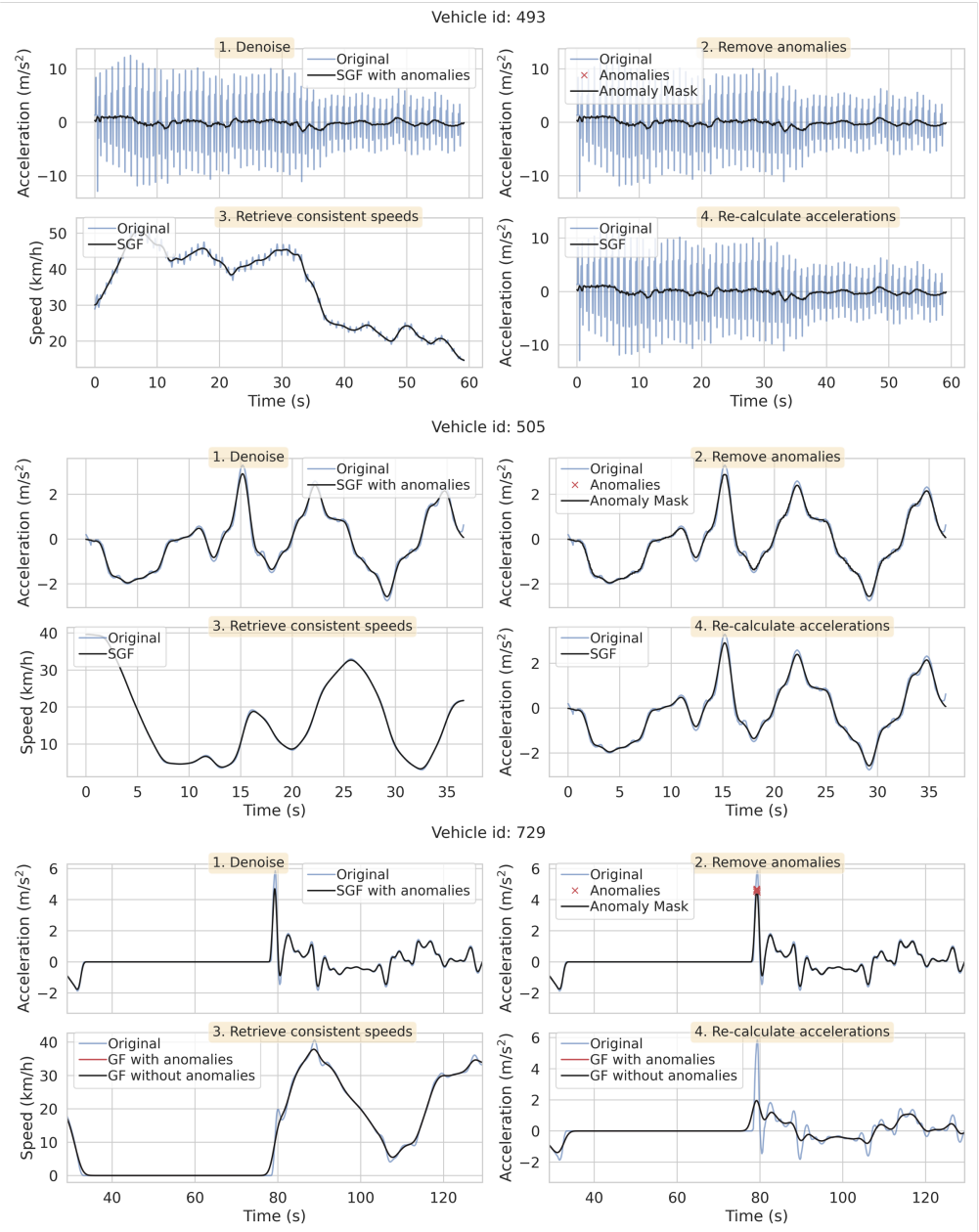


Figure 4.14: Treatment examples when trajectory has (top) only noise, (middle) neither noise nor anomalies, and (bottom) only anomalies. ©2023 IEEE.

are less than 4 m/s^2. Still, for a few samples, values go up to 9 m/s^2. Figure 4.15 should be compared with Figure 4.3 to see the effect of noise and anomaly treatment.

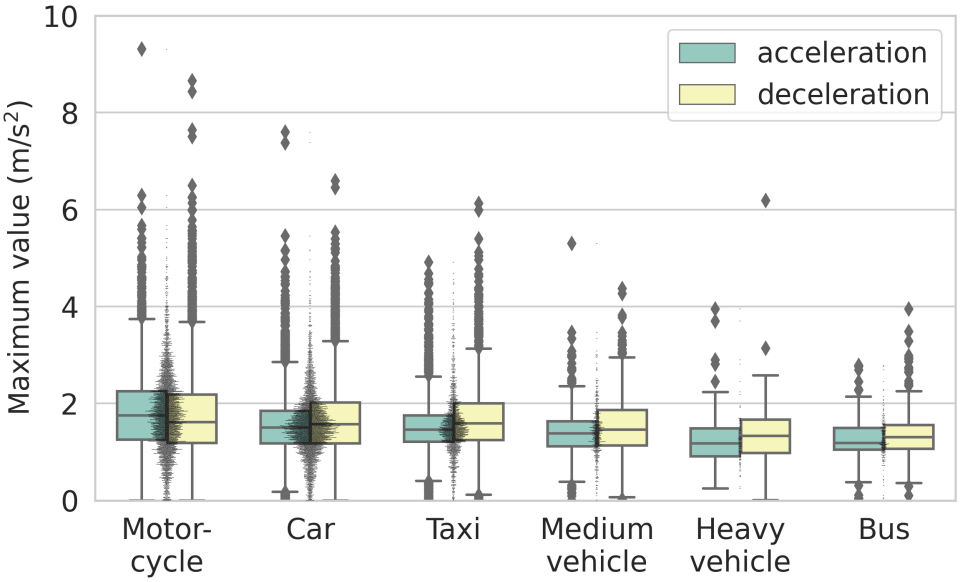


Figure 4.15: Distribution of the maximum acceleration and deceleration for vehicles in the dataset after treating anomalies and noise. The same sample is used here as in Figure 4.3. ©2023 IEEE.

In acceleration-speed plots (Figure 4.16), we show the step-wise treatment process on 1000 vehicle trajectories with extreme acceleration values. The top row in Figure 4.16 shows unrealistically high acceleration values due to noise and anomalies. The output after interim noise treatment using the SG filter (window size: 25 frames) is shown in the panel's second row (from the top). This is followed by the anomaly treatment in the third row. The output after the final smoothing (after the removal of anomalies) by Gaussian filter (window size: 25 frames) is shown in the bottom row of Figure 4.16. The final data in the speed-acceleration plot shows that the range of accelerations is confined within the reasonable range. Over-smoothing can distort the time-space diagram by drastically changing the speed or distance traveled compared to the pretreated values. To verify this, we relied on time-space diagrams of a sample of the vehicles. We did not find significant differences between the distance traveled with pre-treatment and that with post-treatment speeds.

It is also relevant to provide processing time statistics, which can depend on many factors, such as the hardware specifications, parallelization of the algorithms, and the number of samples in each trajectory. We used an HP desktop Machine with eight physical cores (i7-11700F @ 2.50GHz) and 16 GB RAM. We run our method sequentially, i.e., all vehicles are treated one by one. We use two cores for the XGBoost model via the parameter *n_jobs*. In our case, the computation mainly involves the calculation of

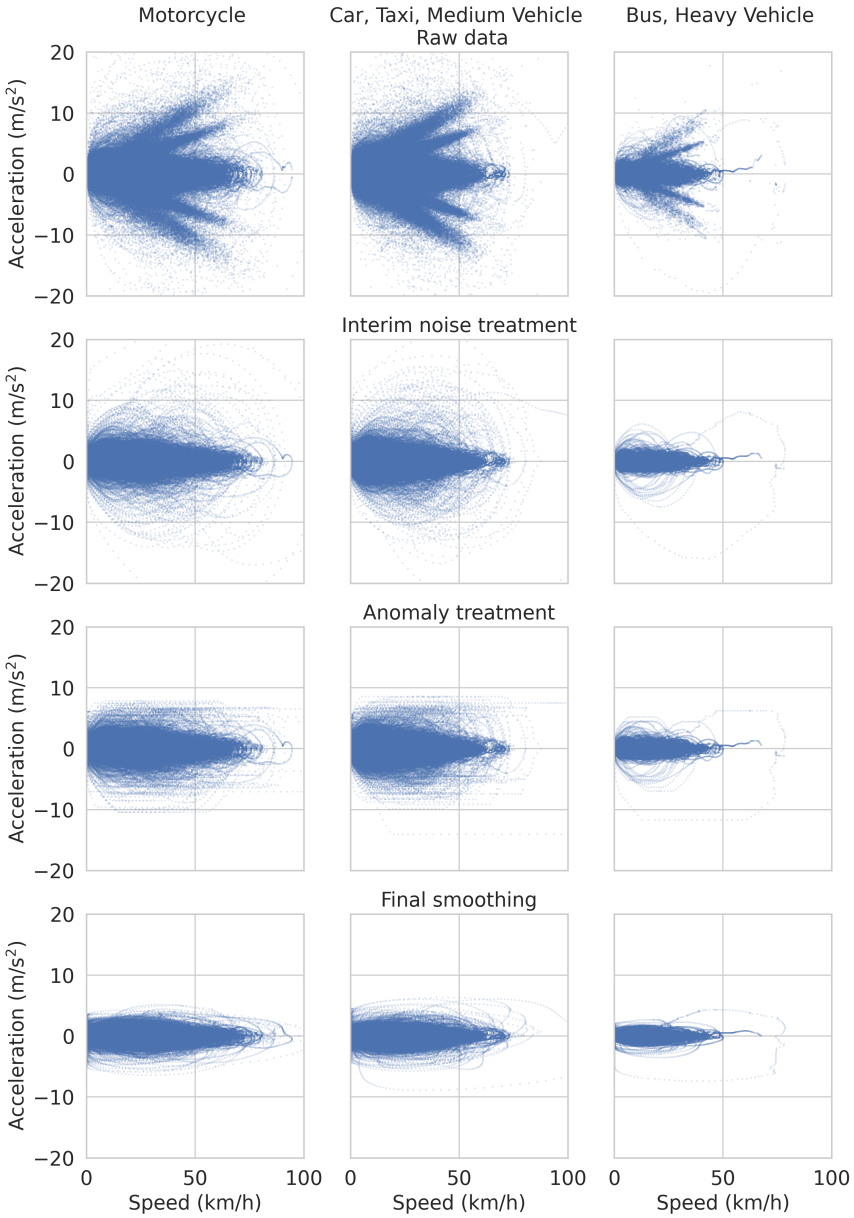


Figure 4.16: Step-wise treatment output and errors. ©2023 IEEE.

numerical gradients, data frame, array operations, XGboost model training, and applying low-pass filters. We record the run-time statistics per vehicle trajectory and find that the run-time mean, median and standard deviation are 0.77 s, 0.59 s, and 1.02 s, respectively.

4.7 Summary

We discussed the challenges of emerging traffic data from drone videography. Validating data quality is crucial before using it, and data processing is necessary to remove errors and improve data quality. In this chapter, we demonstrated the application of noise smoothing and anomaly detection models on the pNEUMA dataset. Techniques like SG filter, XGBoost with adaptive regularization, and GF are used to remove noise and anomalies, effectively identifying unrealistic transient peaks. The treated data is much more suitable for subsequent applications in traffic research and can thus help accelerate future research.

5 Explaining demand patterns during special events using opportunistic data

Contents

The content of this chapter has been presented in the following work:

Mahajan, V., Cantelmo, G., & Antoniou, C. (2021). Explaining demand patterns during COVID-19 using opportunistic data: a case study of the city of Munich. European Transport Research Review, 13(1), 26. doi:10.1186/s12544-021-00485-3

5.1 Introduction

The anticipation or announcement of movement restrictions due to COVID-19 caused specific changes in people's lifestyles, routines, and consumption patterns, such as panic buying during the early pandemic or lockdown stage (Arafat et al., 2020), working from home, or a decline in non-essential retail consumption (Nicola et al., 2020). With people spending, on average, around 40% less, these new trends also risk generating an economic slowdown that could last for a long time (Nicola et al., 2020). If significant, such changes in behavior and attitudes can reveal a pattern exhibited through changes in the number, types, and spatial-temporal extent of the activities. For example, crowding at some locations or imbalanced use of transport facilities, like roads and transport modes, can be observed. Planners must understand these behavioral changes and, more importantly, the spatial-temporal patterns of the population's activities for an effective response. The scale and speed of these changes have left cities, transport operators, and research communities with several unanswered questions on how to respond so that a basic service level is efficiently maintained.

This chapter is structured as follows: the following section lists the research contributions, followed by a section presenting the study's methodology, followed by sections concerning data collection and data analysis, and finally followed by the results section.

5.2 Research contributions

There is a lack of research on applying disaggregated POI data to analyze activity and demand patterns during special events. Using a case study of COVID-19, we also uncover insights into the effects of the pandemic on real-time demand patterns at the POI level. These results provide for both theoretical understanding and practical applications because the pandemic phenomenon was quite new at the time of the study and has not been experienced at the same scale in the last 100 years. This chapter shows that publicly available crowdsensed information could provide useful insights into the spatial-temporal distribution of activities or demand during the pandemic. Subsequently, we propose a model that breaks down POI demand patterns into a set of crucial spatial and other attributes, which are assumed to explain the POI demand. [SRQ(6)]

The extended version of the codes (based on the implementation by m-wrzr and riedmaph (2018) was developed to collect real-time popularity data. Due to data restrictions, these codes are uploaded to the private repository on GitHub. The codes can be shared for research purposes upon request.

5.3 Methodology

We use POI visitation data (response variable, denoted by P) and check their correlation with the spatial and other attributes (explanatory variables) of the POI. Firstly, we define a bounding box for the study area and identify the POIs within that area. For these identified POIs, we collect the historical and live popularity data on different days

to capture the before-lockdown and during-lockdown situations, respectively. Some of the prominent spatial attributes that could affect customer visits at a POI are population density (Rolph, 1932), parking facilities (van der Waerden et al., 1998), and public transit stop (Rolph, 1932) nearby the POI. As we aim to capture the spatial variability among the POIs, the selected spatial attributes should capture the local variation, e.g., the threshold distance for calculating the population around a POI should not be too large. Because of this, a square bounding box of side 600 m (two times the assumed neighborhood distance of 300 m) is used to calculate the population living within the catchment of a POI.

Similarly, for the parking area, the catchment corresponds to a square bounding box of side 50 m. Here the distance threshold is selected to characterize short walking distance because parking far from the supermarket is discouraging for the customers (van der Waerden et al., 1998). We adopt the same point for all POI types, but in doing so, we ignore the effect of the POI type on the catchment distance, as it is treated uniformly for all POIs. To compute the average distance between a POI and transit stops, we identify the stops within a straight-line distance ("as the crow flies") 400 m of a POI. This selection of straight-line distance threshold could even result in walking distances greater than 400 m in some cases because the actual route length may be longer depending on the street network. Commonly transit agencies use a walking distance of 400 m as a thumb rule for measuring neighborhood accessibility and transit accessibility, as reflected in previous studies in accessibility research (Achuthan et al., 2010; Aultman-Hall et al., 1997). However, it is relevant to point out that walking behavior is determined by many factors such as trip purpose, built-up environment, mode type, and population demographics (Daniels & Mulley, 2013; Islam et al., 2019), and thus a detailed sensitivity analysis is beyond the scope of this paper. Average transit stop distance is the average distance of a POI from all identified stops. Finally, weather-specific features such as temperature and precipitation can also be relevant for studying demand (Horanont et al., 2013).

Non-spatial attributes are the POI type (e.g., supermarket or chemist); the number of reviews and ratings of a POI, e.g., the supermarket's temporal demand pattern, could differ from that of a fast-food outlet shown in (Capponi et al., 2019). Further, a POI with a high positive rating could imply its high likeability or customer satisfaction. Similarly, a large number of reviews by customers could be indicative of latent characteristics of a POI. These features, such as rating and the number of posted reviews, are also used in the demand trend modeling (Möhring et al., 2020; Timokhin et al., 2020). It is pertinent to mention that other demographic factors, such as average income in the locality, could also play an essential role in retail consumption (Rolph, 1932). However, this study did not consider the same because such a dataset was unavailable.

5.3.1 Data sources

To demonstrate our methodology, we turn to popularity trend data that have been used previously to predict venue popularity (Timokhin et al., 2020), calculate demand expansion factors (MacKenzie & Cho, 2020), classify activities (Capponi et al., 2019),

and investigate consumer behaviors (Möhring et al., 2020). These varied applications of POI demand patterns from popularity trends suggest their potential in other unexplored avenues, such as disruptive events. However, specific crowdsensed data, such as popularity trends, provide only relative or normalized values of the demand for certain activities in specific locations (Capponi et al., 2019). This is a limitation because it prevents one from inferring the corresponding number of absolute check-ins for which the data is recorded, and thus should be considered in analyzing and interpreting the results based on these data.

First, the POI data is collected from OSM via Overpass-Turbo (Raifer, 2020). Google's Popular time graph data (Google, 2020b) is collected as a measure of the demand patterns at all the identified POIs. A popular time graph shows the busyness (workload or saturation) of a POI during the day, relative to the busiest time during the week (Google, 2020b). Historical busyness is quantified on a relative scale of [0,100], where 100 indicates the busiest hour. This information is derived from the anonymized and aggregated data from the POI visitor's location history (Google, 2020b). As per Google, if the number of such users (who have opted for the location service) visiting a POI is insufficient, then the popular time graph and the place's live visit data may not be available (Google, 2020b). For a given POI, a Popular time graph is averaged over the last few months (Google, 2020b), which could be referred to as "historical popular time". Live visit data shows the popularity in real-time, which in some cases could be greater than 100 depending on its busyness or crowding relative to past trends. The popular time data for a particular POI is publicly viewable on the Google Maps website (GoogleMaps, 2020). Due to the smartphone-based passive data collection, Popular time data could also suffer from sample bias. As mentioned above, Popular time data is relative information and cannot infer the number of visitors without extrinsic information. Based on the above, we argue in this paper that live data could be an important indicator for measuring changes in the demand as, for each POI, they provide a measure of the deviation between the current and the average venue popularity.

Population data are obtained from the publicly available High-Resolution Population Density Maps provided by Facebook (2020). Facebook used state-of-the-art Computer vision techniques to process satellite imagery and generate this data. Population data provide human population distribution at a 30-meter spatial resolution. Parking area (size and locations) and transit stop locations are obtained from the OSM data [obtained via Overpass (Raifer, 2020)] and GTFS (2020), respectively. Python library OSMNX (Boeing, 2017) is used for processing and analyzing OSM data.

5.3.2 Modeling approach

The study examines the effect of the lockdown restrictions on the popularity of POIs. This is a problem of the causal inference framework, where lockdown acts as a treatment variable. With pre-treatment and post-treatment data, a preferred modeling approach based on a causal inference framework could be adopted by controlling for the treatment (lockdown) and confounding (day-of-the-week, POI type) variables. Herein we check the significance of covariates in explaining the day-specific popularity before and during

the lockdown. Möhring et al. (2020) and Timokhin et al. (2020) have also modeled popularity as a dependent variable in regression formulation. The dependent variable is the day-specific popularity of a POI, which is to be mapped to a set of explanatory variables, represented analytically as follows:

$$P_{i-d} = f(p_i, pa_i, sd_i, r_i, nr_i, type_i, L_d, D_d, T_d, Pr_d) + \epsilon_{i-d} \tag{5.1}$$

Where,

- P_{i-d} is the response variable in terms of popularity of i^{th} POI on day d
- p_i is population within the defined catchment of i^{th} POI
- pa_i is the total parking area within the defined catchment of i^{th} POI
- sd_i is the average distance to the transit stops within the defined catchment of i^{th} POI
- r_i is the rating of i^{th} POI
- nr_i is total number of reviews of i^{th} POI
- $type_i$ is the dummy variable of i^{th} POI type namely, supermarkets, chemists and fast-food
- L_d is the lockdown dummy variable, wherein during lockdown $L_d = 1$; for historical data $L_d = 0$
- D_d is the dummy variable representing the day of the week e.g., Monday, Tuesday, etc.
- T_d and Pr_d are weather-specific covariates for temperature and precipitation, respectively, on day d
- ϵ_{i-d} is the residual term

POI type (*type*), lockdown (L), and day (D) are categorical variables. They are used as dummy variables after one-hot encoding, e.g., for a supermarket POI, *supermarket* = 1, whereas *fast-food* and *chemist* are assigned 0 values. Similarly, during the lockdown, *lockdown* = 1; else *lockdown* = 0; and on a Monday, only *monday* = 1, while other day-of-the-week dummy variables are equal to zero.

Linear regression models are simple and intuitive as they help understand the features' average or global effects. However, these models depend heavily on the explicit analytical formulation and thus could introduce model bias. To counter this, we use regularized Gradient Boosting (GB) (T. Chen & Guestrin, 2016) for regression, inspired by previous studies (Timokhin et al., 2020). GB, a machine learning technique, is based on training weak learners in an additive manner. Unlike linear models, GB models do not need an

analytical specification and are less sensitive to outliers. GB can work well with small data while avoiding overfitting. Using regularized objective function in Equation 5.2 helps control overfitting. We refer to the regularized GB as Gradient Boosting Regression (GBR) model in this paper.

$$\mathcal{L}^{(t)} = \sum_{i=1}^{n} l\left(y_i, \hat{y}_i^{(t-1)} + f_t\left(\mathbf{x}_i\right)\right) + \Omega\left(f_t\right) \tag{5.2}$$

where $\mathbf{x}_i$ denotes the *i-th* instance of the dataset of size n; f_t is the current model fit; l is the loss function which measures the difference between the target (y_i) and the prediction $\hat{y}_i$, at *t-th* iteration; $\Omega(f)$ is the regularization term to check over-fitting. The details of the GBR are given in Friedman (2001) and T. Chen and Guestrin (2016).

The best GBR model is selected based on the lowest Mean Squared Error (Equation 5.3) on the training data (90% split), using 10-fold cross-validation. The main tunable parameters for the GBR model are the number of estimators and the tree depth (XGBoost, 2020). To handle overfitting, we check the MSE on the test data (10%) to ensure that training and test errors are close.

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - \hat{y}_i\right)^2 \tag{5.3}$$

The interpretation of tree-based models, like the GBR model, is not straightforward since single coefficients (as in linear regression models) for attributes are unavailable. There are many tools for global interpretation, i.e., the average impact of the features on the model output. Recently, work has been done on the local explanations of these models to uncover the role of each feature for every model instance. The combined behavior of these local explanations can also infer global behavior. In this regard, SHapley Additive exPlanations (SHAP) is a recently developed Python tool for explaining a machine learning model's outputs using the game-theoretic approach (Lundberg et al., 2020). TreeExplainer method from SHAP calculates classic Shapley values [a concept from the game theory (Shapley, 1988)] and assigns importance or credit to the input features based on their role in the particular model prediction (Lundberg & Lee, 2017). Similarly, local interaction effects are captured based on the Shapley interaction index from game theory by allocating the credit to a pair of features (Lundberg et al., 2020). A novel advantage of TreeExplainer is that it can compute Shapley values for tree-based models in polynomial time (Lundberg et al., 2020), which makes them highly efficient for practical applications. For details on SHAP, we refer the reader to Lundberg et al. (2020); Lundberg and Lee (2017).

We use Ordinary Least Squares (OLS) regression (linear regression), as a reference model for checking the consistency in the interpretation of the global effect of the features (Equation 5.4). It can be seen that, in addition to the main effects, we also include the interaction effects of lockdown (L_d) with all the other variables. It is pointed out that in the linear model, the coefficient (β_8) of *lockdown* gives the effect of lockdown on the

chemist POIs, conditional on the other covariates. Thus, the coefficient (β_8) actually represents interaction effect of *lockdown-chemist*. We do not include *chemist* dummy explicitly in Equation 5.4, as it is highly negatively correlated with *supermarket*.

$$\begin{aligned} P_{i-d} = \beta_0 + \beta_1 \cdot p_i + \beta_2 \cdot pa_i + \beta_3 \cdot sd_i + \beta_4 \cdot r_i + \beta_5 \cdot nr_i + \beta_6 \cdot supermarket_i + \\ \beta_7 \cdot fast{-}food_i + \beta_8 \cdot L_d + \beta_9 \cdot Monday_d + \beta_{10} \cdot T_d + \beta_{11} \cdot Pr_d + \\ \beta_{12} \cdot L_d \cdot p_i + \beta_{13} \cdot L_d \cdot pa_i + \beta_{14} \cdot L_d \cdot sd_i + \beta_{15} \cdot L_d \cdot r_i + \beta_{16} \cdot L_d \cdot nr_i + \\ \beta_{17} \cdot L_d \cdot supermarket_i + \beta_{18} \cdot L_d \cdot fast{-}food_i + \beta_{19} \cdot L_d \cdot Monday_d + \\ \beta_{20} \cdot L_d \cdot T_d + \beta_{21} \cdot L_d \cdot Pr_d + \epsilon_{i-d} \end{aligned} \tag{5.4}$$

We also use the Robust regression or Robust Linear Model (RLM) or M-Estimation with Huber objective function (Huber, 1973). This objective function uses two different formulations: least squares (in the center) and least absolute values (in the tails), basically underweighting the high-influence observations or outliers in the dependent variable. Finally, it is noteworthy to refer to a recently published study using the GBR and linear regression for modeling and SHAP to explain building energy performance (Arjunan et al., 2020) due to inherent similarity in our modeling approaches. We develop the above models using Python libraries statsmodels (2020) and XGBoost (2020).

5.4 Data collection and processing

We select Munich (the Free State of Bavaria's capital city in Germany) as the study area. Even though many countries in the world are affected by COVID-19, the extent of the impact depends on multiple factors, such as first COVID-19 incidence (Böhmer et al., 2020), rate of spread, travel restrictions (Chinazzi et al., 2020), testing and contact tracing and containment (Lorch et al., 2020), amongst many others. Therefore, the data for before-during scenarios were collected based on the restriction or lockdown timeline. In Germany, the need for social distancing was announced on 12.03.2020, followed by the announcement of the temporary closure of schools on 14.03.2020 and the non-essential travel ban on 18.03.2020 (Robert Koch Institute & Humboldt University of Berlin, 2020). The Federal States took up state-specific measures depending on their needs, such as imposing a full lockdown in Bavaria on 20.03.2020 (Robert Koch Institute & Humboldt University of Berlin, 2020). Therefore, it can be concluded that the second and third weeks of March were the transition period from pre-lockdown to the lockdown period. We are also interested in exploring how the demand pattern at a POI evolves during different stages of the lockdown (e.g., during the early lockdown in the third week of March viz-à-viz during the late lockdown in the last week of April).

3283 POIs were initially identified in the bounding box around Munich [Latitude: 48.137585±0.1125, Longitude: 11.575444±0.175]. The POI attributes, namely location (latitude and longitude), type, rating (on a scale of 1-5), and the number of reviews, are collected. For these POIs, we use the Python library (m-wrzr & riedmaph, 2018) to obtain hourly historical data (Table 5.1). The live data is collected bi-hourly, e.g.,

Table 5.1: Popular time data collection

Date	Cumulative COVID-19 Cases[a]	Type of data (Period)	Description
13-02-2020	-	Historical average (0000-2400)	Before lockdown
20.03.2020	878	Live (1200-1400)	Start of lockdown
03.04.2020	3304	Live (1200-1400)	Middle of lockdown
14.04.2020	4714	Live (0800-2000)	Late lockdown
22.04.2020	5332	Live (0800-2000)	Late lockdown
27.04.2020	5607	Live (0900-2100)	Late lockdown

[a] *StadtMuenchen* (2020)

Table 5.2: Number of identified POIs with historical data and live data

POI type	with historical data	with live data[a]
Supermarkets	262	137
Fast-food	170	8
Chemist	73	35

[a]live data availability varies per day

1200 H, 1400 H, 1600 H, etc. Not all of these POIs are found to have live popularity information during the lockdown, possibly due to the temporary closure of such POIs due to restrictions or insufficient users visiting such POIs. The availability of the live data varies per hour-day. Therefore, POIs without live data during 1200-1800 hours are dropped for the subsequent analysis and modeling. The analysis period of 1200-1800 is chosen to represent the consistent working time for all the POIs, away from the opening (around 0800-1000 H) and closing hours (1900-2000 H). Only POI types with at least five samples are selected to ensure representativeness, which leaves a total of about 180 POIs for three categories (Figure 5.1), namely supermarket, fast-food, and chemist (Table 5.2). The low number of POIs makes sense because several retail and leisure POIs, such as restaurants, stores, and barber shops, were closed and severely affected due to the lockdown restrictions, and that is why we suppose no popular time data were available for such POIs.

5.5 Data analysis

The hourly trends of average historical popularity in three types of POIs, namely supermarkets, chemists, and fast-foods, are shown in Figure 5.2. In the historical trend, supermarkets show a prominent peak during the evening hours, coinciding with the evening commute. Chemists also show a similar pattern. The trend is absent on Sundays, as most supermarkets and chemists are closed on Sundays in Munich. The fast-food category trend shows two prominent peaks during the weekdays, which can be attributed

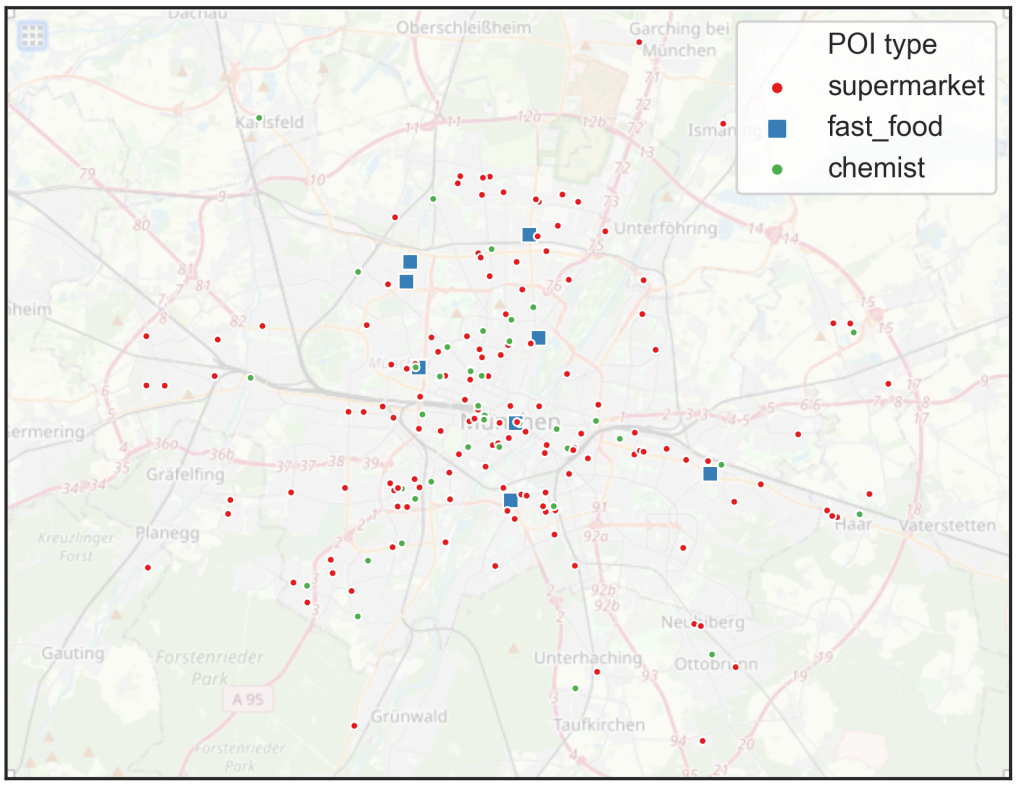


Figure 5.1: Spatial distribution of POIs with Live data. Fast-food POI's symbol is enhanced for better visibility. Basemap source: OpenStreetMaps.

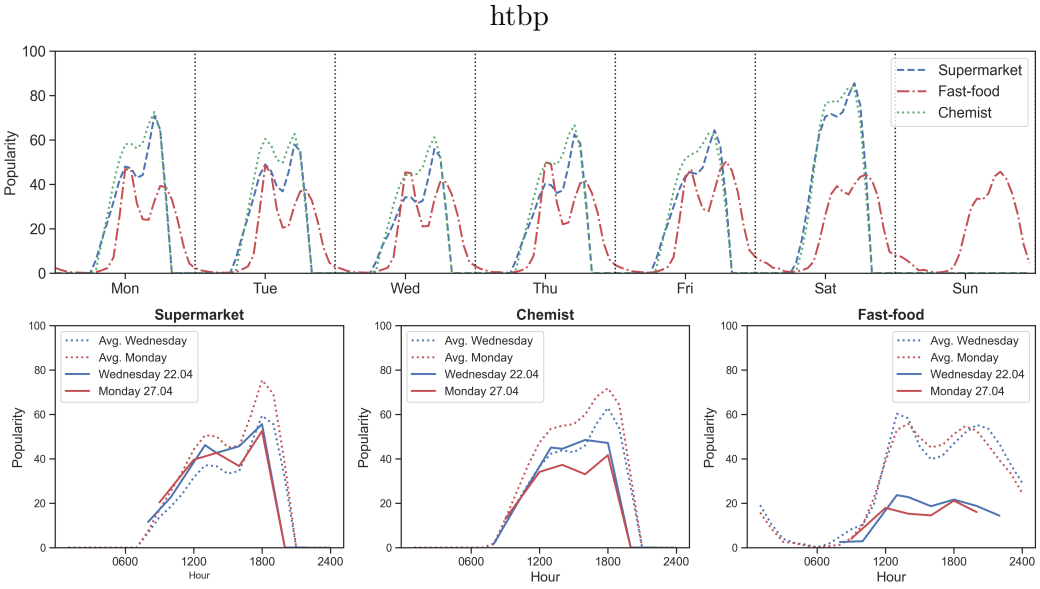


Figure 5.2: (Top) Historical average popular time trend and (bottom) live popular time trend for the three POI types.

to busyness during lunch and late-evening hours. The demand trend on the weekend shows a high demand from lunch to late-evening hours.

Figure 5.2 shows the average live trends on two days of the week during the lockdown, 22nd April 2020 (Wednesday) and 27th April 2020 (Monday). Compared to the average historical popularity, the drop in the peak popularity and the general trend is evident. Interestingly, the drop in the fast-food category is more significant and characterizes the lockdown's adverse effect on similar POIs. It can also be recognized that the shape of the historical popularity trend differs on Monday and Wednesday for supermarkets and chemists, indicating variations during the week. The trend of the afternoon (1400 H) popularity on a few selected weekdays also shows the effect of lockdown on the three POI categories (Figure 5.3). Again, supermarkets and chemists show similar trends with average historical popularity at around 40-50 % of maximum popularity, but markedly increasing on 20th March, i.e., the day lockdown was announced. This increase (57 % for supermarkets and 10 % for chemists) could result from panic buying for groceries and health retail because of the uncertainty in the early days of the lockdown and pandemic. During the later lockdown period in April, a gradual recovery of the popularity of the supermarkets and the chemists' POIs is observed. The fast-food category trend is distinct by a fall in its popularity, which did not wholly recover in April, although it shows small signs of recovery. It can also be seen that there is no panic buying in the fast-food category on the day of the lockdown announcement, unlike the other two types of POIs.

The summary of the explanatory variables is given in Table 5.3. The parking area locations in OSM correspond to different parking types, such as surface, multi-level, and underground parking. The composition of the parking areas in our sample is surface

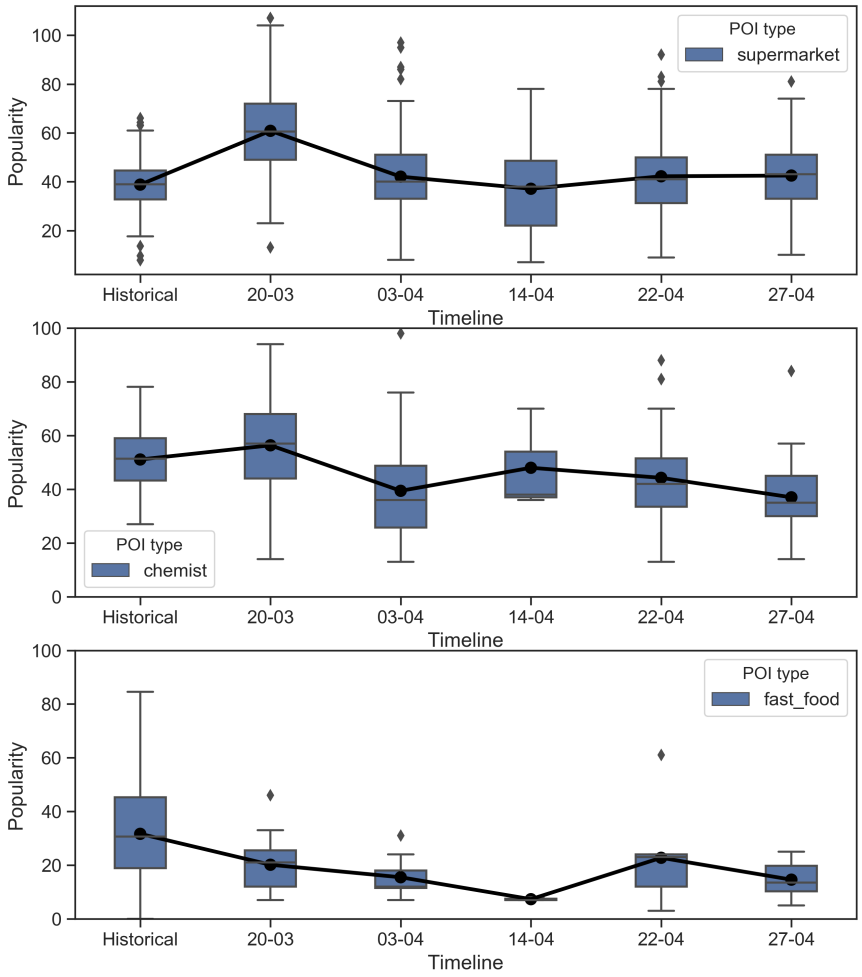


Figure 5.3: Live popular time trend at 1400 H on different days during the lockdown.

Table 5.3: Summary of the explanatory variables

Statistic	Population (<300 m)	Parking area m^2 (<50 m)	Transit stops (<400 m)	Avg. Stop distance (m)	Rating (1-5)	Reviews
minimum	195	0	1	105.2	2.5	8
mean	2614	414.3	8	324.8	3.9	371
maximum	4340	4503.7	28	794.0	4.9	4742
σ	727	850.1	5	87.4	0.3	551

(70%), underground (3%), multi-story (1%), and missing label (26%). We use the historical data and live data (on 22.04.2020 and 27.04.2020) for modeling pre-lockdown and during-lockdown scenarios, respectively (Table 5.1). The response variable in the regression models is the average of the popularity at two-hour intervals over the period of 1200-1800 H, as follows:

$$P_{i-d} = (P_{i-d}^{1200} + P_{i-d}^{1400} + P_{i-d}^{1600} + P_{i-d}^{1800})/4 \tag{5.5}$$

where, P_{i-d}^{t} is the popularity at time t. The features such as rating and the number of reviews change with time as new users rate and review a specific POI. In our case, the change is marginal, i.e., the mean percentage change in rating and reviews during the analysis period is 0.2% and 0.0%, respectively. We do not control the weather-specific covariates due to the panel's limited dimension (two days of live data). The weather for these two days was similar as characterized by sunny or partial cloudy (Time and Date, 2020), which makes it reasonable to not control for weather-specific covariates. With sufficient panel data, we recommend controlling for weather covariates for precise model estimation.

5.6 Results

Using cross-validation, we identify the best parameters for the GBR model (number of estimators: 20, maximum tree depth: 4). With these parameters, the model achieves an R^2 of 0.63. The Mean Squared Errors (MSEs) obtained on the training (7.4) and test data (9.6) are close, which implies no over-fitting. In the SHAP summary plot (Figure 5.4), the feature impact on the output of the GBR model is shown with the distribution of SHAP values. In these plots, each point corresponds to one POI instance in the dataset and the corresponding SHAP values of the features. The color represents the feature value (blue for low value and pink for high value). The features in these sub-plots are ordered by the sum of the SHAP values' magnitude over the training dataset. If high SHAP values are observed for corresponding high values of the feature, it means an increase in that particular feature results in an increase in popularity and vice-versa. If SHAP values for a feature are concentrated near 0, that particular feature does not play much importance in predicting its popularity.

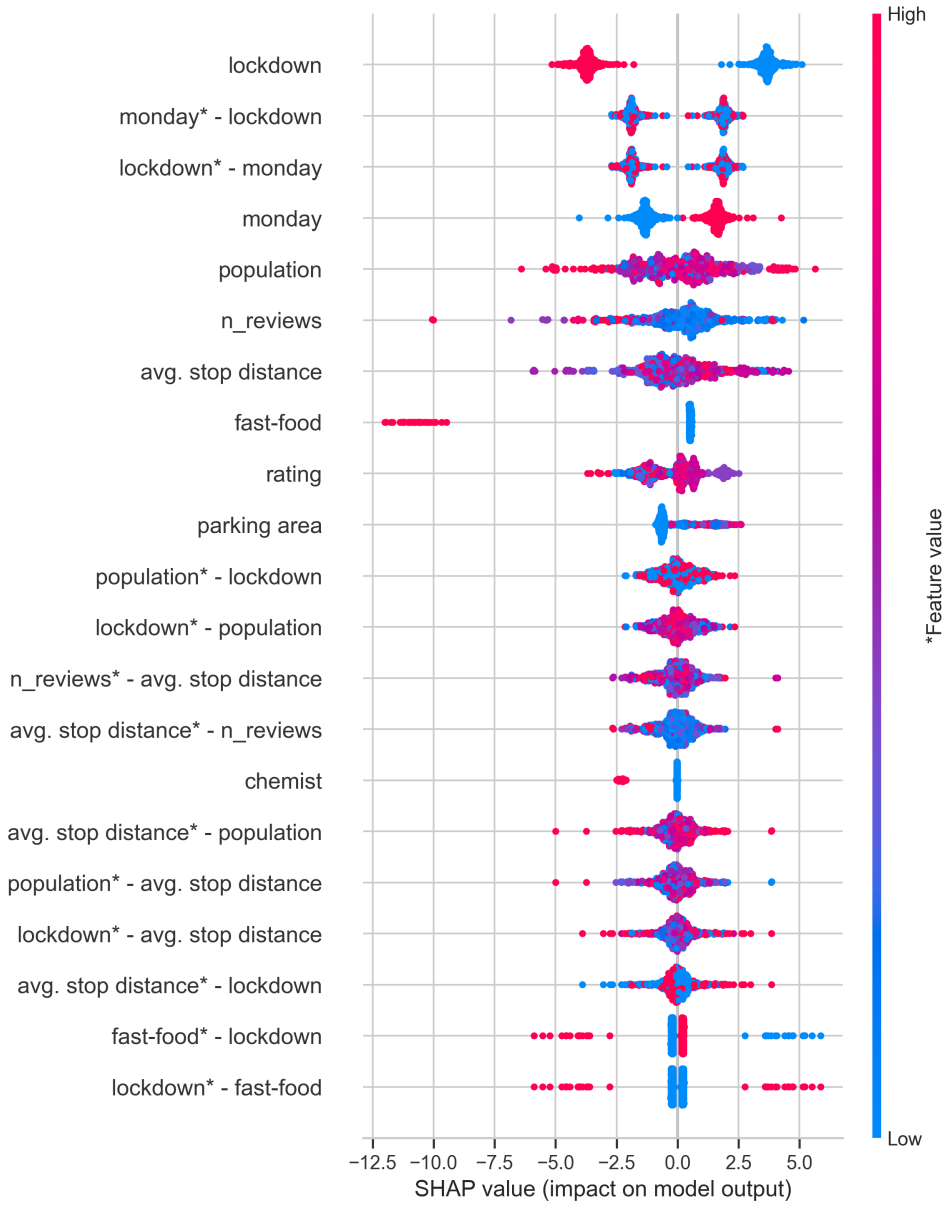


Figure 5.4: Feature impact based on SHAP values for the 15 largest main and interaction effects. (Lundberg et al., 2020)

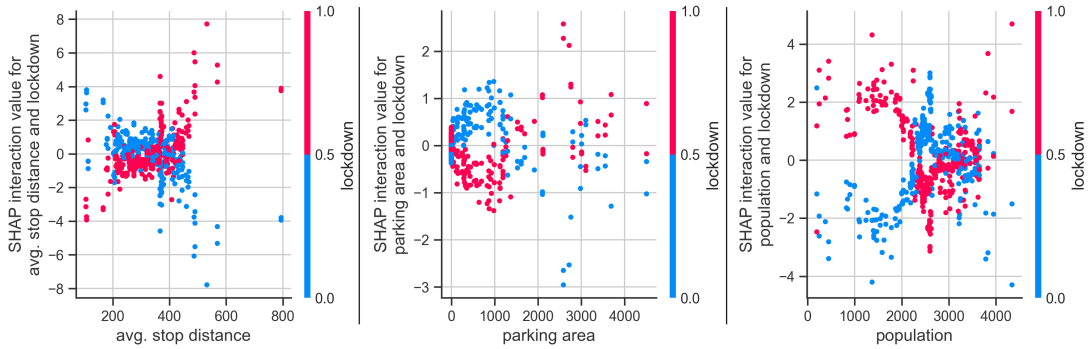


Figure 5.5: SHAP dependence plot based on local explanations for the spatial features' interaction with the lockdown.

The main effects of *lockdown*, *monday*, and *fast-food* are clear due to distinct distribution SHAP values for low and high feature values. The *lockdown* feature is found to be correlated with the drop in popularity. The popularity on Monday is found to be higher than that on Wednesday. It is interesting to note that the POI type plays an important role, especially for fast foods. The *fast-food* attribute is found to be correlated with low SHAP values (i.e., *fast-food* =1), which pushes the popularity to the lower side. The impact of the *population* and the number of reviews (*rating_n_x*) is not clearly correlated with the popularity value, as evident by overlapping pink and blue points. The low values of the *parking area* feature show low SHAP values, whereas high *parking area* is associated with high SHAP values (albeit with some overlap); i.e., it pushes the POI popularity to the higher side. Similarly, the type of POI, namely *chemist*, is correlated with the decrease in popularity, as evident from negative SHAP values. Hence, the features viz. *lockdown*, *day-of-the-week*, *POI-type*, and *parking area* show a clear correlation with popularity.

Figure 5.4 also shows the interaction effects, where the superscript * indicates which feature is represented by the color bar. The interaction effects of *lockdown* and *fast-food* features also show clear effects, implying the adverse effect of lockdown on the fast-food POIs in terms of popularity, also seen in Figure 5.2. Spatial factors, *population*, and *avg. stop distance* are found to have mixed effects (overlap of pink and blue points), and thus their global effects on popularity are not clear in Figure 5.4. However, the interaction effect of *lockdown* - *avg. stop distance* (see feature *avg. stop distance** - *lockdown*) shows high SHAP values for some longer stop distances, and vice-versa. This effectively means that POIs close to the transit stops had lower popularity than those farther from a stop during a lockdown. This is even clearer in the local explanation plot in Figure 5.5, wherein the interaction effects of *lockdown* - *avg. stop distance* are inverted during the lockdown.

The fit of the OLS and RLM models is not as good as that of the GBR model, as evidenced by lower Adjusted R^2 values (Table 5.4), which also justifies the use of the GBR model as it introduces less bias as compared to the linear models. Nevertheless,

Table 5.4: Results of Linear Regression

	Dependent variable: P_{i-d}	
	1:OLS	2: RLM
Intercept	48.74***	53.37***
	(6.93)	(7.04)
fast-food	-2.56	-3.57
	(3.75)	(3.81)
lockdown	3.66	-3.50
	(9.82)	(9.98)
lockdown:fast-food	-19.47***	-16.73***
	(5.31)	(5.39)
lockdown:monday	-16.45***	-16.24***
	(1.52)	(1.55)
lockdown:average stop distance/1000	19.49**	20.19**
	(8.93)	(9.07)
lockdown:number of reviews/1000	0.13	-0.97
	(1.88)	(1.91)
lockdown:parking area/1000	0.18	0.13
	(0.95)	(0.96)
lockdown:population/1000	-1.93*	-1.93*
	(1.10)	(1.12)
lockdown:rating	-1.68	-0.10
	(2.26)	(2.30)
lockdown:supermarket	4.53**	4.84**
	(2.02)	(2.05)
Monday	11.20***	11.29***
	(1.08)	(1.09)
average stop distance/1000	-10.52*	-11.40*
	(6.32)	(6.42)
number of reviews/1000	-1.52	-1.41
	(1.33)	(1.35)
parking area/1000	0.67	0.64
	(0.67)	(0.68)
population/1000	1.18	1.20
	(0.78)	(0.79)
rating	-0.40	-1.41
	(1.60)	(1.62)
supermarket	-1.96	-2.12
	(1.43)	(1.45)
Observations	718	718
R^2	0.34	
Adjusted R^2	0.32	
Residual Std. Error	10.18	3.21
F Statistic	21.20***	

*p<0.1; **p<0.05; ***p<0.01

the results of the OLS model (the sign and magnitude of the coefficients) show that the average behavior of the features is consistent with that of the GBR models. The intercept term is significant in the OLS and RLM models, with a value close to 50. The main effect of *lockdown* is not found to be significant, unlike in Figure 5.4. In the linear model (Equation 5.4), it represents the interaction of *lockdown-chemist*. The main effects of only *monday* and *average stop distance* are found to be significant. The positive coefficient of *monday* shows that the average historical popularity of POIs on Monday is more than that on a Wednesday due to variations in the daily demand patterns (Figure 5.2). The negative coefficient of the *average stop distance* implies that popularity decreases with increased distance to a transit stop. The *lockdown* has significant interactions with other features. The popularity during lockdown depends on the type of POI, as the interaction of *fast-food* type POI has a greater negative coefficient than the other two types of POIs, whereas the popularity of the supermarkets is marginally greater than the chemist type POIs. During the lockdown, Monday's popularity is lower than on Wednesday, which provides evidence of the daily temporal variations in popularity during the lockdown. Interestingly, popularity is positively correlated with the *lockdown - avg. stop distance* interaction. A possible explanation is a drop in transit ridership during the lockdown, as shown in Figure 2.4. Specifically, passenger ridership dropped by around 70% during April 2020 in Munich [Münchner Verkehrsgesellschaft mbH (MVG) (2021)], and thus POIs closed to transit stops observed a greater reduction in popularity than others located far from the transit stops. The *lockdown-population* interaction also has a negative coefficient, albeit with a weak significance. One thing to note is that in the linear models, the main and interaction coefficients of *parking area* are not found to be significantly correlated.

5.7 Summary

We analyzed the demand patterns at POIs in Munich during special events, such as the COVID-19 lockdown, using publicly available crowdsensed data. Demand patterns uncovered in this study match the expectations of viz-à-viz restrictions during the COVID-19 lockdown in Munich. We explained the effect of features in the GBR regression model using SHAP. The behavior of coefficients is consistent with previous studies to some extent, wherein transit stop connectivity is associated with the demand at retail locations (Rolph, 1932; van der Waerden et al., 1998). Significance of *POI type* (fast-food) during COVID-19 confirms the dominance of POI type in explaining the lockdown impact, possibly as the lockdown was directed to reduce non-essential retail consumption and crowding. POI types are significant in explaining the dip in the POI's popularity, as *POI-type* captures latent consumer behavior.

This marks the end of Part II of this dissertation. In the next Part III, we focus on developing efficient methods to address the data insufficiency challenges in traffic prediction and calibration.

Part III

Efficient methods to tackle data scarcity

6 Ensembling and heuristics for efficient traffic simulation calibration

Contents

The content of this chapter has been presented in the following work:

> Mahajan, V., Cantelmo, G., & Antoniou, C. (in review). One-shot heuristic and ensembling for automated calibration of large-scale traffic simulations. In Review. Retrieved from https://mediatum.ub.tum.de/doc/1701188/document.pdf

6.1 Introduction

In this chapter, we propose an efficient methodology using conventional data for calibrating large-scale traffic simulations. We address the challenges in Origin-Destination (OD) estimation in a unified methodology with simple heuristics, ensemble techniques, and Bayesian optimization. In this pursuit, we apply simple yet effective heuristics and ensemble techniques (borrowed from the machine learning field) to demand (OD estimation) and supply calibration. Using multiple experiments, we show that ensembling effectively reduces the variance in the final demand estimates. Further, the averaged estimates are much closer to the true or desired estimates and, thus, use the results of multiple local optimizers to land closer to the desired solution. In addition, we propose automatic tuning of Simultaneous Perturbation Stochastic Approximation (SPSA), a gradient approximation algorithm, and thus reduce the manual effort and time spent in doing so hitherto.

The remainder of the chapter is structured as follows: in the following section, we list the contributions of this chapter, followed by an introduction to indirect OD estimation and supply calibration, followed by the methodology of our study, followed by details on experimental design and calibration platform description, followed by a section on results. Finally, we summarize the findings of this chapter.

6.2 Research contributions

In this work, we address the challenges in OD estimation in a unified methodology with simple heuristics, ensemble techniques, and Bayesian optimization. Our contributions are summarized as follows:

- We develop a methodology to fine-tune the calibration algorithm parameters automatically. Substantial research shows that these hyperparameters play a crucial role, but they are usually determined manually, which is time-consuming and unreliable. To the author's knowledge, no methodology exists to estimate them automatically. Our method helps to push the calibration process towards an automated approach. [SRQ(7)]

- We find that applying Bagging and Stochastic Parameter Averaging (SPA) techniques can improve the robustness of the results. This is important since, typically, solutions obtained by local search calibration algorithms have high variance, and these ensemble techniques can help to reduce such variance in the estimates.[SRQ(8)]

- We also provide two additional contributions, which from a methodological standpoint, are minor, but have substantial impacts on the calibration output in practice. First, we develop a one-shot heuristic system that reduces intrinsic bias, reducing computational time. Second, we apply a Bayesian optimization framework that effectively estimates the supply parameters.

- The above approaches are developed using open-source tools and software and made available[1] to advance the research in traffic simulation calibration.

6.3 Indirect OD estimation

6.3.1 Problem formulation

The offline calibration problem can be formulated using the notation in Table 6.1, inspired by Antoniou et al. (2015). Indirect Dynamic Origin-destination Demand Estimation (DODE) is a specific case of transport demand calibration where values of time-dependent OD matrices are the demand calibration parameters. This can be formulated as follows:

$$\underset{X,Y}{\text{minimize}} \quad L\left(\boldsymbol{M^o}, \boldsymbol{M}^s, \boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{X}^a, \boldsymbol{Y}^a\right) \tag{6.1}$$

which can be operationalized as follows:

$$\underset{\boldsymbol{X},\boldsymbol{Y}}{\text{minimize}} \sum_{t=1}^{T} \left[\boldsymbol{w_1} Z_1\left(\boldsymbol{M_t^o}, \boldsymbol{M_t^s}\right) + \boldsymbol{w_2} Z_2\left(\boldsymbol{X_t}, \boldsymbol{X_t^a}\right) + \boldsymbol{w_3} Z_3\left(\boldsymbol{Y_t}, \boldsymbol{Y_t^a}\right)\right] \tag{6.2}$$

subject to:

$$\boldsymbol{M_t^s} = f\left(\boldsymbol{X}_1, \ldots, \boldsymbol{X}_t; \boldsymbol{Y}_1, \ldots, \boldsymbol{Y}_t; \boldsymbol{G}\right) \tag{6.3}$$

$$\boldsymbol{l_x} \leq \boldsymbol{X} \leq \boldsymbol{u_x} \qquad \boldsymbol{l_y} \leq \boldsymbol{Y} \leq \boldsymbol{u_y} \tag{6.4}$$

and Z measures the discrepancy between the two quantities and is called Goodness of Fit (GOF) function or distance metric. In the case of measurements, the two quantities are the simulated and observed measurements, whereas, in the case of parameters, they are the parameter's current value and the parameter's prior value. Equation 6.2 is a type of multi-objective optimization, and $\boldsymbol{w_1}$, $\boldsymbol{w_2}$, and $\boldsymbol{w_3}$ are the assigned weights for these objectives. Equation 6.3 captures the dependence between simulated outputs and the input parameters, which is directly obtained from the Dynamic Traffic Assignment (DTA) traffic simulator.

Z_2 contributes to the discrepancy of the current estimates from the initial or historical demand estimates, so the optimization algorithm is penalized for exploring far from the initial OD demand values. Furthermore, if the initial values are biased, dependence on initial values in the objective function can prevent the optimization algorithm from reaching the desired optimum. In other words, a misleading specification of OD prior will restrict the algorithm from recovering the desired values. The same is true for prior values of supply parameters. Thus, when prior parameters are heavily biased or unreliable, Z_2 and Z_3 should be set to a small value. But still, the prior demand matrix has certain structural information, such as the relative magnitude of the demand flows among the zones. Prior information about parameters needs to be provided to narrow down the possible solutions.

[1] `https://github.com/vishalmhjn/actrys`

Table 6.1: Symbols used in the chapter

Symbol	Description
T	Number of time intervals
ΔT	Duration of each time interval
X	Time-dependent demand parameters, e.g., time-dependent OD flows in our case, $X = \{X_t\} \forall t \in T$. In this work, we use the terms dynamic OD matrix and demand parameters interchangeably since they are identical.
$\boldsymbol{X}^a$	A priori or initial or given time-dependent parameter values, $\boldsymbol{X}^a = \{\boldsymbol{X}_t^a\}$
p	Number of OD pairs
Y	Selected supply parameters
Y^a	A priori or initial of selected supply parameters
q	Number of supply parameters
G	Road network and other fixed supply parameters, $G = \{G\}$
f	Traffic simulation model
M^o	Observed time-dependent sensor measurements, $M^o = \{M_t^o\}$, e.g., $M_t^o = \{C_t^o, V_t^o\}$ for count C and speed V measurements
M^s	Simulated time-dependent measurements, $M^s = \{M_t^s\}$, e.g., $M_t^s = \{C_t^s, V_t^s\}$ for count C and speed V measurements
m	Number of link measurements
Z_1, Z_2, Z_3	Goodness of fit function between simulated and observed measurements, simulated and prior OD estimates, simulated and prior supply parameters, respectively
$\boldsymbol{w_1}, \boldsymbol{w_2}, \boldsymbol{w_3}$	Decision weights for error functions Z_1, Z_2, Z_3, respectively in the multi-objective optimization
L	Weighted overall objective function
B^x	Bias factor for OD matrices X
R^x	Randomness factor for OD matrices X
u	Acquisition function for Bayesian optimization
$\boldsymbol{A}$	OD flow-Link counts assignment matrix
W	Weight-matrix for W-SPSA, $W = J(\boldsymbol{A})$, where J is a non-linear function
$w_{cut-off}$	threshold value below which the correlation is set as zero
$w_{round-off}$	boolean variable, if True, then the non-zero correlation between the parameter and the sensor is set to 1
a, c	SPSA gain coefficients
A, γ, δ	other SPSA parameters
K, S, B, E	Number of iterations for W-SPSA, sequential calibration, Bayesian optimization, and ensembles, respectively
τ	Error level, which is acceptable and hence defines successful convergence

Since calibration is a constrained optimization problem (Equation 6.4), we must specify the domain of the decision variables, i.e., values in the OD matrices. The equation 6.4 specifies the domain of the demand and supply parameters; if the domain for the demand variables is wide, the local search algorithm has more flexibility to find solutions, leading to a higher variance in the results. On the other hand, narrow domain specification restricts the search space. These constraints help provide additional information to the optimization algorithm regarding the search space of parameters.

6.3.2 Stochastic search and approximation using SPSA

6.3.2.1 Stochastic Approximation

Equation 6.2 is a form of an iterative optimization problem where the analytical form of the objective function is unknown. To handle this, we move to Stochastic Approximation (SA), which is a family of iterative stochastic optimization algorithms used for the minimization of objective functions without an analytical form. Such objective functions can only be estimated from noisy observations or noisy function evaluations, such as in black box systems. In black box systems, only inputs and outputs can be viewed but not the inner mechanism of the system (Bunge, 1963). A general form of SA is:

$$\hat{\theta}_{k+1} = \hat{\theta}_k - a_k \hat{g}_k(\hat{\theta}_k) \tag{6.5}$$

where $\hat{\theta}_k$ is the decision vector for the k^{th} iteration and $\hat{g}_k(\hat{\theta}_k)$ is the estimate of gradient at $\hat{\theta}_k$. a_k is the step size or gain sequence. There are different approaches to estimating the gradient of the objective function from limited observations or function evaluations. The naïve gradient estimation can be done using finite differences; the gradient is estimated by perturbing the parameters in the decision vector sequentially, i.e., one at a time, evaluating the objective function twice as many times as there are the number of parameters, and estimating the gradient. Sequential perturbation of the elements of decision vector and function evaluation at those points has a high time complexity due to the large number of parameters and long run-time of large-scale traffic simulators.

6.3.2.2 Simultaneous Perturbation Stochastic Approximation (SPSA)

SPSA, by Spall (1998a, 1998b), is a gradient approximation-based optimization algorithm for stochastic optimization. In SPSA, the gradient is approximated by perturbing all the parameters simultaneously. This leads to only two function evaluations of the objective function per gradient evaluation. SPSA reduces the computation time by order of p, where p is the number of dimensions or, in our case, the number of OD parameters. Due to this advantage, SPSA is favored for simulation-based OD estimation since function evaluation is expensive and the number of OD parameters is large (of the order of thousands). The gradient vector in SPSA is approximated as follows:

$$\hat{g}_k\left(\hat{\theta}_k\right) = \frac{L\left(\hat{\theta}_k + c_k\Delta_k\right) - L\left(\hat{\theta}_k - c_k\Delta_k\right)}{2c_k\Delta_k} = \frac{L\left(\hat{\theta}_k^+\right) - L\left(\hat{\theta}_k^-\right)}{2c_k\Delta_k} \tag{6.6}$$

where $\hat{\theta}_k^+ = \hat{\theta}_k + c_k\Delta_k$, and $\hat{\theta}_k^- = \hat{\theta}_k - c_k\Delta_k$. Gain sequences are given by $c_k = c/(k+1)^\gamma$ and $a_k = a/(A+k+1)^\alpha$, where c, γ, a, α and A are the SPSA parameters. c and a can be scaled relative to the magnitude of the θ. The magnitude of gain sequences reduces with k. Δ_k is a random perturbation vector sampled from the Bernoulli distribution with values of $+1$ and -1 with equal probabilities.

6.3.2.3 Weighted - SPSA (W-SPSA)

SPSA does not account for domain information and parameter correlations while propagating gradients from objective function to parameters. Thus, various extensions of SPSA for DODE are proposed in the literature, as discussed in Section 2.5.3. Of the proposed extensions, the W-SPSA exploits the simulator knowledge to map the correlations of the gradients with the parameters. W-SPSA (Antoniou et al., 2015; Lu et al., 2015) uses a weight matrix to account for the correlation of the errors in MOP with the parameters (OD flows) during gradient approximation. This enables the use of information from the traffic simulator to discard the gradient signal from uncorrelated measurements. W-SPSA can also be seen as splitting the original problem into multiple smaller SPSA problems (Antoniou et al., 2015). To show how W-SPSA works, we re-write the loss function (Equation 6.2) by omitting the constants (observed measurements and prior values of the parameters), using θ to denote the demand and supply parameters, and setting $w_2 = w_3$, $Z_2 = Z_3$, and $P = p + q$ for the sake of verbosity:

$$L(\theta) = \sum_{t=1}^{T} \left[\boldsymbol{w_1} Z_1\left(f(\boldsymbol{\theta})\right) + \boldsymbol{w_2} Z_2\left(\boldsymbol{\theta}\right)\right] \tag{6.7}$$

Now, the additive elements of L can be arranged in a $(m+P)T$ array $\boldsymbol{\mathcal{Z}}$:

$$\boldsymbol{\mathcal{Z}} = \left[\begin{array}{cccccccc} \boldsymbol{w_1} z_{1,1}(\theta) & \dots & \boldsymbol{w_1} z_{1,mT}(\theta) & \boldsymbol{w_2} z_{2,mT+1}(\theta) & \dots & \boldsymbol{w_2} z_{2,(m+P)T}(\theta) \end{array}\right] \tag{6.8}$$

Where z corresponds to the element-wise error function for each parameter or measurement. The gradient estimation in W-SPSA makes use of the correlation between parameters and measurements based on the following $(PT \times (m+P)T)$ dimensional matrix :

$$\boldsymbol{W} = \left[\begin{array}{ccccccccc} w_{1,1} & w_{1,2} & \dots & w_{1,m} & \dots & w_{1,mT} & \dots & w_{1,(m+P)T} \\ w_{2,1} & w_{2,2} & \dots & w_{2,m} & \dots & w_{2,mT} & \dots & w_{2,(m+P)T} \\ \vdots & \vdots & & \vdots & & \vdots & & \vdots \\ w_{P,1} & w_{P,2} & \dots & w_{P,m} & \dots & w_{P,mT} & \dots & w_{P,(m+P)T} \\ \vdots & \vdots & & \vdots & & \vdots & & \vdots \\ w_{PT,1} & w_{PT,2} & \dots & w_{PT,m} & \dots & w_{PT,mT} & \dots & w_{PT,(m+P)T} \end{array}\right]$$

where $w_{i,j}$ is the correlation of i^{th} parameter with j^{th} measurement or parameter. Note that these weights $w_{i,j}$ are different from the weights of multi-objective optimization (as in the Equation 6.2), which are denoted by bold symbol $\boldsymbol{w}$. The gradient calculation steps for the i^{th} parameter can be written as follows:

$$\hat{g}_{ki}\left(\hat{\boldsymbol{\theta}}_k\right) = \frac{\sum_{j=1}^{(m+P)T} w_{ij}\left[\mathcal{Z}_j^+ - \mathcal{Z}_j^-\right]}{2c_k\Delta_{ki}} = \frac{1}{2c_k\Delta_{ki}}\boldsymbol{W}_i^\top\left[\boldsymbol{\mathcal{Z}}^+ - \boldsymbol{\mathcal{Z}}^-\right] \tag{6.9}$$

where $\boldsymbol{W}_i$ is the i^{th} row of the weight matrix, and

$$\boldsymbol{\mathcal{Z}}^+ = \left[\begin{array}{cccccccc} \boldsymbol{w_1}z_{1,1}(\theta^+) & \dots & \boldsymbol{w_1}z_{mT}(\theta^+) & \boldsymbol{w_2}z_{mT+1}(\theta^+) & \dots & \boldsymbol{w_2}z_{(m+P)T}(\theta^+) \end{array}\right] \tag{6.10}$$

$$\boldsymbol{\mathcal{Z}}^- = \left[\begin{array}{cccccccc} \boldsymbol{w_1}z_{1,1}(\theta^-) & \dots & \boldsymbol{w_1}z_{mT}(\theta^-) & \boldsymbol{w_2}z_{mT+1}(\theta^-) & \dots & \boldsymbol{w_2}z_{(m+P)T}(\theta^-) \end{array}\right] \tag{6.11}$$

It can be seen that the gradient for each parameter is computed differently (Equation 6.9) in W-SPSA instead of a single gradient value for all parameters as in the case of SPSA (Equation 6.6). The gradient matrix for all the parameters can be written as follows:

$$\hat{G}_k = \frac{1}{2}\boldsymbol{W}^\top\left[\boldsymbol{\mathcal{Z}}^+ - \boldsymbol{\mathcal{Z}}^-\right] \oslash c_k\Delta_k \tag{6.12}$$

where $\oslash$ is the operator for the element-wise division of matrices, for further details on W-SPSA, we refer the reader to Antoniou et al. (2015); Lu et al. (2015). Finally, momentum can be used with W-SPSA to obtain the running average of the gradients across iterations for efficient convergence. Thus, the update step (Equation 6.5) can be replaced with the following:

$$\begin{aligned} v^{k+1} &= \beta v^k - a_k\hat{g}_k \\ \theta^{k+1} &= \theta^k + v^{k+1} \end{aligned} \tag{6.13}$$

Where β is the momentum factor with a value between 0 and 1.

6.4 Methodology

6.4.1 Overview

The complete methodological framework for off-line calibration is summarized in Figure 6.1. The figure shows the application of the bias-correction heuristic on the initialized parameters. This is followed by automatic SPSA parameter tuning and, finally, ensembling of W-SPSA with sequential demand calibration and supply calibration (only in case of real data scenario). The following sub-sections provide the details on these aspects.

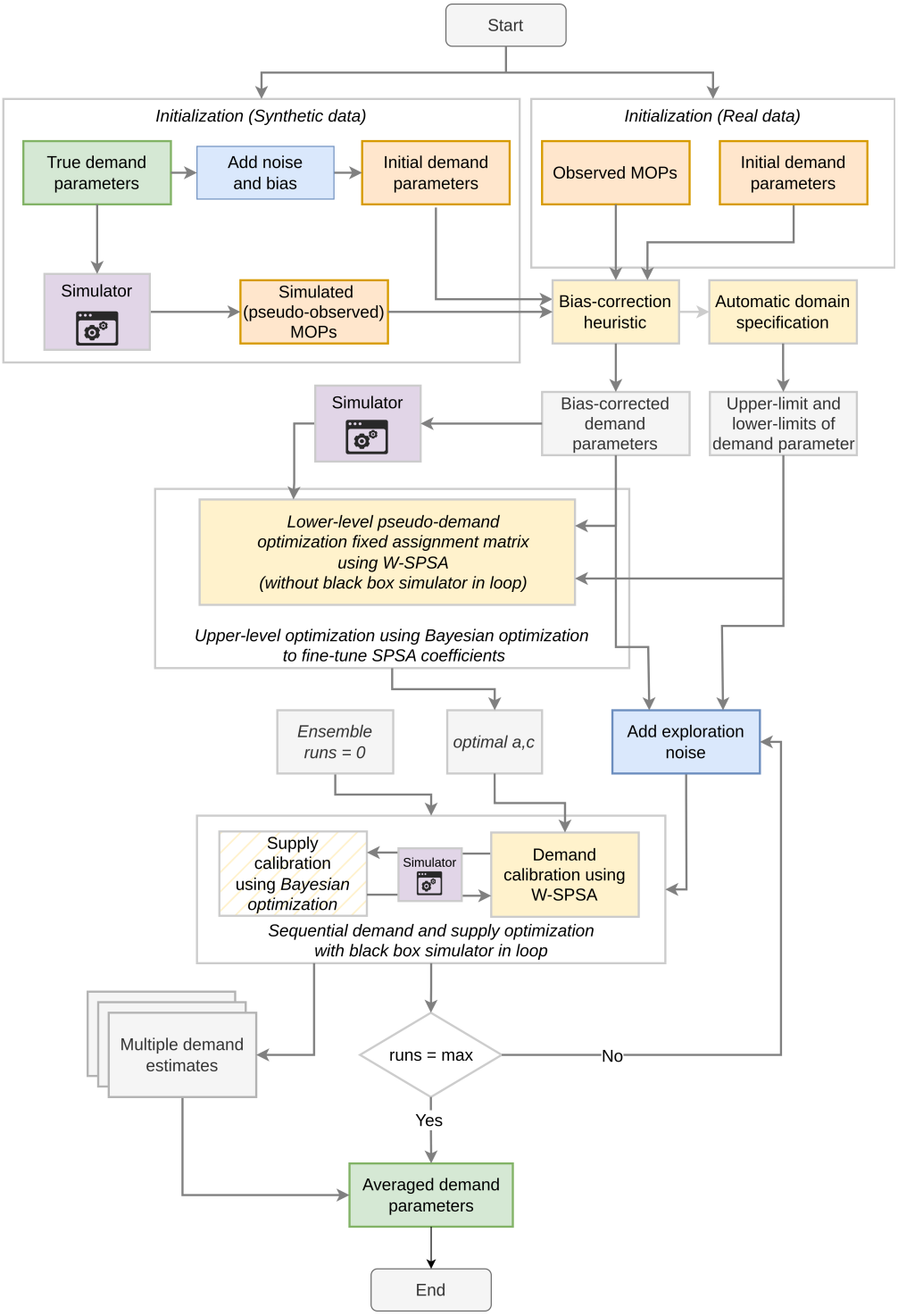


Figure 6.1: Proposed demand-supply offline calibration framework

Algorithm 1 W-SPSA, source: Antoniou et al. (2015); Lu et al. (2015)

Require: SPSA gain coefficients $\{a, c\}$ and other parameters $\{\gamma,\ \alpha,\ A\}$, number of iterations K or error tolerance τ, Initial parameter θ_0

Ensure: $\theta^{\dagger}$

1: $L_0 \leftarrow L(\theta_0)$
2: **for** k $\leftarrow$ 1, 2, ..., K **do**
3: $a_k \leftarrow a/(k + A)^{\alpha}$
4: $c_k \leftarrow c/(k)^{\delta}$
5: $\boldsymbol{W} \leftarrow W_k$
6: $\hat{G}_k \leftarrow \frac{1}{2c_k}\boldsymbol{W}^{\top}\left[\boldsymbol{\mathcal{Z}}^{+} - \boldsymbol{\mathcal{Z}}^{-}\right] \oslash \Delta_k$
7: $\theta_{k+1} \leftarrow \theta_k$ - $a_k\hat{G}_k(\theta_k)$
8: $L_k \leftarrow \sum_{t=1}^{T}\left[\boldsymbol{w_1}Z_1\left(f(\boldsymbol{\theta_{k+1}})\right) + \boldsymbol{w_2}Z_2\left(\boldsymbol{\theta_{k+1}}\right)\right]$
9: **if** $L_k \leq L_{k-1}$ **then**
10: $\theta^{\dagger} \leftarrow \theta_k$
11: **end if**
12: **if** $L_k < \tau$ **then break**
13: **end if**
14: **end for**

6.4.2 Sequential calibration

Equation 6.2 implies simultaneous calibration of demand and supply parameters since both sets of parameters are optimized simultaneously in a single objective function. Even though simultaneous calibration of demand and supply parameters provides efficient estimates (Toledo et al., 2014) (since at every iteration, both sets of parameters are consistent), it comes with additional computational complexity and more degrees of freedom. On the contrary, in sequential calibration, demand, and supply parameters are calibrated sequentially. It means the demand parameters are initially calibrated while keeping supply parameters fixed, followed by calibrating supply parameters while keeping the demand parameters fixed. Although this helps to reduce the complexity, this could be time-consuming since the process is repeated till estimates of both sets of parameters are consistent. Therefore, in sequential calibration, Equation 6.2 can be decomposed into two parts: demand (Line 2 in Algorithm 2) and supply (Line 4) calibration.

Sequential calibration provides the advantages of computational simplification of a large optimization problem into two smaller problems. Also, optimization can be flexibly adapted for the demand and supply parameters. This is important because demand and supply have distinct properties, such as a number of parameters, their range of possible values, and parameter sensitivity (Ciuffo et al., 2014) toward simulation outputs. This reason motivates the selection of suitable optimization techniques for each class of parameters. For instance, optimization algorithms scalable to high dimensions, such as SPSA, make sense for demand parameters that are large in number. On the other hand, if the number of supply parameters to be tuned is fewer, other state-of-the-art optimization techniques, such as Bayesian optimization, can be applied.

Algorithm 2 Sequential demand and supply calibration

Require: weights for sensor counts and prior OD matrices w_1 and w_2, prior parameters X^a and Y^a, number of sequential iterations S

Ensure: X, Y

1: **for** s $\leftarrow$ 1, 2, ..., S **do**
2: $X_s^\dagger \leftarrow \underset{X}{\text{minimize}} \sum_{t=1}^{T} \left[\boldsymbol{w_1} Z_1\left(M_t^o, M_t^s\right) + \boldsymbol{w_2} Z_2\left(X_t, X_t^a\right)\right]$ ▷ Demand calibration
3: $X_t \leftarrow X_s^\dagger$
4: $Y_s^\dagger \leftarrow \underset{Y}{\text{minimize}} \sum_{t=1}^{T} \left[\boldsymbol{w_1} Z_1\left(M_t^o, M_t^s\right) + \boldsymbol{w_3} Z_3\left(Y_t, Y_t^a\right)\right]$ ▷ Supply calibration
5: $Y_t \leftarrow Y_s^\dagger$
6: **end for**

6.4.3 Bias-variance decomposition

DODE can be seen as determining the optimal demand and supply parameters based on the given initial conditions (starting parameters) and the search process. Due to the estimation process, there will be an error between the estimated demand (or supply) parameters and optimal demand parameters. Now, we define:

- Let $h(\mathbf{x})$ represent the (family of) estimators to be learned from sequential minimization in Algorithm 2, where $\mathbf{x} = \{X, Y\}$ are the possible solutions.
- Let $h^*(\mathbf{x})$ be the best estimator, i.e., which provides the best values of parameters.
- $\mathcal{U}$ represents the stochasticity of the search process, which affects the outcome. This stochasticity can arise due to the characteristics of the optimization algorithm and black box simulator.
- Then, **bias** is the error between the average estimator (averaged over $\mathcal{U}$) and the best estimator $h^*(\mathbf{x})$
- Randomness due to $\mathcal{U}$ will give rise to **variance** of a single estimator $h(\mathbf{x})$
- Finally, we have the **noise** or irreducible error, which is the difference between the unobserved true estimator $\mathcal{H}$ and the best estimator $h^*(\mathbf{x})$

Using the Bias-Variance decomposition, the error can be written as:

$$\text{expected error} = (\text{bias})^2 + \text{variance} + \text{noise} \tag{6.14}$$

where

$$\begin{aligned}
(\text{ bias })^2 &= \int \left\{\mathbb{E}_{\mathcal{U}}[h(\mathbf{x};\mathcal{U})] - h^*(\mathbf{x})\right\}^2 p(\mathbf{x}) \mathrm{d}\mathbf{x} \\
\text{variance} &= \int \mathbb{E}_{\mathcal{U}}\left[\left\{h(\mathbf{x};\mathcal{U}) - \mathbb{E}_{\mathcal{U}}[h(\mathbf{x};\mathcal{U})]\right\}^2\right] p(\mathbf{x}) \mathrm{d}\mathbf{x} \\
\text{noise} &= \int \left\{h^*(\mathbf{x}) - \mathcal{H}\right\}^2 p(\mathbf{x}, \mathcal{H}) \mathrm{d}\mathbf{x} \mathrm{d}\mathcal{H}
\end{aligned}$$

Initial values or the given parameter values can be seen as belonging to the sub-optimal estimator that needs improvement. Thus, DODE aims to correct initial parameter values to recover the "true" or desired values. If X^* is the best estimate (corresponding to $h^*(\mathbf{x})$) and X^a is the initial/ current or given estimate, then:

$$X^a = X^*((1 - B^x) + R^x\epsilon) \tag{6.15}$$

where B^x and R^x & ϵ control the systematic bias and randomness in each parameter value, respectively. Here R^x is the contribution due to the estimator variance and noise. Thus, the selected estimator should be the one that leads to minimum error. In the following subsection, we provide a step-wise approach to addressing the bias and variance of the estimators:

6.4.4 One-shot bias correction heuristic

As the true estimator $\mathcal{H}$ is unknown, it is impossible to compute the expected error. Therefore, this sub-section introduces an alternative approach for bias correction (Algorithm 3) applicable when the count data from links is available. The functional relationship between the OD flows and sensor counts can be then represented using the following equation (ignoring measurement errors):

$$C = \boldsymbol{A}^\top X \tag{6.16}$$

where, C is the $(mT \times 1)$ dimensional column matrix for the sensor counts, X is the $(pT \times 1)$ dimensional OD demand demand column matrix, and A is the $(pT \times mT)$ dimensional assignment matrix of demand onto the sensors. In uncongested networks, link flows depend linearly on the demand because the link costs or assignment matrix in uncongested networks do not depend on the demand. Now we can write the above equation for both the simulation and real scenarios:

$$C_o = \boldsymbol{A_r}^\top X_r \qquad\qquad C_s = \boldsymbol{A_s}^\top X_s \tag{6.17}$$

Under the assumptions of the uncongested network and similar demand-link assignment in real-world and simulation, combining the above two equations gives us the following:

$$X_r = BX_s \tag{6.18}$$

Where B is the factor based on the sensor counts in simulation and measurements. Under assumptions of an uncongested network, we only use a single simulation run to upscale or downscale the OD demand matrix for a given time interval. Therefore, it is called a "one-shot". We approximate B in two ways, as shown in Algorithm 3. In the first case, we simulate the initial demand X^a and calculate the ratio of the cumulative simulated counts with the cumulative measured counts (Line 5 in Algorithm 3) where, $C^s_{t,m}$ and $C^o_{t,m}$ are the simulated counts and observed counts during period t for the m^{th}

sensor, respectively, and N_c is the number of sensors in the network. This scalar value is termed the Naïve bias factor, which is used to upscale or downscale the initial values and estimate the intermediate "bias-corrected" OD matrix $\{\hat{X}_t\}$.

Algorithm 3 Bias correction heuristic

Require: Initial OD parameters X^a, Other parameters including supply parameters Y, Road network and other fixed supply parameters G, Observed sensor counts C^o
Ensure: $\hat{X}^a$
1: $M_t^s \leftarrow f\left(X_t^a; Y; G\right)$
2: $C_t^s, S_t^s \leftarrow M_t^s$
3: **if** method=*Naïve* **then**
4: **for** $t \leftarrow 1, 2, \ldots, T$ **do**
5: $\hat{B}_t^x \leftarrow \frac{\sum_{m \in N_c} C_{t,m}^s}{\sum_{m \in N_c} C_{t,m}^o}$
6: $\hat{X}_t^a \leftarrow \frac{X_t^a}{\hat{B}_t^x}$
7: **end for**
8: **end if**
9: **if** method=*weighted* **then**
10: $\hat{B}^x \leftarrow C^s \oslash C^o \cdot W^\top$
11: $\hat{X}^a \leftarrow X^a \oslash \hat{B}^x$
12: **end if**

The above factor has limitations as it assumes that demand for the current interval only influences the link incidence of the same interval and ignores the correlation of count sensors with the demand. However, in practice, this is not true. To address this, we can also use the simulator knowledge, i.e., the assignment matrix, to obtain an accurate Bias factor. The idea is to estimate the bias factor for the demand flows based on the count sensors that fall along the routes or paths during specific periods for the given demand flows. Thus, the contribution of the uncorrelated count sensors and periods can be omitted. We use the weight matrix (same as the weight matrix in W-SPSA) in Line 10 of Algorithm 3.

Due to the simplicity of the above heuristic, there is no guarantee that $\hat{X}_t^a$ will lead to a better fit of sensor counts. The proposed method can be applied to the demand corresponding to the off-peak hours before calibrating the demand for the peak hours due to the possibility of congestion. If most of the network during peak hours is uncongested, then the above relationship can be expected to approximate the upscaling or downscaling factor. Therefore, the accuracy of the correction depends on the actual state of the network and how the congestion affects the demand-link assignment within the calibration intervals. Nevertheless, this step is only an intermediate step and provides a principle for initial adjustment in the given estimates. Further fine-tuning is performed by calibration algorithms, which are discussed in the following sections.

Using initial demand i.e., X^a for domain specification can be ineffective since initial values are disturbed due to bias and noise, as shown in Equation 6.15. Instead, we use $\hat{X}^a$

for specifying the domain since they have been partially adjusted for the bias. Further, we specify a domain flexibly depending on each of the values of the parameter, using the $\hat{X}^a l_x \leq X \leq \hat{X}^a u_x$, where, l_x and u_x are the multiplicative factors for specifying the lower bound and upper bound on the parameters. Thus, at least two parameters (l_x and u_x) are needed to specify the domain for the complete set of demand parameters. By using the $\hat{X}^a$, we take into account the (corrected) prior knowledge about the magnitude of the parameters. The domain specification leads to a fan-shaped domain specification, where the domain is narrow for the smaller values of the parameters, and vice-versa.

6.4.5 Automatic tuning of SPSA parameters using analytical model

We use an analytical assignment method approximated from the initial simulation run to automatically fine-tune the calibration or optimization algorithm's (such as SPSA) parameters. In this way, we avoid iterating over the computationally expensive simulation-based dynamic assignment. Thus we call it a "simulator out of the loop," i.e., the calibration algorithm does not use DTA or simulation assignment but uses an alternate analytical assignment method. Therefore, we do away with the need to fine-tune the algorithm's parameters with the simulator in the loop, thus reducing the computational burden and saving time. After tuning the calibration algorithm's parameters, we run the calibration with the simulator and similarly call it a "simulator in the loop," i.e., The calibration algorithm involves iteration or looping over the DTA simulator for traffic assignment.

To develop the analytical model, we only use an initial simulation-based assignment to derive the assignment matrix. An assignment matrix is endogenous to the simulator based on the time-dependent OD flows and route choice model and is derived from the incidence of the OD flows on the edges with count sensors. The functional relationship between the OD flows and link counts can be then represented using the following equation:

$$\hat{C}^s = \boldsymbol{A}^\top \hat{X} \tag{6.19}$$

where, $\hat{C}$ are the sensor counts from the analytical assignment, A is the assignment matrix derived from the simulator. We use Equation 6.19, as an approximation of the simulator to fine-tune the algorithm's parameters. This analytical assignment is way faster than running the simulator. This equation can also be seen as a meta-model of the simulation model. This method does not use the sensor or link speeds, since the complex relationship between the link speeds and OD flows is non-linear and cannot be analytically approximated using just the assignment matrix. Thus, to use this approach, sensor counts must be used as MOP in the GOF function. The parameter (ϕ) tuning can be formulated as an optimization problem (Equation 6.20), keeping demand and supply parameters fixed, where, $\hat{C}^s$ is given by equation 6.19.

$$\phi^\dagger \leftarrow \underset{\phi}{\text{minimize}} \left[\underset{\hat{X}_t}{\text{minimize}} \sum_{t=1}^{T} \left[\boldsymbol{w_1} Z_1 \left(C_t^o, \hat{C}_t^s \right) + \boldsymbol{w_2} Z_2 \left(\hat{X}_t, \hat{X}_t^a \right) \right] \right] \tag{6.20}$$

Overall, the automatic parameters tuning module can be viewed as a hierarchical optimization framework consisting of the following:

1. First-level or inner optimization using calibration algorithm with an analytical model to calibrate the pseudo demand parameters ($\hat{X}_t$) with a given set of parameters. This is shown by the inner part of the Equation 6.20. We cannot ensure the consistency between the demand and assignment matrix during optimization by using the analytical model (Equation 6.19) instead of the simulator. This is because when there is a change in the demand parameters ($\hat{X}$), the assignment matrix ($\boldsymbol{A}$) is considered fixed during the inner minimization in Equation 6.20. Thus, the calibrated demand parameters here are referred to as pseudo-demand parameters ($\hat{X}_t$) for the algorithm's parameter tuning. Still, they help decide the appropriate gain coefficient values for optimization based on the magnitude of the parameters.

2. Second-level or outer optimization with Bayesian learning to fine-tune the algorithm's parameters (a_k, c_k) based on the first-level optimization. The reason for using Bayesian optimization is that it is a powerful optimization technique when the objective function is not observed, function evaluations are expensive, and the number of parameters is limited. In this case, the objective function is shown by the outside part of the Equation 6.20. A simple Bayesian optimization algorithm adapted from (Brochu et al., 2010) is presented in Algorithm 4. Bayesian optimization uses an acquisition function u to sample the next data point, deciding between exploration and exploitation (Brochu et al., 2010). By specifying a smooth prior belief, such as the Gaussian Process (GP), we can calculate the posterior distribution of the GP by sampling the new data points iteratively. The posterior distribution is the surrogate model of our unobserved objective function (Equation 6.20). The acquisition function samples the points by evaluating the expected value of a surrogate function and selecting the point that maximizes it. We refer the reader to the tutorial on Bayesian Optimization for further details (Brochu et al., 2010).

Algorithm 4 Bayesian optimization adapted from Brochu et al. (2010)

1: **for** $b \leftarrow 1, 2, \ldots, B$ **do**
2: Let x represent the gain coefficients $\{a, c\}$, then find x_b by optimizing the acquisition function over the GP: $x_b = \underset{x}{\operatorname{argmax}}\, u(x|D_{1:b-1})$
3: Sample the objective function: $y_b = z(x_b)$
4: Augment the data $D_{1:b} = \{D_{1:b-1}, (x_b, y_b)\}$
5: **end for**

Subsequently, the sequential optimization of demand (and supply) parameters (Algorithm 2) is done using the optimal calibration algorithm's parameters obtained by the above hierarchical optimization module.

6.4.6 Ensembling for variance reduction

Due to the under-determined nature of OD estimation, there can be multiple solutions for a given optimization formulation (Line 2 in Algorithm 2) and local-search algorithms, such as SPSA, can result in the distinct local minima resulting in parameters with considerable variance. Due to variance in the spatiotemporal demand patterns, variance in sampling distribution or measurement errors, and simulation behavior stochasticity, some of these solutions can be hypothesized as a manifestation of the desired or "true" solution. Parameter averaging, such as in the bagging technique and Stochastic Weight Averaging (SWA), can help to cancel out some of the variance in the individual solution so that the averaged solution is closer to the desired solution.

Algorithm 5 W-SPSA with Bagging

Require: Bias-corrected dynamic OD matrices $\hat{X}_t^a$, number of bagging ensembles E, exploration parameter σ^2

Ensure: $X^\dagger$ ▷ Averaged or "bagged" estimate

1: **for** e ← 1, 2, . . . , E *in parallel* **do** ▷ Bagging cycles

2: $\epsilon \leftarrow \mathcal{N}(0,\, \sigma^2)$

3: $\hat{X}^a \leftarrow \hat{X}^a + \epsilon$

4: $X_e^\dagger \leftarrow \underset{X}{\text{minimize}} \sum_{t=1}^{T} \left[\boldsymbol{w_1} Z_1 \left(M_t^o, M_t^s \right) + \boldsymbol{w_2} Z_2 \left(X_t, \hat{X}_t^a \right) \right]$ ▷ Demand calibration using W-SPSA

5: **end for**

6: $X^\dagger \leftarrow \frac{1}{E} \sum X_e^\dagger$

6.4.6.1 Bagging (ensembling with cold restart)

Here we run multiple estimators, such as W-SPSA (in parallel or serial order), and record the final estimates of each run or cycle. Since SPSA is stochastic due to the nature of its search process (see Equation 6.6, where Δ_k is a random vector). Thus, different runs of SPSA with different seeds can lead to different local optima, even if SPSA parameters are kept the same. In all the cycles, the same initial estimate is used, which is why this can be referred to as "cold restart" (Algorithm 5), since knowledge from the previous cycle is not used to influence the current cycle. However, we add a small exploratory noise in the initial OD vector to promote the optimization algorithm to find new solutions. With the cold restart, the algorithm has more freedom to explore other possible solutions that are scattered around the desired solution. The final "bagged" estimate is the simple average of all the final estimates from all the W-SPSA cycles. Further, specifically in bagging, individual models can be trained in parallel, thus offsetting the time cost of multiple optimization cycles. For further details on bagging, we refer the reader to Breiman (1996); Dietterich (2000).

6.4.6.2 Stochastic parameter averaging (ensembling with warm restart)

We propose ensembling with warm restart and refer to this approach as SPA (Stochastic Parameter Averaging), inspired by SWA (Izmailov et al., 2018), snapshot ensembling (G. Huang et al., 2017), and Stochastic Gradient Descent (SGD) with warm restarts (Loshchilov & Hutter, 2016), for W-SPSA. We use the term "parameter" instead of weight since the former term is more common in traffic calibration literature. In SPA, the gain coefficients are reset after fixed iterations or when the objective function fitness is not changing much. The next optimization cycle uses the iterate from the previous cycle as the initial parameters (Algorithm 6); hence, it is referred to as "warm restart". The resetting of SPSA gain coefficients resembles the cyclic learning rate. The idea is after initial convergence around a probable solution, W-SPSA is further pushed to explore the other solutions for improvement, but in the vicinity of the estimate from the previous cycle. Finally, we take the simple average of cycle estimates to obtain the final "SPA" estimate.

Algorithm 6 W-SPSA with Stochastic Parameter Averaging [based on SWA (Izmailov et al., 2018)]

Require: bias corrected dynamic OD matrices $\hat{X}_t$, number of SPA cycles E
Ensure: X_{SPA} {Averaged SPA estimate}
1: **for** e $\leftarrow$ 1, 2, ..., E **do**
2: $\quad X_e^{\dagger} \leftarrow \underset{X}{\text{minimize}} \sum_{t=1}^{T} \left[\boldsymbol{w_1} Z_1 \left(M_t^o, M_t^s \right) + \boldsymbol{w_2} Z_2 \left(X_t, \hat{X}_t^a \right) \right]$
3: $\quad X_{SPA} = \frac{(e-1) \cdot X_{SPA} + X_e^{\dagger}}{e}$
4: $\quad \hat{X}^a = X_e^{\dagger}$
5: **end for**
6: $X^{\dagger} \leftarrow \frac{1}{E} \sum X_e^{\dagger}$

6.4.7 Calibration of supply parameters

We use Bayesian optimization (Algorithm 4) for calibrating the selected supply parameters (Line 4 in Algorithm 2). Different data sources, such as point-based, edge-based, and network-based, can be used to calibrate the parameters. The type of supply parameters can vary based on the specific simulator. However, Bayesian optimization is a kind of black-box optimization and thus accesses only the inputs (parameters) and outputs of the objective function. Therefore, supply parameters to be calibrated are selected based on their sensitivity to the output data or corresponding MOPs. If certain parameters are not very sensitive to the outputs, it is not possible to calibrate them with the given data.

6.5 Experiment design and set-up

6.5.1 Overview

In this research, the demand parameters are the time-dependent OD matrices. Supply parameters control the traffic propagation and route choice behavior. The details of scenarios with different simulation and data combinations for varying levels of simulation complexity and data are as follows:

1. **Scenario 1**: Analytical assignment with synthetic sensor counts: A randomly generated demand-link assignment matrix is used for mapping OD flows (randomly sampled using a distribution function) to sensor counts using Equation 6.19. In the case of synthetic experiments, where true OD parameters are generated/ known, the algorithm is also validated by the error between the calibrated OD parameters and true OD parameters. The method's performance is evaluated on the fitness of sensor counts and OD matrices. This scenario focuses on obtaining accurate demand estimates (Line 2 of Algorithm 2), which is why supply parameters are considered fixed. Hence, this scenario is just restricted to demand calibration.

2. **Scenario 2**: SUMO and Munich network with synthetic sensor counts data: Given OD flows (Moeckel et al., 2020) are simulated and corresponding sensor counts are recorded as desired counts. In this case, supply parameters are kept constant and thus not part of the calibration. The method's performance is evaluated on the fitness to measurements (counts, speeds) and OD matrices.

3. **Scenario 3**: SUMO and Munich network with real-world sensor counts: Given OD flows (Moeckel et al., 2020) are used with sensor counts from real-world data sources (BAST: Bundesanstalt für Straßenwesen, 2023). We use the best-performing approaches in the above scenarios and apply them here. In this case, true OD matrices are unknown, and the algorithm's performance is only evaluated on sensor count fitness. To achieve the best fitness, we calibrate the demand and the supply parameters sequentially (Algorithm 2).

6.5.2 Initialization

A "true" OD is sampled from an underlying distribution for the experiments with synthetic data. Based on the empirical findings, we select a right-skewed distribution for sampling the OD demand, so a few OD pairs have many trips mirroring large and dominating zones (such as external zones) within the study area. On the other hand, most zones have a relatively smaller number of trips. The sampled demand matrix (in case of synthetic experiments) or initial OD demand matrix (in case of real scenarios) is given as input to a traffic simulator (Algorithm 7), and corresponding simulation outputs (link counts and link speeds) are recorded.

Subsequently, bias and randomness, proportional to the OD parameter's magnitude, are added to the true demand values according to Equation 6.15. In this way, a "true"

Algorithm 7 Initialization

Require: Initial OD parameters X_t (real case) or distribution D_X (synthetic case), Other parameters including supply parameters Y, Road network and other fixed supply parameters G, Observed sensor measurements M_t^o (real case)
Ensure: X_a, C_t^o, S_t^o

```
if scenario = synthetic then                          ▷ Synthetic data scenario
    X*_t ~ D_X                                        ▷ Generate true OD matrix parameters
    M^s_t ← f(X*_t; Y; G)                             ▷ Generate true sensor measurements
    X^a ← X*((1 − B^x) + R^x ε)                       ▷ Perturb original parameters
    C^o_t, S^o_t ← M^s_t                              ▷ Assign observed measurements
else                                                  ▷ Real data scenario
    X_a ← X_t                                         ▷ Assign seed matrix
    C^o_t, S^o_t ← M^o_t
end if
```

or desired OD matrix is corrupted or disturbed by adding artificial bias and noise. This disturbed OD matrix is used as the initial or given OD matrix (X^a), similar to practical situations where the actual or "true" OD matrix is unknown. However, instead, an error-prone prior estimate is available. Due to errors in the prior demand matrix, we do not use it in the calibration objective function, i.e., we set $\boldsymbol{w_2}$=0 and $\boldsymbol{w_3}$=0 in all the above scenarios (Algorithm 2). Thus, optimization is guided by the fitness of count or speed measurements ($\boldsymbol{w_1}$=1), but the search is restricted within the domain or structure specified using bias-corrected prior estimates.

6.5.3 Gradient and performance evaluation

We use primarily Weighted Average Percentage Error (WAPE) (Equation 6.21) as our evaluation criteria for OD fitness and count fitness:

$$\text{WAPE} = \frac{\sum(|x - \hat{x}|)}{\sum x} \tag{6.21}$$

where x and $\hat{x}$ are actual and predicted values. WAPE weights the percentage errors based on their magnitude since the scale of the parameters can vary across a wide range. WAPE, also called MAD/ mean ratio, is a preferred alternative over MAPE (Kolassa & Schütz, 2007). This is crucial since the costs of inaccurate estimation of large OD demand flows can be more adverse and thus need to be minimized. Apart from WAPE, we use Root Mean Squared Error (RMSE) for performance evaluation.

In W-SPSA gradient calculation steps, we scale the estimators (z_1 and z_2) relative to each other using the following method (He et al., 2021):

$$\widetilde{z_2} = z_2 \cdot \frac{max\{z_1\}}{max\{z_2\}} = z_2 \cdot \eta_2 \tag{6.22}$$

where η_2 is the scaling factor. Alternatively, the measurements or parameters can be normalized or standardized before evaluating the estimator. Similar scaling is used for speed measurements if included in the objective function.

6.5.4 Experiments

We conduct the grid-based evaluation of the effect of the parameters B^x and N^x on the effectiveness of our proposed approach. Since we expect ensembling to be beneficial when the individual estimates are in the neighborhood of each other, by averaging some of the variance can be canceled, and the mean of the estimates is closer to optimal values, as compared to the individual estimates. We hypothesize that with the increase in the magnitude of bias and noise in the initial OD values (X^a), the resulting calibrated estimates can be far from each other, leading to reduced effectiveness of the ensembling. This grid-based evaluation helps to define the value of N^x for Scenarios 2 and 3. We also add randomness to the sensor count measurements and check the impact on the calibrated estimates. The noise is added to mimic random data errors according to $\hat{C}^o = C^o(1 + R^c\epsilon)$. We incrementally add the proposed methodological components to the baseline W-SPSA method and evaluate the improvement. The possibilities are enumerated as follows:

1. **W**: Baseline, using only **W**-SPSA and manual specification of SPSA parameters.
2. **BC**: **B**ias **C**orrection heuristic
3. **A-W**: W-SPSA with **A**utomatic SPSA's parameter tuning.
4. **W-B**: W-SPSA with **B**agging.
5. **W-SPA**: W-SPSA with SPA.
6. **A-W-B**: W-SPSA with Automatic SPSA tuning, followed by Bagging.
7. **BC-A-W**: Bias correction heuristic, followed by W-SPSA with Automatic SPSA tuning.
8. **BC-A-W-B**: Bias correction heuristic, followed by Automatic SPSA tuning for W-SPSA, with bagging

6.5.5 Computation burden

The computation requirement for convergence of the algorithm depends on many factors. We quantify the computation requirement of our approach in terms of the number of objective function evaluations or traffic assignment instances. Depending on the type of scenario, the type of traffic assignment (analytically or simulation-based) and its time burden can be different. If all the methods are used, then the minimum number of times function evaluation is done can be expressed as $1 + S((3 \cdot E \cdot K) + B)$. This is because we need one evaluation of BC and three evaluations for each W-SPSA (ignoring gradient

and simulation replications). We used a desktop PC (8 i7-11700F @ 2.50GHz physical cores and 50 GB RAM) and a workstation (36 Intel Xeon @ 2.60 GHz physical cores and 156 GB RAM). A single analytical traffic assignment requires less than 5 seconds due to its simplicity, whereas a single simulation-based traffic assignment takes around 31 minutes.

6.5.6 Calibration platform description

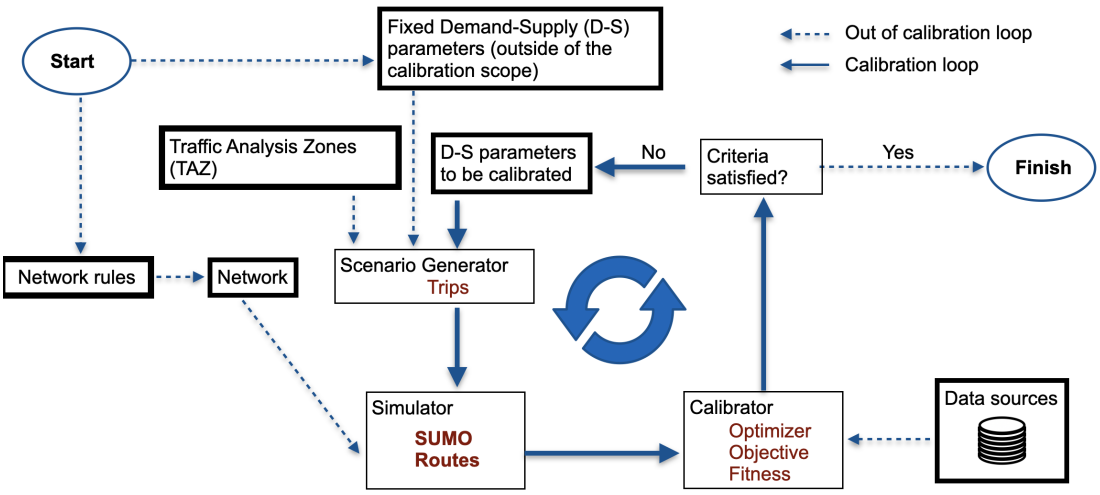


Figure 6.2: Calibration platform and SUMO simulator coupling in Python

We developed a Python-based platform for the sequential calibration of the demand and supply parameters of the large-scale mesoscopic traffic simulation in Simulation of Urban Mobility (SUMO) (Lopez et al., 2018). Figure 6.2 shows a schematic representation of the platform. Given the simulation inputs (network, traffic analysis zones, detector locations) and parameters' priors, the platform calibrates the demand according to the proposed methodology. Other parameters that are fixed are, therefore, not part of calibration or outside of the scope of calibration. An initial OD matrix is used to generate trips between edges in different Traffic Analysis Zones (TAZs). The routing algorithm in SUMO assigns routes to these trips. We select a few supply parameters that influence traffic flow, junction delays, and route choice behavior. These parameters are defined below:

1. Automatic or online routing is used for the traffic assignment. According to SUMO (2023a), this routing approach works by giving some or all vehicles the capability to re-compute their route periodically based on the traffic conditions in the network. This kind of routing is also called a "flexible one-shot assignment" (Castiglione et al., 2014). The parameters influencing the routing of vehicles are:
   a) *re-routing probability*: The probability for a vehicle to have a routing device.

Table 6.2: Enumeration of calibration parameters

Simulator →	Analytical	Black box	
Data →	1. Synthetic	2. Synthetic	3. Real
Network parameters			
Number of OD pairs	2500	5256	5256
Number of intervals	3	5	5
Duration of interval (hours)	1	1	1
Number of count sensors	500	1166	450
SPSA parameters			
γ	0.01	0.101	0.101
α	0.7	0.602	0.602
Range for c	$(0.01, 10)$	$(0.01, 1)$	$(0.01, 1)$
Range for a	($1x10^{-6}$, $1x10^{-3}$)	($1x10^{-5}$, $1x10^{-3}$)	($1x10^{-7}$, $1x10^{-2}$)
Maximum number of SPSA iterations	100	50	50
W-SPSA weight parameters			
Weight binary rounding	True	True	True
Weight cut-off	0.01	0.01	0.01
Other parameters			
S	upto 5		
B	upto 100		
E	upto 20		

b) *re-routing period*: The period with which the vehicle shall be rerouted.

c) *re-routing adaptation steps*: The number of adaptation steps for averaging.

2. To influence the routing decision, the travel time of different types of edges can be scaled depending on their priority, using the parameter *edge priority factor* (SUMO, 2023c). Consequently, low-priority edges will receive a penalty and have increased travel times, whereas high-priority edges receive little or no penalty.

3. The parameters which affect other delays (SUMO, 2023b) are:

   a) *tls_travel-time_penalty*: This is a headway penalty to reduce the maximum flow across a signalized intersection.

   b) *meso_minor_penalty*: This is a fixed time penalty when passing a prioritized link.

We implement the W-SPSA by extending the Python SPSA implementation by Mayer (2017). All inputs pertaining to the network specification, count detectors, demand zones, SPSA parameters, etc, for three scenarios, are shown in Table 6.2. Values of SPSA parameters γ and α are fixed based on initial sensitivity analysis. We select SPSA gain coefficient a and perturbation c parameter for the automatic tuning module and

thus $\phi = \{a, c\}$. Their search space is specified in Table 6.2. The complete platform is implemented using Python and is available on GitHub (see footnote in section 6.2).

6.6 Results

6.6.1 Automatic SPSA parameter Tuning

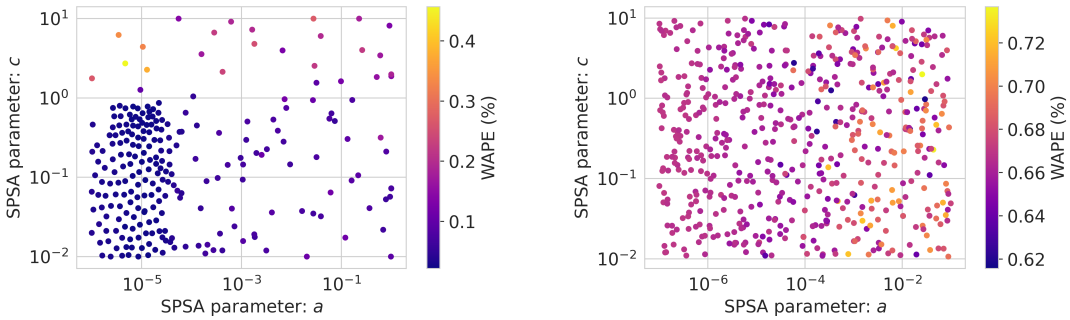


Figure 6.3: Automatic tuning of SPSA gain coefficients using Bayesian optimization for (left) scenario 1: synthetic simulator and (right) scenario 3: SUMO with real data

As discussed in Section 6.4.5, the automatic tuning procedure is solved as a hierarchical optimization process. The first step deploys W-SPSA to calibrate the pseudo demand parameters, while the second step uses Bayesian learning to fine-tune the SPSA parameters (a_k, c_k). For the Bayesian learning model, we use Matérn kernel as the Gaussian prior, and Upper Confidence Bound (UCB) as the acquisition function (Brochu et al., 2010). In all scenarios, we specified the parameter space for c as (1e-2, 1e1). The space for a is set to (1e-6, 1e0) for scenario 1, whereas it is set to (1e-7, 1e1) for scenarios 2 and 3. The points are randomly sampled on the logarithmic scale for initial probing, followed by Bayesian optimization. The number of iterations for initial probing/ exploration and number of iterations for Bayesian optimization was set to $\{50, 100\}$ for scenario 1, and $\{100, 200\}$ for scenarios 2 and 3. In Figure 6.3, we show the results of automatic SPSA parameter tuning for scenario 1 and scenario 3. The WAPE is lower for scenario 1 (scale of the color bar in Figure 6.3), compared to scenario 3 since the former involves synthetic data and the analytical simulator has a simpler loss surface without stochasticity. The approximated assignment matrix, in this case, is the same as the true assignment matrix, which is static. In scenario 1, we see that the points are initially probed randomly over the specified space of parameters during exploration, followed by a focused search based on the acquisition function. For scenario 1, we find that values of c and a in the ranges of (1e-2, 1e0) and (1e-6, 1e-4) are effective.

For scenario 3, the loss region is noisy due to errors from real data and analytical approximation of the assignment matrix in place of the actual simulator. This is why parameter combinations do not have a clear boundary of lower error and errors are also high. This stochasticity can be addressed by increasing the number of output averaging

and SPSA replications at the cost of additional computation. Still, a fuzzy pattern is evident for c and a in the range of (1e0, 1e1) and (1e-5, 1e-4), where lower errors are predominant. Thus, we conclude that automatic tuning of the SPSA parameters using an analytical approximation of the simulator is practical. With these insights, scenarios 2 and 3 are instrumented with the above settings of the automatic tuning module.

6.6.2 Scenario 1: Synthetic data with analytical assignment

We show the results of the grid-based evaluation for bagging effectiveness in Figure 6.4. At lower levels of randomness (R^x in 30-40%), initial error in demand (X^a) and sensor counts (M^c) are about the same. At higher levels of R^x, the initial error in X^a (OD parameters) increases. During the initial increase in R^x for $R^x < 30\%$, there is a rapid increase in the error for higher values of B^x, whereas the error increase is gradual for smaller values of B^x. The gradual error increase continues for higher values of R^x in the case of lower B^x, but the error is stable for higher B^x. For the sensor counts, the initial error stabilizes or even drops with an increase in R^x. This is because counts are the weighted sum of the demand flows between respective OD zones. Thus, additional randomness in the OD flows is canceled due to weighted summation. There is no strong correlation between the initial error in OD demand flows and corresponding counts in this range. Secondly, an increase in randomness cancels out the initial bias in some of the parameters and thus results in a small drop in the initial count WAPE.

We notice that W-SPSA can minimize the objective function in all cases of bias and randomness. This is because the sensor counts are used as MOP in the objective function, and it is evident that the final count error is lower than the initial count error. Further, the total error computed by equally weighing the error in sensor counts and OD parameters is also lower. For low values of randomness (R^x), the error is dominated by the factor B^x. Results indicate that the algorithm can correct even high bias values in OD parameters if R^x is small. This is why the initial and final total error gap is highest for low values of R^x.

The box plots in the middle column (Figure 6.4) show the WAPE of each individual estimate. The fitness of bagged OD estimates is consistently lower than the individual estimates in all cases, which supports the effectiveness of the bagging. However, the calibrated estimates are only better than the desired estimates for smaller values of R^x (0-20%) in all the ranges of B^x. This observation implies that the algorithm can only move closer to the desired ODs X^* for lower values of the R^x. This is because, firstly, increased randomness in the initial estimates will deteriorate the structure of the initial demand specification and the quality of the domain specification of the demand parameters. At high randomness values, the initial point and domain misguide the calibration algorithm to a local optimum which is even far from the starting point resulting in higher error. Secondly, high R^x does not translate to a higher error in sensor counts due to the cancellation of the random errors. Thus, gradients relying on the sensor counts cannot effectively guide the search. The conclusion is that desired OD parameters are only recoverable when R^x is small since, at higher values, the essential structure of the X^* in X^a starts to disappear. However, the Bagging approach effectively improves the weighted

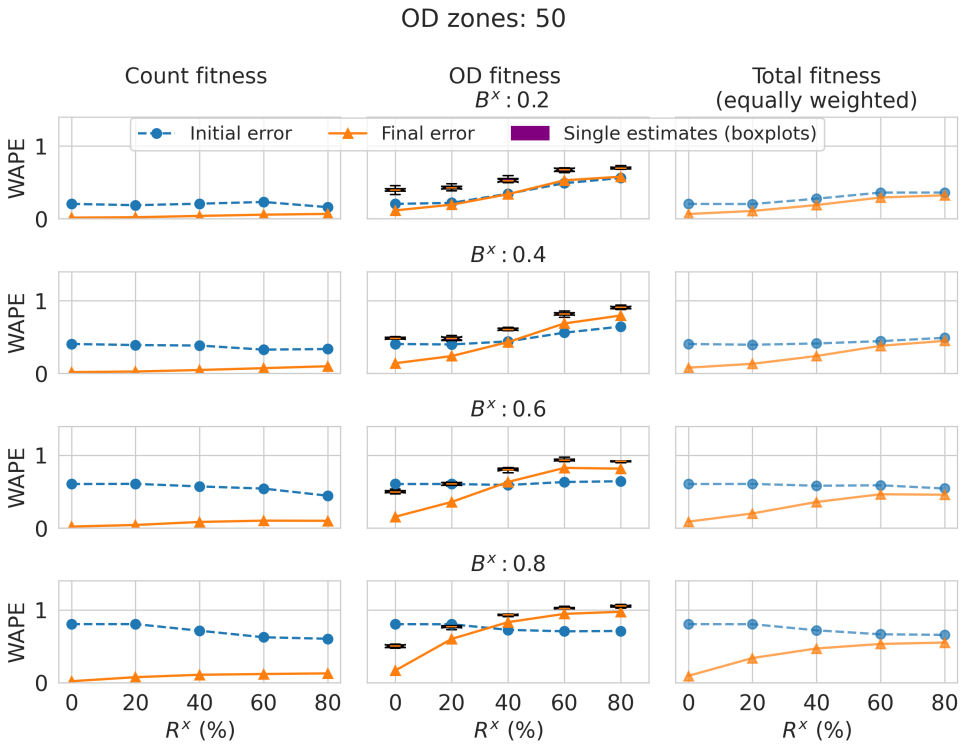


Figure 6.4: Scenario 1: Error at varying levels of B^x and R^x, for a Synthetic scenario with 50 OD zones. The *Final error* includes equally weighted sensor counts and OD demand estimates.

fitness of both the demand and count parameters. Based on these findings, in the black box simulation experiments i.e., scenario 2 and scenario 3, we set the randomness values as R^x=20%. This randomness value is similar to those used in the existing literature (Antoniou et al., 2016).

In Figure 6.5, we show the OD fitness error contours for single W-SPSA, SPA, and bagged estimates. Due to high dimensional optimization, fitness error is influenced by thousands of demand parameters. The plot shows the conditional error (because it depends on multiple parameters) region with the values of the pair of zones on X and Y-axes. The columns in this figure correspond to two levels of R^x 30% and 90%, both at B^x=0.6. Fitness error increases with the increase in R^x. The single estimates are scattered in the region. However, the averaged estimates from SPA and bagging lie with the region of lower errors than the single W-SPSA estimates. Thus, bagging and SPA help reduce the variance from single W-SPSA estimates.

We compare the performance of bagging and Stochastic Parameter Averaging (SPA) in Figure 6.6, where B^x=0.6 and R^x=30%. Here we report bagged estimates from 20 W-SPSA runs, each for 100 iterations. Further, we also show the results of multiple SPA runs, each running for 2000 iterations. It is pointed out that function evaluations in bagging (with 20 different W-SPSA cold restarts, each running for 100 iterations) are equivalent to those in a single SPA run of 2000 iterations with warm restarts. Thus, the comparison between them is fair. The final count WAPE for individual W-SPSA estimates (Column 1 in Figure 6.6) stops to reduce at 0.05 after a few iterations. On the other hand, count fitness WAPE for SPA continues to reduce up to a value of more than 0.025. In the SPA loss curve, we see that each warm restart of the cyclic learning rate pushes the loss curve down faster than before the restart of the learning rate. Individual estimates achieve an OD fitness WAPE of 0.70, whereas individual SPA achieves a WAPE of about 0.55. We find that averaging helps improve the final W-SPSA solution, compared to the single solutions from each. Both bagging and SPA provide better OD estimates than the individual estimate from each W-SPSA run. However, the averaged estimates from bagging show superior performance with a WAPE of 0.38 compared to the averaged SPA estimates with a WAPE of 0.49. This implies that even though individual SPA estimates are more effective in fitting the counts and ODs than individual W-SPSA estimates, the averaged estimates of bagging are better than those of SPA. This could be because SPA prioritizes exploration around the initial local optima. If the initial local optima is not good enough, SPA does not explore sufficiently, and SPA averaging fails to reduce the variance. In the case of bagging, each estimate is obtained from exploration in a broader region. Thus, averaging the estimates has a superior result. The results of averaged estimates from bagging are not too sensitive to the momentum parameter β, as compared to those from the SPA. In bagging, five individual estimates reduce a significant part of the OD fitness error, whereas, for SPA, the error reduction is gradual. This implies that a small number of cold restarts as in bagging can give major benefits. Due to these reasons, we only used bagging or ensembling with cold restarts for the following experimentation.

We compare the performance of different components of our methodology for OD parameter fitness and sensor count fitness in Figure 6.7. In this case, we set $B^x = 0.8$,

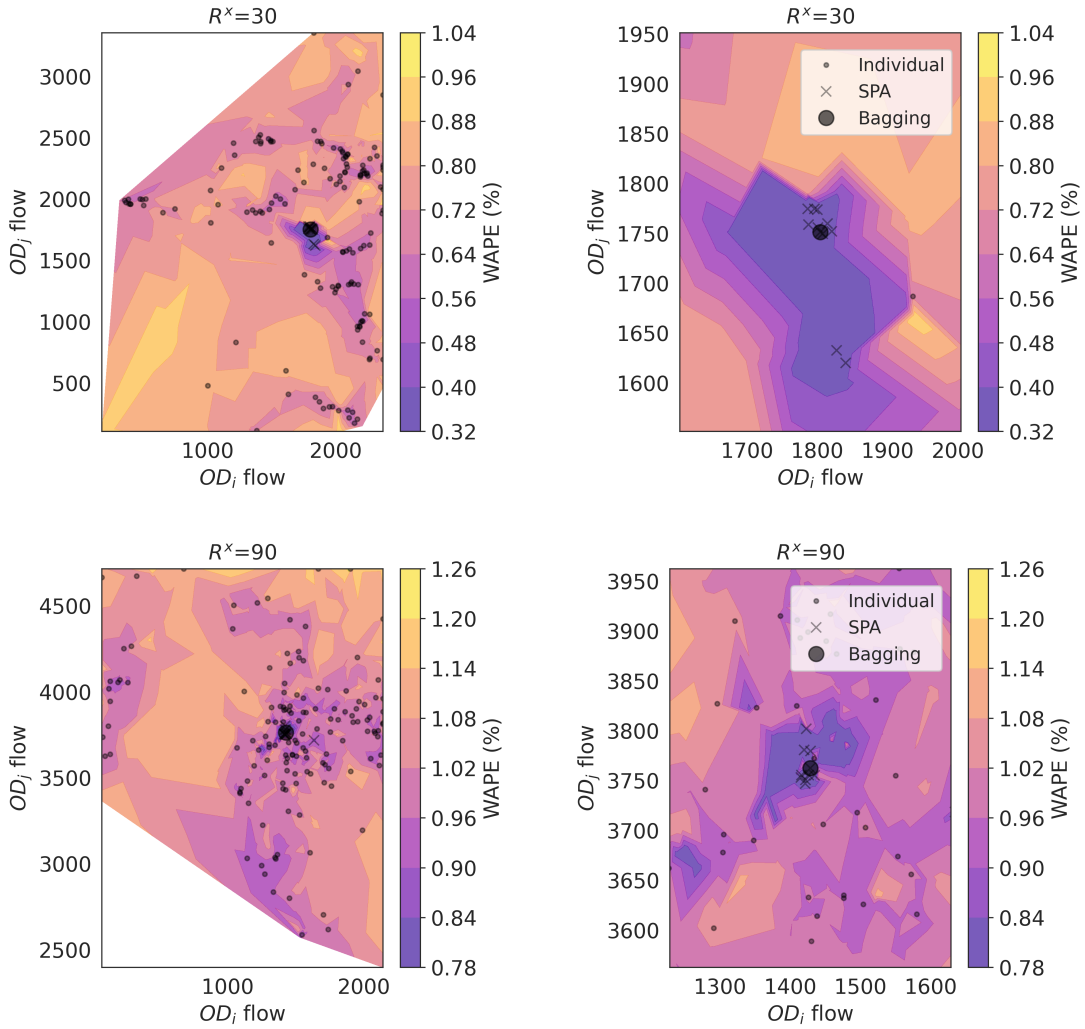


Figure 6.5: Scenario 1: Contour plots showing the parameter values for selected pair of the zones with B^x=0.6, at different values of the R^x. It can be seen that bagged (●) or SPA (×) estimates lie in the lower error region as compared to the single SPSA iterate (•). The right column is the zoomed-in version of the plots in the left column. The plot shows the conditional error region with the selected two OD zone pairs on X and Y-axes

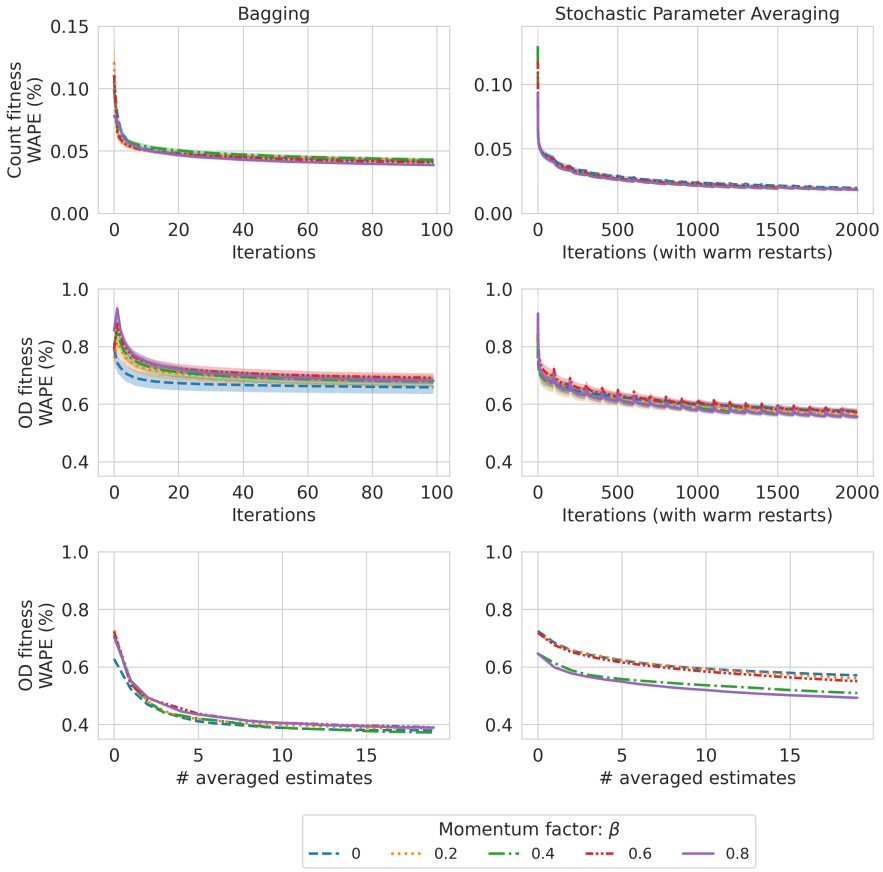


Figure 6.6: Scenario 1: OD and count fitness curves for bagging and stochastic parameter averaging with B^x=0.6 and R^x=30%.

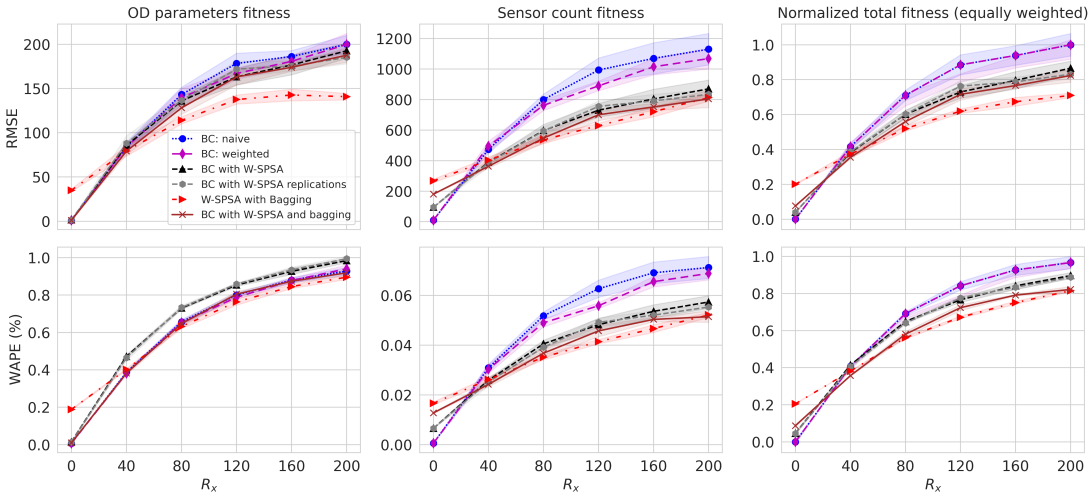


Figure 6.7: Scenario 1: OD and count fitness (RMSE and WAPE) sensitivity with the change in the randomness parameter, with different approaches using an analytical simulator (B^x=0.8)

test the performance for values of N^x ranging from 10% to 200%, and show WAPE and RMSE. The approaches compared are *Bias correction (BC) using naive method*, *BC with weighted method*, *BC with W-SPSA*, *W-SPSA with bagging*, and *BC with W-SPSA and bagging*. Although a high randomness factor leads to higher corresponding errors, the problem becomes more challenging since the structure of the desired estimate is not identifiable from the initial matrix.

We find that the performance of the approaches depends on the R^x. All approaches with bias correction perform equally well at low randomness values. This is an interesting finding since it implies a simple and computationally inexpensive heuristic can achieve similar or better error performance as the W-SPSA optimization process for small randomness in the initial OD matrix. At $R^x > 40\%$, bagging performs better than the other approaches, specifically as seen from *W-SPSA with bagging*. This implies that for OD fitness, the bias-correction heuristic dominates at small randomness, whereas bagging dominates at high randomness. For high randomness, initial estimates are unreliable; thus, averaging multiple estimates helps provide better results. In the case of WAPE, OD fitness of *BC with W-SPSA without bagging* shows higher errors than just using *BC*. Intuitively, the SPSA model works better when the objective function has a clear descent direction. This is often the case when the objective function has a lower/ higher demand with respect to the true demand (Cantelmo et al., 2015). However, as the *BC* heuristic removes bias related to, e.g., overestimation or underestimation, the performance of W-SPSA may be affected.

Looking at the fitness for sensor counts, we find that W-SPSA outperforms simple heuristics in matching the sensor counts regarding both WAPE and RMSE. This is

understandable since *BC* heuristics only adjust the OD parameters without ensuring consistency with the true sensor counts. Simple heuristics work equally well if the randomness in initial estimates is small ($20\% < R^x < 30\%$), meaning that initial estimates sufficiently capture the structure of the true estimates. The normalized total fitness shows that W-SPSA and bagging approaches achieve lower errors than the *BC* heuristics even in high randomness. Thus approaches using W-SPSA and bagging are best when ensuring the overall fitness of the counts and OD demand parameters. To speed up the convergence, *BC* can be used to adjust the initial values of the OD parameters, followed by W-SPSA with bagging to ensure consistency with the MOPs, such as counts.

6.6.3 Scenario 2: Munich scenario with SUMO simulator and synthetic data

We show the results of the calibration for the Munich scenario using the SUMO platform with synthetic counts and speeds in Table 6.3. The first set of results corresponds to $B^x = 0.6$ and a relatively smaller factor for randomness ($R^x = 20\%$) and uses only sensor counts or both sensor counts and link speeds in the objective function. We also add artificial randomness to the sensor counts to mirror data errors. We perform an ablation study by using one or more of the components of our methodology, namely W-SPSA (W), Bias Correction (BC), Automatic SPSA Tuning (A), and Bagging (B). The initial WAPE errors in count, speed, and OD are 0.42, 0.03, and 0.45, respectively. Similarly, the initial RMSE errors in count, speed, and OD are 288, 1.71, and 10.80, respectively. We define *baseline* as the scenario using sensor counts as MOP, with only W-SPSA, where count fitness WAPE is 0.14. The corresponding final speed and OD WAPE are 0.02 and 0.72, respectively. In this case, although counts and speeds fit better, the estimated OD is worse than the initial OD values. This is because in the objective function minimization, W-SPSA can converge to fit better to counts, but it lands in undesired local optima for the OD estimates, which is still away from the desired optima. Thus, individual estimates from W-SPSA have worse OD fitness due to induced randomness in the parameters during the optimization path. When using only BC, the OD fitness, count fitness, and speed WAPE are 0.28, 0.10, and 0.02, respectively. When using W-SPSA with bagging (*A-W-B*), we obtain OD fitness of 0.42, whereas count and speed fitness are 0.10 and 0.02. Thus, we find that bagging helps to provide improved count and OD estimates over initial values as well as Baseline scenarios. In this case, we find speed and count fitness comparable to the *BC* approach. Using *BC-A-W* provides better results than the baseline in terms of improvement over count and OD fitness, but still, the estimated ODs are worse than the initial estimates in terms of WMAPE and RMSE. Adding Bagging helps to address this variance in the estimated OD parameters since the approaches *A-W-B* and *BC-A-W-B* have superior OD fitness than the baseline scenario. Only the latter approach outperforms the *BC* approach in terms of count fitness. This means that at small levels of randomness in the initial estimates, a simple heuristic such as *BC* can provide equal or better OD estimates than other approaches. However, we cannot simultaneously minimize fitness with respect to MOPs. This is why the combination of BC, W-SPSA, and Bagging helps to obtain the estimates while ensuring optimal fitness with respect to counts and speeds. For the given scenario, speed errors are low in all the

Approach	**Which MOPs in objective?**		**Count sensor noise**	**Final error value WAPE / RMSE**		
	Count	Speed		Count	Speed	OD
Low noise ($B^x = 0.6$ & $N^x = 20$)						
Initial estimate	-	-	-	0.42 / 288.12	0.03 / 1.71	0.45 / 10.80
W (baseline)	Yes	No	0	0.14 / 89.34	0.02 / 0.85	0.72 / 19.75
BC	Yes	No	0	0.10 / 71.15	0.02 / 0.97	**0.28 / 08.55**
A-W-B	Yes	No	0	0.10 / 58.34	0.02 / 0.70	0.42 / 11.95
BC-A-W	Yes	No	0	0.11 / 72.96	0.02 / 1.00	0.63 / 18.81
BC-A-W-B	Yes	No	0	**0.07 / 44.03**	0.02 / 0.75	0.35 / 09.44
BC-A-W-B	Yes	Yes	0	**0.11 / 81.75**	0.02 / 0.89	**0.41 / 11.35**
BC-A-W-B	Yes	Yes	15	0.16 / 124.96	0.02 / 0.61	**0.41 / 11.31**
BC-A-W-B	Yes	Yes	30	0.28 / 230.30	0.02 / 0.72	0.46 / 14.41
BC-A-W-B	Yes	Yes	45	0.39 / 340.48	0.02 / 0.91	0.45 / 13.90
High noise ($B^x = 0.6$ & $N^x = 200$)						
BC	Yes	No	0	0.17 / 123.12	0.02 / 0.82	1.01 / 27.10
A-W-B	Yes	No	0	0.16 / 123.75	0.02 / 0.91	**0.97 / 27.55**
BC-A-W-B	Yes	No	0	**0.14 / 95.15**	0.02 / 1.06	1.03 / 30.26

W: **W**-SPSA; **BC**: **B**ias-**C**orrection; **A**: **A**utomatic SPSA tuning; **B**: **B**agging

Table 6.3: Scenario 2: Results (WAPE and RMSE) of the Munich scenario with synthetic data

cases and are not sensitive to the count errors/ approach used. This is possible because most of the network is uncongested. Therefore, they add little value to the calibration process.

When we add randomness to the sensor counts, we expect a reduction in fitness to the OD counts since the signal-to-noise ratio of the gradients from MOPs reduces. Therefore, for different sensor noise levels, we see a gradual reduction of OD fitness. Thus, the quality of sensor counts has important implications for the fitness of the OD parameters. Another finding is that in our experiments, using speeds in MOPs leads to higher errors in estimates as compared to using only counts in the MOPs. Since speed error is already low, they do not provide additional signals to the calibration process. W-SPSA essentially decomposes the original problem into multiple smaller sub-SPSA problems. By inclusion of speeds in the objective function, the number of MOPs increases, and due to the non-linear dependence between speeds and OD flows, the complexity of sub-SPSA problems also increases, leading to a drop in the accuracy of the estimates. However, speeds can provide additional context for better convergence in cases where the network is significantly congested. We suppose that the trade-off between additional context from speed data and complexity depends on the level and spread of congestion/ spill-back in the network and could be a matter of future research.

Then we set the OD randomness value to a high value ($N^x = 200\%$) to simulate situations where the initial demand estimates are of poor quality and, thus, the essential structure of the demand is lost. In the existing literature, such extreme scenarios are not considered and tested in OD estimation. We observe the adverse effect of using the *BC* approach in these situations. This is because the BC approach is unreliable when the initial estimates have high random errors; thus, the bias correction is ineffective. Therefore in these cases, *A-W-B* gives the best fitness for OD parameters. Using *BC-A-W-B* provides the best count fitness in this case as well. However, the final estimates are still far from the desired values. When the initial estimates have high errors, there is little hope of recovering the desired estimates using the local search since the proposed methods will tend to converge to the local optima but far from the desired optima.

The effects of bagging on the calibrated OD estimates are shown in Figure 6.8. The two plots on top and bottom correspond to initial estimates with a good initial estimate (low randomness R^x=20%) and poor initial estimates (high randomness R^x=200%). Bagging can benefit both cases, as the OD fitness improves with the number of estimates used for averaging. We can see that averaging four individual estimates leads to most of the improvement in OD fitness. However, the final OD fitness errors are much lower than initial estimates with low random errors. The calibrated estimates in the case of bagging have lower variance, especially in case of low randomness, and is evident by calibrated estimates closer to the 45° line. Another interesting thing to note in Figure 6.8 is that even though OD parameters have a lot of scatter, counts have limited scatter. This implies that the variance in the OD parameters does not proportionally translate into variance in link counts since counts are the weighted sum of the OD flows. Thus, even if the ODs have significant random errors in case of poor estimates, the sensor counts will not have proportionately larger errors. Thus the optimization algorithm will struggle to converge to a locally optimal solution using only counts as MOP, which is undesirable. In

case of poor estimates, the domain specification (l_x, u_x) also needs to be broad enough to include the desired solutions, which will further increase the complexity of the calibration and the possibility of undesirable solutions. Thus, the quality of good initial estimates from auxiliary sources cannot be overstated in the case of OD estimation.

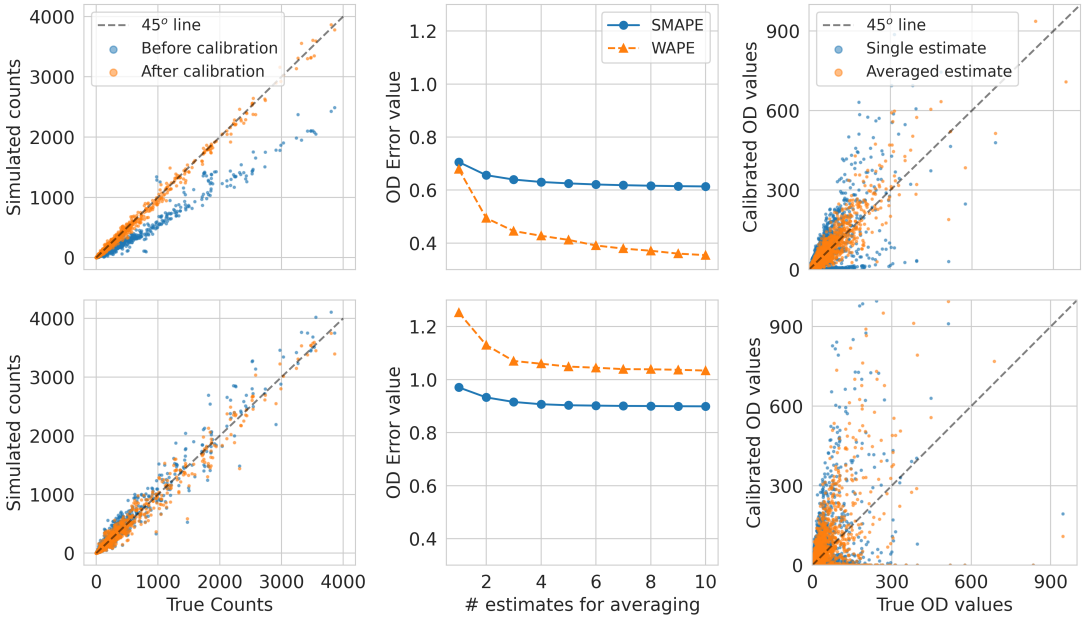


Figure 6.8: Scenario 2: Effects of bagging with initial estimates with (top) small randomness ($R^x = 20\%$) and (bottom) high randomness ($R^x = 200\%$)

6.6.4 Scenario 3: Munich scenario with SUMO and real-world data

This scenario requires a minimum of 400 function evaluations ($S = 2$, $E = 5$, $K = 10$, $B = 50$) or 5-6 days (reduced to 2-3 days if using parallelized W-SPSA and Bagging) to converge. However, these estimates can vary depending on the preciseness of the initial demand and supply parameters. Regressing the error with the supply parameters (R^2=0.90) shows that only *priority factor*, *meso-minor penalty*, *rerouting adaptation*, and *tls travel-time penalty* are significant. We also visually inspect the error surface. Figure 6.9 shows the error surface with supply parameters. A *meso-minor penalty* of less than 10 gives optimal results. The optimal *tls travel-time penalty* is close to 1, and *rerouting adaptation* is less than 5. The optimal *priority factor* lies between 0.35 to 0.60, and *rerouting probability* lies between 0.40 to 0.50; however, lower values of the rerouting probability, such as close to 0.10 are also feasible, conditional on other parameters. Based on the results, we select values of flow penalty, travel-time penalty, and minor junction penalty are 0.57, 0.00, and 0.00, respectively. Rerouting probability, period, and adaptation interval are 0.10, 80, and 1, respectively. We see that multiple values of the combination of supply parameters give the desired or good fitness of the

sensor counts. This points to the fact that additional MOPs from other data sources, such as inter-zone travel times, queue lengths, trajectory data, and travel speeds, should be considered for the further calibration of these parameters.

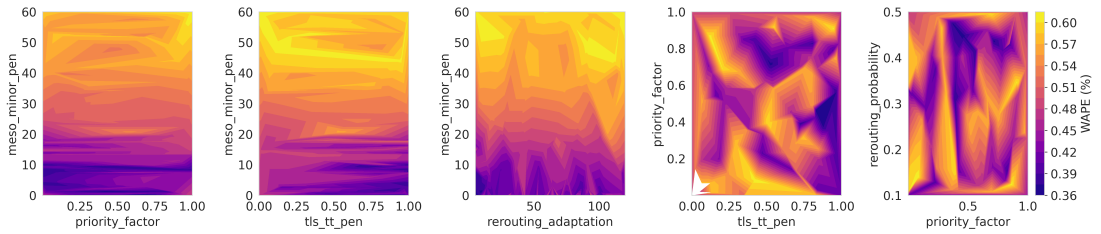


Figure 6.9: Scenario 3: Error surface with the supply parameters

Figure 6.10 shows the plot of simulated and observed link sensor counts during 0700-1000 at different stages of sequential demand and supply calibration. Before demand calibration, the scatter plot is not centered around a 45-degree line for the counts on *other* link types (trunk and primary links), which implies room for improvement. WAPE for *other* links ranges between 1.49 and 1.74. After demand calibration, WAPE for federal (motorway links) is in the range of 0.24-0.40 during 0700-1000 hours. WAPE for other links (trunk and primary links) is in the 0.51-0.53 for the same time interval. Overall, WAPE varies between 0.39 and 0.47, which is lower than the corresponding WAPE before calibration. Simulated counts for federal links during 0800-1000 hours are lower than the corresponding observed counts. After supply calibration, WAPE for federal (motorway) links ranges from 0.19-0.25 for 0700-1000 hours. WAPE for other links (trunk and primary links) ranges from 0.40-0.48 for the same time interval. The overall WAPE varies between 0.32-0.37. Calibration of supply parameters substantially reduces the overall error. The improved match for the federal links during the 0800-1000 is also evident.

We also show the hourly link volumes (Figure 6.11) on the network for the 0800-0900 hour, highlighting the comparison between the uncalibrated and final calibrated models. The difference between the distribution of the flows between the two cases is evident. In the uncalibrated model, there is lesser traffic on the links corresponding to the outer Autobahn ring road (German translation: Äußerer Ring), as well as the middle ring road (Mittlerer Ring), whereas the share of traffic on inner city links is higher. This points to lower impedance on inner roads, so a major share of the traffic selects the routes through these links for their trips. On the contrary, in the calibrated model, traffic distribution is consistent with the observed counts, with a major chunk of trips routed through the outer ring, middle ring roads, and major radial roads. In Figure 6.12, we compare the uncalibrated, calibrated, and observed link speeds in the network. The changes in the speeds between uncalibrated and calibrated models show that certain links (in red) in the former model were congested but not in the latter. Further, we see a reasonable match of link speeds between the observed data and the calibrated model.

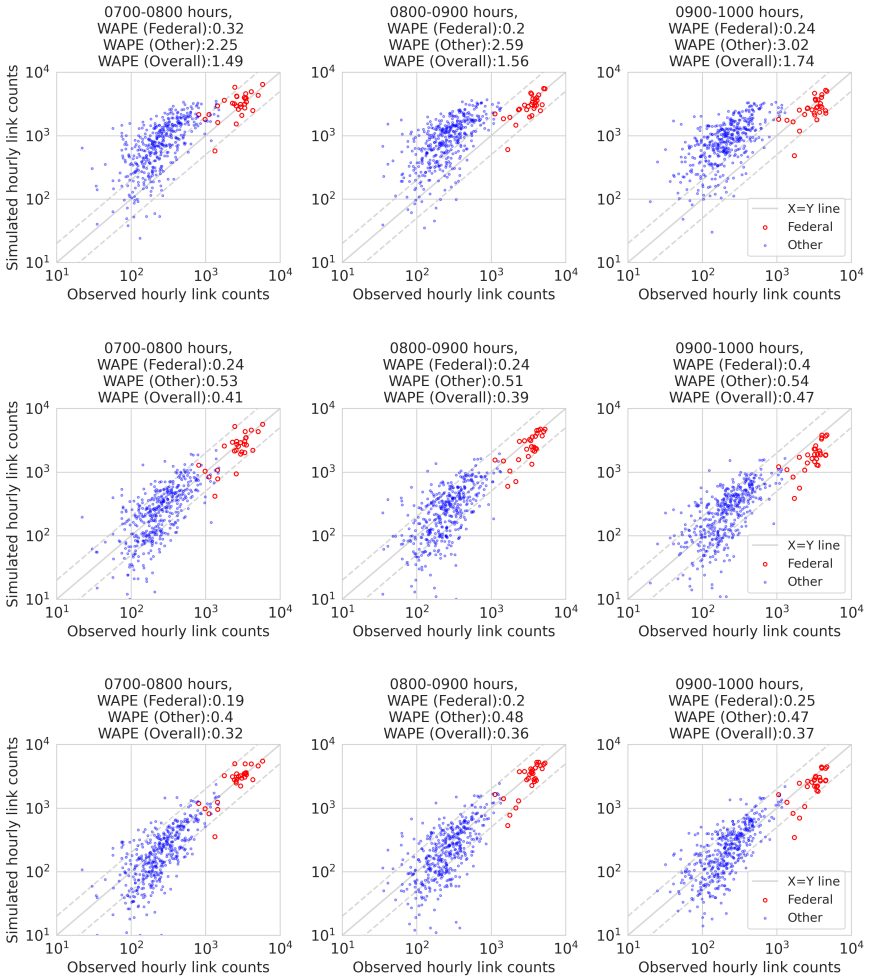


Figure 6.10: Scenario 3: Fitness of link sensor counts before demand calibration (top), after demand calibration (middle), and followed by supply calibration (bottom). Data corresponding to federal links (motorways) and other links (trunk, primary, and secondary links) are highlighted in red and blue, respectively. The Centre line is 45°, or the Y=X line and the lower and upper dotted lines are at Y=X/2 and Y=2X, respectively.

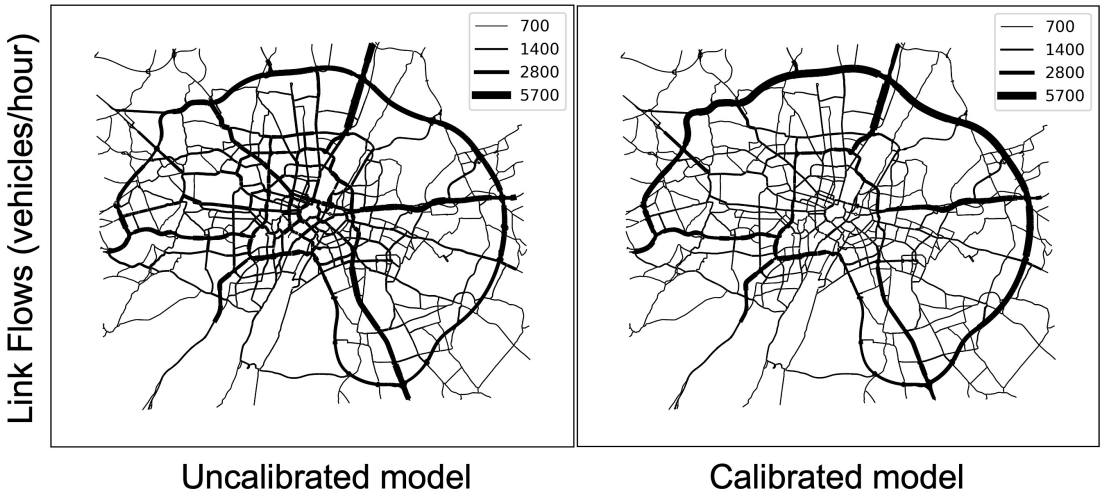


Figure 6.11: Scenario 3: Simulated link volumes during 0800-0900 hours (left) before calibration and (right) after calibration.

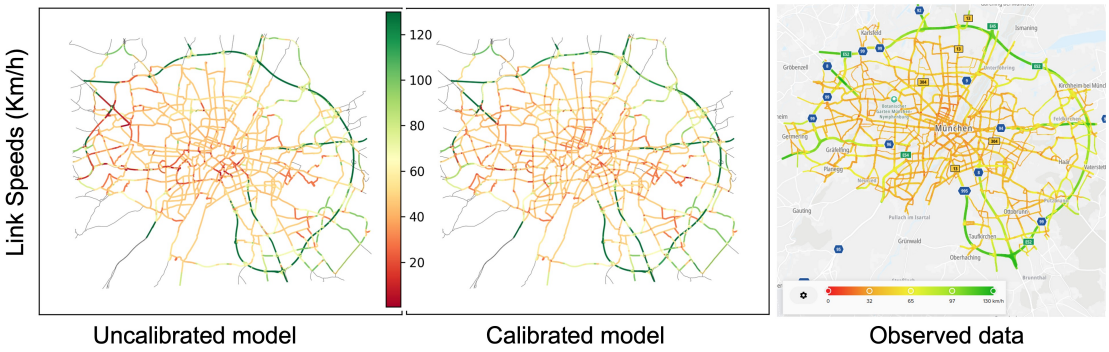


Figure 6.12: Scenario 3: Simulated link speeds during 0800-0900 hours (left) before calibration, (middle) after calibration, and (right) observed data (source: TomTom).

6.7 Summary

The calibration of large-scale traffic simulation models is daunting due to high dimensional objective function, under-determined system, computational burden, and manual effort for parameter tuning. We proposed and tested different approaches to address these challenges. We find that BC-A-W-B provides the best fit of counts in both low and high-noise scenarios with simulation-based assignments. In low-noise scenarios, BC works well to fit ODs and counts (second to BC-A-W-B), but in high-noise scenarios, an approach with bagging provides a better fit. If the information from speed data does not conflict with that from count data, then using them does not lead to additional benefits or even a reduction in accuracy. Further, in high randomness scenarios, count data is insufficient for reliable OD estimation.

Practically, the advantage of bagging is that it can be in parallel, and thus, with parallel compute nodes, it does not cause substantial time overhead. Our approach can help modelers to calibrate their simulation models with little manual effort. By releasing the codes, we also make a practical contribution to OD estimation; there is a large gap between literature and open-source tools.

7 Tackling sparsity of network traffic flows

Contents

The content of this chapter has been presented in the following work:

Mahajan, V., Cantelmo, G., Rothfeld, R., Antoniou, C. (2023). Predicting network flows from speeds using open data and transfer learning. IET Intell. Transp. Syst. 17, 804– 824. doi:10.1049/itr2.12305

7.1 Introduction

We aim to address the scarcity of network-wide dynamic traffic flows/ volumes data using exogenous information from publicly available data. We tackle this challenge using indirect traffic estimation and transfer learning. We aim to model relevant geometric, temporal, and contextual features and prevalent speed data from floating cars to forecast traffic flows reasonably. Further, we want to check if transfer learning helps obtain accurate flow predictions in case of data insufficiency.

The rest of the chapter is structured as follows: the next section lists the research contributions, the following section describes the methodology of the study describing the data processing and prediction methodology, and the following section presents the data collection, followed by data analysis, the next section presents the results of the study, followed by chapter summary.

7.2 Research contributions

Our contributions are as follows:

- We download and curate traffic data for the cities of Paris and Madrid from heterogeneous and publicly available sources. We fuse the collected data for training models to predict traffic flows. Using these data, we also analyze traffic fundamental diagrams and macroscopic fundamental diagrams for in-depth analysis of different link categories. These data are shared with the community for further research.

- We use machine learning (including deep learning) for indirect flow estimation and explore the effect of link types on prediction error, feature combinations, the temporal distribution of errors, and models' uncertainties. We also conduct a systematic sensitivity analysis for different input feature sequences and output flow sequence lengths to identify the best input-output length configuration. [SRQ(9)]

- We transfer the model to a new location and identify the best transfer learning configuration. We also show what proportions of the data transfer learning outperform training a machine learning model from scratch. [SRQ(10)]

- Data collected and processed in this study are shared via the link provided on the GitHub[1] repository.

[1] `https://github.com/vishalmhjn/indirect-traffic-flows`

7.3 Methodology

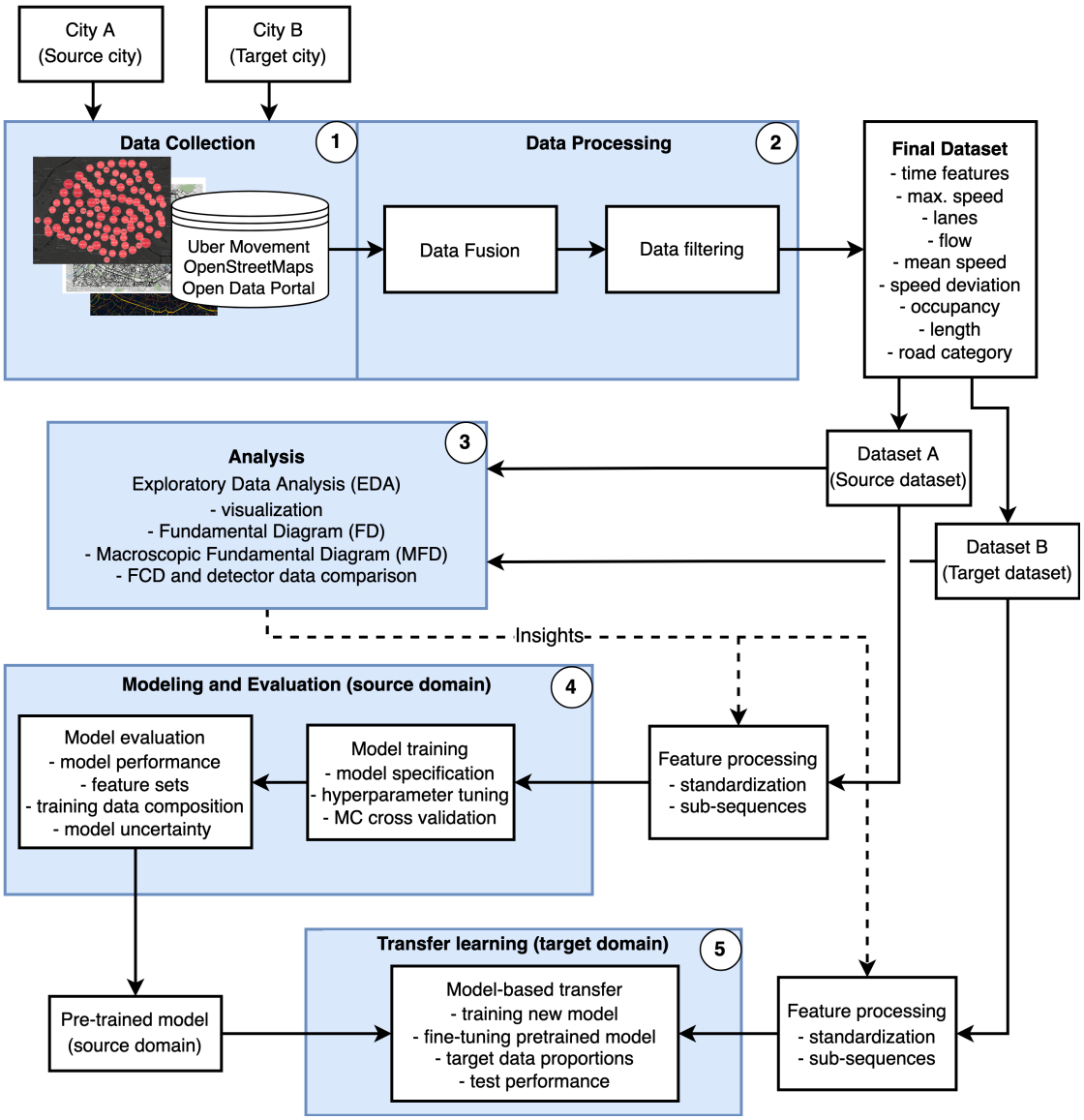


Figure 7.1: Methodological flow showing data collection, analysis, and modeling.

The overall methodology of the study is shown in Figure 7.1. As mentioned above, We rely on the publicly available data in this study. Our research needs traffic flow counts and speed data to train and validate our models for the exact links. We identify prospective study areas based on availability and retrieve the data to achieve this. Since it is expected that the traffic count and speed data are in different formats, we fuse these datasets. Samples of links' speed and traffic counts are matched according to the link/ location in data fusion.

We plot the Macroscopic Fundamental Diagram (MFD) (Daganzo & Geroliminis, 2008) or Network Fundamental Diagram (NFD) (Mahmassani et al., 2013) and traffic fundamental diagrams to understand the traffic state dynamics. We use the weighted average formulation by Geroliminis and Daganzo (2008) to represent the MFD mentioned below:

$$q_t^w = \frac{\Sigma_i q_{it} l_i}{\Sigma_i l_i} \qquad (7.1) \qquad o_t^w = \frac{\Sigma_i o_{it} l_i}{\Sigma_i l_i} \qquad (7.2) \qquad s_t^w = \frac{\Sigma_i s_{it} l_i}{\Sigma_i l_i} \qquad (7.3)$$

where, q^w, o^w, and s^w are average lane-flow, occupancy and speed, respectively, l_i is link length, q_{it}, o_{it}, and s_{it} are flow, occupancy and speed for the i^{th} link at time t. Separate MFD is estimated for each link-type category. Factors such as spatial distribution of the congestion and location of detectors can affect the shape, scatter, and the existence of a well-defined MFD (Geroliminis & Sun, 2011).

Further, this paper aims to develop a model that predicts traffic flow in transport networks exclusively from given link speeds and other relevant time and contextual covariates. Thus, we adopt a formulation for indirect traffic estimation and assume the unavailability of time-lagged flow data as a predictor. Inspired by Mallick et al. (2021), we borrow and adapt their problem formulation for our case. Given, a set $\mathcal{S}$ of d links, the traffic flow at time step t for a d^{th} link is $Y_t^d \in \mathbb{R}^1$. For the d^{th} link, given static predictors $W^d \in \mathbb{R}^E$, where E is the number of such predictors (length, number of lanes, road width, type, speed limit) and H historical observations of dynamic exogenous predictors (speed, time, speed deviation) $X^d = \left(X_{t_1}^d, X_{t_2}^d, \ldots, X_{t_H}^d\right) \in \mathbb{R}^{H \times F}$, H historical observations of the traffic flow $Y^d = \left(Y_{t_1}^d, Y_{t_2}^d, \ldots, Y_{t_H}^d\right) \in \mathbb{R}^H$, P current observations of the same predictors $X^d = \left(X_{t_1}^d, X_{t_2}^d, \ldots, X_{t_P}^d\right) \in \mathbb{R}^{P \times F}$, where F is the number of such predictors and $H >> P$, we want to forecast the traffic flow of the next Q time steps, $\hat{Y^d} = \left(\hat{Y^d}_{t_{P+1}}, \hat{Y^d}_{t_{P+2}}, \ldots, \hat{Y^d}_{t_{P+Q}}\right) \in \mathbb{R}^Q$. Thereafter, Let $\mathcal{S}'$ be the set of j links for which we do not have the historical time series data. Given static predictors $W^{j\prime} \in \mathbb{R}^E$, and P observations of the current exogenous predictors for j^{th} link $X^{j\prime} = \left(X_{t_1}^{j\prime}, X_{t_2}^{j\prime}, \ldots, X_{t_P}^{j\prime}\right) \in \mathbb{R}^{P \times F}$, the goal is to develop a model that can forecast the traffic flow of the next Q time steps for all the links in $\mathcal{S}'$, $\hat{Y^{j\prime}} = \left(\hat{Y^{j\prime}}_{t_{P+1}}, \hat{Y^{j\prime}}_{t_{P+2}}, \ldots, \hat{Y^{j\prime}}_{t_{P+0}}\right) \in \mathbb{R}^{Q \times F}$.

7.3.1 LSTM model

Our task can be formulated as supervised machine learning because the model learns the mapping from features to the given targets. We select LSTM networks as our primary model. LSTM is appropriate for modeling time-series data such as traffic flow or speed, where correlations between time intervals have a long lag. As discussed above, other techniques (such as GNN or CNN) could introduce network dependencies. Still, LSTM is a popular sequential deep learning model. Recent studies have used LSTMs (Li et al., 2021; Mallick et al., 2021) and indicate that they are still a competitive model choice. In studies (Abdelraouf et al., 2022; Mallick et al., 2021) using GNN for short-term traffic

forecasting, we observed that even though graph-based models (DCRNN) give the best performance, LSTM's performance is still competitive.

LSTM is considered a more advanced version of the standard vanilla RNN. Traditional RNNs can model a sequence of events by propagating information through time. RNNs achieve this by using the output from the previous time interval as an input to predict the system's state in the current time interval. In equations:

$$h_t = \tanh\left(W_{ih}x_t + b_{ih} + W_{hh}h_{t-1} + b_{hh}\right) \tag{7.4}$$

Where h_t is the hidden state at time interval t, concerning the other variables, x_t represents the input vectors - i.e., input data at the current time interval - while W_{ih}, W_{hh}, b_{ih}, and b_{hh} are model specific parameter matrices and vectors that dictate how each element of the input data (or the hidden state) contributes to the prediction. In Equation 7.4, we show a tanh activation function as this is the most common one in the case of RNNs, but other activation functions can also be used instead. The critical aspect of an RNN is, therefore, that the hidden state works similarly to the lagged variable in an autoregressive model, meaning that the predictions at the current time interval h_t depend on the hidden state at the previous time interval h_{t-1}.

Although a successful architecture, RNNs suffer from vanishing/ exploding gradients and short-term memory problems. Furthermore, because of their simple structure, RNNs cannot memorize long data sequences and begin to forget previous inputs. As stated, the hidden state h_{t-1} carries much information about the previous time interval, but RNNs cannot capture correlations with very long time lags. To alleviate the issues above, LSTM introduces the cell state c_t in addition to the existing hidden state of RNNs. LSTM consists of a memory cell that controls the flow of information by using input, forget, and output gate layers that discard non-essential information and memorize only essential information. This complex architecture updates the cell state c_t and selects which information should be preserved and lost. Therefore, LSTM is a deep learning architecture that uses the cell state c_t next to the already mentioned hidden state h_t to provide the model with longer memory over past events. The operations in the LSTM model (Hochreiter, 1991; Paszke et al., 2019) can be represented by the following set of equations:

$$\begin{aligned}
i_t &= \sigma\left(W_{ii}x_t + b_{ii} + W_{hi}h_{t-1} + b_{hi}\right) \\
f_t &= \sigma\left(W_{if}x_t + b_{if} + W_{hf}h_{t-1} + b_{hf}\right) \\
g_t &= \tanh\left(W_{ig}x_t + b_{ig} + W_{hg}h_{t-1} + b_{hg}\right) \\
o_t &= \sigma\left(W_{io}x_t + b_{io} + W_{ho}h_{t-1} + b_{ho}\right) \\
c_t &= f_t \odot c_{t-1} + i_t \odot g_t \\
h_t &= o_t \odot \tanh\left(c_t\right)
\end{aligned} \tag{7.5}$$

Where h_t: hidden state of layer, c_t: cell state, x_t: input, all at time t; h_{t-1}: hidden state of the layer at time $t-1$; and i_t, f_t, g_t, o_t are the input, forget, cell, and output gates, respectively. σ is the sigmoid function, and $\odot$ is the Hadamard product. Equations are sourced from Torch Contributors (2019).

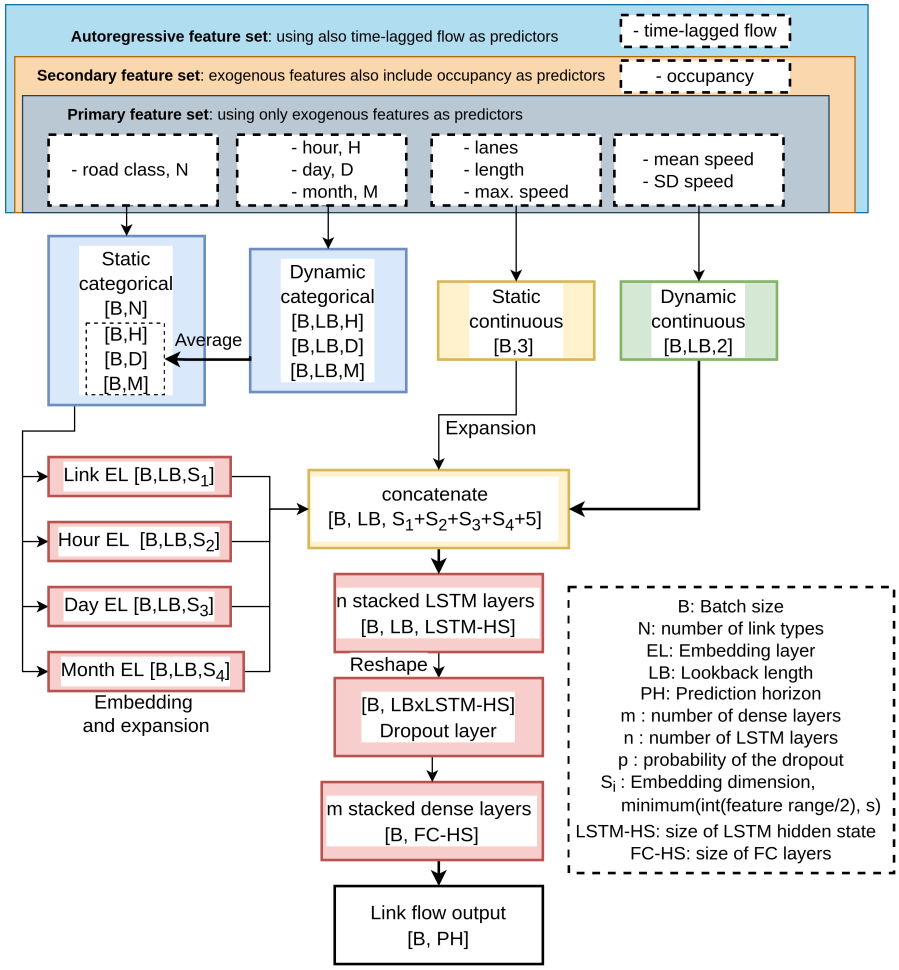


Figure 7.2: Architecture of the deep learning model using embedding and LSTM layers

7.3.2 Model architecture

The input data for the LSTM model can be static or dynamic and continuous or categorical, leading to four different input types (Figure 7.2): static categorical inputs (links type with N classes), dynamic categorical inputs (an hour with H classes, a day with D classes, the month with M classes), continuous static inputs (link length, number of lanes, maximum speed), continuous dynamic inputs (hourly speed). In our model (Figure 7.2), we use *embedding* layers to process the categorical features. Embedding layers use a fixed-length continuous vector to represent a categorical feature. The embeddings are learned during model training, and similar feature categories will have closer representations in the embedding space. To reduce the need to learn the multiple embeddings for dynamic categorical features such as *hour*, *weekday*, and *month*, we convert the dynamic categorical features to static features by taking their average value over the lookback length. This gives us single values of these features for each training sample, irrespective of the lookback length. The single value provides the time context for the day, weekday, and

month. The size of the feature's embedding layer (S) is a minimum of s and half the number of unique values of a feature, where s is a hyperparameter. This step reduces the number of feature embeddings to be learned and, thus, reduces the complexity of the model. The dynamic features are provided for a certain past/ lookback length, so the static embeddings and features are expanded along the time dimension. The *concatenate* layer combines then continuous and embedding outputs to form a single time-dependent input for the next LSTM layers. This input is passed through a stacked n LSTM layer(s) sequence. LSTM layer(s) compute the function in Equation 7.5 at each time step or hidden state to produce the output equal to the latent dimension of hidden state.

Dropout is applied to the output of the final LSTM layer to randomly switch off neurons with a probability p. Dropout is a regularization technique to reduce overfitting (Srivastava et al., 2014) and helps to improve the model's robustness. Dropout can also be used to understand the model's uncertainty (Gal & Ghahramani, 2016). Later during the model testing, we used dropout to obtain the uncertainty estimates by running the model several times (e.g., ten times) and obtaining the confidence intervals for the model predictions. The output of the dropout layer is fed to the stacked m fully connected or dense layer(s) to give a single output or a sequential output (in the case of multi-step forecasting), as shown in Figure 7.2.

7.3.3 Feature sets

We construct three sets of features to observe the model performance differences, as shown in Figure 7.2. The first set of features for the primary model is only the *exogenous* features, which can be either static or dynamic. This set of features is the main focus of this paper. These features exclude flow and occupancy data from the detectors and thus only use easily available data. In the second set, we add time-lagged occupancy (o) data from detectors to the exogenous set (*exogenous covariates,* o). In the third set, we further add time-lagged flow (q) to the second set resulting in (*exogenous covariates,* o, q). When using the third feature, our model follows autoregressive formulation. We hypothesize that model performance will improve by including o and q into the feature set since the model has direct and endogenous information to predict the flow. However, this information cannot be used when the objective is to predict traffic volumes on links not equipped with sensors. Before training, features in source and target data are standardized (removing the mean and scaling to unit variance) independently.

7.3.4 Model evaluation

We use the XGBoost regression model as the benchmark model due to its lesser computation burden as compared to deep learning models. XGBoost has shown superior performance on tabular datasets in research and practice (Shwartz-Ziv & Armon, 2021). Therefore, we use XGBoost with non-sequential inputs. This means we only provide the input data for the current time step but not the past intervals while predicting one step in the future. The XGBoost model (T. Chen & Guestrin, 2016) is based on the GBM concept. In boosting, observations with high residuals generally receive ever-increasing

influence with each iteration (Hastie et al., 2001). Boosting models are generally considered "off-the-shelf classifiers" (Hastie et al., 2001) and need less feature preprocessing and parameter tuning than deep learning models such as neural networks.

The dataset corresponding to all links is split into a train set (85%) and a test set (15%). We use group-based splitting (using detector ID) of the data, which means that data from each detector or link can be into either a train set or test set to avoid overestimating the model performance due to temporal correlations within the data from one detector. This also mirrors the scenario of data availability for partial links in the network. The hyperparameters of the LSTM model used in this paper are the size of the embedding layers, learning rate, batch size, maximum epochs, number of LSTM layers (n), size of the LSTM hidden state, dropout rate (p), and weight decay or L-2 type regularization of weights (w), and number (m) and size of the dense layers. The primary hyperparameters of the boosting model are the number of iterations and the size of each constituent tree (number of leaves in the tree) (Hastie et al., 2001). The model training is stopped when the validation error does not improve for twenty iterations. This is also known as early stopping. We use Bayesian optimization (Nogueira, 2014) to tune the hyperparameters of the XGBoost and LSTM models. During tuning, the Monte Carlo cross-validation (MCCV) error is used. In MCCV, the model is trained by randomly splitting the training data into training (85%) and validation data (15%) for ten runs. The average error on the validation data is used to select the best hyperparameters.

The choice of forecasting metric is crucial and varies from task to task. In our case, we have a time series corresponding to each detector. The target variable (flow) scale can vary between links in different categories. Thus, we use percentage error metric (Rink, 2021). Mean Absolute Percentage Error (MAPE) is one of the popular percentage error metrics. However, we do not use MAPE because it has no upper bound and can be problematic when the actual values are close to zero. Instead, we select Symmetric Mean Absolute Percentage Error (SMAPE), shown in Equation 7.6, as the model training and evaluation criterion due to the time-series nature of the input data.

$$SMAPE_d = \frac{1}{n} \sum_{i=1}^{N} \frac{|x_i - \hat{x}_i|}{(|x_i| + |\hat{x}_i|)/2} \tag{7.6}$$

Where SMAPE_d is the SMAPE for the d^{th} link, $\hat{x}_i$ is the predicted value, and x_i is the observed value. In contrast to MAPE, SMAPE has both a lower and upper bound. It is noted that the data within each detector is correlated. Thus we use mean error over detectors instead of the sample means to prevent the dominance of detectors with large samples. In other words, we first estimate the error for each detector and then estimate the mean error over all the detectors for reporting the model performance. One of the drawbacks of SMAPE is that it does not treat large positive and negative errors equally and thus is not "symmetric" as its name suggests (Goodwin & Lawton, 1999). No single metric is sufficient for forecasting, so we also use RMSE for model evaluation. However, RMSE is only used when the target scale is similar, so the evaluation is fair, e.g., when the data for a single link type is used. We use the Python frameworks XGBoost (T. Chen

& Guestrin, 2016) and Pytorch (Paszke et al., 2019) for developing the XGBoost and the LSTM model, respectively.

7.3.5 Model transferability

We select the best-trained model on the source data and check its generalizability if the model can be applied to study areas without or insufficient traffic flow data. This is done using transfer learning. We collect and prepare the source data ($\mathcal{D}_S$, $\mathcal{T}_S$) and target data ($\mathcal{D}_T$, $\mathcal{T}_T$) with same set of features ($\mathcal{X}_S = \mathcal{X}_T$) and same type of labels ($\mathcal{Y}_S = \mathcal{Y}_T$). Still, there is no guarantee that the feature's marginal distributions ($P_S(X) \neq P_T(X)$) and the label conditional probability distributions ($P\left(Y_S \mid X_S\right) \neq P\left(Y_T \mid X_T\right)$) are similar among the source data and target data. For instance, link attributes and traffic flow patterns can vary between locations. Thus, this is a case of transductive transfer learning (S. J. Pan & Yang, 2010), or domain adaptation (Redko et al., 2019). We use model-based transfer where a pre-trained model is used. This means that the weights of parameters in the pre-trained model are used as priors or initial values for the target task. Further, the model can be used for the target task without any changes or further retraining or fine-tuning on the sample of target data. We devise a systematic method to transfer the trained model as listed below and check model performances to find the best transfer learning scenario.

1. Baseline model without transfer learning. Here, the model has the same architecture as the source task, but randomly initialized parameters are used. This model is trained and tested on the target data only; thus, it has no knowledge transfer from the source task.

2. Transferring the model pre-trained for source task without retraining on the target data. Model architecture is also not changed.

3. Transferring the model trained for the source task by fine-tuning one or more Fully Connected (FC) layers, LSTM layers, and embedding layers, but the rest of the model is frozen. For instance, LSTM layers with parameters from the source task are fine-tuned on target data. In contrast, the parameters of the rest of the layers remain fixed. Model architecture remains the same as in the source task.

For the target domain, we test our model under different proportions of training data, simulating the scenarios of limited training data availability. For evaluating the model transferability, we use twenty runs of MCCV. First, target data is divided into training and test sets. In each MCCV run, target training set detectors are further divided into the training and validation set according to the training proportion, e.g., if the training proportion is 0.65, then 65% of detectors (excluding detectors corresponding to test links) are assigned for re-training the pre-trained model, and rest are used for the model validation (e.g., early stopping). After model training, a test set is used to evaluate all the models from all the runs. Thus, our approach can capture the sampling variability during model transfer.

Figure 7.3: Full Road network within the ring road of Paris (left) i.e., the source domain, and Madrid (right) i.e., target domain. Maps created using Python library OSMnx (Boeing, 2017)

.

7.4 Data collection

We use Paris (region within Paris' ring road or Boulevard Peripherique) as our primary study area or source domain/ city (Figure 7.3), as the Paris open data portal (Open Data Paris, 2020) provides historical traffic flow/volume and occupancy data. For this study, we assume we have a sufficient source dataset for training and testing our model. We train our original model using these data.

Traffic flow data (dependent variable) are collected from the traffic sensors (loop detectors) installed on the road. The data for the full year of 2019 was retrieved. We use these data to train our machine and deep learning models. The raw dataset from the portal is at the aggregation interval of one hour, and it defines the predictive resolution of our models. Our models cannot predict for a horizon of less than one hour.

We use link speed data from the Uber Movement portal for the same study area and period. Uber, a Transportation Network Company (TNC), provides aggregated speeds by road segments at hourly granularity (Uber Movement, 2020b). The speed values are derived from average speed readings from on-trip ride-hailing vehicles associated with the Uber (Uber Movement, 2020b). The raw data in GPS pings are ingested in real-time every four seconds. Uber performs map matching based on a Hidden Markov chain Model (HMM) to assign the GPS pings to a road segment. These map-matched data are used to calculate traversal speed per segment (Uber Movement, 2020b). Speed is given by dividing the length of the road by the time a vehicle takes to traverse it. Uber does not publish speed if the number of traversals is below a minimum threshold to safeguard privacy. Finally, the speed traversals are aggregated into time windows during a time

interval. We retrieved the data from Uber Movement (2020a). In the retrieved data, each road segment has a mean speed and a speed deviation at hourly granularity in 2019.

We use Madrid as the secondary study area or target domain/ city for investigating transfer learning performance. Open Data Madrid provides historical data from flow, occupancy, and speed (only for inter-city roads) at an aggregation interval of 15 minutes (Open Data Madrid, 2022). The data are aggregated at intervals of one hour so that the attributes are consistent with the model trained using Paris data. Link speeds for Madrid are also available from Uber Movement (one-hour intervals). For Madrid, therefore, we have link speed data from two sources, and we can compare these two sources to check the plausibility of the FCD data (from Uber Movement).

To match the flow and speed data, we use Shared Streets, a standard for streets, i.e., roads are assigned a unique identifier for referencing. Sharedstreets (2018) also provides a tool to match the geographic objects (in the form of points and edges) using a probabilistic HMM map matching (Sharedstreets, 2017). Since the flow and speed datasets are geo-referenced, we utilize this tool to map traffic flow and speed data to a common Shared Streets standard and then merge them. We also retrieve the road's static features from the OSM. These features include length, type, number of lanes, and speed limit. During data fusion, we dropped the roads/ links if static features (number of lanes or speed limit) were missing.

In the OSM data, highway segments are classified into *motorway*, *trunk*, *primary*, *secondary* and *tertiary*, *unclassified* and *residential*, according to their importance in the road network. Further, the segment or link types such as *motorway_link*, *trunk_link*, *primary_link* and *secondary_link* refer to the slip roads/ramps and physically separated at-grade turning lanes in the OSM data. For definitions of these links, we refer the reader to OSM documentation (OpenStreetMap, 2021, 2022). Finally, we do not consider the effects of dynamic traffic management on features (such as dynamic speed limits) derived from OSM data because such data are unavailable. The flow-speed-OSM matched data consist of a time series for each of the road segments with static (geometric and contextual) and dynamic (speed and flow) features (Figure 7.1).

7.5 Data analysis

We show the trends of the mean flow, speed, and occupancy for different link types during the day in Figure 7.4 (top row). Trunk-type links show the highest average flows. Figure 7.4 (middle and bottom row) also shows the average traffic states or MFD (flow, occupancy, speed) for all links within the source city. Each point in the plot corresponds to a time interval of one hour. In this figure, we have not adjusted or filtered the data to account for the homogeneity of the congestion since it is not the focus of this study. Still, the existence of MFD is evident, albeit some link types show more scatter than others; for instance, the trunk type links show well-defined MFD over a wide range of speed values and occupancy (Figure 7.4). This is a crucial element for our experiment, as we focus only on temporal aspects and do not explicitly consider network characteristics (for example, through a GNN). MFD for other link types (residential-type links) shows

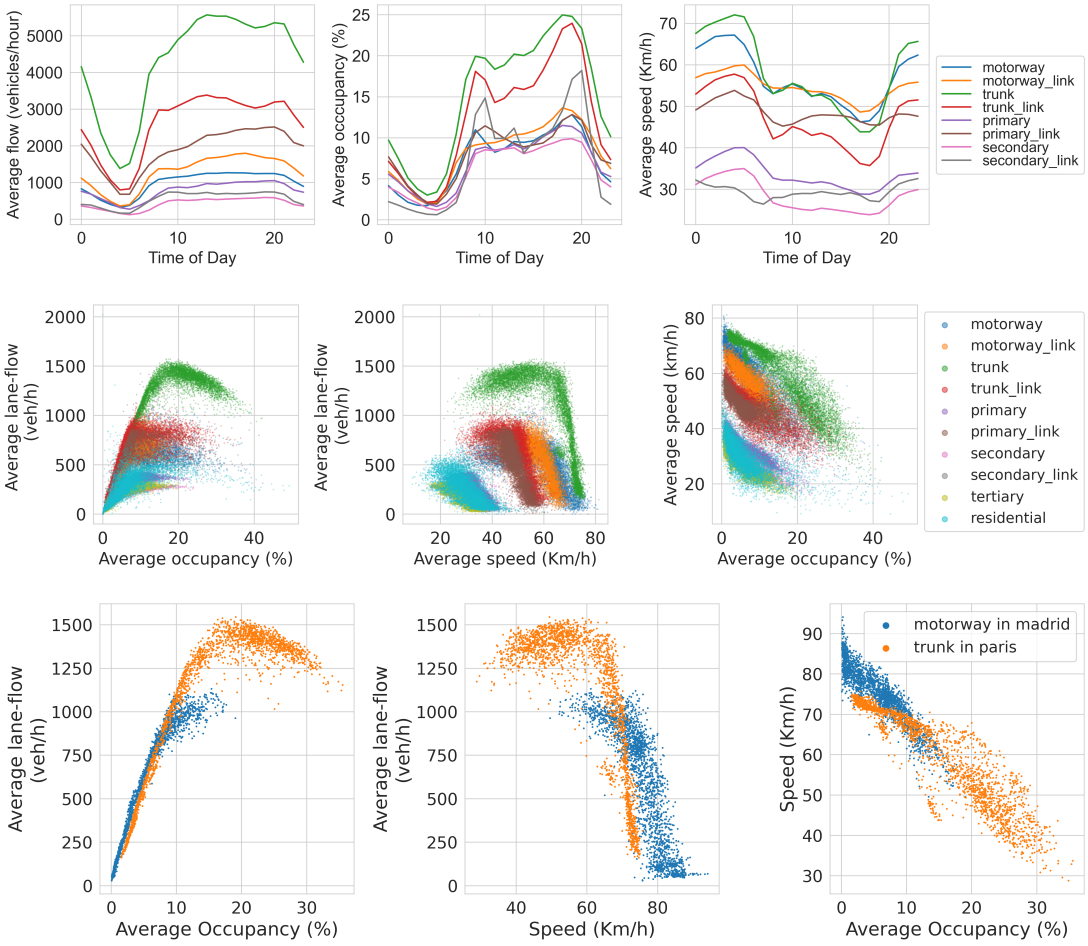


Figure 7.4: Average trends (top) of the speed, flow, and occupancy for the links from source city, MFD shows average flow, occupancy, and speed for (middle) all links from source (Paris) city. MFD (bottom) for links from source and target (Madrid) cities for trunk and motorway links, respectively.

more scatter and is mostly confined over a narrow range of speed values. We suppose this is typical for urban roads with speed limits on the lower side (50 km/h or 30 km/h). Further, the scatter is prominent in all link types during high average occupancy or low average speeds. This could be due to heterogeneity in the congestion patterns leading to the loss of well-defined MFD.

Data for time instances, when any of the link's dynamic (flow or speed data) features were missing were also dropped. Low-importance roads have more missing speed data values. The remaining Paris data has median completeness of 68%, 63%, and 96% over a full year (8670 hours) for primary, secondary, and trunk links (Table 7.1), respectively. For Madrid, the median data completeness is 81% (motorway-type links). Due to the generally high rate of data completeness for these link types, we have enough samples (Table 7.1) for training and testing our models. Otherwise, as a pre-processing step, data imputation can be needed if data completeness is low (X. Chen et al., 2022).

Table 7.1: Descriptive statistics of the features in the pre-filtered Paris and Madrid data

	Max. speed (km/h)	Length (m)	Lanes (no.)	Hourly speed (km/h)	Speed SD (km/h)	Hourly flow (vehicles/h)
Pre-filtered Paris data						
Link type: **Primary**; detectors: 809; samples: 4.28×10^6						
Min	30.00	10.05	1.00	0.39	0.07	0.00
Mean	48.85	187.94	2.77	29.89	10.27	654.19
Median	50.00	133.97	3.00	29.62	9.81	540.00
Max	60.00	933.01	5.00	126.36	70.99	11152.00
SD	4.33	153.59	1.05	10.24	3.84	502.94
Link type: **Secondary**; detectors: 363; samples: 1.88×10^6						
Min	30.00	13.37	1.00	0.47	0.06	0.00
Mean	46.42	143.42	2.50	27.57	9.71	450.58
Median	50.00	108.87	2.00	27.71	9.32	364.00
Max	50.00	607.46	5.00	107.36	63.44	6152.00
SD	7.28	108.49	0.89	8.25	3.64	342.72
Link type: **Trunk**; detectors: 122; samples: 9.60×10^5						
Min	50.00	97.20	2.00	0.83	0.11	0.00
Mean	69.80	609.13	3.77	58.13	10.28	4149.65
Median	70.00	581.63	4.00	65.17	9.38	4378.00
Max	70.00	1362.56	5.00	92.38	62.97	9021.00
SD	1.90	243.62	0.58	16.86	3.56	1865.58
Pre-filtered Madrid data						
Motorway; detectors: 129; samples: 7.27×10^5						
Min	50.00	27.06	3.00	3.06	0.27	0.00
Mean	81.84	843.82	3.41	75.02	11.04	2299.09
Median	90.00	558.58	3.00	78.89	9.67	2117.75
Max	100.00	4658.27	5.00	141.77	67.55	8632.00
SD	12.42	925.01	0.62	15.56	5.77	1561.87

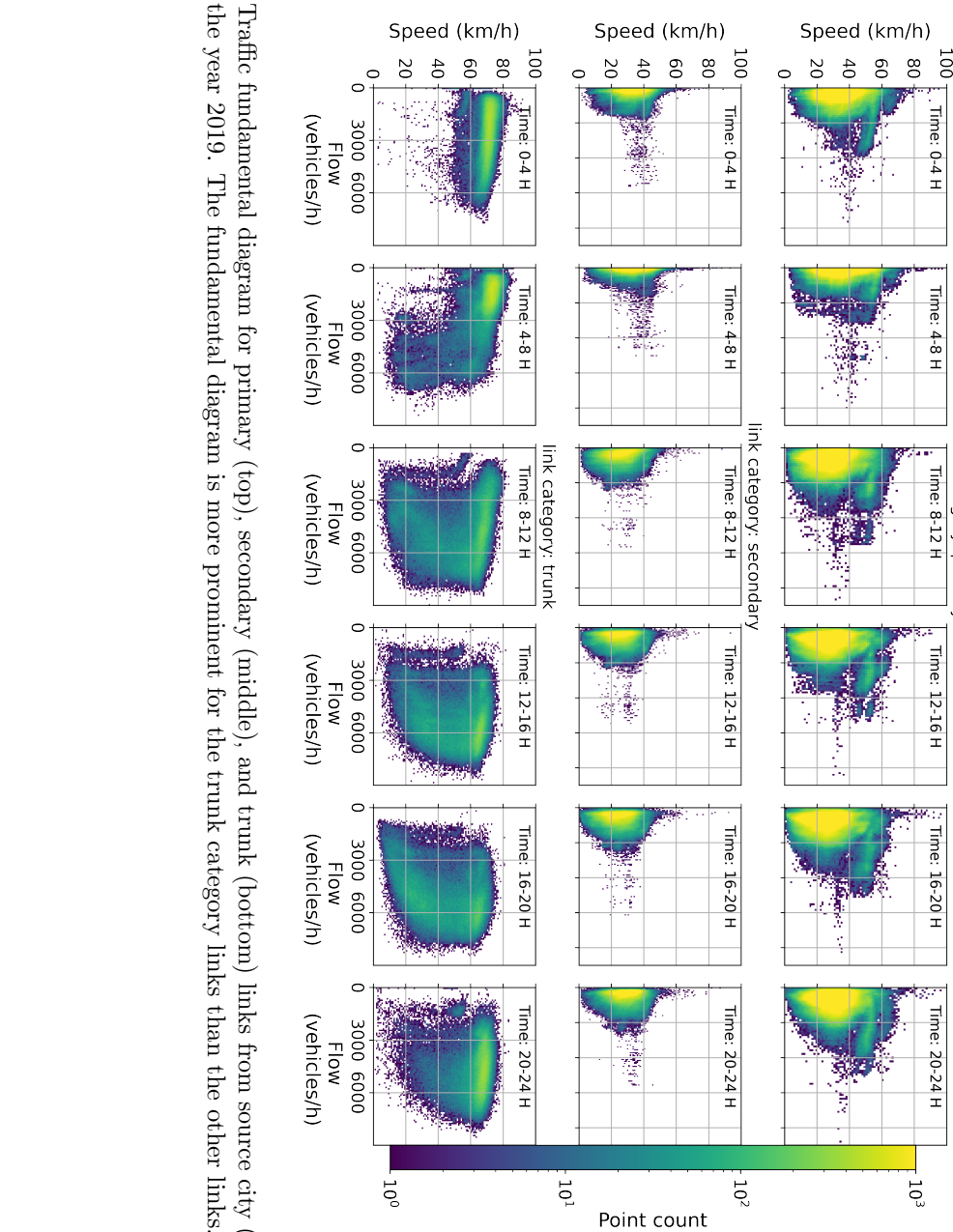


Figure 7.5: Traffic fundamental diagram for primary (top), secondary (middle), and trunk (bottom) links from source city (Paris) during the year 2019. The fundamental diagram is more prominent for the trunk category links than the other links.

We provide the descriptive statistics for the selected links in Table 7.1. For the source city, flow data from primary, secondary, and trunk link types constitute 90% of the dataset, with about 8.2 million samples for 1290 unique links. Source data for Paris consists of data of 809, 363, and 122 links for primary, secondary, and trunk links, respectively. Primary and secondary links show similar speed characteristics with a mean speed of around 27-30 Km/h, whereas trunk links have a mean speed of 58 Km/h. Both primary and secondary links are shorter than the trunk links and have, on average fewer lanes. The mean hourly flow on primary and secondary links is about 650 and 450 vehicles/hour, respectively, whereas trunk links have a significantly higher mean flow of 4150 vehicles/hour. In Table 7.1, it can be seen that the scale, as seen from the mean and standard deviation (SD), of the flow values, are different for the primary, secondary, and trunk link types. Also, flow variance in primary (503 vehicles/hour) and secondary (342 vehicles/hour) links is lower than that of trunk links (1865 vehicles/hour). This is why we do not report RMSE when all links are considered in the training data because RMSE is a scale-dependent metric.

In the traffic fundamental diagram in Figure 7.5, it is evident that primary and secondary type links have more scatter than others. For instance, the trunk links display the evolving relationship between the flow and speed over a wide range of values at different times. From midnight to early morning, traffic remains largely in a free-flow regime. We see the typical fundamental diagram from the morning to the evening hours, wherein links are either in free-flow, transitions, or congestion regimes. This finding is essential for exogenous flow modeling because speed is clearly correlated with the flow for the trunk-type links in the dataset. On the other hand, the same is not valid for the primary and secondary type links as their speed-flow plot is only confined within a limited range. One of the possible explanations is the existence of speed limits on the lower side (50 km/h or 30 km/h) which prevents the manifestation of the fundamental diagram over a wide range of values. Some scattering is also because plots are not controlled for variables such as speed limit and lanes.

For the target data, we have data from 129 motorway links. Figure 7.6 shows the fundamental diagram for these links. The mean speed on these links is higher (75 Km/h), and the mean flow is lower (2300 vehicles/hour) than those on the trunk links in source data. This shows that features in source and target data have different distributions. However, their standard deviation of flow (1561 vs. 1865 vehicles/hour) and speed (15.5 vs. 16.8 Km/h) are similar. Although trunk and motorway link features have different distributions, their ranges overlap significantly (Figure 7.7). This makes us confident that a model trained using source trunk links is a better candidate for transfer than using a model trained with all link types.

While comparing the speed data from detectors and FCD sources for Madrid, we find that the data are not uniformly consistent across the traffic states, as shown in Figure 7.8. The mean error is high in regions of low data density (low speeds, very low flows, and high occupancy). This shows that FCD data is not reliable in these ranges. For flow values greater than 400 vehicles per hour, the mean percentage error stays within -10% to 5%. Speed data from UBER movement is more trustworthy in the flow higher than 400 vehicles per hour. The high percentage error occurs for low values of the speed,

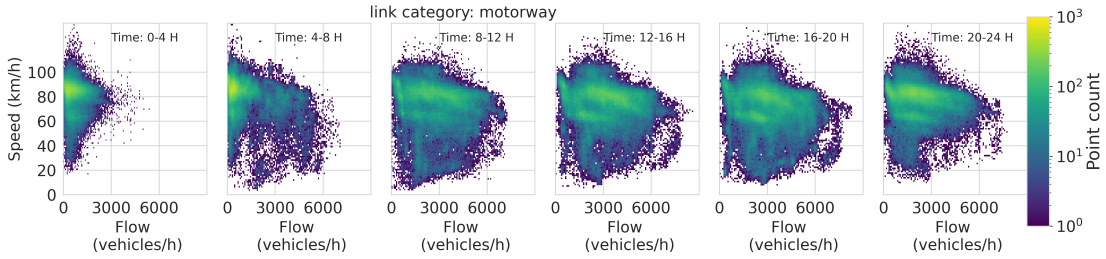


Figure 7.6: Traffic fundamental diagram for motorway links from target city (Madrid).

i.e., less than 50 Kmph, and for higher values of link occupancy, i.e., greater than 30%. Aggregated data in low-speed regimes from conventional detectors are found to be noisy (Coifman, 2014). We conclude that link speed data from the UBER movement dataset is consistent with the detector-measured speeds for high flows and low occupancy or higher speeds.

Based on the above analysis, we conclude that it makes sense to develop two types of models from source data based on the link types in the input data. The first model type considers datasets from all three source link types for training. The second model only uses data from source trunk-type links. Contrasting between the two models helps us confirm our belief regarding the adverse effects of the scattering in the fundamental diagram, distinct feature statistics, and speed data errors on the flow prediction model's performance.

7.6 Results

7.6.1 Indirect estimation performance

The list of hyperparameters and their range of possible values for search is shown in Table 7.2. The best parameters for the XGboost and LSTM models are also shown in Table 7.2 and are based on the best SMAPE on the validation dataset. When the input data contains all links (primary, secondary, and trunk), both models fail to achieve good SMAPE on the test data and are under-fitting (Table 7.3). SMAPE of XGBoost and LSTM models are 51.76% and 40.17%, respectively. This finding was expected due to the lack of structure in the fundamental diagram for primary and secondary links and thus, a weak correlation between speed and flow. Still, the LSTM model fits better than the XGBoost. For the input data with only trunk type links, the LSTM model again performs better than the XGBoost model in terms of both SMAPE and RMSE on test data (Table 7.3). The LSTM model outperforms the XGBoost model in test SMAPE and RMSE by approximately 21% and 13%, respectively. SMAPE and RMSE of the LSTM model on the test data are comparable to those on the training data, showing that the LSTM model can better generalize on the unseen data. Test SMAPE of the XGBoost model is much higher than on the training set, indicating an overfitting problem. Lastly, the confidence intervals of the LSTM model (Table 7.3) are narrower than the confidence

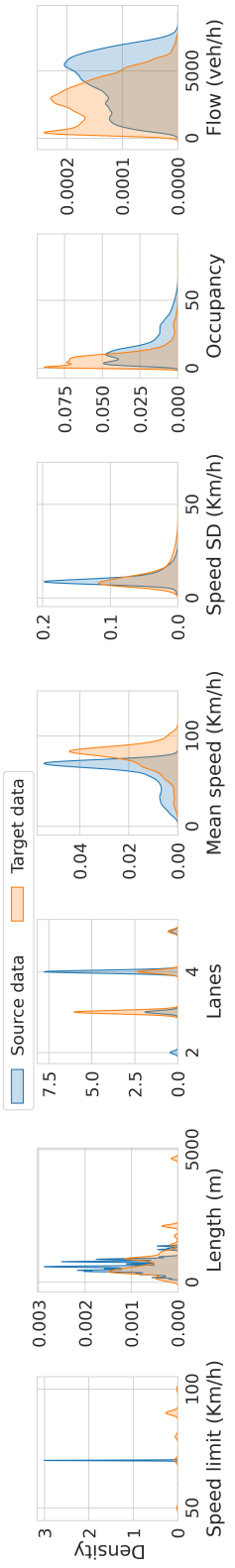


Figure 7.7: Feature distributions for source and target data.

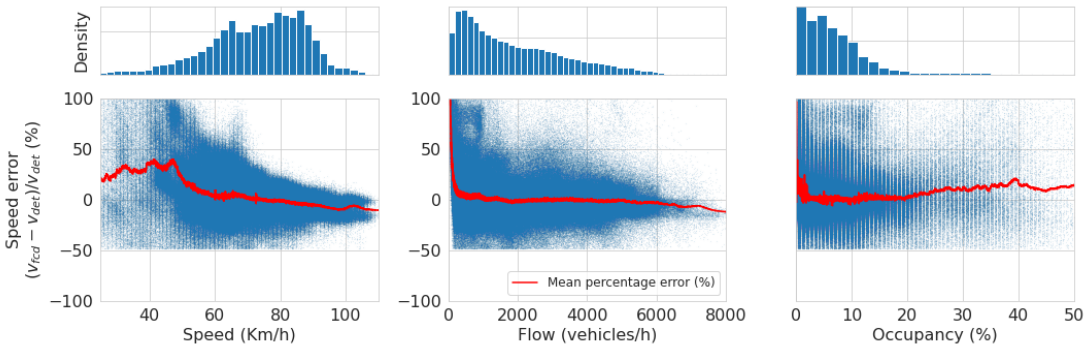


Figure 7.8: Speed error between stationary detector data and FCD data.

intervals of the XGBoost model's error, indicating that the LSTM model has a lower variance.

We also show the effect of the lookback length and prediction horizon on the performance of the LSTM model in Figure 7.9. In Figure 7.9 and Table 7.4, model performance degrades with the increase in the prediction horizon. This degradation is expected since it becomes challenging to predict accurately with increasing prediction horizons. The model shows good performance with a prediction horizon shorter than three hours. However, the error increases rapidly when the prediction horizon is over six hours, as shown in Table 7.4. On the other hand, an increase in lookback length does not show a major change in the model performance, but the lookback length of six steps shows the best performance in our experimental setting (Table 7.4).

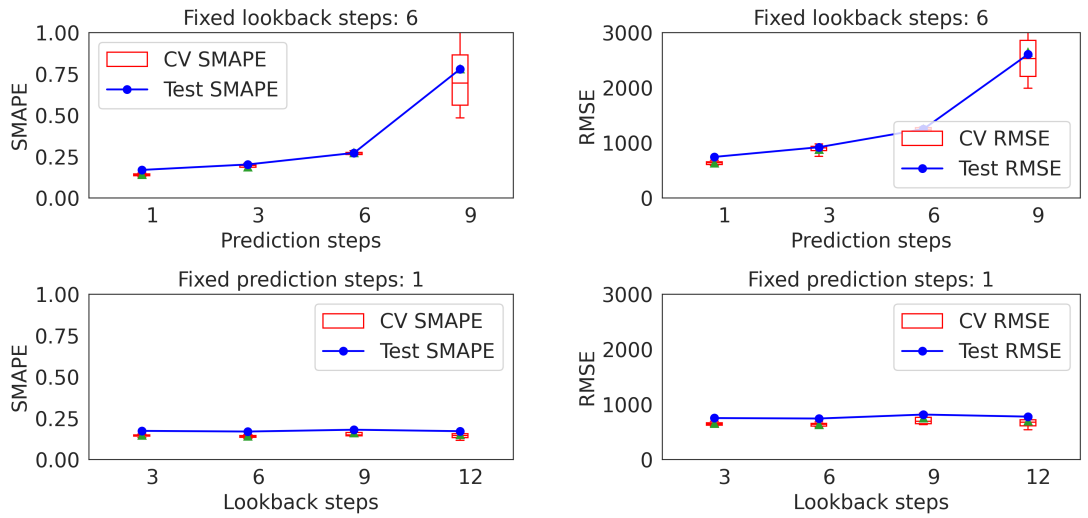


Figure 7.9: Cross-validation error and test error for different performance metrics, lookback length, and prediction step.

Table 7.2: Parameter ranges for tuning hyperparameters of machine learning models using Monte Carlo cross-validation (MCCV)

Model	Parameter	Search range	Best value
XGBoost	Learning rate	(1e-4, 0.9)	0.4
	Maximum depth of tree	{2,3,4,5,6}	5
	Column sub-sampling	[0, 1]	1
	Sub-sampling	[0, 1]	0.6
	Iterations	up to 4000	varies[a]
LSTM	Learning rate	(1e-6, 1e-1)	1.8e-4
	Batch size	{1024,2048,4096,8192}	2048
	Weight decay	(1e-7, 1e-4)	1e-5
	maximum size of embedding	5,10,15	10
	Dropout rate	(0, 1)	0.5
	Number of LSTM layers	{1,2,3,4}	3
	Size of LSTM hidden state	{40,50,100,200}	50
	Number of dense layers	{1,2,3}	2
	Size of penultimate hidden layer	{30,50,100,200}	50
	Size of last hidden layer	{5,10,20,40}	{5,10}[b]
	Epochs	up to 4000	varies[a]

[a]early stopping based [b]based on output size

Table 7.3: Model performance on different metrics.

Model	Link types	Loss criteria	Performance metric	Training data	Test data
XGBoost	all	SMAPE (%)	SMAPE (%)	45.15 ±2.02	51.76 ±5.28
	trunk		SMAPE (%)	14.04 ±1.39	21.65 ±2.93
			RMSE	725 ±88	862 ±157
LSTM	all	SMAPE (%)	SMAPE (%)	40.75 ±0.51	**40.17 ±0.90**
	trunk		SMAPE (%)	14.05 ±0.47	**16.89 ±0.31**
			RMSE	634 ±19	**743 ±14**

Note: RMSE is not reported for link types "all" since the scale of the target variable largely varies across the primary, secondary, and trunk link types.

Table 7.4: Effect of lookback length and prediction horizon on test data. The best test performance is shown in **bold**.

Model - Input features	Lookback length (hour)	Prediction horizon (hour)	SMAPE (%)	RMSE (vehicles/hour)
LSTM - (exogenous)	**6**	**1**	**16.89 ±0.31**	**743 ±14**
	6	3	20.20 ±1.06	919 ±42
	6	6	27.18 ±0.60	1247 ±20
	6	9	77.80 ±16.45	2602 ±317
	3	1	17.30 ±0.47	751 ±24
	6	**1**	**16.89 ±0.31**	**743 ±14**
	9	1	18.00 ±1.38	815 ±73
	12	1	17.11 ±1.08	777 ±56
LSTM - (exogenous, o)	6	1	9.95 ±2.01	466 ±92
LSTM - (exogenous, o, q)	6	1	6.47 ± 0.97	321 ±37

Note: o: occupancy, q: time-lagged flow.

In Table 7.4, we compare the model's performance (with lookback length and prediction horizon of six steps and one step, respectively) with the different feature compositions. We show the effect of step-wise addition of features from the loop detectors, namely, the occupancy (o) and flow (q), to the exogenous set of features (link attributes, speed attributes). We find that the model SMAPE reduces by approx. 41% when we add o to the input features. When we use the past value(s) of q as an input feature, the model formulation resembles autoregressive forecasting with exogenous inputs. In this setting, SMAPE reduces by more than 61% over the baseline indirect estimation model. This improvement is expected because past target variables provide direct information about the scale or range for future predictions due to the autocorrelation among the target variables. Thus, the autoregressive forecasting setting provides more accurate predictions than the purely exogenous or indirect estimation setting, indicating that the latter is more challenging.

In Figure 7.10, we show specimens of the model predictions for different detectors. There are a few noticeable things. First, the model can capture the flow periodicity, i.e., the ascent and descent of the flow trend during the day. This signifies good predictions when the flow transitions from off-peak to peak flow and vice-versa. The model performs well for detectors 5380 and 5299, as the predicted peak flow is closer to the true (actual or measured) value. The model performs reasonably well with a SMAPE of less than 17% on the test data, considering exclusive exogenous input features. In a few instances, the model struggles to capture peak and off-peak flow (e.g., crests for detector 5169 and troughs for detector 5273), where the model either over-predicts or under-predicts the flow compared to the actual value. The dropout during the model testing provides insights into the prediction uncertainty. For this, we show the 95% confidence interval of the model predictions. The uncertainty is higher near the peak flows, as seen from the wider prediction intervals at the peaks. During off-peak hours, the predictions have low

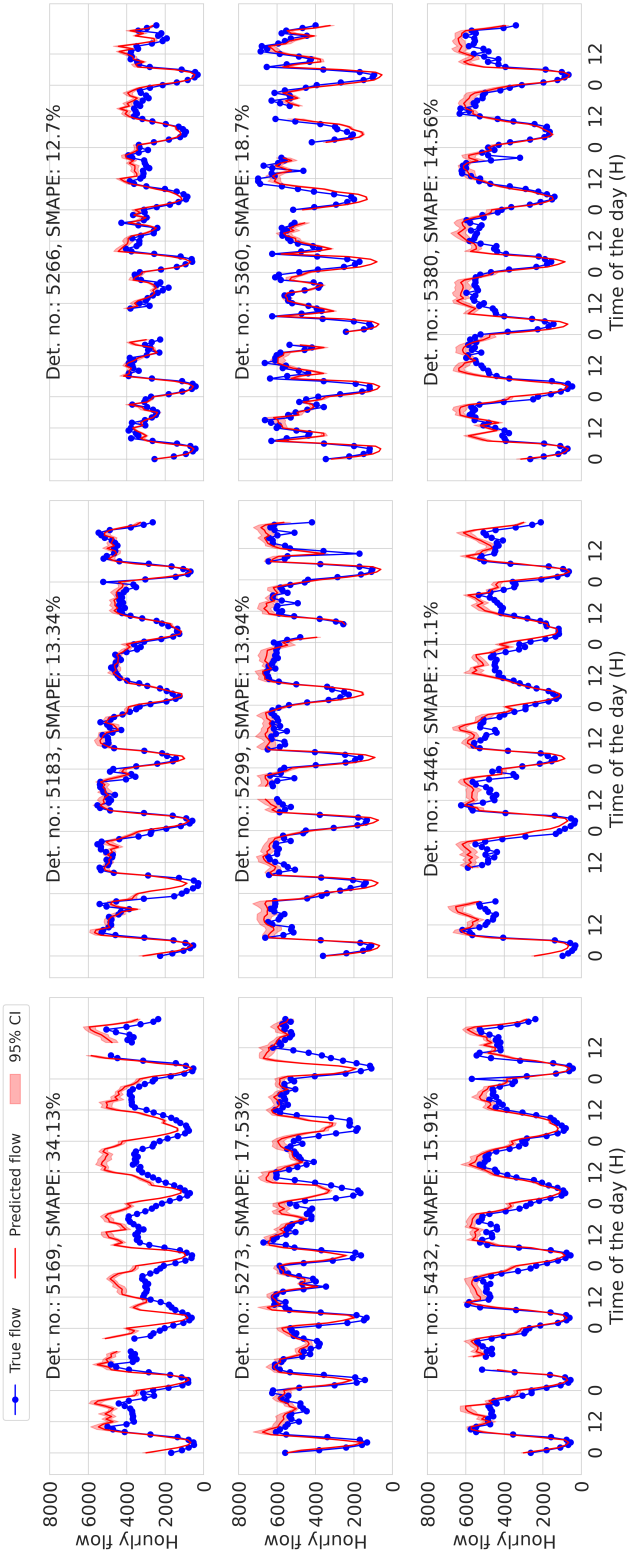


Figure 7.10: Examples showing flow predictions for detectors in test data from Paris.

variance. The uncertainty estimates are as important as the predictions since they help to understand how much the model predictions can be trusted.

We also show the distribution of the errors, namely, SMAPE and RMSE, across time of the day, weekday, and month to identify any error correlations. In Figure 7.11, it is seen that the mean SMAPE for night and morning hours (during 2100-0900 hours) is higher than during the rest of the day. On the other hand, the RMSE during the corresponding hours is lower than the rest of the day. One of the plausible explanations is the small magnitude of the flows during the off-peak hours, which pushes the SMAPE to higher values. Finally, the SMAPE and RMSE are almost constant across different weekdays. For August, errors are slightly higher than in the rest of the months, possibly due to distinct traffic patterns during the vacation period in Paris.

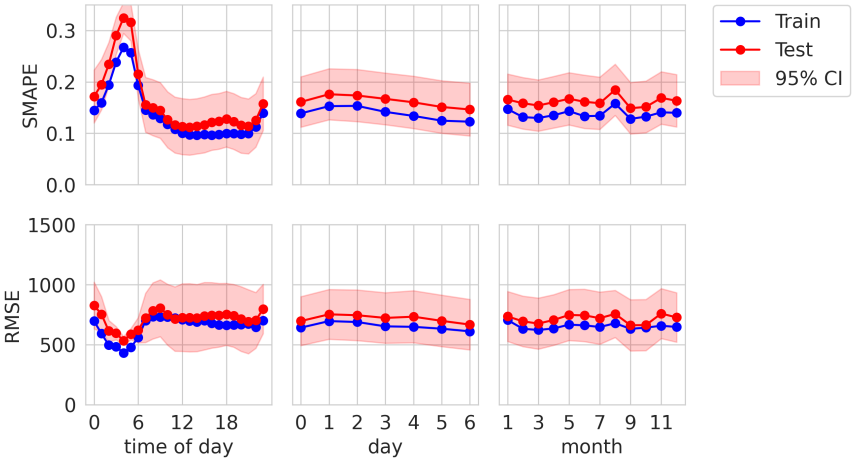


Figure 7.11: Trend of SMAPE and RMSE with the time of day, weekday, and month.

7.6.2 Model transferability

In Table 7.5, we show the model performances on test data using high (0.65) and low (0.10) proportions of training data for the target domain. When using 65% of the target data for training, the baseline model with randomly initialized parameters achieves a SMAPE of 22.24%. A pre-trained model without fine-tuning the target data does not lead to accurate predictions, as evidenced by its higher SMAPE. However, selective fine-tuning of the pre-trained model on the target data helps achieve even better results than the new model. Out of the different combinations of unfrozen layers in the pre-trained model, the model with all the unfrozen layers achieves the best SMAPE of 20.5%, which is a marginal improvement of 8% over the baseline model. A model with only LSTM layers as unfrozen layers also performs well with a SMAPE of 21.49%. Thus, fine-tuning the LSTM layer is essential when transferring knowledge from the source to the target domain. This is due to the difference in temporal patterns between the target and the source city. Thus, the model relearns the new patterns from the target dataset.

Table 7.5: Performance comparison between training new model and fine-tuning pre-trained model with the change in the proportion of training data.

LSTM Model-type	Weight initialization	Proportion of target data for training	Unfrozen/fine-tuned layers	Test SMAPE (%)	improvement over baseline (%)
Baseline	random	0.65	all	22.24 ±1.96	-
Transfer	pre-trained		None	65.82 ±2.74	-195
			FC3	46.54 ±0.39	-109
			FC2-3	46.55 ±0.34	-109
			FC1-3	45.87 ±0.29	-106
			LSTM, FC	21.24 ±0.99	4
			LSTM	21.49 ±0.90	3
			E	22.75 ±0.73	-2
			E, LSTM, FC	**20.50 ±1.10**	**8**
Baseline	random	0.10	all	55.30 ±16.06	-
Transfer	pre-trained		None	65.82 ±2.74	-19
			FC3	47.15 ±1.30	15
			FC2-3	47.29 ±1.00	15
			FC1-3	47.60 ±1.00	14
			LSTM, FC	29.07 ±1.70	47
			LSTM	**27.41 ±1.68**	**50**
			E	29.14 ±1.61	47
			E, LSTM, FC	30.92 ±1.89	44

Note: FC: fully connected layer, E: Embedding layer.

The benefits of transfer learning are prominent in low data availability scenarios. When using 10% of the target data for training, the new model has a high bias and variance, as seen from its higher SMAPE and 95% confidence interval. The pre-trained model (with a fine-tuned LSTM layer) can achieve a SMAPE of 27.4% even in the case of data insufficiency.

In Figure 7.12, we show the performance differences between the baseline or new model and fine-tuned pre-trained model with the different proportions (from 0.04 to 0.96 at a spacing of 0.04) of the target data used for fine-tuning. We fine-tune only the LSTM layer since it is crucial for successful transfer learning. We find that with sufficient training data, i.e. when more than 40% of the target domain data is used for re-training, both models perform equally well with the test SMAPE of around 20%. When the proportion of training data falls below 40%, we notice two trends. First, the variance of the performance of the baseline model increases considerably. This is due to the high variance in sampling training datasets at low proportions since smaller training datasets do not capture complete distribution over the target domain. In contrast, the variance of the fine-tuned model is low and about consistent, which points to the advantage of transfer learning over the baseline model.

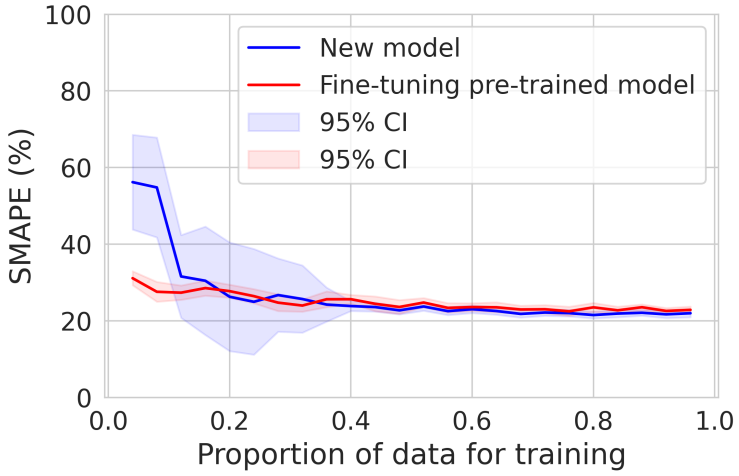


Figure 7.12: Comparison between training new model and fine-tuning pre-trained model with the change in the proportion of training data

Second, when the proportion of training data is very low such as less than 20% (Figure 7.12), the baseline model's bias also increases, and it performs poorly compared to the pre-trained model. This is due to the model's over-fitting of the small training data distribution, which is very different from the test data distribution. In contrast, the performance of the fine-tuned pre-trained model is stable. Thus, we conclude that transfer learning performs equally well when the data is sufficient. More importantly, transfer learning outperforms the baseline model when the data for the target domain is insufficient since source knowledge helps overcome the lack of data in the target domain.

Figure 7.13 shows example predictions on test target data from the new and pre-trained models at different training data proportions. When using less than 10% of the training data, the pre-trained model's predictions are more accurate and consistent than the new model's predictions. In these examples, when using insufficient training data, high bias and high variance in the predictions by the new model are evident. An increase in training data helps the model to make accurate predictions. However, the pre-trained model can make accurate predictions using even a small amount of data.

7.7 Summary

This chapter was motivated by the sparsity of traffic flow data and the need to infer it from other readily available data. We collected the publicly available traffic data for Paris and Madrid from heterogeneous sources to form a year-long longitudinal traffic dataset. Using data from Paris, we trained an LSTM model, which performs well when the fundamental diagram is well-formed, like trunk-type links. The LSTM model outperforms the XGBoost model in exogenous flow prediction with a test SMAPE of about 17% for the source trunk links. The trunk type links belong to high-speed category links, and

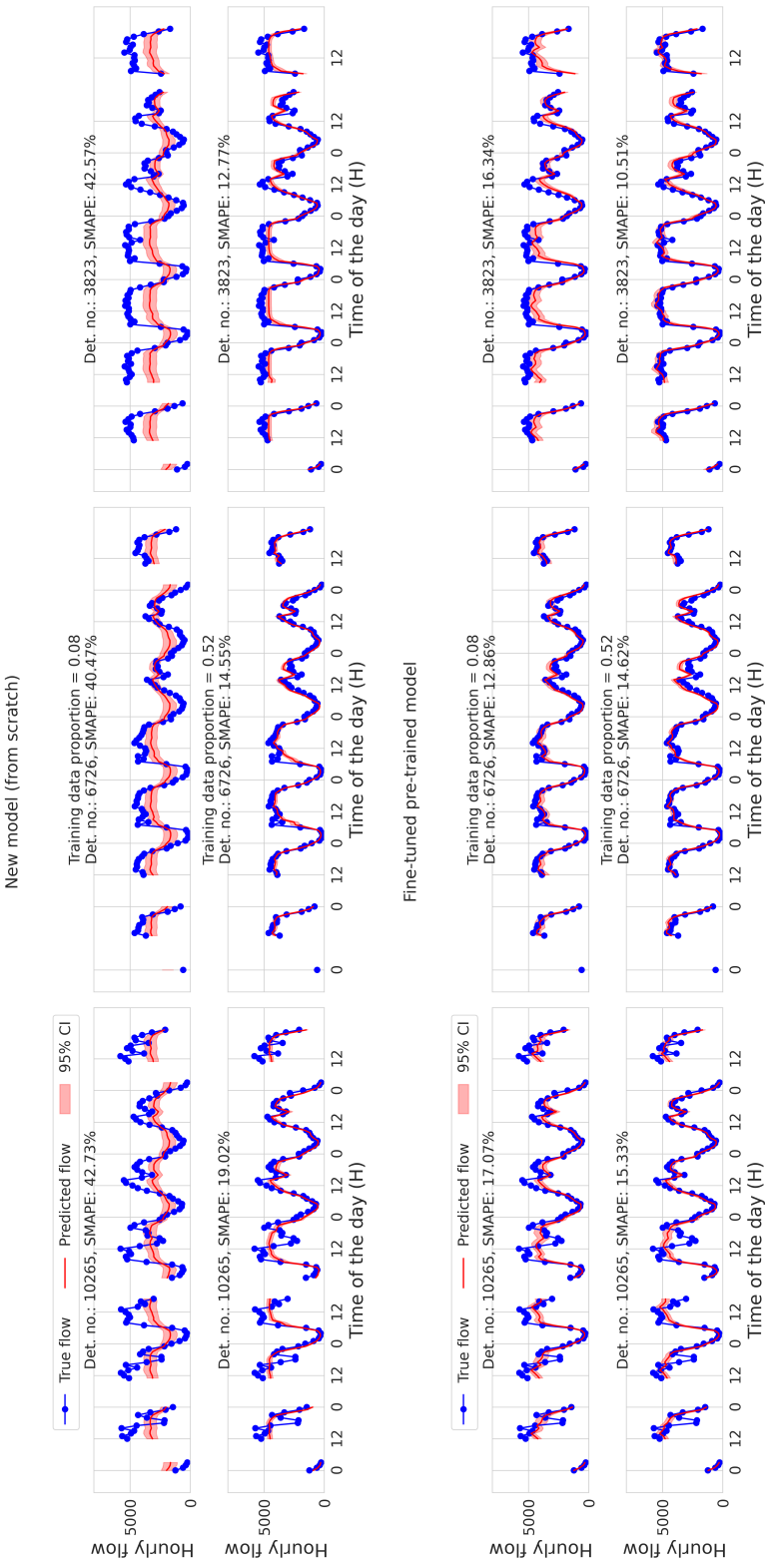


Figure 7.13: Flow predictions on test data in target domain using new model and fine-tuned pre-trained model. Both models are trained using different training proportions of training data.

thus our results somewhat address the limitation of previous work by Neumann et al. (2013), where the model predictions for high-speed links were found to be less accurate. It is fundamental not only to predict these values but, even more notably, to estimate how precise these predictions are. The proposed deep learning architecture also answers this question using a dropout mechanism. We also show that pre-trained models outperform the new model using transfer learning when the data for the target task is insufficient. The pre-trained model needs less data to predict link flows for the target city accurately. Thus transfer learning and indirect flow estimation can help to tackle the traffic flow data scarcity in transport modeling and traffic management applications.

Part IV

Conclusion

8 Research findings and future work

Contents

8.1 Summary of main research findings and implications

In this dissertation started with three primary questions aimed at addressing data insufficiency in transport research. To answer those research questions, this dissertation provides a unified account of the five studies in three parts. Part I provided an introduction and conceptual framework for identifying and classifying the data based on their openness. In Part II, we focused on improving the data quality from emerging sources and applying them for opportunistic applications. Part III of this dissertation focused on developing data-efficient methods, such as efficient calibration, indirect flow estimation, and transfer learning, on bridging the gap between conventional and non-conventional data. Overall, we analyzed traffic and mobility data from diverse traditional, non-conventional, and emerging sources such as stationary traffic detectors, floating car data, trajectory data from drone videography, and mobile crowdsensing. These data were applied to solve extant challenges due to the lack of data in the context of transport data mining and modeling, such as analysis of POI demand patterns during interventions, treatment of noise and anomalies in emerging data, prediction of network-scale flows using transfer learning, and automated calibration of large-scale traffic simulations.

We revisit the PRQs formulated in Chapter 1 and provide a discussion on how the goals of this dissertation to address **data insufficiency in transport analysis and modeling** were achieved by answering these questions:

8.1.1 Systematic understanding of transport data openness

Before using data for modeling, it is important to understand which kinds of data are relatively easier to access. Motivated by PRQ(1), we develop the conceptual framework (Chapter 3) for classifying transport data by examining whether data are attainable, affordable, accessible, usable, and redistributable. Simultaneously, we also showed how non-conventional open and not-so-open data are used in transport modeling applications. Most data types are applied for either supply-side (e.g., GTFS) or demand-side modeling (e.g., social media data). However, no single data excel in all the applications, and thus, data complementarity is vital in transport research. Therefore, modelers and authorities must plan and invest in developing or acquiring complementary data sources.

We find that Mobile phone data, social media data, and even smart card data collected by public and private organizations come with challenges, including proprietary ownership and privacy risks. These data can become publicly available with restricted or free-use licensing if concerns regarding commercial competition, privacy protection, and revenue loss are allayed. For instance, this could be overcome if private companies, communities, and the government defined and followed a common objective and shared data cooperatively based on reciprocity. This was partially demonstrated over the last fifteen years as the OGD and open standards matured. The transition towards OGD can help bridge the data availability gap by pushing the PSI or government data from being inaccessible to publicly available. A lesson can be learned from the few road or public transport authorities who have publicly made the aggregate traffic or passenger data available in their cities. Therefore, this chapter provides a conceptual direction that can

help cities and modelers understand and prioritize data openness and thus help address data insufficiency, and thus provides the answer to PRQ(1).

8.1.2 Creating value from emerging data

PRQ(2) and PRQ(3) are focused on getting more out of non-conventional data by different means. Emerging traffic data collection methods have their benefits and challenges. Before using these data, data quality needs to be validated so that the efficacy of data is established. Data processing for error removal or minimizing implausible values is essential to improve data quality. In response to PRQ(2), Chapter 4 shows the development and application of the data processing framework on the pNEUMA dataset from the aerial footage. We used SG filter, XGBoost with adaptive regularization, and GF to remove the noise and anomalies. We show that our approach can accurately detect anomalies in the form of unrealistic transient peaks in the data. Adaptive regularization adjusts itself based on the maximum acceleration value and simplifies anomaly detection to fewer tunable parameters. Using an off-the-shelf model such as XGBoost reduces the number and effort of tuning the model. When processing vehicle trajectories, a balance should be maintained between filtering the data and retaining naturalistic driving behavior. Our approach is adaptable to other trajectory or sequence datasets corrupted by noise and anomalies. However, for successful transfer, anomalies in new data should be similar to those in the pNEUMA data, i.e., a few unrealistic transient acceleration peaks. The treated data is much more suitable for microscopic traffic analysis, such as road safety analysis using surrogate measures or driving behavior modeling (car-following or lane-changing), and can thus help accelerate future research.

Further, novel avenues for applying these data must be identified where existing data or practices are not suitable or insufficient. Chapter 5 answers PRQ(3) by showing that POI check-ins are a potential source of information during dynamic events like COVID-19. While doing so, we also show how machine learning methods can help model and better understand the data compared to traditional methods. We find that POI-level data and features help to understand the underlying interactions of spatial and non-spatial features in detail and identify the spatial variability (if any) and the influencing factors thereof. Demand patterns aligned with the restrictions implemented during the lockdown. The impact of features on POI demand was explained using the GBR regression model and SHAP. The significance of certain factors, such as POI type (fast-food), confirmed the influence of lockdown measures on non-essential retail consumption. The study also highlighted the vulnerability of businesses near transit hubs during the lockdown, suggesting the need for further investigation and mitigation measures.

These findings are also interesting to the transport planners and operators as they provide insights into the effect of transport variables such as parking area and transit-stop distance on the POI popularity. We provide empirical evidence of the disproportionate impact of the lockdown restrictions on the POIs in Munich, depending on their distance from a transit stop. Businesses near or in the transit hubs are more vulnerable to these disruptions due to reduced commuters and potential customers, possibly due to reduced travel (home office) or changed travel behavior (customers avoiding public

transport). These insights point to the lack of resilience of transit-near POIs due to excessive dependence on commuting customers. Policymakers can look into or even adapt the transit-oriented development principles to diversify the customers of near-transit POIs.

8.1.3 Data efficient methods

PRQ(4) stresses using advanced models to learn more from conventional data. Towards this end, In Chapter 6, we presented an end-to-end sequential demand and supply calibration approach. Our approach has components automating certain aspects, such as SPSA and tuning of supply parameters to save manual effort. We also tackled bias and variance in the initial estimates by proposing methods for each. We proposed a bias correction heuristic to correct the initial bias and thus reduce the burden on the following optimization algorithm, i.e., W-SPSA. However, W-SPSA will stop improving the errors after most of the overall bias has been corrected, i.e., beyond the limit where noise starts to dominate. This happens due to the cancellation of the random gradients dominated by the noise. We applied ensembling with bagging and SPA to address the variance in the calibration estimates due to stochasticity in the optimization and simulation. Bagging helps to cancel the variance among the individual estimates, scattered near the desired estimate and thus brings us closer to the desired estimate. Due to the algorithm-agnostic nature of the proposed enhancements, such as automated tuning and ensembling, our approach is not just restricted to the SPSA class of algorithms.

Further, In Chapter 7, we presented the indirect traffic state estimation model (for predicting flow) using open data and transfer learning. The proposed model uses only exogenous features as input for the prediction. The objective of the proposed framework is to predict traffic counts on links that are not equipped with a traffic sensor. The transferred model can help when the data is insufficient to develop models from scratch. The LSTM model only fits well for trunk-type links, whereas models did not perform well for primary and secondary-type links. We conclude that the manifestation of a traffic fundamental diagram and reliable FCD data over a wide range of speed and flow is essential for indirect dynamic flow estimation from speed. If a fundamental diagram is not well-formed, models struggle to learn link speed and flow mapping.

Our experiments conclude that the indirect traffic flow estimation task has two components: a) transferable and b) non-transferable patterns. Training a new model on minimal target data will lead to high bias and variance in prediction. Thus, transfer learning can help to bridge this gap. Distributions of source and target data are prone to be distinct across different cities. Thus, applying a pre-trained model without fine-tuning target data does not give accurate results. The pre-trained model contains insights from the source domain, thus eliminating the need for a model to re-learn the transferable patterns. Some data is still required to learn the nontransferable patterns. Nevertheless, the overall data requirements are lesser than the scenario without transfer learning, and the pre-trained model outperforms the newly trained model when data is scarce. The practical implications of our research are that the exogenous flow prediction can help fill the flow data unavailability for traffic management or transport model validation.

Further, open data and transfer learning can help address this challenge by reducing data acquisition costs. Flow predictions with uncertainty estimates can help lessen the practitioners' hesitancy to apply these models. The proposed approach can help overcome the challenges researchers and practitioners face in traffic management and transportation modeling due to data scarcity.

8.2 Limitations and recommendations for future work

8.2.1 Limitations

It is important to acknowledge the limitations of this study. By acknowledging the following limitations, we can pave the way for future research to overcome them and enhance our understanding of the subject matter.

The first set of limitations relates to the **scope and focus** of the studies done in this dissertation:

1. The conceptual framework in Chapter 3 does not cover all data applications in detail but is intended to present a broad overview of the most prominent transport data. The public availability landscape of specific datasets could vary depending on the location, policy ecosystem, and technology penetration. Therefore, a location-specific analysis or case studies for selected cities could be avenues for future research.

2. In Chapter 4, we only address the anomalies of the unrealistic-peak character but not the anomalies of other characters, which may also be present in the dataset.

3. Demonstration of the calibration approach in Chapter 6 is limited to mesoscopic simulation and few supply parameters.

The second set of limitations comes from the **data** used in this work.

1. In Chapter 4, the absence of ground truth labels prevents us from verifying the driving behavior or causes behind the detected anomalies in the acceleration.

2. In Chapter 5, relative popularity or demand from the popular times' data fails to capture the population's effect on the POIs. Adding more features, like land-use type (residential vs. workplace), could improve the results, mainly because during COVID-19, generally, work from home was recommended. We also found that live popularity is unavailable for most POIs during the lockdown, limiting the data for modeling and adding to sampling bias. Sensitivity analysis on the effect of sampling variation and feature threshold could be an interesting topic for the future. We do not account for the marketing strategies which influence consumers.

3. The distribution of crowdsensed data are expected to differ among cities due to different city-specific factors. The role of the spatial factors might vary depending on city-specific factors like the impact of events or interventions on mobility. Thus,

the behavior of factors influencing the demand patterns uncovered in our study might not be directly transferable to other cities.

4. POI busyness can be influenced by marketing decisions which could itself be motivated by complex factors such as weather, time, day, and month. Thus to some extent, the overall effects can be captured by collecting time-series data and controlling for an hour, month, and day. However, at an individual POI level, marketing-specific data could be hard to collect as the marketing strategies could be diverse and highly dynamic even across similar POI types.

5. In Chapter 7, data issues such as data inaccuracy due to noise and anomalies, non-perfect data matching, information loss due to aggregation, and heterogeneity of data sources can lead to distortion of the same information from different sources. For instance, unreliable FCD speed data in specific ranges of traffic variables will degrade the model performance. This directly affects the quality of the training data and decreases the signal-to-noise ratio, thus making it challenging for the model to learn the underlying correlations. Further, the static features, such as maximum speed and the number of lanes, ignore the effect of dynamic traffic management (such as dynamic speed limits or lane closures) in the recorded flow values.

The third set of limitations is related to the **methodological approach** used for error treatment, indirect estimation, and calibration.

1. In Chapter 4, machine learning model training can be ineffective in extremely short trajectories due to a lack of data. Although XGBoost is less data-hungry when compared to deep neural networks, XGBoost's performance to detect anomalies can still be affected when data is scarce.

2. The performance of our calibration framework in Chapter 6 is limited by the quality of the initial estimate. If the initial estimate has very high randomness, then the specified constraints on the demand will not be precise and could be far from the plausible estimates. Our proposed OD estimation framework can be further augmented with any auxiliary OD demand data sources in the objective function to counter this.

3. Methodological components such as BC in Chapter 6 are specific to traffic count data, and thus they cannot be applied when such data are unavailable for calibration. Future work can be done to apply Probe Vehicle data (Antoniou et al., 2004) or Speed data for initial bias adjustment.

4. In Chapter 7, we tackled temporal correlations using LSTM-based architecture, which does not explicitly capture spatial or topological correlations.

8.2.2 Future work

We provide recommendations that can guide future works in mitigating these limitations and further advancing the field. The directions for future research, including those which could help to address the above limitations, are discussed below:

1. Future works could be done to adjust the trajectory positions from drone videography as per the treated speed to ensure internal consistency (Montanino & Punzo, 2015; Punzo et al., 2011) (between position and speed) and platoon consistency (with leader vehicle and follower vehicle).

2. There exists a range of parameters for anomaly detection and smoothing, which provide acceptable results. This indicates that any subsequent analysis (traffic emissions, crash safety analysis) using trajectory data will also be sensitive to these parameters. Thus, the researchers should estimate confidence intervals to quantify their results' uncertainty.

3. Another crucial future work is decomposing the processed speed and acceleration vectors into longitudinal and lateral components relative to the street's orientation for analyzing lateral driving maneuvers.

4. The use of publicly available data sources for demand pattern analysis increases the transferability of the methods in Chapter 5 to other study areas, which could be the subject of future works. Popularity data used in this work can also be tested and applied to analyze other interventions that could lead to changes in travel or mobility patterns, such as short-term free-fare transit policies.

5. A time-series crowd-sensed data over a longer duration is suitable for causal inference to conduct the policy impact evaluation of lockdowns or other interventions.

6. Further research should be done on POI busyness data to infer latent features such as consumer preferences and socializing behavior during disruptive events.

7. During such interventions or disruptive events, popularity patterns should be correlated with other exogenous factors such as public transport, land use, and demographic attributes to check their influence on crowding behavior. Researchers could use crowd-sensed information to analyze if the POI visitation trend has changed and returned to normal levels, thus indicating the system's resilience.

8. Ensembling techniques such as bagging and SPA have proven effective in machine learning and thus should be explored for other simulation-based optimization problems. For instance, ensemble methods should also be explored for application to calibrate parameters, even in car-following or lane-changing models.

9. Future works can also experiment with the ensembling aspects, such as different types of gain coefficient restart techniques and intermediate estimates during each cycle.

10. Stochastic simulations involving high-dimensional inputs do not guarantee a unique calibration parameter set. The possibility of a multiple-parameter set arises from the unobservable/ indeterminate system, wherein many solutions for given conditions are possible. However, some of these parameters can be practically reasonable in real-world scenarios due to the stochasticity of the real system. Thus, having a

single set of parameters is insufficient. Here, multiple estimates during ensembling cycles can also be used to quantify the uncertainty in parameters.

11. The location of sensors can influence the quality of the estimated ODs. This is related to the coverage or network observability the sensors provide. In our calibration experiments using synthetic or analytical simulators, we used multiple random variations of detector configurations for each run and thus help to tackle the variance due to such sensor location settings. For the experiments with SUMO simulators, doing this is computationally expensive, and we consider investigation of this aspect a matter of future work.

12. The calibration framework can be tested and applied for online calibration at shorter periods (5 minutes or 15 minutes) where the fluctuations in the demand and traffic flow are prominent and challenging to handle.

13. Using additional data sources for MOPs will help reap other benefits, especially in the case of real scenarios where true or global parameters are unknown. In these cases, W-SPSA may need to be adapted according to the data source to reap benefits. For instance, the weight matrix based on the link assignment matrix may not be the best choice for non-linear variables such as speed and thus need further enhancements.

14. We used a mesoscopic simulation model to ease the computation burden, and thus future studies should consider applying our methods to microscopic simulations. Using microscopic simulations will also expand the scope of calibrating supply parameters.

15. There is still scope for further research using state-of-the-art deep learning models for indirect estimation. For instance, GNNs (Jiang & Luo, 2021; Lin et al., 2018) or Temporal Fusion Transformer (Beitner, 2020; Lim et al., 2021) are more specialized to learn network or topological data and temporal sequences, respectively. Therefore, they can further help to reduce the forecasting error.

16. For future research in indirect flow estimation, we see potential in integrating the model and predicted forecasts for improving transport demand model calibration. Specifically, the link flows provided by the flow prediction model can increase the transport network observability; thus, their impacts on the calibrated demand estimates could be analyzed.

17. Future works should explore other data sources, such as real-time traffic updates, and extract the relevant features to augment the training data. Using the enriched data, it can be possible to apply transfer learning for long-term flow estimation like daily traffic flow estimation (D. Ma et al., 2021) or Annual Average Daily Traffic (AADT) flows.

18. Special events, planned or unplanned, lead to changes in the traffic patterns (Polson & Sokolov, 2017) and thus can be added as an additional feature to improve the model performance.

19. Another challenging work is investigating additional features, especially for the primary and secondary type links, to address their scatter in their fundamental diagram. Other features that help address the scatter can help obtain accurate predictions, especially for lower-category link types (primary and secondary). Additional features could help explain variance in the fundamental diagram and thus provide an improved signal to train the model.

Bibliography

Abdelraouf, A., Abdel-Aty, M., & Mahmoud, N. (2022). Sequence-to-sequence recurrent graph convolutional networks for traffic estimation and prediction using connected probe vehicle data. *IEEE Transactions on Intelligent Transportation Systems*, 1-11. doi: 10.1109/TITS.2022.3168865

Acheampong, R. A., & Silva, E. (2015, Jul.). Land use–transport interaction modeling: A review of the literature and future research directions. *Journal of Transport and Land Use*, *8*(3). doi: 10.5198/jtlu.2015.806

Achuthan, K., Titheridge, H., & Mackett, R. L. (2010). Mapping accessibility differences for the whole journey and for socially excluded groups of people. *Journal of Maps*, *6*(1), 220-229. doi: 10.4113/jom.2010.1077

Adadi, A. (2021, Jan 26). A survey on data-efficient algorithms in big data era. *Journal of Big Data*, *8*(1), 24. Retrieved from `10.1186/s40537-021-00419-9` doi: 10.1186/s40537-021-00419-9

Alzubaidi, L., Bai, J., Al-Sabaawi, A., Santamaría, J., Albahri, A. S., Al-dabbagh, B. S. N., ... Gu, Y. (2023, Apr 14). A survey on deep learning tools dealing with data scarcity: definitions, challenges, solutions, tips, and applications. *Journal of Big Data*, *10*(1), 46. doi: 10.1186/s40537-023-00727-2

Anda, C., Erath, A., & Fourie, P. J. (2017). Transport modelling in the age of big data. *International Journal of Urban Sciences*, *21*(sup1), 19-42. doi: 10.1080/12265934.2017.1281150

Antoniou, C., Balakrishna, R., & Koutsopoulos, H. N. (2011, Nov 01). A synthesis of emerging data collection technologies and their impact on traffic management applications. *European Transport Research Review*, *3*(3), 139-148. doi: 10.1007/s12544-011-0058-1

Antoniou, C., Barceló, J., Breen, M., Bullejos, M., Casas, J., Cipriani, E., ... others (2016). Towards a generic benchmarking platform for origin–destination flows estimation/updating algorithms: Design, demonstration and validation. *Transportation Research Part C: Emerging Technologies*, *66*, 79–98. doi: 10.1016/j.trc.2015.08.009

Antoniou, C., Ben-Akiva, M., & Koutsopoulos, H. N. (2004). Incorporating automated vehicle identification data into origin-destination estimation. *Transportation Research Record*, *1882*(1), 37-44. Retrieved from `https://doi.org/10.3141/1882-05` doi: 10.3141/1882-05

Antoniou, C., Ben-Akiva, M., & Koutsopoulos, H. N. (2005). Online calibration of traffic prediction models. *Transportation Research Record*, *1934*(1), 235-245. doi: 10.1177/0361198105193400125

Antoniou, C., Lima Azevedo, C., Lu, L., Pereira, F., & Ben-Akiva, M. (2015). W-SPSA in practice: Approximation of weight matrices and calibration of traffic simulation

models. *Transportation Research Part C: Emerging Technologies*, *59*, 129-146. (Special Issue on International Symposium on Transportation and Traffic Theory) doi: 10.1016/j.trc.2015.04.030

Apple. (2020). *Mobility trends reports.* Retrieved from `https://www.apple.com/covid19/mobility` (Accessed on 25.07.2020)

Arafat, S. M. Y., Kar, S. K., & Kabir, R. (2020, May 21). Possible controlling measures of panic buying during covid-19. *International Journal of Mental Health and Addiction*, 1-3. doi: 10.1007/s11469-020-00320-1

Araghi, B. N., Hammershøj Olesen, J., Krishnan, R., Tørholm Christensen, L., & Lahrmann, H. (2015). Reliability of Bluetooth Technology for Travel Time Estimation. *Journal of Intelligent Transportation Systems: Technology, Planning, and Operations*, *19*(3), 240–255. doi: 10.1080/15472450.2013.856727

Ariffin, A., Hamzah, A., Solah, M., Paiman, N., Jawi, Z. M., & Isa, M. M. (2017). Comparative analysis of motorcycle braking performance in emergency situation. *Journal of the Society of Automotive Engineers Malaysia*, *1*(2). Retrieved from `http://jsaem.saemalaysia.org.my/index.php/jsaem/article/view/53`

Arjunan, P., Poolla, K., & Miller, C. (2020). Energystar++: Towards more accurate and explanatory building energy benchmarking. *Applied Energy*, *276*, 115413. doi: 10.1016/j.apenergy.2020.115413

Arzberger, P., Schroeder, P., Beaulieu, A., Bowker, G., Casey, K., Laaksonen, L., ... Wouters, P. (2004). An International Framework to Promote Access to Data. *Science: Policy Forum*, *303*(March), 1777–1779. doi: 10.1126/science.1095958

Aslam, J., Lim, S., Pan, X., & Rus, D. (2012). City-scale traffic estimation from a roving sensor network. In *Proceedings of the 10th acm conference on embedded network sensor systems* (p. 141–154). New York, NY, USA: Association for Computing Machinery. doi: 10.1145/2426656.2426671

Aultman-Hall, L., Roorda, M., & Baetz, B. W. (1997). Using gis for evaluation of neighborhood pedestrian accessibility. *Journal of Urban Planning and Development*, *123*(1), 10-17. doi: 10.1061/(ASCE)0733-9488(1997)123:1(10)

Backx, M. (2003). *Gebouwen redden levens. Toegankelijkheidseisen van gebouwgegevens in het kader van de openbare orde en veiligheid.*

Badr, H. S., Du, H., Marshall, M., Dong, E., Squire, M. M., & Gardner, L. M. (2020, Nov 01). Association between mobility patterns and covid-19 transmission in the usa: a mathematical modelling study. *The Lancet Infectious Diseases*, *20*(11), 1247-1254. doi: 10.1016/S1473-3099(20)30553-3

Bagchi, M., & White, P. R. (2005). The potential of public transport smart card data. *Transport Policy*, *12*(5), 464–474. doi: 10.1016/j.tranpol.2005.06.008

Bagrow, J. P., Wang, D., & Barabási, A.-L. (2011, 03). Collective response of human populations to large-scale emergencies. *PLOS ONE*, *6*(3), 1-8. doi: 10.1371/journal.pone.0017680

Bakillah, M., Liang, S., Mobasheri, A., Jokar Arsanjani, J., & Zipf, A. (2014). Fine-resolution population mapping using OpenStreetMap points-of-interest. *International Journal of Geographical Information Science*, *28*(9), 1940–1963. doi: 10.1080/13658816.2014.909045

Balakrishna, R., Ben-Akiva, M., & Koutsopoulos, H. N. (2007). Offline calibration of dynamic traffic assignment: Simultaneous demand-and-supply estimation. *Transportation Research Record*, *2003*(1), 50-58. doi: 10.3141/2003-07

Barceló, J., Montero, L., Marqués, L., & Carmona, C. (2010). Travel time forecasting and dynamic origin-destination estimation for freeways based on bluetooth traffic monitoring. *Transportation Research Record*(2175), 19–27. doi: 10.3141/2175-03

Barmpounakis, E., & Geroliminis, N. (2020). On the new era of urban traffic monitoring with massive drone data: The pNEUMA large-scale field experiment. *Transportation Research Part C: Emerging Technologies*, *111*(October 2019), 50–71. doi: 10.1016/j.trc.2019.11.023

Barmpounakis, E., Sauvin, G. M., & Geroliminis, N. (2020). Lane detection and lane-changing identification with high-resolution data from a swarm of drones. *Transportation Research Record*, *2674*(7), 1-15. doi: 10.1177/0361198120920627

Barmpounakis, E. N., Vlahogianni, E. I., & Golias, J. C. (2016). Intelligent transportation systems and powered two wheelers traffic. *IEEE Transactions on Intelligent Transportation Systems*, *17*(4), 908-916. doi: 10.1109/TITS.2015.2497406

Barmpounakis, E. N., Vlahogianni, E. I., & Golias, J. C. (2016). Unmanned aerial aircraft systems for transportation engineering: Current practice and future challenges. *International Journal of Transportation Science and Technology*, *5*(3), 111 - 122. (Unmanned Aerial Vehicles and Remote Sensing) doi: 10.1016/j.ijtst.2017.02.001

Barrington-Leigh, C., & Millard-Ball, A. (2017, 08). The world's user-generated road map is more than 80% complete. *PLOS ONE*, *12*(8), 1-20. doi: 10.1371/journal.pone.0180698

Barry, J. J., Newhouser, R., Rahbee, A., & Sayeda, S. (2002). Origin and destination estimation in New York City with automated fare system data. *Transportation Research Record*(1817), 183–187. doi: 10.3141/1817-24

Bast, H., Storandt, S., & Weidner, S. (2015). Fine-grained population estimation. In *Gis: Proceedings of the acm international symposium on advances in geographic information systems* (Vol. 03-06-Nove). doi: 10.1145/2820783.2820828

BAST: Bundesanstalt für Straßenwesen. (2023). *Automatic permanent counting points: raw data.* `https://www.bast.de/DE/Publikationen/Daten/Verkehrstechnik/DZ.html?nn=1954870`. (Accessed: 2023-02-01)

Bathaee, N., Mohseni, A., Park, S. J., Porter, J. D., & Kim, D. S. (2018). A cluster analysis approach for differentiating transportation modes using Bluetooth sensor data. *Journal of Intelligent Transportation Systems: Technology, Planning, and Operations*, *22*(4), 353–364. doi: 10.1080/15472450.2018.1457444

Beitner, J. (2020). *PyTorch Forecasting: Time series forecasting with PyTorch.* Retrieved from `https://pytorch-forecasting.readthedocs.io/`

Ben-Akiva, M. E., Gao, S., Wei, Z., & Wen, Y. (2012). A dynamic traffic assignment model for highly congested urban networks. *Transportation Research Part C: Emerging Technologies*, *24*, 62-82. doi: 10.1016/j.trc.2012.02.006

Bhaskar, A., & Chung, E. (2013). Fundamental understanding on the use of Bluetooth scanner as a complementary transport data. *Transportation Research Part C: Emerging Technologies*, *37*, 42–72. doi: 10.1016/j.trc.2013.09.013

Bhaskar, A., Qu, M., & Chung, E. (2015). Bluetooth vehicle trajectory by fusing bluetooth and loops: Motorway travel time statistics. *IEEE Transactions on Intelligent Transportation Systems*, *16*(1), 113–122. doi: 10.1109/TITS.2014.2328373

Bienzeisler, L., Lelke, T., Wage, O., Thiel, F., & Friedrich, B. (2020). Development of an Agent-Based Transport Model for the City of Hanover Using Empirical Mobility Data and Data Fusion. In *Transportation research procedia* (Vol. 47, pp. 99–106). doi: 10.1016/j.trpro.2020.03.073

Boeing, G. (2017). Osmnx: New methods for acquiring, constructing, analyzing, and visualizing complex street networks. *Computers, Environment and Urban Systems*, *65*, 126-139. doi: 10.1016/j.compenvurbsys.2017.05.004

Böhmer, M. M., Buchholz, U., Corman, V. M., Hoch, M., Katz, K., Marosevic, D. V., ... Zapf, A. (2020, 6). Investigation of a covid-19 outbreak in germany resulting from a single travel-associated primary case: a case series. *The Lancet Infectious Diseases*. doi: 10.1016/S1473-3099(20)30314-5

Bokare, P., & Maurya, A. (2017). Acceleration-deceleration behaviour of various vehicle types. *Transportation Research Procedia*, *25*, 4733-4749. (World Conference on Transport Research - WCTR 2016 Shanghai. 10-15 July 2016) doi: 10.1016/j.trpro.2017.05.486

Bonnel, P., Fekih, M., & Smoreda, Z. (2018). Origin-Destination estimation using mobile network probe data. *Transportation Research Procedia*, *32*, 69–81. doi: 10.1016/j.trpro.2018.10.013

Braunschweig, K., Eberius, J., Thiele, M., & Lehner, W. (2012). The State of Open Data: Limits of Current Open Data Platforms Categories and Subject Descriptors. *WWW*. Retrieved from `https://wwwdb.inf.tu-dresden.de/opendatasurvey/www2012_short.pdf`

Breiman, L. (1996, Aug 01). Bagging predictors. *Machine Learning*, *24*(2), 123-140. Retrieved from `10.1007/BF00058655` doi: 10.1007/BF00058655

Brinkman, J., & Mangum, K. (2020). Travel behavior and the coronavirus outbreak. *Economic Insights*, *5*(3), 23–26. Retrieved from `https://fedinprint.org/item/fedpei/88742`

Broach, J., Dill, J., & Gliebe, J. (2012). Where do cyclists ride? A route choice model developed with revealed preference GPS data. *Transportation Research Part A: Policy and Practice*, *46*(10), 1730–1740. doi: 10.1016/j.tra.2012.07.005

Brochu, E., Cora, V. M., & de Freitas, N. (2010). *A tutorial on bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning.* arXiv. Retrieved from `https://arxiv.org/abs/1012.2599` doi: 10.48550/ARXIV.1012.2599

Buisson, C., Daamen, W., Punzo, V., Wagner, P., Montanino, M., & Ciuffo, B. (2014). Chapter 4: Calibration and validation principles. In W. Daamen, C. Buisson, & S. P. Hoogendoorn (Eds.), *Traffic Simulation and Data: Validation Methods and Applications* (p. 89-118). Boca Raton, FL (U.S.): CRC Press. doi: 10.1201/b17440

Bunge, M. (1963, 2023/08/14/). A general black box theory. *Philosophy of Science*, *30*(4), 346-358. Retrieved from `http://www.jstor.org/stable/186066` (Full publication date: Oct., 1963)

Buroni, G., Lebichot, B., & Bontempi, G. (2021). Ast-mtl: An attention-based multi-task learning strategy for traffic forecasting. *IEEE Access*, *9*, 77359-77370. doi: 10.1109/ACCESS.2021.3083412

Bwambale, A., Choudhury, C. F., & Hess, S. (2019). Modelling departure time choice using mobile phone data. *Transportation Research Part A: Policy and Practice*, *130*(September), 424–439. doi: 10.1016/j.tra.2019.09.054

Caceres, N., Wideberg, J. P., & Benitez, F. G. (2007). Deriving origin–destination data from a mobile phone network. *IET Intelligent Transport Systems*, *1*(1), 15–26. doi: 10.1049/iet-its:20060020

California Department of Transportation. (2020, 02). *PEMS USER GUIDE* (Tech. Rep. No. Version 6). Retrieved from `https://pems.dot.ca.gov/Papers/PeMS_Intro_User_Guide_v6.pdf`

Canning, D., & Fay, M. (1993). The effects of transportation networks on economic growth.
doi: 10.7916/D80K2H4N

Cantelmo, G., Cipriani, E., Gemma, A., & Nigro, M. (2014). An adaptive bi-level gradient procedure for the estimation of dynamic traffic demand. *IEEE Transactions on Intelligent Transportation Systems*, *15*(3), 1348–1361. doi: 10.1109/TITS.2014.2299734

Cantelmo, G., Kucharski, R., & Antoniou, C. (2020). Low-dimensional model for bike-sharing demand forecasting that explicitly accounts for weather data. *Transportation Research Record*, *2674*(8), 132-144. doi: 10.1177/0361198120932160

Cantelmo, G., Viti, F., Cipriani, E., & Nigro, M. (2015). A two-steps dynamic demand estimation approach sequentially adjusting generations and distributions. In *2015 ieee 18th international conference on intelligent transportation systems* (pp. 1477–1482). doi: 10.1109/ITSC.2015.241

Cantelmo, G., Viti, F., Cipriani, E., & Nigro, M. (2018). A utility-based dynamic demand estimation model that explicitly accounts for activity scheduling and duration. *Transportation Research Part A: Policy and Practice*, *114*, 303–320. doi: doi.org/10.1016/j.trpro.2017.05.025

Cantelmo, G., Viti, F., Tampère, C. M., Cipriani, E., & Nigro, M. (2014). Two-step approach for correction of seed matrix in dynamic demand estimation. *Transportation Research Record*, *2466*(1), 125–133. doi: 10.3141/2466-14

Capponi, A., Vitello, P., Fiandrino, C., Cantelmo, G., Kliazovich, D., Sorger, U., & Bouvry, P. (2019). Crowdsensed data learning-driven prediction of local businesses attractiveness in smart cities. In *2019 IEEE Symposium on Computers and Communications (ISCC)* (p. 1-6). doi: 10.1109/ISCC47284.2019.8969771

Cascetta, E. (2001). Transportation systems engineering: Theory and methods. Springer New York, NY. doi: 10.1007/978-1-4757-6873-2

Cascetta, E., Inaudi, D., & Marquis, G. (1993). Dynamic estimators of origin-destination matrices using traffic counts. *Transportation Science*, *27*(4), 363-373. Retrieved from `10.1287/trsc.27.4.363` doi: 10.1287/trsc.27.4.363

Cascetta, E., Papola, A., Marzano, V., Simonelli, F., & Vitiello, I. (2013). Quasi-dynamic estimation of O-D flows from traffic counts: Formulation, statistical validation and

performance analysis on real data. *Transportation Research Part B: Methodological*, *55*, 171–187. doi: 10.1016/j.trb.2013.06.007

Castiglione, J., Bradley, M., & Gliebe, J. (2014). *Activity-Based Travel Demand Models: A Primer* (Tech. Rep.). Washington, DC. doi: 10.17226/22357

Chandola, V., Banerjee, A., & Kumar, V. (2009, July). Anomaly detection: A survey. *ACM Comput. Surv.*, *41*(3). doi: 10.1145/1541880.1541882

Chaniotakis, E., & Antoniou, C. (2015). Use of Geotagged Social Media in Urban Settings: Empirical Evidence on Its Potential from Twitter. *IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC*, *2015-Octob*(1), 214–219. doi: 10.1109/ITSC.2015.44

Chaniotakis, E., Antoniou, C., Aifadopoulou, G., & Dimitriou, L. (2017). Inferring activities from social media data. *Transportation Research Record*, *2666*, 29–37. doi: 10.3141/2666-04

Chaniotakis, E., Antoniou, C., & Pereira, F. (2016). Mapping Social media for transportation studies. *IEEE Intelligent Systems*, *31*(6), 64–70. doi: 10.1109/MIS.2016.98

Chaniotakis, E., Antoniou, C., & Pereira, F. C. (2017). Enhancing resilience to disasters using social media. In *2017 5th IEEE International Conference on Models and Technologies for Intelligent Transportation Systems (MT-ITS)* (p. 699-703). doi: 10.1109/MTITS.2017.8005602

Chapman, P., Clinton, J., Kerber, R., Khabaza, T., Reinartz, T., Shearer, C., & Wirth, R. (2000). *CRISP-DM 1.0 Step-by-step data mining guide* (Tech. Rep.). The CRISP-DM consortium. Retrieved from `http://www.crisp-dm.org/CRISPWP-0800.pdf`

Charalabidis, Y., Alexopoulos, C., & Loukis, E. (2016). A taxonomy of open government data research areas and topics. *Journal of Organizational Computing and Electronic Commerce*, *26*(1-2), 41–63. doi: 10.1080/10919392.2015.1124720

Chen, C., Bian, L., & Ma, J. (2014). From traces to trajectories: How well can we guess activity locations from mobile phone traces? *Transportation Research Part C: Emerging Technologies*, *46*, 326–337. doi: 10.1016/j.trc.2014.07.001

Chen, C., Ma, J., Susilo, Y., Liu, Y., & Wang, M. (2016). The promises of big data and small data for travel behavior (aka human mobility) analysis. *Transportation Research Part C: Emerging Technologies*, *68*, 285-299. doi: 10.1016/j.trc.2016.04.005

Chen, T., & Guestrin, C. (2016). XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining* (pp. 785–794). New York, NY, USA: ACM. doi: 10.1145/2939672.2939785

Chen, X., Zhang, C., Zhao, X.-L., Saunier, N., & Sun, L. (2022). *Nonstationary temporal matrix factorization for multivariate time series forecasting.* arXiv. doi: 10.48550/ARXIV.2203.10651

Chen, Y., Mahmassani, H. S., & Frei, A. (2018). Incorporating social media in travel and activity choice models: conceptual framework and exploratory analysis. *International Journal of Urban Sciences*, *22*(2), 180–200. doi: 10.1080/12265934.2017.1331749

Childs, S., McLeod, J., Lomas, E., & Cook, G. (2014). Opening research data: issues and opportunities. *Records Management Journal*, *24*(2). doi: 10.1108/RMJ-01-2014-0005

Chinazzi, M., Davis, J. T., Ajelli, M., Gioannini, C., Litvinova, M., Merler, S., ... Vespignani, A. (2020). The effect of travel restrictions on the spread of the 2019 novel coronavirus (covid-19) outbreak. *Science*, *368*(6489), 395–400. doi: 10.1126/science.aba9757

Ciuffo, B., Punzo, V., & Montanino, M. (2014). Global sensitivity analysis techniques to simplify the calibration of traffic simulation models. methodology and application to the idm car-following model. *IET Intelligent Transport Systems*, *8*(5), 479-489. doi: 10.1049/iet-its.2013.0064

Claussmann, L., Revilloud, M., Gruyer, D., & Glaser, S. (2020). A review of motion planning for highway autonomous driving. *IEEE Transactions on Intelligent Transportation Systems*, *21*, 1826-1848. doi: 10.1109/TITS.2019.2913998

Coifman, B. (2014). Jam occupancy and other lingering problems with empirical fundamental relationships. *Transportation Research Record*, *2422*(1), 104-112. doi: 10.3141/2422-12

Coifman, B., & Li, L. (2017). A critical evaluation of the next generation simulation (ngsim) vehicle trajectory dataset. *Transportation Research Part B: Methodological*, *105*, 362-377. doi: 10.1016/j.trb.2017.09.018

Crawford, F., Watling, D. P., & Connors, R. D. (2018). Identifying road user classes based on repeated trip behaviour using Bluetooth data. *Transportation Research Part A: Policy and Practice*, *113*, 55–74. doi: 10.1016/j.tra.2018.03.027

Creative Commons. (2023). *Attribution 4.0 International.* Retrieved from `https://creativecommons.org/licenses/by/4.0/legalcode` (Accessed on 2023.03.28)

Cui, Y., Meng, C., He, Q., & Gao, J. (2018). Forecasting current and next trip purpose with social media data and Google Places. *Transportation Research Part C: Emerging Technologies*, *97*(September), 159–174. doi: 10.1016/j.trc.2018.10.017

Cui, Z., Ke, R., Pu, Z., & Wang, Y. (2020, 09). Stacked bidirectional and unidirectional lstm recurrent neural network for forecasting network-wide traffic state with missing values. *Transportation Research Part C: Emerging Technologies*, *118*, 102674. doi: 10.1016/j.trc.2020.102674

Daamen, W., Buisson, C., & Hoogendoorn, S. P. (Eds.). (2014). *Traffic simulation and data: Validation methods and applications.* Boca Raton, FL (U.S.): CRC Press. doi: 10.1201/b17440

Daganzo, C. F., & Geroliminis, N. (2008). An analytical approximation for the macroscopic fundamental diagram of urban traffic. *Transportation Research Part B: Methodological*, *42*(9), 771-781. doi: 10.1016/j.trb.2008.06.008

Daniels, R., & Mulley, C. (2013, Aug.). Explaining walking distance to public transport: The dominance of public transport supply. *Journal of Transport and Land Use*, *6*(2), 5-20. doi: 10.5198/jtlu.v6i2.308

Deligianni, S. P., Quddus, M., Morris, A., Anvuur, A., & Reed, S. (2017). Analyzing and modeling drivers' deceleration behavior from normal driving. *Transportation Research Record*, *2663*(1), 134-141. doi: 10.3141/2663-17

A Dictionary of Computing (6th ed.). (2008). Oxford University Press. (Machine learning)

Dietterich, T. G. (2000). Ensemble methods in machine learning. In *Multiple classifier systems* (pp. 1–15). Springer Berlin Heidelberg. doi: 10.1007/3-540-45014-9_1

Djukic, T., Lint, J. W. C. V., & Hoogendoorn, S. P. (2012). Application of principal component analysis to predict dynamic origin–destination matrices. *Transportation Research Record*, *2283*(1), 81-89. doi: 10.3141/2283-09

D'Silva, K., Noulas, A., Musolesi, M., Mascolo, C., & Sklar, M. (2018, May 18). Predicting the temporal activity patterns of new venues. *EPJ Data Science*, *7*(1), 13. doi: 10.1140/epjds/s13688-018-0142-z

Duan, Z., & Wei, Y. (2014). Revealing Taxi Driver Route Choice Characteristics Based on GPS Data. In *Cictp 2014* (pp. 565–573). Reston, VA. doi: 10.1061/9780784413623.055

Dunn, W. (2007). *Managing travel for planned special events handbook.* New York: US Department of Transportation. Retrieved from `https://ops.fhwa.dot.gov/eto_tim_pse/preparedness/pse/handbook.htm` (Accessed on 03.07.2023)

Duran-Rodas, D., Chaniotakis, E., & Antoniou, C. (2019). Built Environment Factors Affecting Bike Sharing Ridership: Data-Driven Approach for Multiple Cities. *Transportation Research Record*, *2673*(12), 55–68. doi: 10.1177/0361198119849908

Efthymiou, D., & Antoniou, C. (2012). Use of social media for transport data collection. *Procedia - Social and Behavioral Sciences*, *48*, 775 - 785. (Transport Research Arena 2012) doi: 10.1016/j.sbspro.2012.06.1055

Eskin, E. (2000). Anomaly detection over noisy data using learned probability distributions. In *Proceedings of the seventeenth international conference on machine learning* (p. 255–262). San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.

European Commission. (2013). *Directive 2013/37/EU of the European Parliament and of the Council.* Retrieved from `https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=celex%3A32013L0037` (Accessed on 11.07.2023)

European Commission. (2020). *Open data.* Retrieved from `https://ec.europa.eu/digital-single-market/en/open-data` (Accessed on 23.06.2020)

European Data Portal. (2018). *Open Data from private companies?* Retrieved from `https://www.europeandataportal.eu/en/news/open-data-private-companies`

Facebook. (2020). *Germany: High resolution population density maps + demographic estimates.* Retrieved from `https://data.humdata.org/organization/facebook?q=germany&ext_page_size=25` (Accessed on 20.07.2020)

Fang, H., Wang, L., & Yang, Y. (2020). Human mobility restrictions and the spread of the novel coronavirus (2019-ncov) in china. *Journal of Public Economics*, *191*, 104272. doi: 10.1016/j.jpubeco.2020.104272

Faroqi, H., Mesbah, M., & Kim, J. (2018). Applications of transit smart cards beyond a fare collection tool: A literature review. *Advances in Transportation Studies*, *45*(September), 107–122. doi: 10.4399/978255166098

Feng, T., & Timmermans, H. J. P. (2013). Transportation mode recognition using GPS and accelerometer data. *Transportation Research Part C: Emerging Technologies*, *37*, 118–130. doi: 10.1016/j.trc.2013.09.014

Fisher, R., Perkins, S., Walker, A., & Wolfart, E. (1997). *Hypermedia Image Processing Reference (HIPR).* http://homepages.inf.ed.ac.uk/rbf/HIPR2/morops.htm. (Last updated on 18.10.2012)

Fransen, K., Neutens, T., Farber, S., De Maeyer, P., Deruyter, G., & Witlox, F. (2015). Identifying public transport gaps using time-dependent accessibility levels. *Journal of Transport Geography*, *48*, 176–187. doi: 10.1016/j.jtrangeo.2015.09.008

Frederix, R., Viti, F., & Tampère, C. M. (2013). Dynamic origin–destination estimation in congested networks: theoretical findings and implications in practice. *Transportmetrica A: Transport Science*, *9*(6), 494–513. doi: 10.1080/18128602.2011.619587

Friedman, J. H. (2001). Greedy function approximation: A gradient boosting machine. *The Annals of Statistics*, *29*(5), 1189–1232. doi: 10.1214/aos/1013203451

Furno, A., Faouzi, E. N. E., Fiore, M., & Stanica, R. (2017). Fusing GPS probe and mobile phone data for enhanced land-use detection. *5th IEEE International Conference on Models and Technologies for Intelligent Transportation Systems, MT-ITS 2017 - Proceedings*, 696–698. doi: 10.1109/MTITS.2017.8005601

Gadziński, J. (2018). Perspectives of the use of smartphones in travel behaviour studies: Findings from a literature review and a pilot study. *Transportation Research Part C: Emerging Technologies*, *88*(July 2017), 74–86. doi: 10.1016/j.trc.2018.01.011

Gal, Y., & Ghahramani, Z. (2016). *Dropout as a bayesian approximation: Representing model uncertainty in deep learning.* Retrieved from `http://proceedings.mlr.press/v48/gal16.pdf`

Geroliminis, N., & Daganzo, C. F. (2008). Existence of urban-scale macroscopic fundamental diagrams: Some experimental findings. *Transportation Research Part B: Methodological*, *42*(9), 759-770. doi: 10.1016/j.trb.2008.02.002

Geroliminis, N., & Sun, J. (2011). Properties of a well-defined macroscopic fundamental diagram for urban traffic. *Transportation Research Part B: Methodological*, *45*(3), 605-617. doi: 10.1016/j.trb.2010.11.004

Geurs, K. T., & van Wee, B. (2004). Accessibility evaluation of land-use and transport strategies: review and research directions. *Journal of Transport Geography*, *12*(2), 127 - 140. doi: 10.1016/j.jtrangeo.2003.10.005

Gkountouna, O., Pfoser, D., & Züfle, A. (2020). Traffic flow estimation using probe vehicle data. In *2020 ieee 7th international conference on data science and advanced analytics (dsaa)* (p. 579-588). doi: 10.1109/DSAA49011.2020.00073

González-Jorge, H., Martínez-Sánchez, J., Bueno, M., & Arias, P. (2017, Jul). Unmanned aerial systems for civil applications: A review. *Drones*, *1*(1), 2. doi: 10.3390/drones1010002

Goodwin, P., & Lawton, R. (1999). On the asymmetry of the symmetric mape. *International Journal of Forecasting*, *15*(4), 405-408. doi: 10.1016/S0169-2070(99)00007-2

Google. (2020a). *Community mobility reports.* Retrieved from `https://www.google.com/covid19/mobility` (Accessed on 25.07.2020)

Google. (2020b). *Popular times, wait times, and visit duration.* Retrieved from `https://support.google.com/business/answer/6263531?hl=en`

GoogleMaps. (2020). *Google maps.* Retrieved from `https://www.google.com/maps/` (Accessed on 20.07.2020)

Grant-Muller, S. M., Gal-Tzur, A., Minkov, E., Nocera, S., Kuflik, T., & Shoor, I. (2015). Enhancing transport data collection through social media sources: Methods, challenges and opportunities for textual data. *IET Intelligent Transport Systems*, *9*(4), 407–417. doi: 10.1049/iet-its.2013.0214

Grantz, K. H., Meredith, H. R., Cummings, D. A. T., Metcalf, C. J. E., Grenfell, B. T., Giles, J. R., ... Wesolowski, A. (2020, Sep 30). The use of mobile phone data to inform analysis of covid-19 pandemic epidemiology. *Nature Communications*, *11*(1), 4961. doi: 10.1038/s41467-020-18190-5

Gray, C. L., & Mueller, V. (2012, Apr 17). Natural disasters and population mobility in bangladesh. *Proceedings of the National Academy of Sciences of the United States of America*, *109*(16), 6000-6005. (22474361[pmid]) doi: 10.1073/pnas.1115944109

Greenshields, B., Bibbins, J., Channing, W., & Miller, H. (1935). A study of traffic capacity. *Highway Research Board proceedings*, *1935*. Retrieved from `https://onlinepubs.trb.org/Onlinepubs/hrbproceedings/14/14P1-023.pdf` (Accessed on 11.07.2023)

Grengs, J., Wang, X., & Kostyniuk, L. (2008). Using GPS Data to Understand Driving Behavior. *Journal of Urban Technology*, *15*(2), 33–53. doi: 10.1080/10630730802401942

GTFS. (2020). *GTFS for Germany.* Retrieved from `https://gtfs.de/de/feeds/` (Accessed on 20.07.2020)

Guo, S., Song, C., Pei, T., Liu, Y., Ma, T., Du, Y., ... Wang, Y. (2019). Accessibility to urban parks for elderly residents: Perspectives from mobile phone data. *Landscape and Urban Planning*, *191*(January), 103642. doi: 10.1016/j.landurbplan.2019.103642

Gupta, A. (2005). *Observability of origin-destination matrices for dynamic traffic assignment* (Master's thesis, Massachusetts Institute of Technology). Retrieved from `https://dspace.mit.edu/handle/1721.1/33692` (Accessed on 20.06.2022)

Hainen, A. M., Wasson, J. S., Hubbard, S. M. L., Remias, S. M., Farnsworth, G. D., & Bullock, D. M. (2011). Estimating route choice and travel time reliability with field observations of bluetooth probe vehicles. *Transportation Research Record*(2256), 43–50. doi: 10.3141/2256-06

Hamstead, Z. A., Fisher, D., Ilieva, R. T., Wood, S. A., McPhearson, T., & Kremer, P. (2018). Geolocated social media as a rapid indicator of park visitation and equitable park access. *Computers, Environment and Urban Systems*, *72*(July 2017), 38–50. doi: 10.1016/j.compenvurbsys.2018.01.007

Hansen, W. G. (1959). How accessibility shapes land use. *Journal of the American Institute of Planners*, *25*(2), 73-76. doi: 10.1080/01944365908978307

Harrison, G., Grant-Muller, S. M., & Hodgson, F. C. (2020). New and emerging data forms in transportation planning and policy: Opportunities and challenges for

"track and trace" data. *Transportation Research Part C: Emerging Technologies*, *117*, 102672. doi: 10.1016/j.trc.2020.102672

Hastie, T., Tibshirani, R., & Friedman, J. (2001). *The elements of statistical learning.* New York, NY, USA: Springer New York Inc.

He, L., Ishibuchi, H., Trivedi, A., Wang, H., Nan, Y., & Srinivasan, D. (2021). A survey of normalization methods in multiobjective evolutionary algorithms. *IEEE Transactions on Evolutionary Computation*, *25*(6), 1028-1048. doi: 10.1109/TEVC.2021.3076514

Ho, M. C., Lim, J. M.-Y., Chong, C. Y., Chua, K. K., & Siah, A. K. L. (2023). High dimensional origin destination calibration using metamodel assisted simultaneous perturbation stochastic approximation. *IEEE Transactions on Intelligent Transportation Systems*, 1-10. doi: 10.1109/TITS.2023.3234615

Hochreiter, S. (1991). *Untersuchungen zu dynamischen neuronalen Netzen. Diploma thesis, Institut für Informatik, Lehrstuhl Prof. Brauer, Technische Universität München.*

Holloway, J. L. (1958). Smoothing and filtering of time series and space fields. In H. Landsberg & J. Van Mieghem (Eds.), (Vol. 4, p. 351-389). Elsevier. doi: 10.1016/S0065-2687(08)60487-2

Hoppe, M., Christ, A., Castro, A., Winter, M., & Seppänen, T.-M. (2014, Nov 20). Transformation in transportation? *European Journal of Futures Research*, *2*(1), 45. doi: 10.1007/s40309-014-0045-6

Horanont, T., Phithakkitnukoon, S., Leong, T. W., Sekimoto, Y., & Shibasaki, R. (2013, 12). Weather effects on the patterns of people's everyday activities: A study using gps traces of mobile phone users. *PLOS ONE*, *8*(12), 1-14. Retrieved from `10.1371/journal.pone.0081153` doi: 10.1371/journal.pone.0081153

Huang, G., Li, Y., Pleiss, G., Liu, Z., Hopcroft, J. E., & Weinberger, K. Q. (2017). *Snapshot ensembles: Train 1, get m for free.* arXiv. doi: 10.48550/ARXIV.1704.00109

Huang, H., Cheng, Y., & Weibel, R. (2019). Transport mode detection based on mobile phone network data: A systematic review. *Transportation Research Part C: Emerging Technologies*, *101*(January), 297–312. doi: 10.1016/j.trc.2019.02.008

Huber, P. J. (1973). Robust regression: Asymptotics, conjectures and monte carlo. *Ann. Statist.*, *1*(5), 799–821. doi: 10.1214/aos/1176342503

Islam, S. R., Markus, M., & Kumar, S. S. (2019, Jan 01). Walking to a public transport station: Empirical evidence on willingness and acceptance in munich, germany. *Smart and Sustainable Built Environment*, *9*(1), 38-53. doi: 10.1108/SASBE-07-2017-0031

ITF. (2021). *Big data for travel demand modelling: Summary and conclusions.* OECD Publishing, Paris: International Transport Forum Discussion Papers. Retrieved from `https://www.itf-oecd.org/big-data-travel-demand-modelling` (Accessed on 20.05.2023)

Izmailov, P., Podoprikhin, D., Garipov, T., Vetrov, D., & Wilson, A. G. (2018). *Averaging weights leads to wider optima and better generalization.* arXiv. doi: 10.48550/ARXIV.1803.05407

Jánošíkova, L., Slavík, J., & Kohání, M. (2014). Estimation of a route choice model for urban public transport using smart card data. *Transportation Planning and Technology*, *37*(7), 638–648. doi: 10.1080/03081060.2014.935570

Janssen, K. (2011). The influence of the PSI directive on open government data: An overview of recent developments. *Government Information Quarterly*, *28*(4), 446–456. doi: 10.1016/j.giq.2011.01.004

Janssen, M., Charalabidis, Y., & Zuiderwijk, A. (2012). Benefits, Adoption Barriers and Myths of Open Data and Open Government. *Information Systems Management*, *29*(4), 258–268. doi: 10.1080/10580530.2012.716740

Japkowicz, N., Myers, C., & Gluck, M. (1995). A novelty detection approach to classification. In *Proceedings of the 14th international joint conference on artificial intelligence - volume 1* (p. 518–523). San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.

Järv, O., Ahas, R., & Witlox, F. (2014). Understanding monthly variability in human activity spaces: A twelve-month study using mobile phone call detail records. *Transportation Research Part C: Emerging Technologies*, *38*, 122–135. doi: 10.1016/j.trc.2013.11.003

Jiang, W., & Luo, J. (2021). Graph Neural Network for Traffic Forecasting: A Survey. *CoRR*, *abs/2101.11174*. Retrieved from `https://arxiv.org/abs/2101.11174`

Kaffash, S., Nguyen, A. T., & Zhu, J. (2021). Big data algorithms and applications in intelligent transportation system: A review and bibliometric analysis. *International Journal of Production Economics*, *231*, 107868. doi: 10.1016/j.ijpe.2020.107868

Kalman, R. E. (1960, 03). A New Approach to Linear Filtering and Prediction Problems. *Journal of Basic Engineering*, *82*(1), 35-45. doi: 10.1115/1.3662552

Kanagaraj, V., Asaithambi, G., Toledo, T., & Lee, T.-C. (2015). Trajectory data and flow characteristics of mixed traffic. *Transportation Research Record*, *2491*(1), 1-11. doi: 10.3141/2491-01

Kelly, P., Krenn, P., Titze, S., Stopher, P., & Foster, C. (2013). Quantifying the Difference Between Self-Reported and Global Positioning Systems-Measured Journey Durations: A Systematic Review. *Transport Reviews*, *33*(4), 443–459. doi: 10.1080/01441647.2013.815288

Kerle, I. (2018, oct). *What is Public Data?* Retrieved from `https://enigma.com/blog/post/what-is-public-data` (Accessed on 11.07.2023)

Kickhöfer, B., Hosse, D., Turner, K., & Tirachinic, A. (2016). *Creating an open MATSim scenario from open data: The case of Santiago de Chile* (Tech. Rep.). Berlin. Retrieved from `https://svn.vsp.tu-berlin.de/repos/public-svn/publications/vspwp/2016/16-02/KickhoeferEtAl2016MatsimSantiago.pdf` (Accessed on 11.07.2023)

Kim, S., Anagnostopoulos, G., Barmpounakis, E., & Geroliminis, N. (2023, Jan). Visual extensions and anomaly detection in the pneuma experiment with a swarm of drones. *Transportation Research Part C: Emerging Technologies*. doi: 10.1016/j.trc.2022.103966

Kolassa, S., & Schütz, W. (2007). Advantages of the mad/mean ratio over the mape. *Foresight: The International Journal of Applied Forecasting*(6), 40-43. Retrieved from `https://EconPapers.repec.org/RePEc:for:ijafaa:y:2007:i:6:p:40-43`

Kostic, B., Gentile, G., & Antoniou, C. (2017). Techniques for improving the effectiveness of the spsa algorithm in dynamic demand calibration. In *2017 5th IEEE International Conference on Models and Technologies for Intelligent Transportation Systems (MT-ITS)* (p. 368-373). doi: 10.1109/MTITS.2017.8005699

Kruber, F., Wurst, J., Chakraborty, S., & Botsch, M. (2019). *Highway traffic data: macroscopic, microscopic and criticality analysis for capturing relevant traffic scenarios and traffic modeling based on the highd data set.* doi: 10.48550/arXiv .1903.04249

Kuehnel, N., Kaddoura, I., & Moeckel, R. (2019). Noise Shielding in an Agent-Based Transport Model Using Volunteered Geographic Data. *Procedia Computer Science*, *151*, 808–813. doi: 10.1016/j.procs.2019.04.110

Kühne, R. (2008). Foundation of traffic flow theory i: Greenshields legacy highway traffic. In *TRB Traffic Flow Theory and Characteristics Committee Summer Meeting and Symposium on the Fundamental Diagram: 75 years ("Greenshields75" Symposium).* Retrieved from `https://elib.dlr.de/54950/`

Kujala, R., Weckstrom, C., Darst, R. K., Mladenovic, M. N., & Saramaki, J. (2018). Data Descriptor: A collection of public transport network data sets for 25 cities. *Scientific Data*, *5*, 1–14. doi: 10.1038/sdata.2018.89

Kumarage, S. P., Rajapaksha, R. P. G. K. S., De Silva, D., & Bandara, J. M. S. J. (2018). Traffic flow estimation for urban roads based on crowdsourced data and machine learning principles. In T. Kováčiková, Ľ. Buzna, G. Pourhashem, G. Lugano, Y. Cornet, & N. Lugano (Eds.), *Intelligent transport systems – from research and development to the market uptake* (pp. 263–273). Cham: Springer International Publishing.

Langenkamp, M., & Yue, D. N. (2022). How Open Source Machine Learning Software Shapes AI. In *Proceedings of the 2022 aaai/acm conference on ai, ethics, and society* (p. 385–395). New York, NY, USA: Association for Computing Machinery. Retrieved from `10.1145/3514094.3534167` doi: 10.1145/3514094.3534167

Lantseva, A. A., & Ivanov, V. S. (2016). Modeling Transport Accessibility with Open Data: Case Study of St. Petersburg. In *Procedia computer science* (Vol. 101, pp. 197–206). doi: 10.1016/j.procs.2016.11.024

Laptev, N. P., Yosinski, J., Li, L. E., & Smyl, S. (2017). Time-series extreme event forecasting with neural networks at Uber.. Retrieved from `http://www.cs.columbia.edu/~lierranli/publications/TSW2017_paper.pdf`

Lara-Benítez, P., Carranza-García, M., & Riquelme, J. C. (2021, Feb). An experimental review on deep learning architectures for time series forecasting. *International Journal of Neural Systems*, *31*(03), 2130001. doi: 10.1142/s0129065721300011

Lefebvre, N., Chen, X., Beauseroy, P., & Zhu, M. (2017). Traffic flow estimation using acoustic signal. *Engineering Applications of Artificial Intelligence*, *64*, 164-171. doi: 10.1016/j.engappai.2017.05.019

Li, J., Guo, F., Sivakumar, A., Dong, Y., & Krishnan, R. (2021). Transferability improvement in short-term traffic prediction using stacked lstm network. *Transportation Research Part C: Emerging Technologies*, *124*, 102977. doi: 10.1016/j.trc.2021.102977

Li, Y., Yu, R., Shahabi, C., & Liu, Y. (2017). Diffusion Convolutional Recurrent Neural Network: Data-Driven Traffic Forecasting. *arXiv preprint arXiv:1707.01926*. doi: 10.48550/arXiv.1707.01926

Lim, B., Arık, S. O., Loeff, N., & Pfister, T. (2021). Temporal fusion transformers for interpretable multi-horizon time series forecasting. *International Journal of Forecasting*, *37*(4), 1748-1764. doi: 10.1016/j.ijforecast.2021.03.012

Lin, L., He, Z., & Peeta, S. (2018). Predicting station-level hourly demand in a large-scale bike-sharing network: A graph convolutional neural network approach. *Transportation Research Part C: Emerging Technologies*, *97*, 258-276. doi: 10.1016/j.trc.2018.10.011

Liu, J., Shen, H., & Zhang, X. (2016). A survey of mobile crowdsensing techniques: A critical component for the internet of things. In *2016 25th International Conference on Computer Communication and Networks (ICCCN)* (p. 1-6). doi: 10.1109/ICCCN.2016.7568484

Llorca, C., Ji, J., Molloy, J., & Moeckel, R. (2018). The usage of location based big data and trip planning services for the estimation of a long-distance travel demand model. Predicting the impacts of a new high speed rail corridor. *Research in Transportation Economics*, *72*(June), 27–36. doi: 10.1016/j.retrec.2018.06.004

Lopes, J., Bento, J., Huang, E., Antoniou, C., & Ben-Akiva, M. (2010). Traffic and mobility data collection for real-time applications. In *13th International IEEE Conference on Intelligent Transportation Systems* (pp. 216–223). doi: 10.1109/ITSC.2010.5625282

Lopez, P. A., Behrisch, M., Bieker-Walz, L., Erdmann, J., Flötteröd, Y.-P., Hilbrich, R., ... Wießner, E. (2018, November). Microscopic traffic simulation using sumo. In *The 21st IEEE International Conference on Intelligent Transportation Systems* (pp. 2575–2582). IEEE. doi: 10.1109/ITSC.2018.8569938

Lorch, L., Trouleau, W., Tsirtsis, S., Szanto, A., Schölkopf, B., & Gomez-Rodriguez, M. (2020). *A spatiotemporal epidemic model to quantify the effects of contact tracing, testing, and containment.* doi: 10.48550/arXiv.2004.07641

Loshchilov, I., & Hutter, F. (2016). *SGDR: Stochastic Gradient Descent with Warm Restarts.* arXiv. doi: 10.48550/ARXIV.1608.03983

Lu, L., Xu, Y., Antoniou, C., & Ben-Akiva, M. (2015). An enhanced SPSA algorithm for the calibration of Dynamic Traffic Assignment models. *Transportation Research Part C: Emerging Technologies*, *51*, 149-166. doi: 10.1016/j.trc.2014.11.006

Luan, J., Guo, F., Polak, J., Hoose, N., & Krishnan, R. (2018). Investigating the transferability of machine learning methods in short-term travel time prediction. In *97th Annual Meeting Transportation Research Board.* Retrieved from `https://trid.trb.org/view/1495445`

Lundberg, S. M., Erion, G., Chen, H., DeGrave, A., Prutkin, J. M., Nair, B., ... Lee, S.-I. (2020, Jan 01). From local explanations to global understanding with explainable ai for trees. *Nature Machine Intelligence*, *2*(1), 56-67. doi: 10.1038/s42256-019-0138-9

Lundberg, S. M., & Lee, S.-I. (2017). A unified approach to interpreting model predictions. In I. Guyon et al. (Eds.), *Advances in Neural Information Processing Systems 30* (pp. 4765–4774). Curran Associates, Inc. doi: 10.48550/arXiv.1705.07874

m-wrzr, & riedmaph. (2018). *populartimes.* `https://github.com/m-wrzr/populartimes`. GitHub. (Accessed on 20.07.2020)

Ma, D., Song, X., & Li, P. (2021). Daily traffic flow forecasting through a contextual convolutional recurrent neural network modeling inter- and intra-day traffic patterns. *IEEE Transactions on Intelligent Transportation Systems*, *22*(5), 2627-2636. doi: 10.1109/TITS.2020.2973279

Ma, T., & Abdulhai, B. (2002). Genetic algorithm-based optimization approach and generic tool for calibrating traffic microscopic simulation parameters. *Transportation Research Record*, *1800*(1), 6-15. Retrieved from `10.3141/1800-02` doi: 10.3141/1800-02

Ma, T., Antoniou, C., & Toledo, T. (2020). Hybrid machine learning algorithm and statistical time series model for networkwide traffic forecast. *Transportation Research Part C: Emerging Technologies*, *111*, 352-372. doi: 10.1016/j.trc.2019.12.022

Máchová, R., Hub, M., & Lnenicka, M. (2018, Jan 01). Usability evaluation of open data portals. *Aslib Journal of Information Management*, *70*(3), 252-268. doi: 10.1108/AJIM-02-2018-0026

Máchová, R., & Lněnička, M. (2017). Evaluating the quality of open data portals on the national level. *Journal of Theoretical and Applied Electronic Commerce Research*, *12*(1), 21–41. doi: 10.4067/S0718-18762017000100003

MacKenzie, D., & Cho, H. (2020). Travel demand and emissions from driving dogs to dog parks. *Transportation Research Record*, *2674*(6), 291-296. doi: 10.1177/0361198120918870

Maghrebi, M., Abbasi, A., & Waller, S. T. (2016). Transportation application of social media: Travel mode extraction. *IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC*, 1648–1653. doi: 10.1109/ITSC.2016.7795779

Mahajan, V., Barmpounakis, E., Alam, M. R., Geroliminis, N., & Antoniou, C. (2023). Treating noise and anomalies in vehicle trajectories from an experiment with a swarm of drones. *IEEE Transactions on Intelligent Transportation Systems. ©2023 IEEE*. doi: 10.1109/TITS.2023.3268712

Mahajan, V., Cantelmo, G., & Antoniou, C. (2021). Explaining demand patterns during covid-19 using opportunistic data: a case study of the city of munich. *European Transport Research Review*, *13*(1), 26. doi: 10.1186/s12544-021-00485-3

Mahajan, V., Cantelmo, G., & Antoniou, C. (in review). One-shot heuristic and ensembling for automated calibration of large-scale traffic simulations. *in review*.

Mahajan, V., Cantelmo, G., Rothfeld, R., & Antoniou, C. (2023). Predicting network flows from speeds using open data and transfer learning. *IET Intelligent Transport Systems*, *17*(4), 804-824. doi: 10.1049/itr2.12305

Mahajan, V., Katrakazas, C., & Antoniou, C. (2020). Prediction of lane-changing maneuvers with automatic labeling and deep learning. *Transportation Research Record*, *2674*(7), 336-347. doi: 10.1177/0361198120922210

Mahajan, V., Kuehnel, N., Intzevidou, A., Cantelmo, G., Moeckel, R., & Antoniou, C. (2022). Data to the people: a review of public and proprietary data for transport models. *Transport Reviews*, *42*(4), 415-440. doi: 10.1080/01441647.2021.1977414

Mahmassani, H. S., Saberi, M., & Zockaie, A. (2013). Urban network gridlock: Theory, characteristics, and dynamics. *Transportation Research Part C: Emerging Technologies*, *36*, 480-497. doi: 10.1016/j.trc.2013.07.002

Makridis, M., Mattas, K., Anesiadou, A., & Ciuffo, B. (2021). Openacc. an open database of car-following experiments to study the properties of commercial acc systems. *Transportation Research Part C: Emerging Technologies*, *125*, 103047. doi: 10.1016/j.trc.2021.103047

Malinovskiy, Y., Saunier, N., & Wang, Y. (2012). Analysis of pedestrian travel with static bluetooth sensors. *Transportation Research Record*(2299), 137–149. doi: 10.3141/2299-15

Mallick, T., Balaprakash, P., Rask, E., & Macfarlane, J. (2021). Transfer learning with graph neural networks for short-term highway traffic forecasting. In *2020 25th International Conference on Pattern Recognition (ICPR)* (pp. 10367–10374). doi: 10.1109/ICPR48806.2021.9413270

Martínez-Plumed, F., Contreras-Ochando, L., Ferri, C., Hernández-Orallo, J., Kull, M., Lachiche, N., ... Flach, P. (2021). Crisp-dm twenty years later: From data mining processes to data science trajectories. *IEEE Transactions on Knowledge and Data Engineering*, *33*(8), 3048-3061. doi: 10.1109/TKDE.2019.2962680

Mayer, A. (2017). Noisyopt: A python library for optimizing noisy functions. *Journal of Open Source Software*, *2*(13), 258. doi: 10.21105/joss.00258

Merriam-Webster. (2020). *Data.* Retrieved from `https://www.merriam-webster.com/dictionary/data`

Milne, D., & Watling, D. (2019). Big data and understanding change in the context of planning transport systems. *Journal of Transport Geography*, *76*(October 2017), 235–244. doi: 10.1016/j.jtrangeo.2017.11.004

Mockus, M., & Palmirani, M. (2015). Open government data licensing framework. In A. Kő & E. Francesconi (Eds.), *Electronic government and the information systems perspective* (pp. 287–301). Cham: Springer International Publishing. doi: 10.1007/978-3-319-22389-6_21

Moeckel, R., Kuehnel, N., Llorca, C., Moreno, A. T., & Rayaprolu, H. (2020, Feb 25). Agent-based simulation to improve policy sensitivity of trip-based models. *Journal of Advanced Transportation*, *2020*, 1902162. doi: 10.1155/2020/1902162

Montanino, M., & Punzo, V. (2013). Making ngsim data usable for studies on traffic flow theory: Multistep method for vehicle trajectory reconstruction. *Transportation Research Record*, *2390*(1), 99-111. doi: 10.3141/2390-11

Montanino, M., & Punzo, V. (2015). Trajectory data reconstruction and simulation-based validation against macroscopic traffic patterns. *Transportation Research Part B: Methodological*, *80*, 82 - 106. doi: 10.1016/j.trb.2015.06.010

Möhring, M., Keller, B., Schmidt, R., & Dacko, S. (2020). Google Popular Times: towards a better understanding of tourist customer patronage behavior. *Tourism Review*, *ahead-of-print*(ahead-of-print). doi: 10.1108/TR-10-2018-0152

Münchner Verkehrsgesellschaft mbH (MVG). (2021). *Together against corona: Passenger traffic.* Retrieved from `https://www.mvg.de/services/aktuelles/coronavirus.html` (Accessed on 15.01.2021)

Narayanan, S., & Antoniou, C. (2022). Electric cargo cycles - a comprehensive review. *Transport Policy*, *116*, 278-303. doi: 10.1016/j.tranpol.2021.12.011

Neumann, T., Böhnke, P. L., & Touko Tcheumadjeu, L. C. (2013). Dynamic representation of the fundamental diagram via bayesian networks for estimating traffic flows from probe vehicle data. In *16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013)* (p. 1870-1875). doi: 10.1109/ITSC.2013.6728501

Ni, M., He, Q., & Gao, J. (2017). Forecasting the Subway Passenger Flow under Event Occurrences with Social Media. *IEEE Transactions on Intelligent Transportation Systems*, *18*(6), 1623–1632. doi: 10.1109/TITS.2016.2611644

Nicola, M., Alsafi, Z., Sohrabi, C., Kerwan, A., Al-Jabir, A., Iosifidis, C., ... Agha, R. (2020). The socio-economic implications of the coronavirus pandemic (COVID-19): A review. *International Journal of Surgery*, *78*, 185–193. doi: 10.1016/j.ijsu.2020.04.018

Nogueira, F. (2014). *Bayesian Optimization: Open source constrained global optimization tool for Python.* Retrieved from `https://github.com/fmfn/BayesianOptimization`

North American Bikeshare Association. (2015, aug). *General Bikeshare Feed Specification.* Retrieved from `https://github.com/NABSA/gbfs`

OECD. (2019). *Enhancing Access to and Sharing of Data.* doi: 10.1787/276aaca8-en

OGP. (2011). *Open Government Partnership.* Retrieved from `https://www.opengovpartnership.org/` (Accessed on 23.04.2022)

O'Haver, T. (2022). Signals and noise. In T. O'Haver (Ed.), *A pragmatic introduction to signal processing* (p. 23). Maryland, USA: Author.

Omrani, R., & Kattan, L. (2012). Demand and supply calibration of dynamic traffic assignment models: Past efforts and future challenges. *Transportation Research Record*, *2283*(1), 100-112. doi: 10.3141/2283-11

Omrani, R., & Kattan, L. (2018). Concurrent estimation of origin-destination flows and calibration of microscopic traffic simulation parameters in a high-performance computing cluster. *Journal of Transportation Engineering, Part A: Systems*, *144*(1), 04017068. doi: 10.1061/JTEPBS.0000093

Open Data Madrid. (2022). *Tráfico. Histórico de datos del tráfico desde 2013.* `https://datos.madrid.es/portal`. (Accessed on 05.04.2022)

Open Data Paris. (2020). *Comptage routier - données trafic issues des capteurs permanents.* `https://opendata.paris.fr/explore/dataset/comptages-routiers-permanents`. (Accessed on 05.04.2022)

OpenStreetMap. (2021). *Highway link.* `https://wiki.openstreetmap.org/wiki/Highway_link/`. (Accessed on 05.04.2022)

OpenStreetMap. (2022). *Highway link.* `https://wiki.openstreetmap.org/wiki/Key:highway/`. (Accessed on 05.04.2022)

OpenStreetMap Contributors. (2018, jul). *OpenStreetMap.* Retrieved from `http://www.openstreetmap.org` (Accessed on 05.04.2022)

Osorio, C. (2019). Dynamic origin-destination matrix calibration for large-scale network simulators. *Transportation Research Part C: Emerging Technologies*, *98*, 186-206. doi: 10.1016/j.trc.2018.09.023

Osorio-Arjona, J., & García-Palomares, J. C. (2019). Social media and urban mobility: Using twitter to calculate home-work travel matrices. *Cities*, *89*(September 2018), 268–280. doi: 10.1016/j.cities.2019.03.006

Owen, A., & Levinson, D. M. (2017). Developing a comprehensive u.s. transit accessibility database. In P. Thakuriah, N. Tilahun, M. B. T. S. C. T. B. D. S. G. Zellner, P. Thakuriah, N. Tilahun, & M. Zellner (Eds.), *Seeing cities through big data. springer geography.* (pp. 279–290). doi: 10.1007/978-3-319-40902-3_16

Pan, G., Qi, G., Wu, Z., Zhang, D., & Li, S. (2013). Land-use classification using taxi GPS traces. *IEEE Transactions on Intelligent Transportation Systems*, *14*(1), 113–123. doi: 10.1109/TITS.2012.2209201

Pan, S. J., & Yang, Q. (2010). A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, *22*(10), 1345-1359. doi: 10.1109/TKDE.2009.191

Papinski, D., Scott, D. M., & Doherty, S. T. (2009). Exploring the route choice decision-making process: A comparison of planned and observed routes obtained using person-based GPS. *Transportation Research Part F: Traffic Psychology and Behaviour*, *12*(4), 347–358. doi: 10.1016/j.trf.2009.04.001

Pasquetto, V. I., Randles, B. M., & Borgman, C. L. (2017). On the reuse of scientific data. *Data Science Journal*, *16*(Borgman 2015), 1–9. doi: 10.5334/dsj-2017-008

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., ... Chintala, S. (2019). Pytorch: An imperative style, high-performance deep learning library. In (pp. 8024–8035). Curran Associates, Inc. doi: 10.48550/arXiv.1912.01703

Pelletier, M. P., Trépanier, M., & Morency, C. (2011). Smart card data use in public transit: A literature review. *Transportation Research Part C: Emerging Technologies*, *19*(4), 557–568. doi: 10.1016/j.trc.2010.12.003

Pham, H. Q., Camey, M., Pham, K. D., Pham, K. V., & Rilett, L. R. (2020). Review of unmanned aerial vehicles (uavs) operation and data collection for driving behavior analysis. In C. Ha-Minh, D. V. Dao, F. Benboudjema, S. Derrible, D. V. K. Huynh, & A. M. Tang (Eds.), *CIGOS 2019, Innovation for Sustainable Infrastructure* (pp. 1111–1116). Singapore: Springer Singapore.

Polson, N. G., & Sokolov, V. O. (2017). Deep learning for short-term traffic flow prediction. *Transportation Research Part C: Emerging Technologies*, *79*, 1-17. doi: 10.1016/j.trc.2017.02.024

Polyak, B. T., & Juditsky, A. B. (1992). Acceleration of stochastic approximation by averaging. *SIAM Journal on Control and Optimization*, *30*(4), 838-855. Retrieved from `10.1137/0330046` doi: 10.1137/0330046

Prakash, A. A., Seshadri, R., Antoniou, C., Pereira, F. C., & Ben-Akiva, M. E. (2017). Reducing the dimension of online calibration in dynamic traffic assignment systems. *Transportation Research Record*, *2667*(1), 96-107. doi: 10.3141/2667-10

Prelipcean, A. C., Gidófalvi, G., & Susilo, Y. O. (2017). Transportation mode detection–an in-depth review of applicability and reliability. *Transport Reviews*, *37*(4), 442–464. doi: 10.1080/01441647.2016.1246489

Pullano, G., Valdano, E., Scarpa, N., Rubrichi, S., & Colizza, V. (2020, 2020/11/05). Evaluating the effect of demographic factors, socioeconomic factors, and risk aversion on mobility during the covid-19 epidemic in france under lockdown: a population-based study. *The Lancet Digital Health*. doi: 10.1016/S2589-7500(20)30243-0

Pun, L., Zhao, P., & Liu, X. (2019). A multiple regression approach for traffic flow estimation. *IEEE Access*, *7*, 35998-36009. doi: 10.1109/ACCESS.2019.2904645

Punzo, V., Borzacchiello, M. T., & Ciuffo, B. (2011). On the assessment of vehicle trajectory data accuracy and application to the next generation simulation (ngsim) program data. *Transportation Research Part C: Emerging Technologies*, *19*(6), 1243-1262. doi: 10.1016/j.trc.2010.12.007

Qurashi, M., Lu, Q.-L., Cantelmo, G., & Antoniou, C. (2022). Dynamic demand estimation on large scale networks using principal component analysis: The case of non-existent or irrelevant historical estimates. *Transportation Research Part C: Emerging Technologies*, *136*, 103504. doi: 10.1016/j.trc.2021.103504

Qurashi, M., Ma, T., Chaniotakis, E., & Antoniou, C. (2020). PC–SPSA: Employing dimensionality reduction to limit spsa search noise in dta model calibration. *IEEE Transactions on Intelligent Transportation Systems*, *21*(4), 1635-1645. doi: 10.1109/TITS.2019.2915273

Rafati Fard, M., Shariat Mohaymany, A., & Shahri, M. (2017). A new methodology for vehicle trajectory reconstruction based on wavelet analysis. *Transportation Research Part C: Emerging Technologies*, *74*, 150-167. doi: 10.1016/j.trc.2016.11.010

Raifer, M. (2020). *Overpass-turbo.* Retrieved from `http://overpass-turbo.eu/` (Accessed on 20.07.2020)

Rashidi, T., Abbasi, A., Maghrebi, M., Hasan, S., & Waller, T. S. (2017). Exploring the capacity of social media data for modelling travel behaviour: Opportunities and challenges. *Transportation Research Part C: Emerging Technologies*, *75*, 197–211. doi: 10.1016/j.trc.2016.12.008

Redko, I., Habrard, A., Morvant, E., Sebban, M., & Bennani, Y. (2019). 2 - domain adaptation problem. In I. Redko, A. Habrard, E. Morvant, M. Sebban, & Y. Bennani (Eds.), *Advances in domain adaptation theory* (p. 21-36). Elsevier. doi: 10.1016/B978-1-78548-236-6.50002-7

Rieck, D., Schünemann, B., & Radusch, I. (2015). Advanced traffic light information in openstreetmap for traffic simulations. In M. Behrisch, M. B. T. M. M. w. O. D. L. N. i. M. Weber, M. Behrisch, & M. Weber (Eds.), *Modeling mobility with open data. lecture notes in mobility.* doi: 10.1007/978-3-319-15024-6_2

Rinaldi, M., & Viti, F. (2020). A cascading Kalman filtering framework for real-time urban network flow estimation. In *2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC)* (p. 1-6). doi: 10.1109/ITSC45102.2020.9294175

Rink, K. (2021). *Time Series Forecast Error Metrics You Should Know.* `https://towardsdatascience.com/time-series-forecast-error-metrics-you-should-\know-cc88b8c67f27`. (Accessed on 05.04.2022)

Robbins, H., & Monro, S. (1951). A stochastic approximation method. *The Annals of Mathematical Statistics*, *22*(3), 400–407. Retrieved 2023-02-09, from `http://www.jstor.org/stable/2236626`

Robert Koch Institute & Humboldt University of Berlin. (2020). *COVID-19 Mobility Project.* Retrieved from `https://www.covid-19-mobility.org/reports/first-report-general-mobility/` (Accessed on 20.07.2020)

Rojas, M. B., Sadeghvaziri, E., & Jin, X. (2016). Comprehensive review of travel behavior and mobility pattern studies that used mobile phone data. *Transportation Research Record*, *2563*(2563), 71–79. doi: 10.3141/2563-11

Rolph, I. K. (1932). The Population Pattern in Relation to Retail Buying: As Exemplified in Baltimore. *American Journal of Sociology*, *38*(3), 368–376. Retrieved from `http://www.jstor.org/stable/2767477`

Roy, A., & Kar, B. (2020). Characterizing the spread of covid-19 from human mobility patterns and sociodemographic indicators. In *Proceedings of the 3rd ACM SIGSPATIAL International Workshop on Advances in Resilient and Intelligent Cities* (p. 39–48). New York, NY, USA: Association for Computing Machinery. doi: 10.1145/3423455.3430303

Ruder, S. (2016). An overview of gradient descent optimization algorithms. *CoRR*, *abs/1609.04747*. doi: 10.48550/arXiv.1609.04747

Ruppert, D. (1988). Efficient estimations from a slowly convergent robbins-monro process.. Retrieved from `https://www.researchgate.net/publication/246031929_Efficient_Estimations_from_a_Slowly_Convergent_Robbins-Monro_Process` (Accessed on 01.07.2023)

Ryeng, E. O., Haugen, T., Grønlund, H., & Overå, S. B. (2016). Evaluating Bluetooth and Wi-Fi Sensors as a Tool for Collecting Bicycle Speed at Varying Gradients. *Transportation Research Procedia*, *14*(2352), 2289–2296. doi: 10.1016/j.trpro.2016.05.245

Sabir, M. (2010). *Impact of weather on daily travel demand.* Amsterdam: VU University, Department of Spatial Economics. Retrieved from `https://research.vu.nl/en/publications/weather-and-travel-behaviour` (Monograph)

Sakurada, M., & Yairi, T. (2014). Anomaly detection using autoencoders with nonlinear dimensionality reduction. In *Proceedings of the mlsda 2014 2nd workshop on machine learning for sensory data analysis* (p. 4–11). New York, NY, USA: Association for Computing Machinery. doi: 10.1145/2689746.2689747

Sandim, M., Rossetti, R. J. F., Moura, D. C., Kokkinogenis, Z., & Ŕubio, T. R. P. M. (2016). Using GPS-based AVL data to calculate and predict traffic network performance metrics: A systematic review. *IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC*, 1692–1699. doi: 10.1109/ITSC.2016.7795786

Sangster, J., Rakha, H., & Du, J. (2013). Application of naturalistic driving data to modeling of driver car-following behavior. *Transportation Research Record*, *2390*(1), 20-33. doi: 10.3141/2390-03

Savitzky, A., & Golay, M. J. E. (1964). Smoothing and differentiation of data by simplified least squares procedures. *Analytical Chemistry*, *36*(8), 1627-1639. doi: 10.1021/ac60214a047

Schafer, R. W. (2011). What is a savitzky-golay filter? [lecture notes]. *IEEE Signal Processing Magazine*, *28*(4), 111-117. doi: 10.1109/MSP.2011.941097

Schlaich, J. (2010). Analyzing route choice behavior with mobile phone trajectories. *Transportation Research Record*(2157), 78–85. doi: 10.3141/2157-10

Senaratne, H., Mobasheri, A., Ali, A. L., Capineri, C., & Haklay, M. M. (2017). A review of volunteered geographic information quality assessment methods. *International Journal of Geographical Information Science*, *31*(1), 139–167. doi: 10.1080/13658816.2016.1189556

Seo, T., Bayen, A. M., Kusakabe, T., & Asakura, Y. (2017). Traffic state estimation on highway: A comprehensive survey. *Annual Reviews in Control*, *43*, 128-151. doi: 10.1016/j.arcontrol.2017.03.005

Shapley, L. S. (1988). A value for n-person games. In A. E. Roth (Ed.), *The shapley value: Essays in honor of lloyd s. shapley* (p. 31–40). Cambridge University Press. doi: 10.1017/CBO9780511528446.003

Sharedstreets. (2017). *sharedstreets-matcher.* GitHub. Retrieved from `https://github.com/sharedstreets/sharedstreets-matcher` (Accessed: 02.09.2021)

Sharedstreets. (2018). *sharedstreets-js.* GitHub. Retrieved from `https://github.com/sharedstreets/sharedstreets-js` (Accessed: 02.09.2021)

Shwartz-Ziv, R., & Armon, A. (2021). Tabular Data: Deep Learning is Not All You Need. *CoRR*, *abs/2106.03253*. doi: 10.48550/arXiv.2106.03253

Smith, S., Berg, I., & Yang, C. (2020). *General Modeling Network Specification: documentation, software and data* (Tech. Rep.). Cambridge. doi: 10.21949/1518740

Spall, J. (1998a). Implementation of the simultaneous perturbation algorithm for stochastic optimization. *IEEE Transactions on Aerospace and Electronic Systems*, *34*(3), 817-823. doi: 10.1109/7.705889

Spall, J. (1998b). An overview of the simultaneous perturbation method for efficient optimization. *Johns Hopkins Apl Technical Digest*, *19*, 482-492. Retrieved from `https://www.jhuapl.edu/SPSA/PDF-SPSA/Spall_An_Overview.PDF`

Spall, J. (2003). Stochastic Approximation for Nonlinear Root-Finding. In *Introduction to Stochastic Search and Optimization* (p. 95-125). John Wiley & Sons, Ltd. doi: 10.1002/0471722138.ch4

Spall, J., & Cristion, J. (1994). Nonlinear adaptive control using neural networks: estimation with a smoothed form of simultaneous perturbation gradient approximation. In *Proceedings of 1994 American Control Conference - ACC '94* (Vol. 3, p. 2560-2564 vol.3). doi: 10.1109/ACC.1994.735021

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, *15*(56), 1929-1958. Retrieved from `http://jmlr.org/papers/v15/srivastava14a.html`

Stadtmuenchen. (2020). Retrieved from `https://twitter.com/StadtMuenchen` (Accessed on 20.11.2020)

Statista. (2023). *Number of social media users worldwide from 2017 to 2027.* Retrieved from `https://www.statista.com/statistics/278414/number-of-worldwide-social-network-users/` (Accessed on 20.04.2023)

statsmodels. (2020). *Linear regression.* Retrieved from `https://www.statsmodels.org/stable/regression.html` (Accessed on 20.07.2020)

Stickel, J. J. (2010). Data smoothing and numerical differentiation by a regularization method. *Computers & Chemical Engineering*, *34*(4), 467-475. doi: 10.1016/j.compchemeng.2009.10.007

Stopher, P., FitzGerald, C., & Xu, M. (2007). Assessing the accuracy of the Sydney Household Travel Survey with GPS. *Transportation*, *34*(6), 723–741. doi: 10.1007/s11116-007-9126-8

Strong, C., & Wolenetz, J. (2005). *Pilot Test of Automatic Vehicle Location on Snow Plows* (Tech. Rep.). Helena: Montana Department of Transportation Maintenance Division. Retrieved from `https://westerntransportationinstitute.org/research_projects/pilot-test-of-automatic-vehicle-location-on-snow-plows/` (Accessed on 02.03.2023)

Suber, P. (2008, 08). Gratis and libre open access. In *SPARC Open Access Newsletter.* Retrieved from `http://nrs.harvard.edu/urn-3:HUL.InstRepos:4322580` (Accessed on 02.07.2023)

SUMO. (2023a). *Automatic routing.* Retrieved from `https://sumo.dlr.de/docs/Demand/Automatic_Routing.html` (Accessed on 29.03.2023)

SUMO. (2023b). *Meso.* Retrieved from `https://sumo.dlr.de/docs/Simulation/Meso.html#tls-penalty` (Accessed on 29.03.2023)

SUMO. (2023c). *Routing.* Retrieved from `https://sumo.dlr.de/docs/Simulation/Routing.html#routing_by_traveltime_and_edge_priority` (Accessed on 29.03.2023)

Sunlight Foundation. (2010). *TEN PRINCIPLES FOR OPENING UP GOVERNMENT INFORMATION.* Retrieved from `https://sunlightfoundation.com/policy/documents/ten-open-data-principles/` (Accessed on 12.06.2023)

Tang, J., Liang, J., Zhang, S., Huang, H., & Liu, F. (2018). Inferring driving trajectories based on probabilistic model from large scale taxi GPS data. *Physica A: Statistical Mechanics and its Applications*, *506*, 566–577. doi: 10.1016/j.physa.2018.04.073

Tavassoli, A., Mesbah, M., & Hickman, M. (2017). Application of smart card data in validating a large-scale multi-modal transit assignment model. *Public Transport*, *10*(1). doi: 10.1007/s12469-017-0171-1

Tedjopurnomo, D. A., Bao, Z., Zheng, B., Choudhury, F., & Qin, A. K. (2020). A survey on modern deep neural network for traffic prediction: Trends, methods and challenges. *IEEE Transactions on Knowledge and Data Engineering*, 1-1. doi: 10.1109/TKDE.2020.3001195

Teh, H. Y., Kempa-Liehr, A. W., & Wang, K. I.-K. (2020, Feb 11). Sensor data quality: a systematic review. *Journal of Big Data*, *7*(1), 11. doi: 10.1186/s40537-020-0285-1

The World Bank. (2019, dec). *Open Data Essentials.* Retrieved from `http://opendatatoolkit.worldbank.org/en/essentials.html` (Accessed on 10.06.2023)

Time and Date. (2020). *Time and date: April 2020 weather in munich.* Retrieved from `https://www.timeanddate.com/weather/germany/munich/historic?month=4&year=2020` (Accessed on 20.11.2020)

Timokhin, S., Sadrani, M., & Antoniou, C. (2020, Aug). Predicting venue popularity using crowd-sourced and passive sensor data. *Smart Cities*, *3*(3), 818–841. Retrieved from `http://dx.doi.org/10.3390/smartcities3030042` doi: 10.3390/smartcities3030042

Tizghadam, A., Khazaei, H., Moghaddam, M. H. Y., & Hassan, Y. (2019, Jun 04). Machine learning in transportation. *Journal of Advanced Transportation*, *2019*, 4359785. doi: 10.1155/2019/4359785

Toledo, T., Kolechkina, T., Wagner, P., Ciuffo, B., Lima Azevedo, C., Marzano, V., & Flötteröd, G. (2014, 09). Network model calibration studies. In W. Daamen, C. Buisson, & S. P. Hoogendoorn (Eds.), *Traffic Simulation and Data: Validation Methods and Applications* (p. 22). Boca Raton, FL (U.S.): CRC Press. doi: 10.1201/b17440

Tolouei, R., Psarras, S., & Prince, R. (2017). Origin-Destination Trip Matrix Development: Conventional Methods versus Mobile Phone Data. *Transportation Research Procedia*, *26*(2016), 39–52. doi: 10.1016/j.trpro.2017.07.007

TomTom. (2021). *Traffic stats.* `https://www.tomtom.com/products/traffic-stats/`. (Accessed on 02.09.2021)

Torch Contributors. (2019). *LSTM.* `https://pytorch.org/docs/stable/generated/torch.nn.LSTM.html`. (Accessed on 02.09.2021)

Torre-Bastida, A. I., Del Ser, J., Laña, I., Ilardia, M., Bilbao, M. N., & Campos-Cordobés, S. (2018). Big data for transportation and mobility: recent advances, trends and challenges. *IET Intelligent Transport Systems*, *12*(8), 742-755. doi: 10.1049/iet-its.2018.5188

Torrey, L., & Shavlik, J. (2010). Transfer learning. In (p. 242-264). Hershey, PA, USA: IGI Global. doi: 10.4018/978-1-60566-766-9.ch011

Travers, J. (2010, jul). *Inside Inrix–How traffic data is collected, what it means to your commute.* Retrieved from `https://www.consumerreports.org/cro/news/2010/04/inside-inrix-how-traffic-data-is-collected-what-it-means-to-your-commute/index.htm` (Accessed on 10.06.2023)

Trépanier, M., Tranchant, N., & Chapleau, R. (2007). Individual trip destination estimation in a transit smart card automated fare collection system. *Journal of Intelligent Transportation Systems: Technology, Planning, and Operations*, *11*(1), 1–14. doi: 10.1080/15472450601122256

Tympakianaki, A., Koutsopoulos, H. N., & Jenelius, E. (2015). c-SPSA: Cluster-wise simultaneous perturbation stochastic approximation algorithm and its application to dynamic origin–destination matrix estimation. *Transportation Research Part C: Emerging Technologies*, *55*, 231-245. doi: 10.1016/j.trc.2015.01.016

Tympakianaki, A., Koutsopoulos, H. N., & Jenelius, E. (2018). Robust SPSA algorithms for dynamic OD matrix estimation. *Procedia Computer Science*, *130*, 57-64. (The 9th International Conference on Ambient Systems, Networks and Technologies

(ANT 2018) / The 8th International Conference on Sustainable Energy Information Technology (SEIT-2018) / Affiliated Workshops) doi: 10.1016/j.procs.2018.04.012

Uber Movement. (2020a). *Movement cities.* `https://movement.uber.com/` and `https://www.npmjs.com/package/movement-data-toolkit`. (Accessed on 02.09.2021)

Uber Movement. (2020b). *Uber movement: Speeds calculation methodology.* `https://movement.uber.com/faqs?lang=en-US`. (Accessed on 02.09.2021)

U.S. Government. (2009, jul). *Open Government Directive.* Retrieved from `https://obamawhitehouse.archives.gov/open/documents/open-government-directive` (Accessed on 11.06.2023)

van der Waerden, P., Borgers, A., & Timmermans, H. (1998, Aug 01). The impact of the parking situation in shopping centres on store choice behaviour. *GeoJournal*, *45*(4), 309-315. doi: 10.1023/A:1006987900394

van Loenen, B., & Grothe, M. (2014). INSPIRE Empowers Re-Use of Public Sector Information. *International Journal of Spatial Data Infrastructures Research*, *9*, 96–106. Retrieved from `https://ijsdir.sadl.kuleuven.be/index.php/ijsdir/article/view/353`

Venthuruthiyil, S. P., & Chunchu, M. (2018). Trajectory reconstruction using locally weighted regression: a new methodology to identify the optimum window size and polynomial order. *Transportmetrica A Transport Science*, *14*(10), 881-900. doi: 10.1080/23249935.2018.1449032

Venthuruthiyil, S. P., & Chunchu, M. (2020). Vehicle path reconstruction using Recursively Ensembled Low-pass filter (RELP) and adaptive tri-cubic kernel smoother. *Transportation Research Part C: Emerging Technologies*, *120*, 102847. doi: 10.1016/j.trc.2020.102847

Vlahogianni, E. I., & Barmpounakis, E. N. (2017). Driving analytics using smartphones: Algorithms, comparisons and challenges. *Transportation Research Part C: Emerging Technologies*, *79*, 196 - 206. doi: 10.1016/j.trc.2017.03.014

Vlahogianni, E. I., Golias, J. C., & Karlaftis, M. G. (2004). Short-term traffic forecasting: Overview of objectives and methods. *Transport Reviews*, *24*(5), 533-557. doi: 10.1080/0144164042000195072

Vlahogianni, E. I., Karlaftis, M. G., & Golias, J. C. (2014). Short-term traffic forecasting: Where we are and where we're going. *Transportation Research Part C: Emerging Technologies*, *43*, 3-19. (Special Issue on Short-term Traffic Flow Forecasting) doi: 10.1016/j.trc.2014.01.005

Wang, H., Calabrese, F., Di Lorenzo, G., & Ratti, C. (2010). Transportation mode inference from anonymized and aggregated mobile phone call detail records. *IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC*, 318–323. doi: 10.1109/ITSC.2010.5625188

Wang, R. Y., & Strong, D. M. (1996). Beyond accuracy: What data quality means to data consumers. *Journal of Management Information Systems*, *12*(4), 5-33. doi: 10.1080/07421222.1996.11518099

Wang, S., Cao, J., & Yu, P. (2019, sep). Deep learning for spatio-temporal data mining: A survey. *IEEE Transactions on Knowledge & Data Engineering*(01), 1-1. doi: 10.1109/TKDE.2020.3025580

Wang, X., Chen, C., Min, Y., He, J., Yang, B., & Zhang, Y. (2018). *Efficient metropolitan traffic prediction based on graph recurrent neural network.* doi: 10.48550/arXiv .1811.00740

Welch, T. F., & Widita, A. (2019). Big data in public transportation: a review of sources and methods. *Transport Reviews*, *39*(6), 795–818. doi: 10.1080/01441647.2019 .1616849

Welle Donker, F., & van Loenen, B. (2016). Sustainable Business Models for Public Sector Open Data Providers. *JeDEM - eJournal of eDemocracy and Open Government*, *8*(1), 28–61. doi: 10.29379/jedem.v8i1.390

Welle Donker, F., & van Loenen, B. (2017). How to assess the success of the open data ecosystem? *International Journal of Digital Earth*, *10*(3), 284–306. doi: 10.1080/17538947.2016.1224938

Westerman, M. (1995, jul). *Probe Vehicle System Concept.* Retrieved from `http://www.wirelesscommunication.nl/reference/chaptr01/roadtrin/ivhsprob.htm` (Accessed on 10.02.2023)

Wikipedia. (2023). *Law of the instrument.* Retrieved from `https://en.wikipedia.org/wiki/Law_of_the_instrument` (Accessed on 04.07.2023)

Willumsen, L. (2021). *Use of Big Data in Transport Modelling* (Tech. Rep.). International Transport Forum. doi: 10.1787/86a128c7-en

Wolf, J., Loechl, M., Thompson, M., & Arce, C. (2003). Trip Rate Analysis in GPS-Enhanced Personal Travel Surveys. In *Transport survey quality and innovation* (pp. 483–498). doi: 10.1108/9781786359551-028

Wood, J. S., & Zhang, S. (2021). Evaluating relationships between perception-reaction times, emergency deceleration rates, and crash outcomes using naturalistic driving data. *Transportation Research Record*, *2675*(1), 0361198120966602. doi: 10.1177/0361198120966602

Wu, T., Chen, F., & Wan, Y. (2018). Graph attention lstm network: A new model for traffic flow forecasting. In *2018 5th International Conference on Information Science and Control Engineering (ICISCE)* (p. 241-245). doi: 10.1109/ICISCE.2018.00058

Wu, X., Lindsey, G., Fisher, D., & Wood, S. A. (2017). Photos, tweets, and trails: Are social media proxies for urban trail use? *Journal of Transport and Land Use*, *10*(1), 789–804. doi: 10.5198/jtlu.2017.943

Wynne-Jones, L. (2019, sep). *Is There a Difference Between Open Data and Public Data?* Retrieved from `https://blog.thinkdataworks.com/open-data-vs-public-data` (Accessed on 03.08.2022)

XGBoost. (2020). *Introduction to boosted trees.* Retrieved from `https://xgboost.readthedocs.io/en/latest/tutorials/model.html` (Accessed on 20.07.2020)

Xu, J., Lin, W., Wang, X., & Shao, Y.-M. (2017). Acceleration and deceleration calibration of operating speed prediction models for two-lane mountain highways. *Journal of Transportation Engineering, Part A: Systems*, *143*(7), 04017024. doi: 10.1061/JTEPBS.0000050

Yabe, T., Sekimoto, Y., Tsubouchi, K., & Ikemoto, S. (2019, 02). Cross-comparative analysis of evacuation behavior after earthquakes using mobile phone data. *PLOS ONE*, *14*(2), 1-12. doi: 10.1371/journal.pone.0211375

Yamamura, E., Tsutsui, Y., Yamane, C., & Yamane, S. (2014). *Effect of major disasters on geographical mobility intentions: The case of the fukushima nuclear accident* (ISER Discussion Paper No. 903). Osaka. Retrieved from `http://hdl.handle.net/10419/127069`

Yang, Y., Yuan, Z., Fu, X., Wang, Y., & Sun, D. (2019). Optimization Model of Taxi Fleet Size Based on GPS Tracking Data. *Sustainability*, *11*(3), 731. doi: 10.3390/su11030731

Young, A., & Verhulst, S. (2016). Understanding the impact of open data. In *The global impact of open data.* O'Reilly Media, Inc.

Zannat, K. E., & Choudhury, C. F. (2019). Emerging Big Data Sources for Public Transport Planning: A Systematic Review on Current State of Art and Future Research Directions. *Journal of the Indian Institute of Science*, *99*(4), 601–619. doi: 10.1007/s41745-019-00125-9

Zhang, Z., He, Q., & Zhu, S. (2017). Potentials of using social media to infer the longitudinal travel behavior: A sequential model-based clustering method. *Transportation Research Part C: Emerging Technologies*, *85*(October), 396–414. doi: 10.1016/j.trc.2017.10.005

Zhang, Z., Li, M., Lin, X., & Wang, Y. (2020). Network-wide traffic flow estimation with insufficient volume detection and crowdsourcing data. *Transportation Research Part C: Emerging Technologies*, *121*, 102870. doi: 10.1016/j.trc.2020.102870

Zhao, L., Song, Y., Zhang, C., Liu, Y., Wang, P., Lin, T., ... Li, H. (2020). T-GCN: A Temporal Graph Convolutional Network for Traffic Prediction. *IEEE Transactions on Intelligent Transportation Systems*, *21*(9), 3848-3858. doi: 10.1109/TITS.2019.2935152

Zhu, W., Chang, A., Jiang, G., & Zhang, W. (2009). Link average speed of traffic flow estimation method based on floating car. In *ICCTP 2009* (p. 1-6). doi: 10.1061/41064(358)229

Ziemke, D., Kaddoura, I., & Nagel, K. (2019). The MATSim Open Berlin Scenario: A multimodal agent-based transport simulation scenario based on synthetic demand modeling and open data. *Procedia Computer Science*, *151*, 870–877. doi: 10.1016/J.PROCS.2019.04.120