TECHNISCHE UNIVERSITÄT MÜNCHEN

FAKULTÄT FÜR INFORMATIK
Lehrstuhl für Datenbanksysteme

Exploratory Knowledge-Mining from Complex Data Contexts in Linear Time

Samuel Joseph Maurus

Vollständiger Abdruck der von der Fakultät für Informatik der Technischen Universität München zur Erlangung des akademischen Grades eines

Doktors der Naturwissenschaften
(Dr. rer. nat.)

genehmigten Dissertation.

Vorsitzender: Univ.-Prof. Dr. Alfons Kemper

Prüfer der Dissertation:
1. Univ.-Prof. Dr. Claudia Plant
Universität Wien, Österreich
2. Univ.-Prof. Dr. Hans-Joachim Bungartz

Die Dissertation wurde am 07.11.2016 bei der Technischen Universität München eingereicht und durch die Fakultät für Informatik am 27.01.2017 angenommen.

Abstract

Automated measurements are being taken in many areas of society. Typically, the scale is large and the structure complex. Even within a single application, data are often collected in various forms (e.g. graph structures, relational tables and multivariate time-series) and over all the fundamental scales of measurement. Despite the heterogeneity, such stores can hold profound insights about patterns found in the domain. This thesis is concerned with *exploratory data mining* – the development of unsupervised and automatic algorithms to extract this knowledge.

Research Approach: Our research is primarily driven by the "top challenges" identified by the data-mining community. These challenges highlight five aspects of practically-observed *complexity* on which we focus: 1) heterogeneous data types and measurement scales, 2) missing information, 3) clutter and noise, 4) high dimensionality, and 5) high-bandwidth time-series data. To help address these concerns, we present novel methods for practical data mining tasks having these complexities. Secondly, driven by the pressing need to develop algorithms that scale efficiently with size of the data, we present *linear-time* algorithms for our problems and discuss their properties. We empirically evaluate each against the state-of-the-art with respect to 1) synthetically-generated data, 2) real-world data, and 3) run-time behavior.

Results: We present problems, frameworks, algorithms and statistical tests to extract knowledge in various forms. We extract latent patterns in the complex context of incomplete heterogeneous measurements using our Ternary Matrix Factorization (TMF) problem and our Matrix Factorizations over Discrete Finite Sets (MDFS) framework. We extract clusters in the complex context of high-dimensional data with "extreme" clutter using our algorithm SkinnyDip. Finally, we extract anomalous system events from the complex context of high-bandwidth time-series data using our approach BenFound.

Contributions: From a *research* perspective, we show how dimensionality-reduction through matrix factorization can be performed over heterogeneous data types and measurement scales, and in doing so complete the theoretical unification of a set of related data-mining techniques. We show how an elegant "mode-hunting" approach can help to cluster data with high dimensionality and extreme levels of clutter. Finally, we show how anomaly-detection can be performed on high-bandwidth time-series data without the need for a parameterized model to describe the underlying process.

From a *practical* perspective, our contributions are fourfold. Firstly, all of our techniques can outperform the state-of-the-art with respect to standard quality metrics. Secondly, each is highly scalable, with aggressive optimization and empirically-demonstrated linear run-time growth in the size of the data. Thirdly, our techniques have no "obscure" parameters and thus contribute to the holy grail of "parameter-free" data mining. Finally, we present numerous examples on real-world data, and provide all prototypes and source code in online, publicly-accessible repositories for direct use. All results are reproducible.

Limitations: None of our proposed algorithms are able to solve optimally in the general case (NP-Hard). Indeed, we are only able to provide an approximation factor for one sub-problem under special conditions. Our algorithms are therefore based on heuristics and their evaluation empirical. A number of additional assumptions and practical limitations exist and are discussed.

Zusammenfassung

In beinahe allen Bereichen der Gesellschaft werden automatisierte Messungen vorgenommen. Typischerweise ist die Datenmenge groß und die Struktur komplex. Sogar innerhalb eines Anwendungsfalls werden die Daten oft in unterschiedlichen Formen (z.B. als Graphstrukturen, relationale Tabellen, multivariate Zeitreihen) und über alle fundamentalen Skalenniveaus hinweg gesammelt. Trotz ihrer Heterogenität können die Datensätze tiefe Einblicke in das Anwendungsgebiet ermöglichen. Diese Dissertation befasst sich mit dem explorativen Datamining, das heißt der Entwicklung von unüberwachten, automatischen Algorithmen zur Extraktion dieses Wissens.

Forschungsansatz: Unsere Forschung wird in erster Linie durch die in der Datamining-Gemeinschaft als „größten Herausforderungen" geltenden Probleme angetrieben. Diese Herausforderungen lenken den Blick auf fünf Aspekte der in der Praxis zu beobachtenden *Komplexität*, auf welche wir uns fokussieren: 1) heterogene Datenarten und Skalenniveaus, 2) fehlende Informationen, 3) Stördaten und Rauschen, 4) hohe Dimensionalität, und 5) Zeitreihendaten mit hoher Bandbreite. Um die Herangehensweise an diese Probleme zu erleichtern, stellen wir neue Methoden für den praktischen Umgang mit Datamining-Aufgaben dieser Komplexität vor. Angetrieben durch den dringenden Bedarf an effizient mit der Datengröße skalierender Methoden, entwickeln wir außerdem lineare Algorithmen für unsere Problemstellungen und diskutieren ihre Eigenschaften. Wir vergleichen die entwickelten Algorithmen empirisch mit wissenschaftlich etablierten Algorithmen hinsichtlich 1) synthetisch generierter Daten, 2) realer Daten, und 3) ihres Laufzeitverhaltens.

Ergebnisse: Wir präsentieren Problemstellungen, Frameworks, Algorithmen und statistische Tests, um Wissen verschiedener Arten zu extrahieren. Wir extrahieren latente Muster im komplexen Kontext unvollständiger heterogener Messungen mittels unserer Ternäre Matrixzerlegung (TMF)-Problemstellung und mittels unseres Frameworks der Matrixzerlegung über diskreten endlichen Mengen (MDFS). Wir extrahieren Cluster im komplexen Kontext hoch-dimensionaler Daten mit „extremen" Stördaten mittels unseres SkinnyDip-Algorithmus. Schließlich extrahieren wir anomale Systemereignisse im komplexen Kontext von Zeitreihendaten mit hoher Bandbreite mittels unseres BenFound-Ansatzes.

Beiträge: Aus Forschungssicht zeigen wir, wie Matrixzerlegung über heterogene Datenarten und Skalenniveaus hinweg durchgeführt werden kann. Wir vervollständigen dadurch die theoretische Vereinigung einer Reihe an verwandten Datamining-Techniken. Wir zeigen, wie ein eleganter „mode-hunting“-Ansatz beim Clustern von Daten mit hoher Dimensionalität und extremem Niveau an Stördaten helfen kann. Schließlich zeigen wir, wie eine Anomalieerkennung bei Zeitreihendaten mit hoher Bandbreite ohne Verwendung eines parametrisierten Models, das den zugrunde liegenden Prozess beschreibt, durchgeführt werden kann.

Aus *praktischer* Sicht zeigen wir empirisch, dass unsere Techniken den „State-of-the-art“ in Bezug auf Standard-Qualitätskriterien übertreffen können. Jede Technik ist außerdem hoch skalierbar und hat keine „verschleierten“ Parameter. Schließlich präsentieren wir zahlreiche auf realen Daten basierende Beispiele und stellen alle Prototypen und den Quellcode in online öffentlich zugänglichen Repositories für den unmittelbaren Gebrauch bereit. Des Weiteren sind alle Ergebnisse reproduzierbar.

Einschränkungen: Keiner der von uns vorgeschlagenen Algorithmen stellt im allgemeinen Fall ein optimales Lösungsverfahren dar. In der Tat können wir nur einen Approximationsfaktor für ein Teilproblem unter speziellen Bedingungen liefern. Unsere Algorithmen basieren somit auf Heuristik und deren empirische Evaluation. Eine Reihe weiterer Annahmen und praktischer Einschränkungen existieren und werden diskutiert.

Acknowledgments

To Professor Claudia Plant, my supervisor. I am indebted to you for your fairness, honesty, flexibility, optimism, motivation, experience and time. Your words were thoughtful and inspiring during our numerous fruitful discussions. It was a pleasure to be a member of your "research start-up" at Helmholtz Zentrum München, attend the premier conferences with you, and to visit and meet your new team for a short research stay in Vienna. You are a successful and relatively young Professor, and I highly respect the spirit and skillfulness you continue to show in the highly-competitive research field of data mining. To Professor Christian Böhm, likewise thank you for all the priceless discussions, manuscript reviews, suggestions, optimism and experienced judgement. I particularly wish to thank you for the passion with which you engaged and "jumped straight in" to many of the thesis ideas.

To PD Dr. Wolfgang zu Castell at Helmholtz Zentrum München. I enjoyed our discussions on a wide range of topics, particularly the challenging research questions that you were pondering in your own projects. I thank you for providing feedback on much of my work from a mathema($x \in \{\mathrm{t}, \mathrm{g}\}$)ician's perspective. Finally, I appreciate your efforts and support at the organizational level of the stack, and particularly for later taking up the role of my Helmholtz thesis-committee supervisor. To Dr. Jens Baumert at Helmholtz: thank you also for being part of my thesis committee, and for allowing me the chance to work with you and your team on one of your diabetes research projects. To Helmholtz Zentrum München in general: thank you for creating a highly-attractive environment in which to complete a PhD (quiet, spacious, green and focused).

To the others with whom I enjoyed my time at Helmholtz over the years: Dr. David Endesfelder, Dr. Marion Engel, Nina Hubig, Annika Tonch, Alexandra Derntl, Sebastian Goebl, Wei Ye, Linfei Zhou, Ruben Seyfried, Kristof Schröder, Hannah Schrenk, Bernhard Tandler, Renate Frieß, Walter Huss, Jürgen Grabow, and numerous interns/students. You all played various positive roles in my development and I thank you. To Professor Alfons Kemper and Professor Hans-Joachim Bungartz at the TUM: many thanks for agreeing to take part in my examination committee. Finally, the last thanks on the "professional" side go to the peers who took the time to respond to my requests for an

implementation of their algorithm(s): Pauli Miettinen, Jilles Vreeken, Radim Belohlavek, Jason Rennie, Greg Hamerly, Junming Shao and Kohei Hayashi.

To Stefanie Ringelhan: thank you for your unwavering care and support through the "ups and downs". You gave me context, taught me to "look over everything first" and take my time. Thank you particularly for the assistance in grasping the German language, and for your particularly motivating comments while I was learning to ski. To Stefanie's parents, Franz and Regine: you are über-generous and always so positive and welcoming. May the Canasta sessions continue for many years to come! To Mark and Verena: you guys are great friends, neighbors and supporters. To your children, Noah and Lea, for the laughs during our hour-long sessions of "Verstecken" and "Pferdespielen"!

To Chopin, Schubert and Mozart: your Nocturnes, Etudes, Impromptus and Sonatas showed me the way to an enchanting artistic escape at the end of each technical day. To my teachers Yamile and Mangfred, pianists studying at the Musikhochschule München: thank you so much for your motivation and guidance as I continue to try and tame these works. To Jon, Ryan, and Derek, my childhood friends from Melbourne: it's great to still be in such close contact after all these years. May our adventures continue for many years to come! To the Geschwister Scholl Wohnheim: you offered me the ideal Munich student accommodation during the majority of my studies, thank you all so much!

To my amazing family in the Allgäu, particularly Ehrentraud and Albert Freudling. Your love and caring is genuine and infectious, and it's always "wunderschön" when we meet! Finally, to my loving family in Australia: our strong bonds render the distances between us insignificant. It's always special when we get together. To my mother Rosemary in particular: I continue to appreciate and be in awe of the extra energy you invested in raising my three sisters and me. Our triumphs are yours as well.

Publication Preface

The contributions of this thesis are based on the following five first-author papers:

A Maurus, S. and Plant, C., 2014, December. Ternary matrix factorization. In Proceedings of the 2014 IEEE International Conference on Data Mining (ICDM '14). IEEE.

Note: This contribution received the single ***Best Paper Award*** *from 727 submissions. The overall acceptance rate for full research-track papers was 9.7%. ICDM is a top-tier data-mining conference.*

B Maurus, S. and Plant, C., 2016, January. Ternary Matrix Factorization: problem definitions and algorithms. Knowledge and Information Systems (KAIS), 46(1). Springer.

Note: This contribution is an extension of the 2014 ICDM paper above. It includes at least 30% new material. KAIS is a leading data-mining journal (2014 impact factor 1.782).

C Maurus, S. and Plant, C., 2016, December. Factorizing Complex Discrete Data "with Finesse". In Proceedings of the 2016 IEEE International Conference on Data Mining (ICDM '16). IEEE.

Note: The acceptance rate for short research-track papers was 11.1% (from 910 submissions).

D Maurus, S. and Plant, C., 2016, August. Skinny-dip: Clustering in a Sea of Noise. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '16). ACM.

Note: KDD is a top-tier data-mining conference. The 2016 acceptance rate for full research-track papers was 8.9% (784 submissions). A poster and video was showcased in addition to an oral presentation.

E Maurus, S. and Plant, C. Let's See your Digits: Anomalous-State Detection using Benford's Law. Submitted to the Research Track of the 2017 SIAM International Conference on Data Mining.

Note: SDM is a top-tier data-mining conference. At the time of writing, this contribution had been submitted for peer review (pending acceptance).

The following paper with major contributions as *second* author was additionally published during the course of the doctoral studies. This publication does not, however, form part of the work at hand.

Ye, W. Maurus, S. Hubig, N and Plant, C., 2016, December. Generalized Independent Subspace Clustering. In Proceedings of the 2016 IEEE International Conference on Data Mining (ICDM '16). IEEE.

Note: The acceptance rate for regular research-track papers was 8.5% (from 910 submissions).

Contents

1 Introduction

1.1 Clarifying the Buzzwords: *Data Mining, Knowledge Discovery in Databases* and *Data Science*

The terms "data mining", "knowledge discovery in databases" (KDD) and "data science" are often used loosely, so it is useful to begin with some clarifications.

We follow the definitions as given in [FPS96]. Specifically, we understand "data mining"[1] as being the application of *specific algorithms* to prepared data for the purpose of either *prediction* or *description*. *Prediction* involves finding patterns that can assist in forecasting the behavior of a phenomenon (or some entities). *Description* involves finding useful explanatory patterns that can be presented to a user in a digestible, understandable form. Of course, the boundaries between these types need not be sharp. For example, a predictive data-mining algorithm that yields a decision-tree with readable branching rules is usually more descriptive and interpretable than, say, a feed-forward artificial neural network containing a cryptic set of numeric weights.

In this thesis we are primarily interested in *descriptive* data mining, that is, the extraction of interpretable and digestible patterns in a given data set. More specifically, this thesis focuses on learning these patterns in an *unsupervised* or *exploratory* way. We assume no a priori "labeled" data. This implies no training phase, in which a system might "learn" about the domain from a representative set of such labeled instances. The knowledge-discovery processes on which we focus are also not *hypothesis driven*, as is often the case in classical statistical analysis. In short, we focus on variants of the fundamental unsupervised and exploratory data-mining problems: *finding associations*, *clustering objects* and *detecting anomalies*.

The term "knowledge discovery in databases" refers to the high-level workflow that transforms raw data into proven domain insights. Data mining is a single step in this workflow. The other steps include selection (of a data subset for analysis), preprocessing, transformation and interpretation/evaluation. The broad workflow is depicted in Figure 1.1. Again, it is important to realize that "data mining" is one of many steps in the KDD process. As the primary contributions of this thesis are novel data-mining methods, we will usually only treat the other tasks of the KDD workflow to the extent necessary for

[1] We use the terms "data mining" and "knowledge mining" interchangeably.

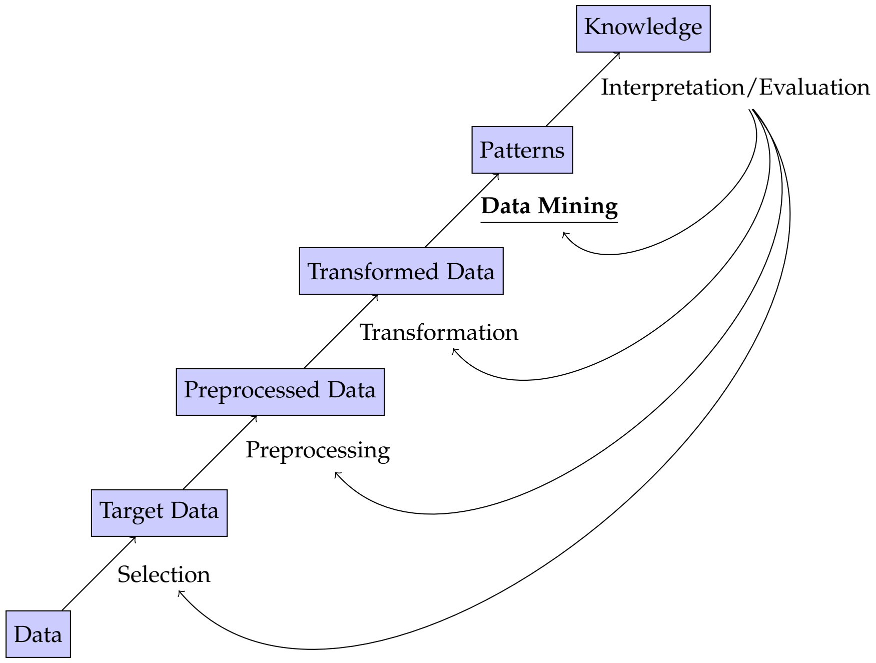


Figure 1.1: The *Knowledge Discovery in Databases* workflow, inspired by Figure 1 in [FPS96].

demonstrating our ideas.

Finally, the term “data science” is typically understood to be more abstract again. It is often defined as the field concerned with processes and systems which extract knowledge or insights from data in various forms. Many techniques from machine learning and classical fields such as statistics, mathematics and signal-processing fit this definition. Indeed, the term “data science” is often argued to be a “buzzword” for statistics. The interested reader can find a detailed discussion between the concepts of “data science” and “statistics” in [Dha13].

In summary, this thesis makes methodological contributions to the field of *data mining*, so to avoid confusion we will mostly refrain from using the terms “data science” and “knowledge discovery in databases”.

1.2 The Role of Exploratory Data Mining in Society

Data mining techniques play an increasingly important role in society [KB11; Kri+07; BY09]. In (e-)commerce, data mining is used as a basis for many purposes, including the generation of cross-selling recommendations [AIS93] and for summarizing customer sentiments based on large numbers of reviews [HL04]. In healthcare, applications range from the detection of health-insurance claims fraud to the evaluation of treatment effectiveness [K+11]. In science and engineering, data mining finds applications in bioinformatics [Wan+05], genetics [Kan+02], medicine [CM02], education [SVM06] and electrical power engineering [McG+02]. In online social networks, data mining is used for many tasks including link-prediction [LK07], community detection [TL10], fraud detection [YWB11] and spam detection [Ben+10].

The contributions of this thesis are application-agnostic, but we do note a number of common properties from the list just mentioned. In many of these applications, the *Four V's* of big data [Buh+13; SS13] are evident. *Volume* refers to the scale of the data, now larger than terabytes and petabytes, which outstrips traditional store and analysis techniques. *Velocity* refers to the bandwidth at which data is being streamed (e.g. 1TB of trade information is collected during each trading session on the New York Stock Exchange). *Variety* refers to the different forms of data, including measurements that are made over fundamentally different scales. Finally, *veracity* refers to the uncertainty and poor quality of data. Humans cannot be expected to manually analyze big data. "Hence, KDD is an attempt to address a problem that the digital information era made a fact of life for all of us: data overload" [FPS96, p. 38]. In this thesis, the *scalability* of our proposed methods to such big data applications in society is a key consideration.

1.3 Current Challenges in Data Mining

As illustrated by the DBLP[2] metrics in Figure 1.2, many fundamental problems in the field of data mining remain in an active state of research.

Take cluster analysis, for example. One commonly-found definition states that *Clustering is the task of partitioning a set of objects such that objects in the same group are more "similar" to each other than those in other groups.* With such an innocent-looking definition, why does this task continue to attract such a growing number of academic publications? That is, why is it so *challenging*? Why haven't we solved it yet?

One answer is that it is difficult to find a more precise definition of the clustering problem that is universally valid. The notion and measurement of "similarity", for example, may vary by application, as may the mechanism for evaluating a candidate

[2] dblp.uni-trier.de, a computer-science bibliography.

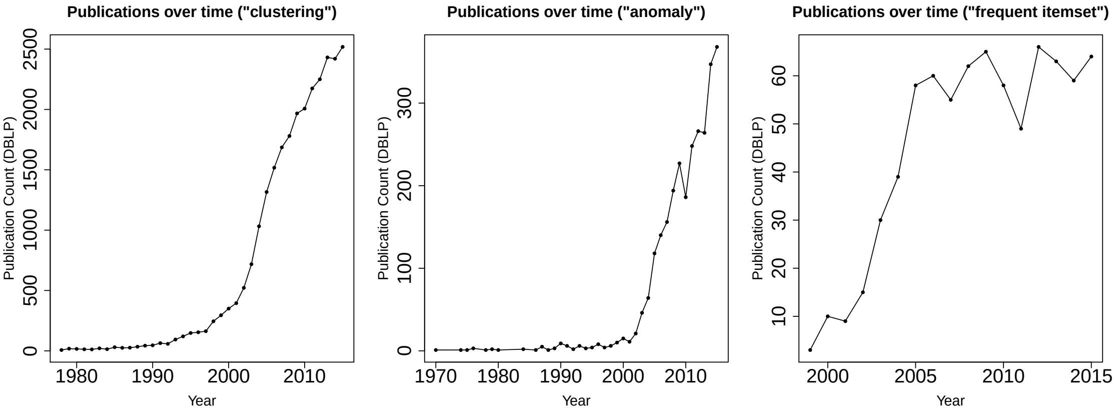


Figure 1.2: Counts of publications over time that include "clustering", "anomaly" or "frequent itemset" in the title respectively (*source: DBLP*).

solution. Other questions soon become evident: *What about the notion of "noise" and "outliers"? Must the partitioning be strict? How do we select the number of clusters?*

To arrive at a well-defined problem for which an algorithm can be designed and deployed in practical situations, we must make assumptions that help us to answer such questions. The validity of those assumptions in turn depends on the nature of the application in question. Different arguments can be made for favoring different assumptions, thus increasing the possibilities for developing more specialized clustering techniques.

Other data mining problems are analogously difficult (anomaly detection, association-rule mining, graph mining). This leads us to perhaps the top challenge in modern data mining, namely the development of a *unified* theory of data mining. In the following subsection we discuss this challenge in more detail. In the subsequent subsections we list a further three "top" challenges on which we focus. Taken together, these four challenges correspond to the top four open challenges for data mining as enumerated by Yang and Wu [YW06].

1.3.1 Challenge 1: Developing a Unified Theory of Data Mining

This challenge was identified as "numero uno" in Yang and Wu's highly-cited work on data-mining challenges [YW06]. Specifically, they note that the current state of data-mining research is often criticized as being too "ad-hoc". Indeed, numerous techniques have been designed for individual problems. A theoretical framework that unifies the various data-mining tasks could therefore be considered as a "holy grail" of research in this field.

Needless to say, this is an ambitious goal that may prove very difficult, if not impossible, to attain in practice. Regardless, we can remain optimistic and be inspired by the various elements of what such a direction would entail. For example, we can take a step closer to this goal if we keep in mind the motto: *induce, deduce and reduce*. This is the motto particularly embraced in **Papers A, B and C** of this thesis. Specifically, we should:

- ***Induce* general frameworks for data-mining problems based on similarities found in *specialized* techniques.** Ideally, all possible instances of the *specialized* problems would be provably *reducible* to an instance of the *framework* problem. If the algorithms for the new framework additionally *outperformed* their specialized counterparts with respect to both efficiency *and* effectiveness, then researchers and practitioners alike would welcome the retirement of a further set of redundant tools from the overwhelming population of data-mining algorithms. Additionally, researchers would be inspired to *deduce* additional useful applications from the framework. Problems from these applications could then be solved by the same algorithm.

- ***Reduce* the number of assumptions made by state-of-the-art data-mining approaches.** It would be naïve to think that we can design a useful data-mining algorithm that is void of assumptions. However, we should take the initiative to review the need for some of the most fundamental assumptions made in data mining, and challenge ourselves to relax them when appropriate. For example, for a given relational data set, many matrix-factorization techniques make the implicit assumption that all features are measured over the same scale (typically the ratio scale). In the area of non-hierarchical, vector-based cluster analysis, *all* techniques known to this author assume a particular multivariate "distance" or "similarity" measure on the space (e.g. Euclidean in k-means, or the Gaussian kernel in spectral clustering). We should exercise our scientific curiosity by questioning these basic assumptions.

- ***Reduce* the need for obscure parameters and excessive "tuning".** Algorithm parameters can be a gift and a curse. When a parameter relates to a concept that a practitioner can readily comprehend (e.g. the number of desired clusters k), its variation can yield a set of potentially-useful results on the same data. At the other extreme, it can be a daunting task to set a parameter that is only used internally, has no obvious relationship to the result and is infinitely variable (we will see that the τ parameter required by the Asso algorithm for Boolean Matrix Factorization is such a parameter). We should take care to design algorithms that reduce the requirement for parameters in this latter category.

1.3.2 Challenge 2: Scaling Up for High-Dimensional Data and High-Speed Data Streams

Challenge number two on Yang and Wu's list is first and foremost related to the **curse of dimensionality** [ZSK12; IM98; BC57; KKZ09]. We understand this term as referring to a number of difficulties encountered when analyzing data in high-dimensional spaces. One difficulty could be termed the "distance concentration effect", and refers to the fact that, as the volume of the space increases with increasing dimensionality (making the available data more sparse), there tends to be little difference in the "similarity" between different pairs of objects [KKZ09]. Coupled with the fact that the presence of *irrelevant* features may conceal *relevant* information, the **effectiveness** of many data-mining algorithms fails to scale to high-dimensional data.

In addition to appropriately filtering attributes (e.g. those with zero entropy) in the preparation phase of the KDD workflow, data-mining algorithms should consider mechanisms through which the curse of dimensionality can be tamed. In clustering, for example, it is seldom useful to apply a full-space clustering method to a data set with a moderate-to-large number of dimensions (e.g. 10 or above) [KKZ09]. An algorithm designed to search for the most useful parts of the data, generally in the form of a low-dimensional subspace, can yield better results. **Paper D** makes contributions in this direction.

Yang and Wu also use the words "high-speed" in their naming of this challenge, which implies an additional need to focus on **efficiency**. It is an unfortunate truth that concrete formulations of many data mining problems imply that finding their *optimal* solution is not computationally tractable. Even k-means, which takes a rather simplified view of the abstract clustering problem, is provably NP-Hard [SI84]. Assuming that computer science will not be blessed any time soon with a favorable result akin to $P = NP$, we must resort to heuristics in order to enable the analysis of any data having non-trivial size.

Of course, the use of a heuristic alone does not automatically classify an algorithm as "scalable". Many heuristic-based data mining techniques have super-linear time complexity in the size of the data set. For example, we will see that a number of non-parametric anomaly- and change-point-detection algorithms have quadratic time complexity in the number n of data objects. For high-speed data streams like Twitter, where over 6000 new objects ("Tweets") arrive every second (see **Paper E**), super-linear time complexity may well be prohibitive.

To help address Challenge 2, we will subscribe to some further "guiding principles" when designing our data-mining algorithms:

- Consider the curse of dimensionality, and work to include mechanisms for mitigating its effects.

- Strive for algorithms that have a practically *linear* run-time complexity in the size of the data.
- Use algorithmic paradigms that lend themselves well to parallelization on high-performance computing infrastructure.

1.3.3 Challenge 3: Mining Time-Series Data

Temporal data with trends, seasonality and noise are commonplace in modern information systems [LAF15]. Particularly in the domains of intrusion detection, credit-card fraud, medical diagnoses and law enforcement, there is often a need to raise a "red flag" when the real-time data deviates from the expected distribution or patterns. The detection of the related change-points, anomalies and "events" in time-series data is an important element for modern information systems with large volumes of traffic and strict "uptime" requirements. At Yahoo!, for example, it is critical for the integrity of the business to perform real-time monitoring of millions of production-system metrics [LAF15].

Along with Challenge 2, Challenge 3 on Yang and Wu's list relates to the increasing requirement for monitoring *live* data for patterns in a *real-time* manner. For applications like Yahoo!, it is not sufficient to periodically schedule offline processing alone.

Interestingly, many time-series mining techniques consider measures relating to the distribution of the *absolute values* of the metrics in question. In tune with our *reduce* motto from Section 1.3.1, it can sometimes be useful to question basic assumptions like these. More precisely, we ask: Is there a lens through which we can view time-series data that focuses specifically on what is "natural" and "unnatural"? This curious question will be investigated in **Paper E**.

1.3.4 Challenge 4: Mining Complex Knowledge from Complex Data

Yu and Wang's fourth problem relates to handling "complex" data. They note that graphs are one form of complex data that has become especially prevalent in social networks, and that more research needs to be performed in this direction. **Paper C** considers a certain kind of graph structure in this light.

Another form of complexity is found in relational data that has heterogeneous features measured over fundamentally different scales. This heterogeneity complicates matters because it rules out the application of techniques that assume completely real-valued measurements, or completely categorical measurements, for example. More research needs to be made into methods that can mine patterns from data sets containing measurements from a variety of the different fundamental scales (nominal, ordinal, interval, ratio). This is the topic of **Papers A, B and C**.

A further form of complexity relates to the amount by which the *signal* in the data is hidden by *noise*. The real world is noisy to varying extents. Noise is usually unstructured and contributes little to understanding the patterns or associations between the variables and phenomena in a domain. In the context of noise, data-mining techniques should remain robust. For example, a clustering technique that assigns cluster membership to large volumes of clutter or noise points is not particularly useful. Complexity in the form of clutter and noise is thus a key consideration in our work (particularly **Paper D**).

Finally, Yang and Wu note that we must make sure that we pay attention to the "interestingness" and interpretability of the patterns that we mine. In line with the definition of *descriptive* data mining (Section 1.1), a user must be able comprehend the meaning of the discovered patterns in the context of their domain. This includes making sure that the set of patterns presented to the end-user has a digestible cardinality (e.g. "top 10"), and that each pattern contains information that can be directly translated to real-world domain concepts. If a data-mining algorithm learns a representation of a system, but that representation is in turn difficult to comprehend, then the algorithm's usefulness is limited. Delivering interpretable patterns is thus a key consideration for our work (particularly in **Papers A, B and C**).

1.4 Goals of this Thesis

To summarize the challenges from the previous section, this thesis focuses on making data-mining contributions that address a number of aspects of complexity. Each helps to advance the state-of-the-art in data mining. Specifically, we aim to develop problems, frameworks, algorithms and statistical tests that

1. can subsume a number of existing approaches without the requirement for additional obscure parameters (**Challenge 1**),

2. can support input data with high dimensionality whilst scaling linearly in time (**Challenge 2**),

3. can be deployed in real-time for high-bandwidth time-series data (**Challenge 3**),

4. can yield interpretable results on heterogeneous data sets containing measurements from a number of scales and high levels of noise (**Challenge 4**), and

5. are demonstrably superior to the state-of-the-art in controlled and real-world settings.

1.5 Remarks on the Document Structure

This is a publication-based dissertation. Each individual publication is embedded as an appendix, accompanied by a short introduction concerning the topic, publication outlet, acceptance status, re-use license and author contributions.

The remainder of this dissertation is organized as follows. Chapter 2 reviews a number of the basic technical results and problems on which this work builds. This includes a brief review of the fundamental scales of measurement, a review of the basic algorithms for Blind-Source Separation and a review of the main paradigms for cluster analysis. Chapter 3 presents a review of the literature that is directly relevant to the techniques proposed in this thesis, including Boolean and Ordinal Matrix Factorization, robust clustering algorithms for high-clutter data contexts, and anomaly- and change-point-detection algorithms. Chapter 4 discusses our research approach. Chapter 5 states and discusses the results of our work and highlights their impact on research and industry. Limitations of our work, as well as directions for future research, are likewise covered in Chapter 5. Chapter 6 gives concluding remarks.

2 Preliminaries

2.1 The Fundamental Scales of Measurement

In this thesis, a number of contributions relate to knowledge-discovery from data that has been collected over heterogeneous scales of measurement. For this reason, and despite the risk of triviality, a short review of the fundamentals of these scales is warranted.

S. S. Stevens notes in his seminal 1946 article that “measurement exists in a variety of forms” and that “scales of measurement fall into certain definite classes” [Ste46, p. 677]. His proposed typology has since become the most widely-adopted classification for levels of measurement in the natural sciences. Importantly, the different scales exhibit different properties and permit different mathematical operations. The following subsections briefly review the necessary preliminaries for each type of measurement scale.

Ratio Scales

Physical quantities are often measured on a ratio scale. For example, measurements of electric charge, rates of flow, distance, temperature (in Kelvin) and mass are all done on the ratio scale. What do they all have in common?

Firstly, each scale has a meaningful *zero* point. A mass of zero corresponds to the absence of matter. A distance of zero corresponds to the separation of a location in space from itself. A temperature of zero (Kelvin) corresponds to the minimum possible thermal motion of atoms and molecules, and so on. This meaningful zero point is valuable because it allows us to make sensible statements about the *ratio* between two measurements on that scale. For example, it is reasonable to state that “20K is twice as hot as 10K”, or “200m is twice as long as 100m”.

We can also reasonably make statements about the “difference” between two measurements made on a ratio scale. “The difference between a mass of 10g and a mass of 6g is 4g” is a sensible statement, for example.

Said differently, the ratio scale is the scale with the greatest amount of “metadata”. Of the four fundamental scales of measurement, it permits the greatest number of mathematical operations that can be sensibly applied. These operations include numerical addition and multiplication.

Interval Scales

Many scales used in daily life are measured on an interval scale. We find two examples in temperature (Celsius) and time (measured after the AD 0 epoch). Compared to the examples from the previous section (ratio scale), we recognize that the zero point on these scales is arbitrary. That is, it is arbitrary to select water's freezing point at atmospheric pressure as the zero point for temperature, and arbitrary to select the nativity date as the zero point for time (indeed, many civilizations have done otherwise).

For this reason, it makes little sense to talk about ratios between two interval-scale measurements. It is not useful to assert that "40 degrees Celsius is twice as hot as 20 degrees Celcius", nor that "the year 2016 is half as old as the year 4032". We should therefore refrain from using numerical multiplication on such measurements.

We can, however, still talk about degrees of difference between values measured on interval scales. It is sensible to say that the difference in temperature between 40 and 30 degrees Celcius is the same as the difference in temperature between 20 and 10 degrees Celcius.

Ordinal Scales

Yet more restrictive is an ordinal scale, perhaps most often used for measurements made in survey research. In this thesis, for example, we work in part with data from survey research questionnaires that are designed for measuring opinions (**Paper C**). Items in such surveys involve a statement and a dichotomy (e.g. "Disagree" or "Agree"), and the participant is tasked to select a value[1] from a scale imposed over that dichotomy (commonly called Likert items). A typical five-level Likert item might be 1) Strongly disagree, 2) Disagree, 3) Neither agree nor disagree, 4) Agree, and 5) Strongly agree.

In contrast to interval scales, we cannot sensibly talk about precise degrees of difference on an ordinal scale. That is, it makes little sense to propose that the difference between "Strongly agree" and "Agree" is exactly the same as the difference between "Agree" and "Neither agree nor disagree". Therefore, even if these scales have numerical labels, we should refrain from using numerical operations such as addition and multiplication. Ordinal scales do, however, impose a rank ordering of their elements.

Nominal (Categorical) Scales

The label for this scale derives from the Latin root *nom*, meaning "name". A scale is hence termed "nominal" if it involves differentiating between items based only on a

[1] For simplicity here we ignore the case where the participant is permitted to leave the answer blank, or select the "Don't know" option.

system of qualitative classification or categorization. Measuring the *religion* of a human being is an example of a measurement done over a nominal scale. Such a scale would likely include the set of labels $\{\text{Christianity}, \text{Islam}, \text{Judaism}\}$.

It makes little sense to "mix" or "add" two religions, compute a quantitative "difference" between them, or impose an objective "ordering". For these reasons, neither addition nor multiplication may be performed on nominal measurements. Assigning a numerical value to each nominal category for purposes of identification should be done with caution; it can lead to a false belief that the measurements may be open to the same interpretation as given to one of the more powerful scales from the previous sections. In general, the only permissible statement regarding the relationship between two measurements made on a nominal scale is that of equality.

A curious observation, and one that is particularly relevant for this thesis, is that the values from two- and three-valued logic (Boolean and Ternary logic) are, by this definition, measured over a *nominal* scale (or a trivial dichotomous ordinal scale). The Boolean values of *false* and *true* are often given alternative labels like *no* and *yes*, *failure* and *success*, and so on. Perhaps the most common labels, however, are 0 and 1. This preference is most likely due to the benefits of compactness (a single character in each case), however we will see in this thesis that this representation can, in the context of data-mining, have disadvantages (Section 3.1).

2.2 Blind Source Separation, Latent Patterns, and Finite Mixtures

Blind Source Separation (BSS) is the abstract task of extracting a set of *source* signals from a set of *mixed* observations [YHX14]. With this definition, a number of concrete techniques fall under the BSS banner. We explicitly note that we do not use the BSS term as a synonym for Independent Component Analysis (as is done in some technical communities). For our purposes, "sources" can be understood as the "patterns" we wish to find, and the common property held by each BSS technique is that observations are formed through the *mixture* of sources (patterns). In the definition of BSS as we consider it, no information is given about the form of the source signals or the mixing mechanism, so the problem is highly underdetermined.

Papers A, B and C of this thesis focus on the problem of BSS for *non*-ratio-scale data, so it is useful to briefly review the fundamental concepts and commonly-used BSS techniques for data that *has* been measured on the ratio-scale.

2.2.1 Independent Component Analysis

Various techniques exist to solve special cases of BSS. Each makes different assumptions regarding the nature of the source signals. The “Cocktail Party effect” is a frequently-used example for motivating the study of BSS in signal processing [Bro00]. It refers to the well-known phenomenon of being able to target one’s attention to a single auditory stimulus at a party (e.g. the monologue of the partner), despite the presence of numerous other significant auditory stimuli (music, speeches, other conversations). The sources at the party are the various stimuli in the room, including musical instruments and human speakers. A given observer (e.g. a microphone, or a human ear) measures sound resulting from the weighted numerical sum (mixture, or “superposition”) of these signals, where the weighting is influenced by factors such as each source’s intensity and the distances of the sources from the observer. Importantly, computing a weighted sum implies “weighting” and “summation”, which can only be done sensibly if the corresponding operators (“multiplication” and “addition”) exist for the associated level of measurement. In the Cocktail Party example, sound intensity is a physical quantity measured on the ratio scale, so a finite linear mixture of audio signals is a sensible concept.

With the Cocktail Party application in mind, Figure 2.1 shows a simple application of Independent Component Analysis (ICA), which assumes that the source signals are statistically independent from each other and non-Gaussian. The concrete mechanism by which statistical independence is measured can vary, which leads to different forms of ICA. Often, an information-theoretical approach is taken which seeks to minimize the mutual information between the sources. Other approaches are also possible, such as maximum likelihood estimation, maximization of non-Gaussianity, and tensorial methods [HKO04].

ICA has numerous other applications. The human brain often emits signals from different regions, which are typically understood to have been “mixed” in a linear way when measuring brain activity using sensors attached to the outside of the head. It is often medically sensible to assume that the sources behave independently, so an ICA can help to uncover the signals emitted by the different brain regions. In econometrics, performing an ICA on parallel time series can help to decompose them into independent components in order to gain an insight into the driving mechanisms of the data. For the task of feature-extraction in image processing, ICA can help to find features that are as independent as possible [HKO04].

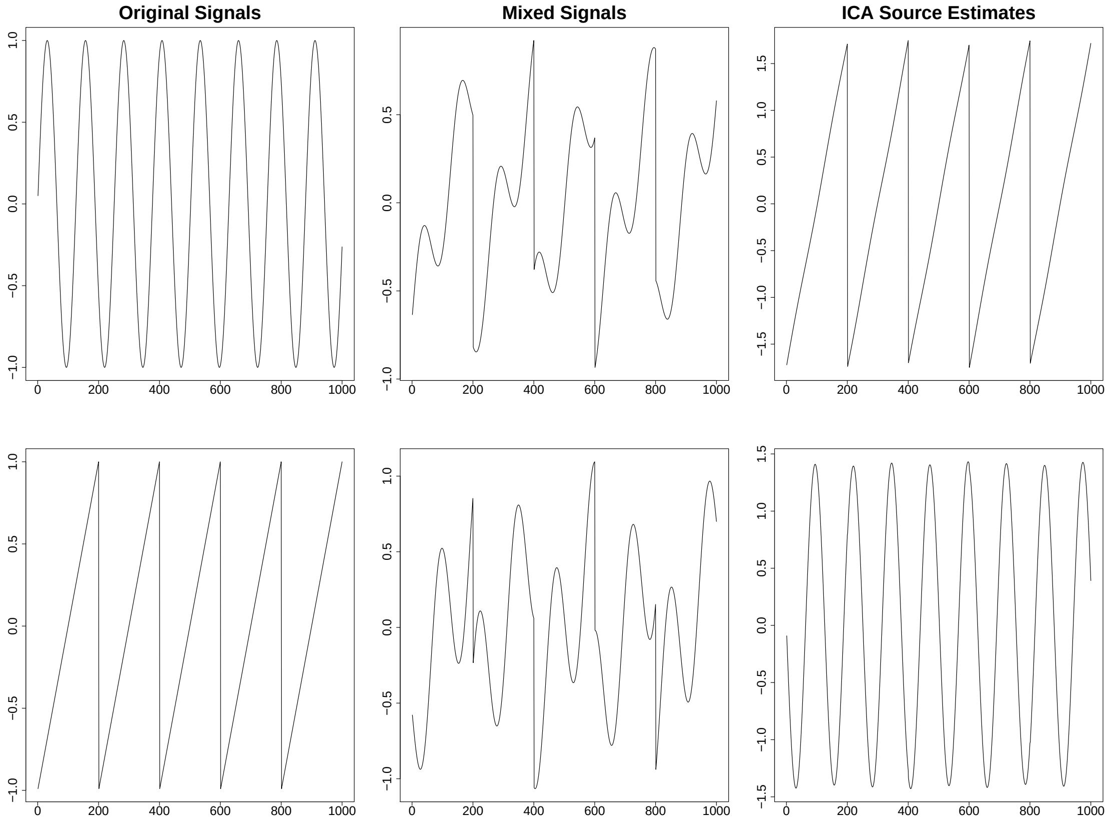


Figure 2.1: An example of Blind-Source Separation using Independent Component Analysis (using the R package *fastICA* [MHR10]). Note the ambiguities: ICA is generally not able to reconstruct the amplitude and the sign of the original signals.

2.2.2 Principal Component Analysis (via the Singular Value Decomposition)

In Figure 2.2 we see an example of Principal Component Analysis (PCA), which can be understood as another kind of BSS technique (based on our definition). In this case (spatial data) a PCA helps us to identify the directions in the data that are most responsible for the variance. Usually, taking just a small subset of these directions gives us the "primary sources" for explaining much of the variability in the data. For visualization tasks, projecting the original data onto these directions can give a digestible (in terms of being low-dimensional and comprehensible by a human) and highly-informative (in terms of being able to visualize the "spread" or variance) view.

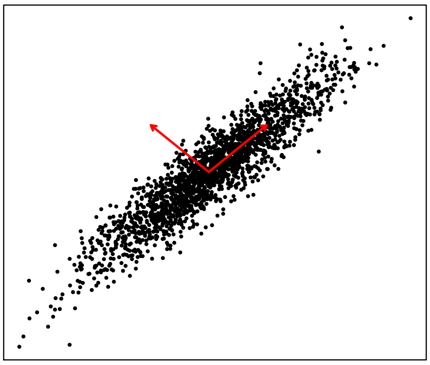


Figure 2.2: PCA finds orthogonal directions in the data, ordered such that each successive direction explains the maximum possible remaining variance.

2.2.3 Non-negative Matrix Factorization

The title of the seminal article on Non-negative Matrix Factorization (NMF) is "Learning the parts of objects by non-negative matrix factorization" [LS99]. Indeed, NMF is a technique that aims to find a *parts-based*, not holistic-based, representation of objects in a data set. Given a data matrix $D \in \mathbb{R}^{n \times m}$ where each of the n rows represents an object with m features, the form of the decomposition is:

$$D \approx H \cdot W. \tag{2.1}$$

The matrix $W \in \mathbb{R}^{k \times m}$ is often named the *basis* matrix because it contains the k most fundamental "parts". The parts are mixed together in various ways to form each observed object in D. The recipe according to which the parts are combined is prescribed by the corresponding row in the "encoding" or "usage" matrix $H \in \mathbb{R}^{n \times k}$. The mixing mechanism is linear and respects the classical matrix product:

$$d_{ij} \approx \sum_{a=1 \ldots k} h_{ia} w_{aj}. \tag{2.2}$$

Performing Blind-Source Separation on *faces* data is often used to intuitively explain the difference between PCA and NMF. Consider the top 25 basis vectors found by two decompositions of the CBCL faces data [HPP00] in Figure 2.3. The basis vectors on the

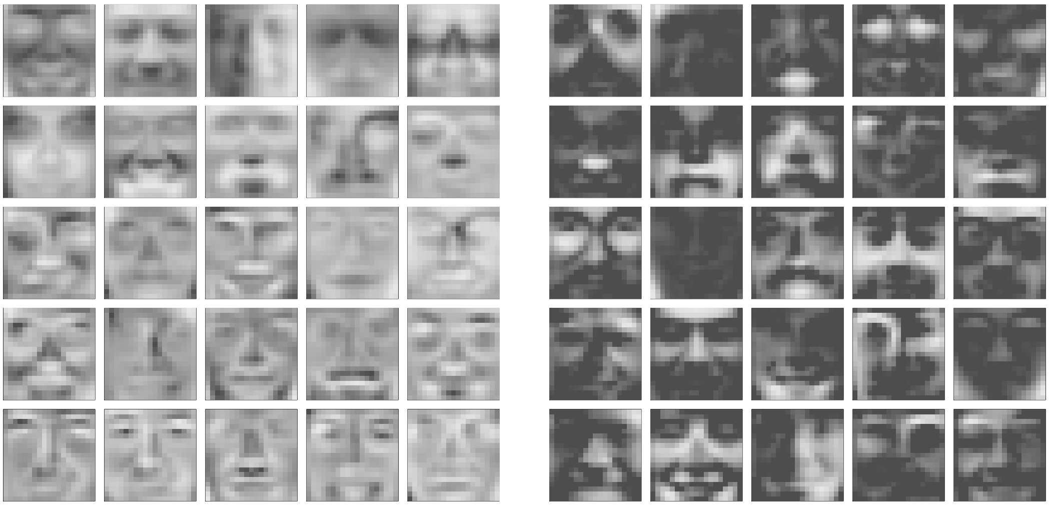


Figure 2.3: Top 25 basis vectors found by performing PCA (left) and NMF (right) on the CBCL faces data [HPP00].

left are found by PCA; those on the right by NMF. This result is a neat visual aid for comprehending the objective of NMF. That is, the NMF basis vectors are more in line with a *parts-based* representation of faces. For example, we can see NMF basis vectors that focus only on the eyes (row 1, column 4), the nose (row 2, column 1), the chin (row 1, column 3) and the cheeks (row 2, column 2). The PCA "eigenfaces" are more holistic in comparison.

Why do approaches like Principal Component Analysis not enable such a parts-based representation? The answer lies in the constraints that NMF enforces on the matrices H and W. As the name suggests, neither H nor W is allowed to contain negative entries. This implies that only *additive* sources and mixtures are allowed. Representations learned by approaches like PCA generally involve cancellations between positive and negative numbers. This complex mechanism lacks an intuitive meaning in such a "faces" example. By forbidding subtractions, NMF is compatible with the intuitive notion of "combining parts to form a whole".

2.3 Cluster Analysis

Cluster analysis involves the abstract task of partitioning a set of objects into groups, or "clusters", such that objects in any given group are more "similar" to one another than they are to objects in other groups.

As discussed in Section 1.3, this definition of "clustering" is a high-level one that requires concrete elaboration. No “silver bullet” technique exists, because the interpretation of the problem typically depends on the application in question. In this thesis, we restrict our focus to *non-hierarchical* clustering of vector data. Figure 2.4 illustrates four different sets of 2D vector-data in which the concept of a “grouping” can differ.

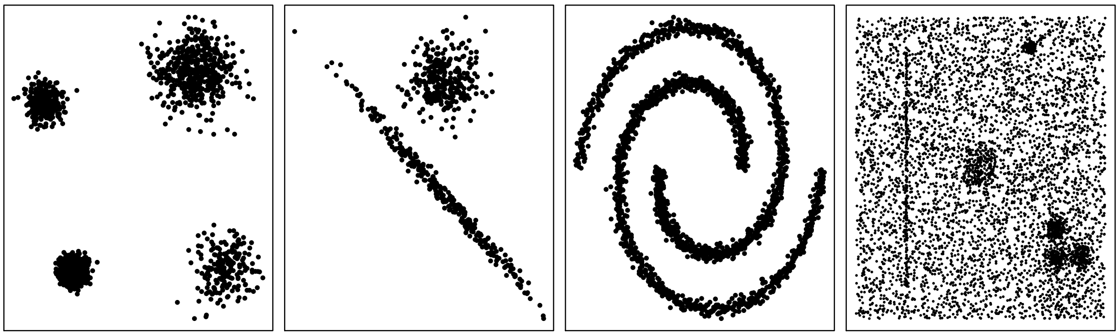


Figure 2.4: Four different two-dimensional spatial data sets.

2.3.1 Partition-Based Clustering

On the left of Figure 2.4 we find perhaps the simplest of the four data sets. This data set was generated by sampling over four bivariate Gaussian distributions (each with a different mean and standard deviation). Although the groups have different sizes, they have comparable spatial extent and are well separated (in the Euclidean sense of the word).

It is clear to the naked eye that it would be useful in this case to group (or “partition”) objects based on their absolute position in space. That is, we could assign a “prototype” vector for each group, and assign each object’s cluster membership based on its closest prototype (using the Euclidean distance). This is one partitioning approach, also known as *vector quantization*. The prototype vectors might also be called “centroids”, hence a further name for this kind of clustering paradigm. In Figure 2.5 we see an example of the popular k-means partition-based clustering algorithm [Mac+67] applied to the data on the left of Figure 2.4.

Vanilla k-means requires that we specify the number k of prototypes (clusters). In general this may not be known a priori. Importantly for the work in this thesis, centroid-based partitioning techniques like k-means are typically also deficient of a *noise* concept (**Challenge 4**). That is, centroid-based partitioning techniques often assume that each object in the data set has a sensible membership in exactly one cluster.

The second data set in Figure 2.4 is not the best fit to the centroid-based partitioning model because the spatial extents of the two clusters vary considerably. We observe in

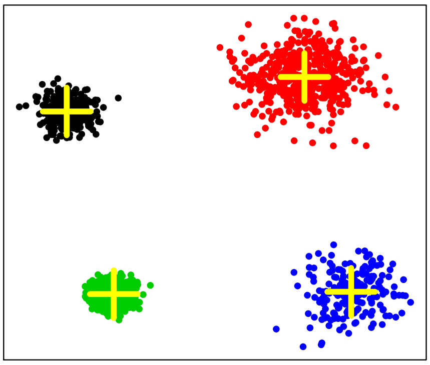


Figure 2.5: An example k-means result. The locations of the cluster prototypes (also known as "centers" or "centroids") are given by the yellow + markers.

Figure 2.6 that this complicates matters somewhat for k-means, so we look for a different approach. One solution is to model each of the k clusters using a statistical distribution. One might call such an approach a "distribution-based" clustering paradigm. For our data we observe that each cluster can be well approximated by a Gaussian distribution (with different parameters). In particular, the top cluster could be described with a scalar variance in each axis direction, and the stretched elliptical cluster on the bottom with an appropriate 2×2 covariance matrix.

Each data object is then assigned a vector of length k, the elements of which represent the membership probabilities for each of the k cluster distributions. This implies a *soft* clustering (no concrete membership assignments), although a hard assignment can trivially be achieved by assigning each object to its most likely cluster.

This approach is clearly parametric. The user is required to specify the number of clusters k and the type of distribution in advance. To practically solve the problem, the popular Expectation-Maximization (EM) [DLR77] algorithm is often deployed. The distribution form of the clusters is typically selected as Gaussian. EM iterates to achieve the (local) maximum likelihood parameters of these distributions.

Although classical distribution-based clustering techniques have no explicit notion of noise, one might interpret an object with consistently "low" membership probabilities as a noise object.

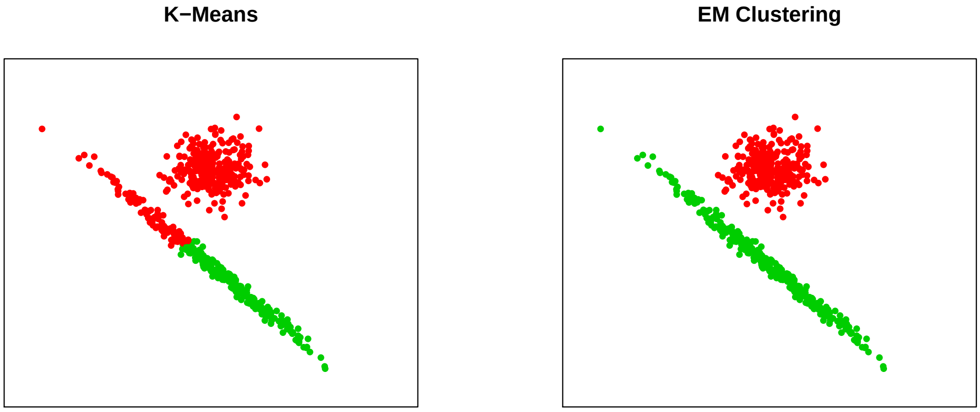


Figure 2.6: *k*-Means (left) and hard-EM (right) clustering algorithms applied to a common data set. EM models each cluster using a bivariate Gaussian with different distribution parameters.

2.3.2 Density- and Spectral-Based Clustering

A rather different clustering paradigm is to consider the localized object density of points, recognizing that such a measure is considerably higher *inside* a cluster than it is *outside* a cluster. The most highly-cited density-based clustering method is DBSCAN [Est+96]. It formalizes this notion using localized concepts like the number of points within a given point's spatial "neighborhood", and whether or not two points can be considered to be "reachable" and "connected" to each other through other local points.

Density-based approaches have a number of advantages. Firstly, operating at a local level, they do not require the user to specify the number of clusters to find. Secondly, they are not restricted to convex-shaped clusters: the notions of "reachability" and "connectivity" enable clusters to grow "naturally" in the direction of all points that satisfy the propagation conditions. Thirdly, they have a clear concept of noise: any point that does not meet the conditions for inclusion in a cluster is labeled as noise. In Figure 2.7 we see an example DBSCAN result on a noisy, synthetic data set with non-convex cluster shapes.

Despite its numerous advantages, DBSCAN its no silver bullet. DBSCAN is not a linear-time algorithm (**Challenge 2**), still requires some measure of distance on the space (the Euclidean distance is used in [Est+96]), needs the user to specify thresholds for

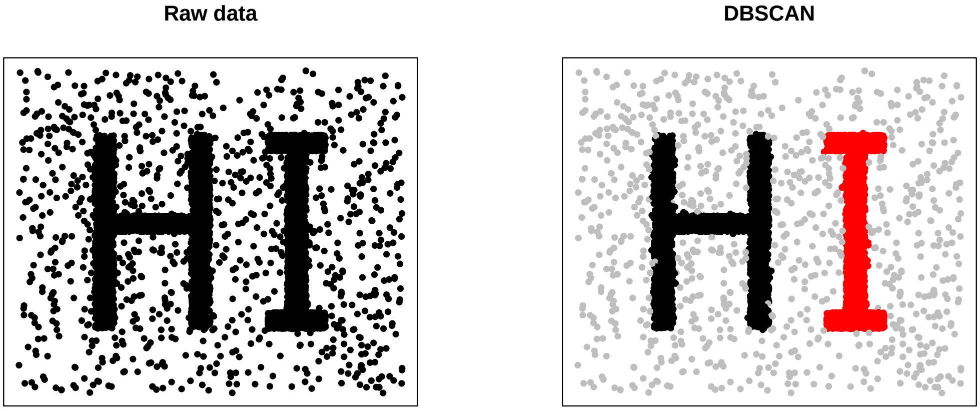


Figure 2.7: A DBSCAN result on a data set (over the unit square) containing clusters of non-convex form. The parameters used were $\epsilon = 0.01$, minPts $= 5$. Light gray is used to represent points labeled by DBSCAN as noise.

minimum cluster sizes and density, and has difficulties extracting clusters of *varying* density. More recent variants of density-based clustering, like OPTICS [Ank+99], help to mitigate the latter two limitations[2].

The third data set in Figure 2.4, commonly known as the "spirals" data, shows another situation in which clusters need not necessarily have convex boundaries. In such cases it is not sensible to assign cluster membership based on prototype vectors or Gaussian distributions. Again, a more local "connectivity"-based approach can prove useful in such situations.

Spectral clustering is the name given to techniques based on graph-theoretical notions, and that exploit the spectral decomposition of matrices derived from the data set. The symmetric matrix typically used is the so-called "affinity matrix" $A \in \mathbb{R}^{n \times n}$, which encapsulates the "similarity" information between all objects in the data. The similarity is typically measured using the Gaussian kernel. Specifically, the affinity between two data points with vectors $\vec{x}_i$ and $\vec{x}_j$ is

$$a_{ij} = e^{\frac{-\left\|\vec{x}_i - \vec{x}_j\right\|^2}{2\sigma^2}}. \tag{2.3}$$

[2] In solving these problems, however, it could be argued that OPTICS introduces others. It produces a hierarchical clustering that typically requires manual interpretation.

σ is a scale parameter, typically interpreted as a measure of when two points are considered "similar". The selection of σ is commonly done manually, however it can also be made to vary in space and can be determined automatically [ZP05].

Spectral clustering can be formulated in various ways, but the classical goal is to find the first k eigenvectors of the symmetric, normalized Graph Laplacian

$$L^{\text{norm}} := I - D^{\frac{-1}{2}} \cdot A \cdot D^{\frac{1}{2}}, \tag{2.4}$$

where D is the graph's *degree* matrix. The k eigenvectors are used as columns in constructing the matrix $V \in \mathbb{R}^{n \times k}$. The rows of V are then interpreted as the new data observations in k-dimensional space. In this space, it turns out that grouping by *compact* clusters corresponds to minimizing the inter-cluster affinity (that is, maximizing the total "separation" of the clusters) in the original graph context. The final step to identify the clusters is hence to apply a traditional compactness-based clustering method, like k-Means or EM, on this k-dimensional space.

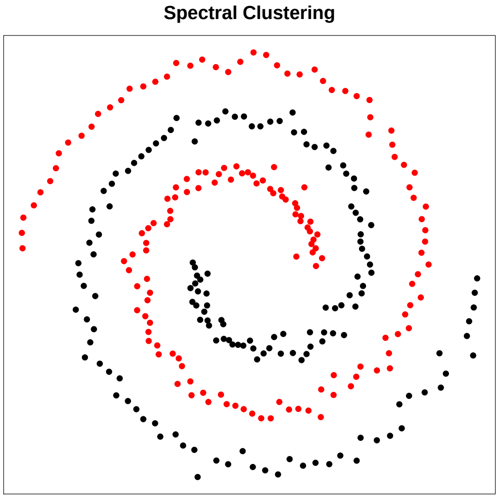


Figure 2.8: The spectral clustering result on the spirals data (using the R *kernlab* package [Kar+04]). We note that density-based algorithms like DBSCAN can find a similar solution.

Although shown to be empirically successful for image segmentation and a variety of exotic spatial data sets (e.g. snakes, letters, and the spirals in Figure 2.8), spectral clustering in its basic form has a number of limitations. These include the difficulty in handling noise (**Challenge 4**, especially if e.g. k-means is used in the final clustering stage), the requirement for the user to specify the number of clusters k and the scale σ (**Challenge 1**), the difficulty in handling multi-scale data, and the high computational cost (**Challenge 2**). Indeed, spectral clustering experiences problems on the DBSCAN-example data in Figure 2.7 because of the noise (note that DBSCAN performs well on the spirals data). More recent approaches, including the highly-cited "self-tuning" (automated) method for spectral clustering [ZP05], have made progress on some of these limitations.

3 Literature Review

In this chapter we reflect on results from the specialized, state-of-the-art literature that is more directly relevant for the work in this thesis.

3.1 Matrix Factorizations over Discrete, Finite Sets

In Section 2.2 we discussed the concept of using matrix factorizations from linear algebra as a tool for Blind-Source Separation. We briefly reviewed Independent Component Analysis, Principal Component Analysis and Non-negative Matrix Factorization as three techniques commonly used to solve concrete realizations of the abstract BSS problem. All three impose restrictions on the nature of the factor matrices in order to be able optimize an objective function that is assumed to coincide with what a practitioner would consider as useful. In the Cocktail Party example, the assumption that the audio sources are statistically independent is reasonable, so ICA's objective function is a sensible choice. In contrast, if we wish to visualize a high-dimensional, real-valued spatial data set, we might obtain a useful view by following PCA's objective (selecting the two or three directions that explain the greatest variance). If a "parts-based" representation is sensible, we might select NMF.

Consider now the simple data matrix in Figure 3.1, which records the courses taken by four students (the courses are Programming 🖥, Electromagnetism 🧲, Mathematics 🧮, Molecular Dynamics ⚗). The first three students study the disciplines of Computer Science, Physics and Chemistry respectively. The last student studies Numerical Simulation, an interdisciplinary program involving a *mix* of the other three "pure" disciplines. To reflect the fact that these measurements are not made on the ratio scale, the entries of the matrix are denoted either $\mathfrak{f}$ or $\mathfrak{t}$ for *false* and *true* respectively (rather than 0 and 1).

As exploratory data miners, we hope to learn something from this data. We pose the question: Can we perform Blind-Source Separation on this data in order to extract some useful and intuitive "sources" (latent patterns)?

Our first (naïve) step in this direction might be to map our nominal labels to numerical values and apply a numerical technique from linear algebra. We select the commonly-used mapping $\mathfrak{f} \mapsto 0$ and $\mathfrak{t} \mapsto 1$. Figure 3.2 shows the effective factors obtained after applying SVD to the data obtained and keeping only $k = 3$ of the most significant eigenvalues.

	🖥	🧲	🧮	⚗
👤$_1$	$\mathfrak{t}$	$\mathfrak{f}$	$\mathfrak{t}$	$\mathfrak{f}$
👤$_2$	$\mathfrak{f}$	$\mathfrak{t}$	$\mathfrak{t}$	$\mathfrak{f}$
👤$_3$	$\mathfrak{f}$	$\mathfrak{f}$	$\mathfrak{t}$	$\mathfrak{t}$
👤$_4$	$\mathfrak{t}$	$\mathfrak{t}$	$\mathfrak{t}$	$\mathfrak{t}$

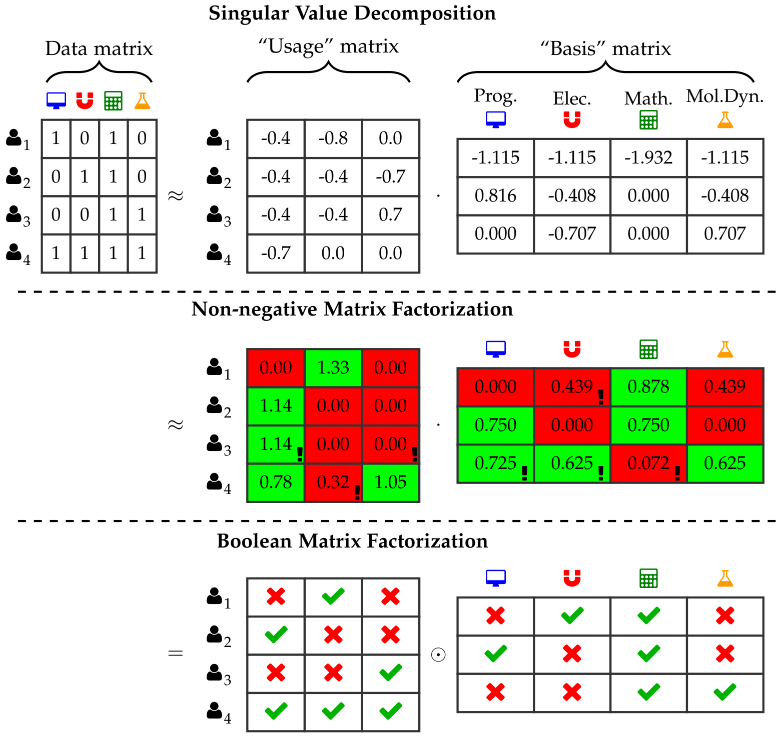
Figure 3.1: A simple data matrix that records the participation of students in courses.

We note two key limitations of the SVD result. Firstly, it is approximate. For this data, there exists no exact rank-three decomposition when the classical matrix product is used. Secondly, the factors contain both negative, zero and positive entries.

From a knowledge-discovery perspective, it is the latter limitation that is perhaps the most critical. It has a direct impact on the *interpretability* of the factors in the context of the data domain (**Challenge 4**). That is, how are we supposed to map the factors found by SVD back into meaningful insights into the context of students and courses? Given the SVD factors here, there is no trivial mapping scheme that will help us learn about our "source" disciplines of Computer Science, Physics and Chemistry.

The Non-negative Matrix Factorization result (also in Figure 3.2) is an improvement. Although it is still only an approximation, it constrains its factors to non-negative entries, which can often aid interpretability. In this case we see entries in the factors ranging from 0 to approximately 1.14. To get a clearer picture, we might use simple rounding to interpret the result in the context of the domain. We color a cell *red* if its entry is closer to zero than it is to 1, and *green* otherwise.

With this rounding, and even if we perform normalization, the factors still fail to reflect our "source" disciplines (errors relative to the upcoming BMF decomposition are labeled with an exclamation mark). Why is this? To answer this question, we reflect on the fundamental assumption that NMF makes about the data we provide. NMF assumes that the data is measured on a ratio scale. NMF, and indeed all techniques from *linear* algebra, rely on the classical matrix product which performs numerical addition and multiplication. The crux is that performing numerical addition and multiplication make little sense for logical data, because logical data is not measured on a ratio scale. Rather, two-valued logical data is more sensibly mixed and weighted using the semantics of Boolean disjunction and conjunction. Concretely, the Boolean axiom of $1 + 1 = 1$ (where 1 represents logical truth and + represents logical disjunction) is in conflict with its arithmetic counterpart $1 + 1 = 2$ (where 1 is a real value and + is numerical addition). When mixing is involved, this has the effect of "pushing down" the values in the NMF


Singular Value Decomposition
Data matrix
"Usage" matrix
"Basis" matrix
Prog.
Elec.
Math.
Mol.Dyn.
≈
·
Non-negative Matrix Factorization
Boolean Matrix Factorization
=
⊙

Singular Value Decomposition

1	1	0	1	0
2	0	1	1	0
3	0	0	1	1
4	1	1	1	1

1	-0.4	-0.8	0.0
2	-0.4	-0.4	-0.7
3	-0.4	-0.4	0.7
4	-0.7	0.0	0.0

Prog.	Elec.	Math.	Mol.Dyn.
-1.115	-1.115	-1.932	-1.115
0.816	-0.408	0.000	-0.408
0.000	-0.707	0.000	0.707

Non-negative Matrix Factorization

1	0.00	1.33	0.00
2	1.14	0.00	0.00
3	1.14 !	0.00	0.00 !
4	0.78	0.32 !	1.05

0.000	0.439 !	0.878	0.439
0.750	0.000	0.750	0.000
0.725 !	0.625 !	0.072 !	0.625

Boolean Matrix Factorization

1	✗	✓	✗
2	✓	✗	✗
3	✗	✗	✓
4	✓	✓	✓

✗	✓	✓	✗
✓	✗	✓	✗
✗	✗	✓	✓

Figure 3.2: Finding latent disciplines from students and courses.

factors, which can lead to misleading results on final interpretation. We hence arrive at the seminal discrete matrix-factorization technique for non-ratio-scale data in the context of data mining: Boolean Matrix Factorization.

3.2 Boolean Matrix Factorization

Let $\mathcal{B} = \{\mathfrak{f}, \mathfrak{t}\}$ be the set of Boolean logic values. Let the binary function $\oslash_{\mathcal{B}} : \mathcal{B} \times \mathcal{B} \mapsto [0,1]$ be a contrast or "dissimilarity" measure over this set such that for all $a, b \in \mathcal{B}$, $\oslash_{\mathcal{B}} (a, b) \mapsto 0$ if $a = b$, otherwise 1 (i.e. the Boolean XOR function). The Boolean matrix product $A \odot_{\mathcal{B}} B$ of two matrices $A \in \mathcal{B}^{n \times k}, B \in \mathcal{B}^{k \times m}$ is an analog to the classical matrix product, whereby arithmetic addition and multiplication are exchanged for logical disjunction $\oplus_{\mathcal{B}}$ and conjunction $\otimes_{\mathcal{B}}$:

$$\left(A \odot_{\mathcal{B}} B\right)_{ij} = \left(a_{i1} \otimes_{\mathcal{B}} b_{1j}\right) \oplus_{\mathcal{B}} \cdots \oplus_{\mathcal{B}} \left(a_{ik} \otimes_{\mathcal{B}} b_{kj}\right) . \tag{3.1}$$

The Boolean Matrix Factorization problem [MV14; Mie09] is stated as follows:

Problem (BMF: Boolean Matrix Factorization). *Given a Boolean data matrix $D \in \mathcal{B}^{n \times m}$ and positive integer k, find "usage" and "basis" matrices $U \in \mathcal{B}^{n \times k}$ and $B \in \mathcal{B}^{k \times m}$ that minimize*

$$\|D \oslash_{\mathcal{B}} (U \odot_{\mathcal{B}} B)\|_1, \tag{3.2}$$

where $\oslash_{\mathcal{B}}$ is applied entry-wise to produce a "residual" matrix, and $\|\cdot\|_1$ is the entry-wise 1-norm (simple sum of all matrix elements).

Returning to Figure 3.2, we see how the BMF treatment of the students and courses data offers numerous advantages. Firstly, we see how BMF's *logic-based* mixture model enables us to achieve an *exact* "rank"[1]-three decomposition (zero reconstruction error with respect to Equation (3.2)). Secondly, and perhaps most importantly, the factor matrices are directly interpretable in the context of the domain (**Challenge 4**). The basis matrix exposes our latent Physics, Computer Science and Chemistry disciplines, and the usage matrix shows how the fourth student learns a *mix* of the core disciplines. Finally, we note that the pre- and post-processing steps (mapping and rounding) necessary for SVD and NMF are not required by BMF. Indeed, such mapping decisions are the root cause of the problems found when applying SVD and NMF to these data: these mappings introduce ratio-scale semantics on non-ratio-scale data.

Asso is a state-of-the-art algorithm specifically designed for solving BMF [Mie+08]. The algorithm is so-named because it involves the computation of Asso*ciation accuracies*. Specifically, an entry a_{ij} of the association matrix A represents the confidence of the association between the ith and jth column, as defined in association-rule mining [AIS93]. Initially, the entries of the non-symmetric matrix A are hence real values between zero and one. Heuristically, A can be converted to a binary matrix using a rounding threshold τ, and its rows then considered as candidate Boolean basis vectors. k of these candidate basis vectors are then selected in a greedy fashion to arrive at an approximate solution. In the best case, Asso computes a solution in time $\mathcal{O}(knm^2)$. The computation of the association matrix is particularly expensive: the association score is calculated for each pairwise combination of columns, requiring time $\mathcal{O}(nm^2)$. Asso requires the values of k and τ to be defined by the user, and depends on floating-point arithmetic for the creation and manipulation of the raw matrix A (despite the fact that the original data is two-valued discrete).

3.2.1 Combinatorics Problems Related to Boolean Matrix Factorization

The BMF problem is related to a number of problems from combinatorics. These combinatorics problems will help us in our work as well. For reference, we hence detail these problems in the following subsections.

[1] See [MV14] for a discussion on the Boolean rank of a matrix.

Set Cover

The Set Cover (SC) problem is one of the classic 21 NP-Complete problems published by Karp [Kar72].

Problem (Set Cover). *Given a "universe" set of elements $\mathcal{U} = \{1, 2, \ldots, m\}$ and a collection $\mathfrak{S} = \{\mathcal{S}_1, \mathcal{S}_2, \ldots, \mathcal{S}_n\}$ of sets such that their union is $\mathcal{U}$, find the smallest sub-collection $\mathfrak{C} \subseteq \mathfrak{S}$ such that the union of its sets is also $\mathcal{U}$.*

Example (Set Cover). *Consider the problem instance with $\mathcal{U} = \{1, 2, 3, 4, 5\}$ and $\mathfrak{S} = \{\{1\}, \{1, 2, 3\}, \{2, 5\}, \{3, 5\}, \{4, 5\}, \{3, 4\}\}$. It is clear that the union of all sets in $\mathfrak{S}$ is equal to $\mathcal{U}$. The smallest sub-collection of $\mathfrak{S}$ which still fully covers $\mathcal{U}$, and hence the solution to the problem, is $\mathfrak{C} = \{\{1, 2, 3\}, \{4, 5\}\}$.*

The *optimization* version of the set-cover problem is NP-Hard [BJ08] (the decision version is NP-complete). The baseline greedy polynomial-time algorithm for approximating set cover simply involves selecting those successive sets which cover the largest number of remaining uncovered elements. The algorithm terminates when all elements are covered. In [Sla96] it is shown that the approximation ratio for this greedy algorithm is exactly $\ln n - \ln \ln n + \mathcal{O}(1)$, making it essentially the best-possible polynomial-time approximation algorithm for set cover.

Red-Blue Set Cover

The Red-Blue Set Cover (RBSC) problem [Car+00] is a generalization of the Set Cover problem. The objective of RBSC differs from that of SC in that the cost of a solution considers not the number of covering sets, but the number of covered "penalty" elements from a second "red" set:

Problem (Red-Blue Set Cover). *Given two disjoint sets $\mathcal{R} = \{r_1, r_2, \ldots, r_\rho\}$ and $\mathcal{B} = \{b_1, b_2, \ldots, b_\beta\}$, as well as a collection of sets $\mathfrak{S} \subseteq 2^{\mathcal{R} \cup \mathcal{B}}$, find the sub-collection $\mathfrak{C} \subseteq \mathfrak{S}$ that covers all elements in $\mathcal{B}$ but covers the minimum possible number of elements in $\mathcal{R}$.*

Example (Red-Blue Set Cover). *Consider the problem instance with $\mathcal{B} = \{1, 2, 3, 4, 5\}$, $\mathcal{R} = \{6, 7, 8\}$ and $\mathfrak{S} = \{\{1\}, \{1, 2, 3, 7, 8\}, \{2, 5\}, \{3, 5, 6\}, \{4, 5, 7\}, \{3, 4, 6\}\}$. The solution is $\mathfrak{C} = \{\{1\}, \{2, 5\}, \{3, 4, 6\}\}$, which has the minimum cost of 1.*

In [Car+00] it is shown that the SC problem can be reduced to the RBSC problem, and that the RBSC problem is at least as hard as the SC problem.

Positive-Negative Partial Set Cover

The Positive-Negative Partial Set Cover (±PSC) problem is a more recent generalization of the RBSC problem (which in turn is a generalization of the SC problem). Its introduction was motivated by the BMF problem [Mie08a]. Compared to RBSC and SC, ±PSC relaxes the strict requirement of complete cover. It searches for a solution that represents the best balance between covering positive elements and *not* covering negative elements.

Problem (Positive-Negative Partial Set Cover). *Given two disjoint sets $\mathcal{P}$ and $\mathcal{N}$, as well as a collection of sets $\mathfrak{S} \subseteq 2^{\mathcal{P}\cup\mathcal{N}}$, find the sub-collection $\mathfrak{C} \subseteq \mathfrak{S}$ that minimizes the cost function*

$$cost_{\pm PSC}(\mathcal{P}, \mathcal{N}, \mathfrak{S}) = \left| \mathcal{P} \setminus \left(\bigcup_{\mathcal{C}\in\mathfrak{C}} \mathcal{C} \right) \right| + \left| \mathcal{N} \cap \left(\bigcup_{\mathcal{C}\in\mathfrak{C}} \mathcal{C} \right) \right|. \tag{3.3}$$

Example (Positive-Negative Partial Set Cover). *Consider the problem instance with $\mathcal{P} = \{2,3,5,7\}$, $\mathcal{N} = \{1,4,6\}$ and $\mathfrak{S} = \{\{1,2,6\},\{2,4\},\{1,4,5,6,7\},\{4,5,7\}\}$. The sub-collection of $\mathfrak{S}$ achieving the minimum cost is $\mathfrak{C} = \{\{2,4\},\{4,5,7\}\}$. The minimum cost is 2, because this solution covers one negative element (4) and fails to cover one positive element (3).*

In [Mie08a] it is shown that RBSC can be reduced to ±PSC, that the ±PSC problem is at least as hard as RBSC, and that a polynomial-time approximation algorithm for ±PSC exists that achieves an approximation factor of $2\sqrt{(|\mathfrak{S}| + |\mathcal{P}|) \log |\mathcal{P}|}$.

3.2.2 Missing-Value Boolean Matrix Factorization

Missing-Value Boolean Matrix Factorization (MVBMF) is a variant of Problem BMF that considers the case where the data matrix is incomplete.

Problem (MVBMF: Missing-Value Boolean Matrix Factorization). *Let $\mathcal{B}_{MV} = \mathcal{B} \cup \{?\}$, where ? indicates a missing value. Given a Boolean data matrix $D \in \mathcal{B}_{MV}^{n\times m}$ and positive integer k, find "usage" and "basis" matrices $U \in \mathcal{B}^{n\times k}$ and $B \in \mathcal{B}^{k\times m}$ that minimize*

$$\|D \oslash_{\mathcal{B}} (U \odot_{\mathcal{B}} B)\|_1, \tag{3.4}$$

where $\oslash_{\mathcal{B}}$ and the norm $\|\cdot\|_1$ this time consider only the known elements in D.

This problem has applications in collaborative filtering [SK09] and the mining of "roles" from incomplete data during an organization's migration to role-based access control (RBAC) [Vav+].

Asso$^{\text{MV}}$ is the name we use to denote the Asso extension that supports missing values [YM12]. The major contribution that this work makes to Asso is in the algorithm step where the association matrix is computed. To handle missing values, the original definition of association confidence is modified to include a probabilistic component. The resulting association matrix is then converted to binary (using the same τ parameter as Asso). The remaining part of the algorithm is similar to Asso, although certain additional data structures are needed to track the elements of the data matrix for which the cover need not be computed. Asso$^{\text{MV}}$ also has quadratic run-time complexity with respect to the dimension m (again due to the need to compute the full association matrix). The method is evaluated on synthetic data with up to 99% missing values in the data matrix [YM12].

One key advantage of Asso$^{\text{MV}}$ is that it addresses both the *prediction* and *description* task simultaneously. That is, the factors U and B are 1) Boolean, which admits their interpretation in the context of the domain to help describe potentially-valuable latent patterns (*description*), and 2) multiplied using the Boolean matrix product, which helps to "fill in the blanks" in the data matrix (*prediction*).

Other techniques exist for the collaborative-filtering problem that focus primarily on *prediction*. Maximum-Margin Matrix Factorization (MMMF) [SRJ04] is such a technique. MMMF finds a factorization of the same basic structure ("usage" and "basis" matrices), however the factors contain real values and are multiplied using the classical matrix product. The main contribution of this work is the formulation of the binary collaborative-filtering problem (in which the data matrix has missing values) in terms of standard optimization problems. To this end, the dimensionality k is kept *unbounded* and the factorization is regularized with a *low-norm* constraint. For the learning process it is shown that, when keeping one of the matrices fixed, each separate linear prediction problem decomposes into a standard support-vector machine problem if the hinge-loss error (appropriate for binary data) is chosen. This technique has been shown to be highly successful on the *prediction* problem, however fails to carefully address the *description* problem for Boolean data. Like SVD and NMF, the use of real-values and a linear mixture model ultimately renders the factors difficult for interpretation in the context of the Boolean domain. Additionally, and *unlike* SVD and NMF, MMMF leaves the decomposition rank k unbounded, which may result in a large and indigestible set of "features" in the factorization.

3.3 Ordinal Matrix Factorization

Ordinal Matrix Factorization was first presented in the context of data-mining research in [BK13]. Like BMF, the authors consider the problem of performing Blind-Source Separation on data measured over a non-ratio scale. In this case, the data under consideration is measured over ordinal scales.

To motivate the need for OMF, Figure 3.3 shows an ordinal example where the "fre-

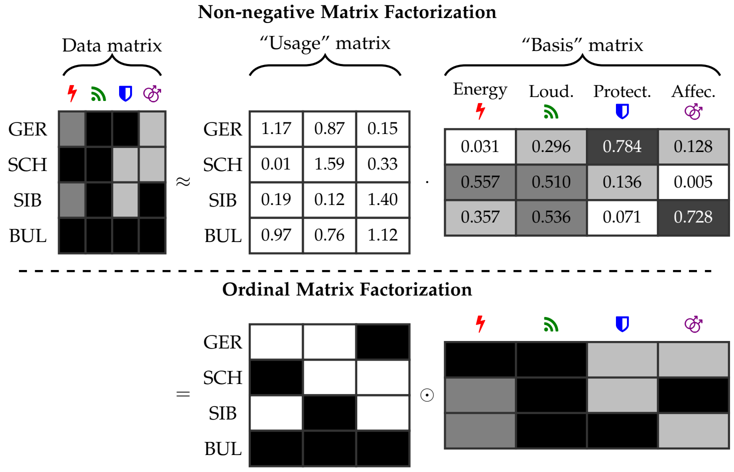


Figure 3.3: Finding latent canine temperaments from dogs and traits.

quency" {Never, . . . , A Great Deal} of dog traits (energy ⚡, loudness 📶, protectiveness 🛡 and affection ⚥) is measured for four breeds (*GER*man Shepherd, *SCH*ipperke, *SIB*erian Husky and *BULL* Terrier). From the domain we are aware of the latent dog temperaments "hyperactive" (energetic and loud), "playful" (loud and affectionate) and "watchful" (loud and protective), which we might hope to uncover with a rank-three decomposition. By imposing a five-element numerical scale $(0, 0.25, 0.5, 0.75, 1)$ on our ordinal data, we might again try NMF. If we again use simple rounding to get a clearer picture, we struggle to clearly see our latent concepts. The first basis vector, for example, fails to pair loudness with protectiveness to highlight "watchfulness". OMF's *exact* decomposition, on the other hand, clearly highlights the temperaments in the basis matrix, with the usage matrix confirming that Bull Terriers strongly exhibit all three. NMF's result is again attributable to its use of a classical linear mixture model (arithmetical addition and multiplication). A more in-depth OMF motivation on real-world canine data is given in [BK13].

Let $\mathcal{L} = \{0, \frac{1}{s+1}, \dots, 1\}$ represent an s-element partially-ordered set bounded by 0 and 1. Scales of this nature are often found in survey-research. For example, we often find "Likert Items" inspired by Miller's Law[2] [Mil56] with the possible values:

□ strongly disagree ░ disagree ▒ neutral ▓ agree ■ strongly agree

When we map such choices to the numerical values $\mathcal{L} = \{0, \frac{1}{4}, \frac{1}{2}, \frac{3}{4}, 1\}$ for the sake of

[2] The observation that the number of objects an average person can hold in working memory is approximately 7 ± 2.

convenience, we should take care not to think that the classical arithmetic operations can then be applied. For example, we cannot make a statement like "*Neutral* is twice as much agreement as *Disagree*" for this data because we have no meaningful "zero" value. It is this mechanism that causes the NMF interpretation problems in Figure 3.3.

Instead, values on such scales can be weighted and mixed using the Łukasiewicz operations [BK13]. The ordinal matrix product $\odot_{\mathcal{L}}$ hence uses the operation $a \oplus_{\mathcal{L}} b = \max(a, b)$ in place of addition, and the operation $a \otimes_{\mathcal{L}} b = \max(0, a + b - 1)$ in place of multiplication. Let the binary function $\oslash_{\mathcal{L}} : \mathcal{L} \times \mathcal{L} \mapsto [0, 1]$ be a contrast or "dissimilarity" measure over $\mathcal{L}$ (in [BK13] the function is defined as $\oslash_{\mathcal{L}}(a, b) = |a - b|$). The formal definition of the OMF problem is then [BK13]:

Problem (OMF: Ordinal Matrix Factorization). *Given an ordinal data matrix $D \in \mathcal{L}^{n\times m}$ and positive integer k, find "usage" and "basis" matrices $U \in \mathcal{L}^{n\times k}$ and $B \in \mathcal{L}^{k\times m}$ that minimize*

$$\|D \oslash_{\mathcal{L}} (U \odot_{\mathcal{L}} B)\|_1, \tag{3.5}$$

where, like BMF, $\oslash_{\mathcal{L}}$ is used entry-wise to produce a "residual" matrix, and $\|\cdot\|$ is the entry-wise 1-norm (simple sum of all matrix elements).

In [BK13] the algorithm GreEss is introduced to solve OMF. The algorithm is based on formal-concept theory. Importantly, we note that GreEss achieves the optimal *from-below* decomposition. That is, GreEss is able to compute the optimal solution from the subset of OMF solutions that give a reconstructed data matrix not exceeding the original matrix D in any entry. Although not mentioned in the original article, our empirical analysis suggests that this effectiveness comes at a price: the time complexity of GreEss is in $\mathcal{O}(sn^2m^3)$.

3.4 Clustering in the Context of Noise/Clutter

Although noise is considered to some extent by a number of clustering techniques (e.g. DBSCAN, or Robust Information-Theoretic Clustering [Böh+06]), there are far fewer that consider the case where global "clutter" *heavily* outweighs the number of clustered points (the right-most case in Figure 2.4). We are aware of only two techniques that specifically focus on this kind of scenario. We discuss these techniques in this section.

3.4.1 Peer Article: *Detecting Features in Spatial Point Processes with Clutter via Model-Based Clustering*

In [DR98] the authors consider the problem of detecting surface-laid minefields on the basis of many images from reconnaissance aircraft (Figure 3.4). Minefields are often laid in a structured way. The authors approach the problem as one of clustering in the presence of significant "clutter". They propose a distribution-based method which sees minefields modeled as multivariate normal distributions, and "clutter" represented by a spatial Poisson process. The EM algorithm is used to find an approximate solution to the problem, with approximate Bayes factors employed to select the number of clusters. The resulting MCLUST-EM algorithm is shown to produce good results for up to two such "minefield" clusters in the presence of considerable "clutter". Examples are also given for detecting seismic faults based on an earthquake catalog, although this data is much less "cluttered".

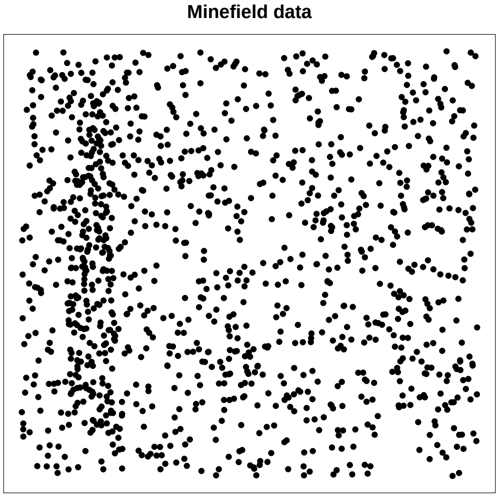


Figure 3.4: A simulated minefield (left) in the presence of "clutter" (like metal objects or rocks). This data set was generated based on similar data presented in [DR98].

Primarily driven by the minefield application, MCLUST-EM is unfortunately only usable for data in two dimensions and is parametric (assumption of Gaussian clusters).

3.4.2 Peer Article: *Efficient Algorithms for Non-Parametric Clustering with Clutter*

Compared to [DR98], the development of the method in this article [WM02] is not driven by a particular application. The authors argue that situations with a combination of noisy background and clusters are found in a *variety* of applications, and that a new approach is needed. One application that the authors *do* use as motivation is the clustering of galaxies (Figure 3.5). Each "bright spot" in such a data set is a galaxy containing billions of stars. Astrophysicists are interested in clustering galaxies, but a noisy background of field galaxies interferes with traditional clustering techniques based on mixture models, vector quantization or graph-theory.

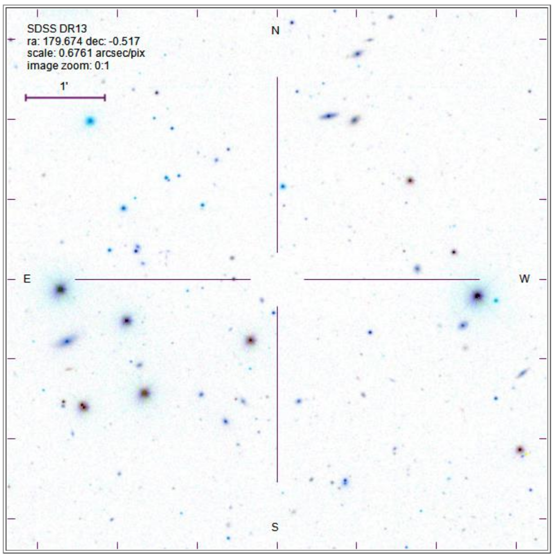


Figure 3.5: A view of galaxy data from the Sloan Digital Sky Survey [Alb+16; Eis+11] as considered in [WM02].

The key contribution of [WM02] is a refined version of the Cuevas, Febrero and Fraiman (CFF) algorithm [CFF00]. The CFF algorithm first determines the subset of data points that are in high-density regions using a non-parametric density estimator. This is followed by a clustering step, whereby such high-density points are agglomerated. The refinements made are aimed at partially addressing the computational problems of the CFF algorithm. However, a key limitation is that it still needs to be executed "hundreds of times" [WM02, p. 4] with different combinations of the three required parameters.

3.5 Anomaly and Change-Point Detection in Time-Series Data

The topics of "change-point detection", "anomaly detection" and "event detection" for time-series data (**Challenge 3**) have seen numerous contributions in the data-mining, signal-processing and statistics literature. In this section we firstly review a range of generic techniques for numerical time series. Afterwards, we review text-based (Natural Language Processing) methods for topic and event detection in social-media and microblogging services like Facebook and Twitter.

3.5.1 Numerical Techniques from Data-Mining and Signal-Processing

A number of recent contributions to the data-mining anomaly-detection literature have been made by research teams of internet-based companies. Twitter, for example, has released two popular techniques: Twitter Anomaly Detection [JKM14] and Twitter Breakout Detection [VHK14]. A result from the former technique is shown in Figure 3.6, where we can see that it identifies intuitive outliers (red points) in a signal that contains noise, short-term seasonality and long-term trend. The underlying algorithm uses statistics based on the energy spectral density of the signals.

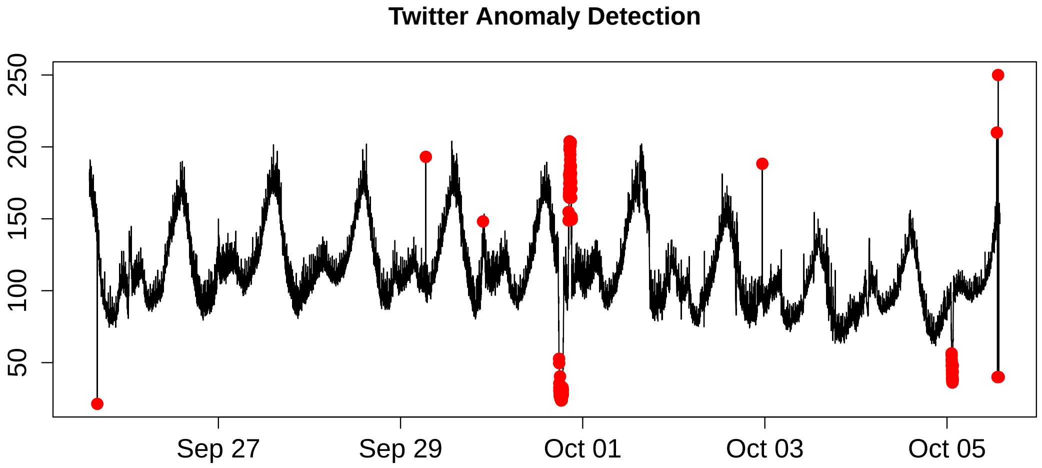


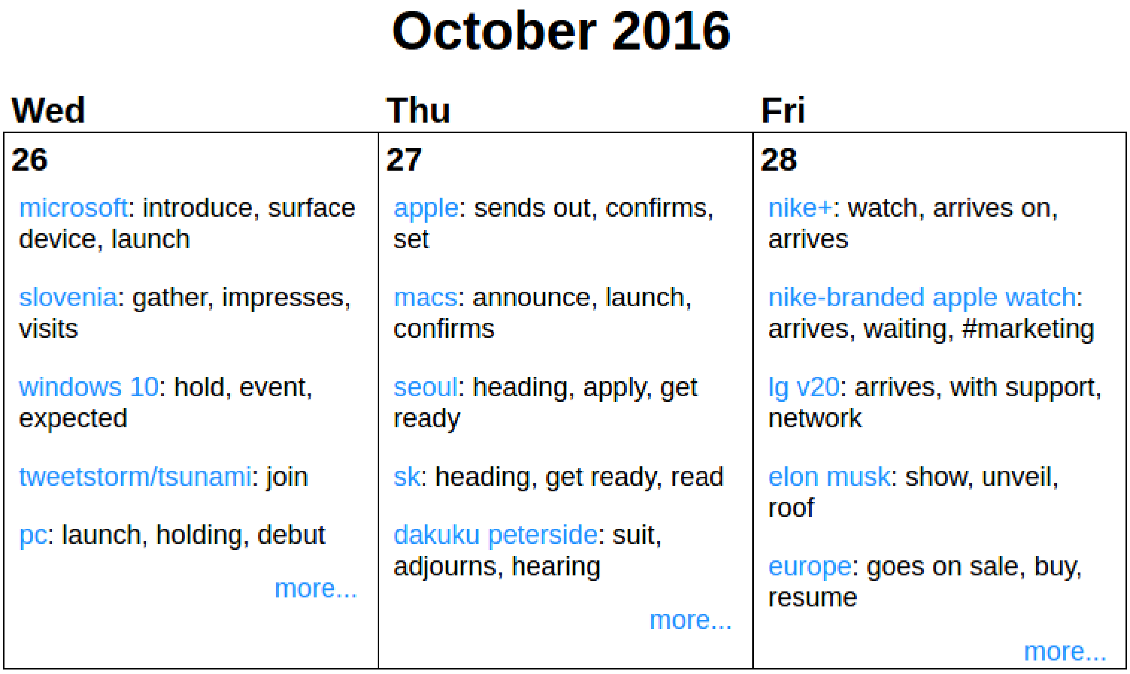
Figure 3.6: Twitter Anomaly Detection finds significant deviations in signals containing noise, short-term seasonality and long-term trend.

Yahoo! is another company which has recently "open sourced" a framework for anomaly detection [LAF15]. It is named the Extensible Generic Anomaly Detection System (EGADS). Subscribing to the view that there is no "silver bullet" approach, it contains a *suite* of anomaly-detection techniques in a single package. The overall architecture consists of two primary components: the time-series modeling module (TMM) and the anomaly-detection module (ADM). The TMM predicts expected future values of a given time series. These predictions are consumed by the ADM, which in turn computes anomaly scores.

The recent statistical literature contains further approaches. The eDiv approach presented in [MJ14] finds segments in time-series data, the boundaries of which represent change points. To this end it employs a binary bisection method and a permutation test. eDiv is primarily an offline method: its computational complexity is in $\mathcal{O}(kn^2)$, where k is the number of estimated change points and n the number of observations. From the same work we find eAgglo, the bottom-up variant of eDiv, and another probabilistic pruning method cp3o. These techniques likewise have quadratic run-time complexity in n. The work presented in [Rig10] considers the same problem and presents the technique pDPA. pDPA is able to find the solution that globally minimizes the sum of the quadratic losses in each segment. pDPA likewise has a worst-case quadratic run-time in n.

3.5.2 Specialized Techniques for Social Media

Motivated by large-scale applications like Twitter and Facebook, the data-mining community has recently been focusing on the task of detecting "events" or "topics" in social media. In [Rit+11; Rit+12] the algorithm TwiCal is presented. TwiCal exploits part-of-speech tagging, named entity recognition, temporal expression resolution (e.g. "next Friday"), event-tagging and event classification in order to generate an open-domain calendar of significant events (see Figure 3.7). In [HTK15] we find an approach that is more focused on the *real-time* detection of topics in Twitter (as opposed to trying to generate a calendar) in the presence of noise. To handle the high bandwidth of Twitter (**Challenge 2**), the authors reformulate the Non-negative Matrix Factorization problem in a stochastic manner. The reformulation enables the use of stochastic gradient descent updates in linear time with respect to the number of non-zero entries in the matrix.



October 2016

Wed	Thu	Fri
26	27	28
microsoft: introduce, surface device, launch slovenia: gather, impresses, visits windows 10: hold, event, expected tweetstorm/tsunami: join pc: launch, holding, debut more...	apple: sends out, confirms, set macs: announce, launch, confirms seoul: heading, apply, get ready sk: heading, get ready, read dakuku peterside: suit, adjourns, hearing more...	nike+: watch, arrives on, arrives nike-branded apple watch: arrives, waiting, #marketing lg v20: arrives, with support, network elon musk: show, unveil, roof europe: goes on sale, buy, resume more...

Figure 3.7: Partial screenshot of the Twitter Status Calendar (statuscalendar.com) – a demonstration of the system described in [Rit+11; Rit+12].

4 Research Approach

This thesis is based on the research and development of practical methods for exploratory, unsupervised data mining. Research questions were initially spawned based on the community-identified challenges enumerated in Section 1.3. Given a research question, literature was reviewed, existing algorithms were collected and example data sets were synthetically generated or harvested from various sources. Given these resources, an iterative phase of method- and algorithm-design was performed to identify and overcome the limitations of existing approaches to the problems in question. Our work did not focus on individual data sets alone, nor a hypothesis for explaining a phenomenon in a particular application domain. The following sections discuss the elements of our research approach in more detail.

4.1 Literature Reviews

Particularly in the accelerating field of data-mining research, it is important to frequently review the literature for a given research question or problem. As with all research, our contributions build on those of others. We adopted the following strategies during our literature reviews:

- Perhaps our primary sources of literature were the proceedings of the premier conferences in the field. These conferences are 1) ACM SIGKDD Conference on Knowledge Discovery and Data Mining, 2) IEEE International Conference on Data Mining, and 3) SIAM Data Mining. Proceedings are published annually, with articles indexed by topic-area (e.g. cluster analysis). These three established conferences are highly-selective and tend to attract the greatest attention and highest-impact contributions from data-mining researchers and practitioners (more so than any data-mining journal, for example).

- "Bottom-up" search was performed by seeking *concrete implementations of algorithms* for the given research question in established data-mining software systems. These systems included 1) the Environment for Developing KDD-Applications Supported by Index-Structures (ELKI) [AKZ08], 2) the Waikato Environment for

Knowledge Analysis (Weka) [Hal+09], and 3) the Comprehensive R Archive Network (CRAN) [Hor12]. Given a software implementation, the source-code or documentation was consulted to extract the relevant publication(s).

- "Top-down" search was performed by seeking *survey articles* for a given research question or problem definition.
- Established researchers in the field were contacted, who often recommended related work. These researchers are listed in Table 4.3 and also in the acknowledgments section of this thesis.

In all cases, identified articles were additionally used as a starting point for backwards search (analyzing the references of each publication in a recursive manner). Taken together, these strategies yielded a representation of the state-of-the-art with respect to a given research question.

4.2 Idea Synthesis, Problems Definitions and Algorithm Design

Having performed literature reviews for a given research question, we began a creative process of idea-generation and brainstorming in an attempt to address the weaknesses of the state-of-the-art. Early in this process, we would typically generate and focus on a "running-example" or "illustrative" problem that highlighted these weaknesses (we make use of such examples in our publications). This exercise helped to remove irrelevant complexity, assisted in refining the problem definition, and served as a springboard for spawning algorithmic ideas for overcoming the weaknesses. In line with our thesis goals, we only pursued ideas that led to *linear-time* algorithmic behavior. Finally, before investing time in writing a software implementation, we manually tested the idea on small ("toy") datasets, and performed a further literature review to determine if a peer had already proposed such an algorithm for the given problem.

4.3 Software Prototypes

Our research made extensive use of software prototypes for the practical implementation of the algorithms presented. Considering that all the problems we investigated cannot be solved optimally in general, a practical implementation gave insight into the performance (both in terms of effectiveness and efficiency) of a proposed heuristic on non-trivial problem instances. Unsurprisingly, researching in this way was often iterative: The implementation was driven by changes to the algorithm, which in turn was driven by investigations into the implementation's performance. Finally, a prototype helped to confirm theoretically-calculated approximation bounds for an algorithm.

The software prototypes developed for this dissertation made use of a variety of programming languages and paradigms. All development work was done on a Linux Ubuntu system. For the initial stages of algorithm development we often selected R, a high-level interpreted programming language and statistical computing environment for rapidly prototyping data-mining ideas [Tip+15]. Java, an object-oriented compiled language, was used for an implementation in **Paper C**. For Java programming we used the IntelliJ IDEA integrated development environment (IDE). For **Papers A, B and C**, further implementations were written in C++ in order to optimize performance. For C++ programming we used the NetBeans IDE. All software prototypes are included in the code repositories referenced from each paper.

4.4 Data Sources

We empirically evaluated our proposed methods on both synthetic and real-world data. Synthetic data was created by building data-generation tools that implemented a generative model. In each case, the generative model was designed to accept a number of parameters. These parameters controlled the properties of the generated problem instance, like the size, ground-truth model order, the density of the patterns, and the amount of noise. The data-generation tools were intentionally non-deterministic, allowing multiple data sets to be generated at random for the same parameter combination. A simple random sample of several such data sets (typically 20) was then used to evaluate each algorithm in focus (more details in Section 4.6.2). We note that all data-generation tools were made available alongside our algorithm implementation in publicly-accessible repositories linked directly from footnotes in each paper.

For our real-world experiments, we used data from various sources. Some of this data was from "benchmark" repositories for machine-learning and data-mining. Table 4.1 shows the data sets used from these repositories. In other cases, we harvested data using the application programming interfaces (APIs) provided by online services. These data sets were subsequently published (with permission) in the respective source repositories (supplementary material) in each case. Table 4.2 gives a description for each of these sources and provides remarks on the data-collection approach.

Repository and Data Sets	Relevant Papers
UCI Machine Learning Repository (archive.ics.uci.edu) Breast Cancer, Congressional Voting Records, Dermatology, Hayes-Roth, Image Segmentation, Lenses, Lymphography, Pen-Based Recognition of Handwritten Digits, Promoter Gene Sequences, Seeds, Soybean, SPECT Heart, Tic-tac-toe, Trains, 3D Road Network (North Jutland, Denmark)	**A,B,D**
Rdatasets (github.com/vincentarelbundock/Rdatasets) Whiteside (MASS), OldMaps (HistData), Coalition2 (Zelig), Motor (Boot), Prestige (Car)	**D**
Personality Tests (personality-testing.info/_rawdata/) Relationships, Feminism, Assertiveness, Personality 1, Occupation, Humor Styles, Mindfulness, Masculinity/Feminism, Personality 2, Sexual Self	**C**
Pew Research Center (pewforum.org) Tolerance and Tension: Islam and Christianity in Sub-Saharan Africa	**A,B**

Table 4.1: Existing real-world data sourced from public repositories

API/Remarks	Relevant Papers
Stack Overflow (api.stackexchange.com) Sourced were 1.02×10^6 answers and their corresponding questions between the dates 2010-01-01 and 2012-12-31. The data set consists of 340×10^3 answers to the most "useful" questions, 340×10^3 answers to the most "non-useful" questions, and 340×10^3 answers to questions where the usefulness is "unknown" (zero votes).	**A,B**
Internet Movie Database (imdb.com/interfaces) Sourced were 14690 films, each having genre flags, rating information and filming locations in the form of raw text strings. The data-collection process explicitly excluded titles with the text "TV". The shooting locations ontology was generated using the Google Places API (developers.google.com/places).	**C**

Yummly (developer.yummly.com) Sourced were the 296 recipes returned from an API-search for main-course recipes of American cuisine and tagged with the holiday *summer*. As the official Yummly ingredients ontology is proprietary, we constructed an ontology by hand based on the tractable number (693) of unique ingredients in these 296 recipes.	**C**
Twitter (dev.twitter.com/rest/public) Sourced were 1) 14698 randomly-sampled Twitter user profiles and their five count-based metrics (follower count, status count etc.), 2) the 7701 tweets made against the hashtag #FathersDay between June 5, 2016 and June 20, 2016, 3) the 10901 tweets made against the hashtag #PokemonGO between July 13, 2016 and July 17, 2016.	**E**
Wikipedia (mediawiki.org/wiki/API:Recent_changes_stream) Sourced were $n > 2$ million edits to Wikipedia pages streamed through stream.wikimedia.org (namespace "/rc") between July 6, 2016 and July 11, 2016. 404,365 of these edits were tagged as *non-bot* edits.	**E**
GitHub (developer.github.com/v3/) Sourced were 100,999 repositories created between January 1 2008 and December 31 2015 with their attributes full_name, stargazers_count and forks_count.	**E**
YouTube (developers.google.com/youtube/v3/) Sourced were 138,529 videos with region code "US", type "video", video type "movie" and published between January 1 2012 and December 31 2016.	**E**

Table 4.2: Real-world data sourced from public Application Programming Interfaces.

4.5 Comparison Techniques

The broad data-mining tasks of *finding associations*, *clustering objects* and *detecting anomalies* have seen a large number of publications (Figure 1.2). Many of these publications present novel data-mining methods and detail a corresponding new algorithm. There are hence many hundreds, if not thousands, of published data-mining algorithms in existence.

To appropriately select comparison techniques for each of our proposed algorithms, we began by clearly defining the problem and/or the kind of data sets in focus. In the case of FasTer and Finesse, we found comparable techniques in Asso, Asso^{MV}, PaNDa, PaNDa+ and GreEss. That is, each searches for k patterns from discrete data that are mixed using non-linear operations (for Asso, Asso^{MV}, PaNDa and PaNDa+ the mixing semantics are Boolean; for GreEss they are ordinal). In the case of SkinnyDip we found only a single implementation of an algorithm focused on the high-clutter case [DR98], so we therefore extended our search to include six parameter-free clustering algorithms from different paradigms (SkinnyDip is likewise parameter-free, so we argued that this was a sensible property on which to base the selection of comparison techniques). Finally, in the case of BenFound, we selected ten techniques (classical benchmark techniques in addition to state-of-the-art approaches) for numerical anomaly-detection, and a set of state-of-the-art techniques specifically for *Twitter-based* topic- and event-detection. The majority of the selected techniques were the result of work presented at one of the premier data-mining conferences (Section 4.1). The techniques were primarily identified through periodic reviews of the literature, however there were also instances in which anonymous peer-reviewers suggested comparison techniques (this was the case for PaNDa+ and Mclust-EM). We additionally thank Pauli Miettinen for making the technique GreEss known to us.

We did not implement the comparison techniques ourselves, but rather obtained existing implementations from the original author or from publicly-accessible repositories. Table 4.3 shows the source for each algorithm implementation that we used in comparing to our work. We implemented "wrapper" scripts where necessary in order to integrate each comparison technique into our experimental testbed.

A subset of the comparison techniques were "parameter-free" in the sense that the only *required* input was the data set. The remaining techniques had hard parameter requirements. To be fair on the competition, we provided correct values for the parameters in each of these cases. For example, we provided the correct number $k = 6$ of clusters to the EM algorithm when evaluating it on our "running example" data in **Paper D**. In the cases where the correct value of a parameter was uncertain ("obscure" parameter), we varied the parameter over a generous range and executed the technique once for each. The concrete cases of this were: DBSCAN (**Paper D**, ϵ parameter) and Asso/Asso^{MV} (**Papers A, B and C**, τ parameter). We subsequently reported the best result from the obtained set.

4.6 Experiments

The evaluation of our proposed methods was primarily empirical. In this section we discuss a number of aspects of the approach taken when performing these experiments.

4.6.1 Evaluation Metrics

In **Papers A, B and C** we adopted the evaluation metric established by the original work on Boolean and Ordinal Matrix Factorization [Mie09; BK13]. This metric can be considered to be a "residual". It is the difference, or "error", between the original data matrix and the reconstruction. It is measured as the sum of the entry-wise contrasts (in the Boolean case it is simply the sum of false positives and false negatives in the reconstruction, i.e. Equation (3.2)).

In **Paper D** we required a metric to compare a computed clustering result against the corresponding set of "ground-truth" cluster labels. We selected Adjusted Mutual Information ($\mathrm{AMI}_{\mathrm{MAX}}$) [VEB09], a state-of-the-art metric that is calculated based on the information-theoretic measures of entropy and mutual information. Although this metric is accepted by the data-mining community, a similar metric known as Normalized Mutual Information (NMI) [Yao03] is likewise used and considered acceptable. For this reason, we additionally presented a subset of our results using the NMI metric (more information in Section 5.1.2).

In **Paper E**, the results of the real-world experiments on Twitter and Wikipedia data were qualitatively evaluated using supplementary data. In the case of the **#FathersDay** and **#PokemonGO** hashtags, we investigated the text of the corresponding tweets. In the case of Wikipedia, we investigated the corresponding editor comments. For the synthetic experiments the evaluation metric was binary: Whether the synthetic process was generating Benford or non-Benford data at a given time.

4.6.2 Controlled Experiments

The controlled experiments in **Papers A,B,C and D** all follow a similar approach. Firstly, a set of *default* generative-model parameters was selected for generating synthetic data sets. The selection of the default parameters was discussed and justified in each case. Given these default parameters, we systematically performed experiments by varying each parameter individually over a range (keeping the other parameters fixed). This approach of designing and implementing a generative model, based on which systematic experiments are performed, is commonly used in the data-mining research community (e.g. [MV14; YM12; Mie09; Mie+08]). We note that it was not tractable to run experiments for *every* combination of generative-model parameters (indeed, many parameters were continuous real numbers with theoretically-infinite variability).

For each algorithm under consideration, as well as each generative-model parameter configuration, we generated 20 random data sets using our generative model tool. Each of the 20 data sets were provided to the algorithm under consideration along with any required algorithm parameters (discussed in Section 4.5). The algorithm was executed independently of all other experiments in each case. We collected 20 results corresponding to the algorithm's output on each data set. We evaluated each of the 20 results using the selected evaluation metric (Section 4.6.1), generating a set of 20 evaluation metric values. Finally, we calculated the mean and standard deviation for this set of 20 evaluation metric values. The mean (first moment) and standard deviation (positive square root of the second central moment) was then used for the plots in the corresponding report. To reduce clutter, we note that we often omitted standard-deviation bars for every second data point in a plot. An example of such a plot is seen in Figure 5.5.

In the case of **Paper B**, we included for reference an indication of the statistical significance of the *difference* between our results and that of the comparison. To this end, we used the conservative Wilcoxon signed-rank test [Wil45] to assess whether a significant difference existed between the mean evaluation-metric value for our proposed method and that of a selected comparison technique. The Wilcoxon signed-rank test is non-parametric (no assumption of normality is made). The resulting p-values for a comparison between FasTer and Asso$^{\text{MV}}$ were made available in Figure 7 of **Paper B**. In all but two cases the results were significant at the $\alpha = 0.05$ level. In the case of **Paper E**, our controlled experiments were evaluated visually by comparing the times at which BenFound and each comparison technique detected an anomaly or change-point.

4.6.3 Real-World Experiments

A suitable decomposition rank k needed to be selected in order to perform experiments on the real-world data for **Papers A and B**. Not knowing the "ground truth" model order, we used a heuristic approach. For the Stack Overflow data we ran FasTer for a range of k values. We then selected $k = 6$ based on the "kink" in the error-k curve, and used this value of k to compare to Asso and PaNDa. For the Africa and Congressional Voting Records data we selected k based on the number of "classes" published in the original data.

For **Paper C** the situation was similar. The decomposition ranks $k = 6$ and $k = 9$ were selected for the Yummly and IMDB data based on the "kink" in the error-k curve (Figure 5.13). For the ten ordinal survey-research data sets, we repeated the experiments for both $k = 10$ and $k = 20$. For **Paper D** we note that the data sets used were "classification-style" data sets, that is, each object included a class label. We thus used the class labels as the "ground truth" reference clustering result during the evaluation of each technique (a common approach adopted by the clustering community). Finally, for BenFound we selected a constant window size of $w = 2000$ for our real-world experiments on Twitter data (based on a trade-off between maximizing statistical power and the ability to promptly capture system dynamics). For the "red-flag" threshold α we selected the commonly-used value of 0.05.

4.6.4 Run-Time (Scalability) Experiments

For all run-time experiments presented, we consistently used the *default* generative-model parameters and systematically varied only the *scale* of the problem (in terms of the number of objects n, number of dimensions m and number of ground-truth patterns k). We note that many of the resulting instances required a number of days to "solve", so it was not tractable to perform 20 repetitions of each experiment. The metric recorded for each experiment was the algorithm's response time in seconds. The run-time experiments were all performed in isolated "jobs" on the compute cluster at Helmholtz Zentrum München. The compute architecture is based on IBM x3650 M3 nodes, each having two Intel Xeon X5690 6-core processors (3.46 GHz) and hyper-threading enabled, giving 24 virtual cores in total. The serial experiments were performed on a single virtual core. The experiments on parallelization in **Papers B and C** used up to 24 virtual cores for strong- and weak-scaling scenarios.

4.7 Reproducibility

In the respective online repositories we provide detailed instructions for reproducing all of our published results. Repository links are provided as footnotes in each publication. All data sets that we sourced ourselves, or that are not publicly-available, are also provided. In the case of **Paper E**, reproducibility is "built in" to the LaTeX report source itself using the *Knitr* approach [Xie15]. That is, compiling the report involves dynamically generating the plots, figures and tables using embedded R code. We advocate this transparent means of conducting research.

Algorithm name	**Source**	**Relevant Papers**
Asso	Pauli Miettinen mpi-inf.mpg.de/~pmiettin/src/DBP-progs/	**A,B,C**
Asso$^{\text{MV}}$	Pauli Miettinen (email request)	**A,B**
GreEss	Radim Belohlavek (email request)	**A,B,C**
PaNDa/PaNDa+	Claudio Lucchese hpc.isti.cnr.it/~claudio	**A,B,C**
NMF	Jean-Philippe Brunet portals.broadinstitute.org	**A,B,C**
MMMF	Jason Rennie (email request)	**A,B**
NMI (evaluation metric)	Nguyen Xuan Vinh sites.google.com/site/vinhnguyenx/publications	**D**
k-Means, Single Link, Complete Link Clustering	R *base* package	**D**
EM Clustering	R *EMCluster* package	**D**
Mclust-EM	R *Mclust* package	**D**
DBSCAN	R *dbscan* package	**D**
STSC	L. Zelnik-Manor and P. Perona vision.caltech.edu	**D**
DipMeans	Argyris Kalogeratos kalogeratos.com/psite/material/dip-means/	**D**
PgMeans	Greg Hamerly (email request)	**D**
RIC	Christian Böhm (email request)	**D**
Sync	Junming Shao (email request)	**D**
FOSSCLU	Sebastian Goebl (email request)	**D**
eAgglo, eDiv, cp3o	R *ecp* package	**E**
pDPA	R *cghseg* package	**E**
BinSeg	R *changepoint* package	**E**
Twitter Anomaly Detection	Twitter github.com/twitter/BreakoutDetection	**E**
Twitter Breakout Detection	Twitter github.com/twitter/AnomalyDetection	**E**
EGADS	Yahoo! https://github.com/yahoo/egads	**E**
Extreme Values	R *extremevalues* package	**E**
Twitter NLP	Alan Ritter github.com/aritter/twitter_nlp	**E**
Twitter Topic Detection (Streaming NMF)	Kohei Hayashi (email request)	**E**

Table 4.3: Sources for implementations of comparison algorithms

5 Results and Discussion

5.1 Summary of Findings

In the following two subsections we discuss our results from two viewpoints. First and foremost, we consider our overall results with respect to the challenges and goals that were laid out in Sections 1.3 and 1.4. Secondly, as some of the publications on which this dissertation is based were space-constrained to an extent, we elaborate for the sake of completeness on each publication's results and arguments separately.

5.1.1 Reflecting on our Challenges and Goals

The goals of this thesis were driven by the community-identified "top challenges" in data mining [YW06] (Sections 1.3 and 1.4). In following sub-sections we reflect on how we have addressed these challenges and goals through our work.

Challenge 1: Developing a Unified Theory of Data Mining

The first challenge was related to the arguably "ad-hoc" nature of data mining. Although conceding that the development of a *unified* theory of unsupervised, exploratory data mining would be difficult, we embraced the elements of what such a direction would entail with the motto *induce, deduce and reduce*. Our first goal in this thesis was thus to contribute to the *induction* of general frameworks, contribute to the *deduction* of further useful applications, contribute to the *reduction* of the number of assumptions made in data-mining approaches, and contribute to the *reduction* in the need for obscure parameters.

We have shown how this goal can be met for a practical set of unsupervised, exploratory data-mining tasks. Specifically, the *induction* of a general framework was successfully completed through the culmination of papers **Papers A, B and C**. The framework is named MDFS, and generalizes the problems of Boolean Matrix Factorization, Ternary Matrix Factorization and Ordinal Matrix Factorization. Each of these specialized problems is provably *reducible* to an instance of MDFS, and its algorithm Finesse was shown to outperform state-of-the-art techniques on many instances of the special cases in terms of both effectiveness and efficiency. Based on the general MDFS framework, we *deduced* additional applications of practical use (in particular, a feature based on tree objects or "itemsets over an ontology").

The *reduction* of commonly-made assumptions in data-mining approaches was demonstrated in **Papers D and E**. Specifically, the assumption that a multivariate distance measure is required when clustering vector data was questioned by our contribution SkinnyDip. SkinnyDip is a unique approach to clustering that exploits the *dip test* of unimodality on systematically-selected univariate projections of the data set, thereby not requiring a multivariate distance measure for clustering objects in a multidimensional vector space. Furthermore, the assumption that anomaly-detection techniques require the *absolute* values of the sample as an input was questioned by BenFound, which showed that anomalous behavior can be detected with just leading-digit (or mantissa) information.

Finally, a *reduction* in the need for "obscure" parameters was demonstrated by all of our methods. The FasTer and Finesse algorithms require the parameter k, however do not require a threshold for association confidence as it is used by Asso. The SkinnyDip algorithm requires a threshold α for statistical significance, however requires neither the number k of clusters, nor a density threshold, nor a strict assumption about the form of the clusters. The BenFound algorithm for anomaly-detection in time-series data requires a window-width w and threshold for statistical significance α, however requires no *parameterized* model from which the measurements are assumed to be taken (the Benford distribution involves no parameters).

Challenge 2: Scaling Up for High-Dimensional Data and High-Speed Data Streams

The second challenge was related to the curse of dimensionality, and additionally to the need for efficient and scalable algorithms. We subscribed to a set of "guiding principles" to address this challenge. Based on these guiding principles, our second goal in this thesis was to 1) consider mitigations for the curse of dimensionality in our work, 2) strive for algorithms that have a practically *linear* run-time complexity in the size of the data, and 3) use algorithmic paradigms that lend themselves well to parallelization.

Again we showed how this goal can be met for unsupervised, exploratory data-mining tasks. Specifically, we presented with SkinnyDip and SparseDip a novel mitigation for the curse of dimensionality in the context of clustering vector data. SparseDip searches for a subspace with coordinate directions that are maximally multimodal, which we argued to be a sensible choice from a clustering perspective. In this way, we are able to focus the actual SkinnyDip clustering on a low-dimensional view of the data, and help to mitigate the various negative effects otherwise faced when clustering in high-dimensional spaces (see Section 1.3.2).

The aim of developing linear-time algorithms was met in all of our work. FasTer, FINESSE, SKINNYDIP and BENFOUND each have a linear run-time complexity in the size of the input data (in the case of BENFOUND the input is the set of measurements made in one time window).

Finally, the ability to parallelize was demonstrated with our TMF algorithm FASTER and our MDFS algorithm FINESSE. We showed that we can achieve near-ideal speedup in weak- and strong-scaling scenarios up to 10 processors.

Challenge 3: Mining Time Series Data

The third challenge was related to the extraction of knowledge from temporal data with trends, seasonality and noise. Our third goal in this thesis was thus to develop approaches that could be deployed in *real-time* for *high-bandwidth* time-series data. More precisely, we posed the question of whether or not it was possible to define an *anomaly-detection* approach that focused, in a non-parametric way, on what is "*(un)natural*".

Again we showed how this goal can be met with a novel contribution. Specifically, we presented with BENFOUND an online, linear-time algorithm for detecting significant deviations from the "natural" state of a system. We identified in Benford's Law an intriguing notion for measuring the "authenticity" of signals from many application domains. By monitoring the conformity of the measured signal to the law, BENFOUND can hence raise a red flag when the signal deviates significantly in an "unnatural" way.

Challenge 4: Mining Complex Knowledge from Complex Data

The final challenge was related to "complex" data in the form of high noise, and heterogeneous features measured over fundamentally different scales. The final goal of our thesis was thus to advance the data-mining state-of-the-art by extracting *interpretable* knowledge from such complex data sets.

Again we showed how this goal can be met for unsupervised, exploratory data-mining tasks. Specifically, we presented with the Matrix Factorizations over Discrete Finite Sets (MDFS) framework an approach which supports interpretable Blind-Source Separation for data with features measured over *heterogeneous scales*. We showed how complex knowledge can be extracted using this framework and its linear-time algorithm FINESSE.

The aim of remaining robust to noise was well-met through our contributions. We presented systematic experiments for FASTER, FINESSE and SKINNYDIP in which the amount of noise was varied over a non-trivial range. In each case, our empirical evaluation showed that our algorithms stand up well to the state-of-the-art with respect to varying levels of noise. SKINNYDIP, in particular, was shown to be highly robust to noise. It is able to extract meaningful clusters in scenarios where the overwhelming majority (e.g. 80%) of data points belong to a global "clutter" distribution.

5.1.2 Elaboration on the Specific Results from Papers A to E

In following sub-sections we summarize and elaborate on the results of **Paper A** to **Paper E** separately.

Paper A: Ternary Matrix Factorization

TMF Shown to be Widely Applicable: We argued that measurements made on the nominal scale of ternary logic are often found in information systems. We listed three practical applications in our work. The **first practical application** corresponds to Boolean data with missing (lost or indeterminable) values, which we termed the Missing Value Boolean Matrix Factorization problem (MVBMF). Approaching this problem using TMF can help to serve two purposes: 1) finding latent descriptive patterns in the data, and 2) imputing the missing values ("filling in the blanks").

Missing values may arise in various ways [VS08]. The *Missing Completely at Random* (MCAR) case that we considered can be found when data collection is done improperly, mistakes are made in data entry, or data is damaged or corrupted.

A concrete example of TMF's application to MVBMF problems is found in an independent study by Phillips Research Europe and the University of Technology in Eindhoven [Vav+]. The authors referenced our work, investigating the TMF problem (as published in this paper) in the context of organizational Role-Based Access Control (RBAC) with missing values. In this context, "roles" need to be mined to help organizations that want to migrate to RBAC from low-level permission-based access systems. The full permissions database is seldom available directly [Vav+], so the set of permissions assigned to users is often obtained by analyzing the system actions they perform. These logs are typically incomplete, hence MVBMF. Importantly, it is typical that a user can take on multiple roles, hence the RBAC problem is best treated with logical mixing semantics (TMF) rather than the classical linear mixing semantics used by techniques from linear algebra. We discuss this study further in Section 5.3.

The **second practical application** of TMF relates to data in which the logical proposition of *unknown* has a meaning *other* than "missing". In **Paper A** we identified two such cases with wide applicability: explicit *Don't Know* responses in questionnaires [Poe+88; FB75], and *null* values in application databases [Zan82; Bis81].

TMF Shown to Subsume BMF: The **third practical application** of TMF relates to the fact that TMF subsumes BMF. Each instance of BMF consists of a data matrix $D \in \mathcal{B}^{n\times m}$ and a positive integer k. An instance of our introduced TMF problem consists of a data matrix $D \in \mathcal{T}^{n\times m}$ and a positive integer k. We presented the reduction from BMF to TMF considering that 1) $\mathcal{B} \subset \mathcal{T}$, 2) the truth table for ternary logic subsumes that of

Boolean logic, and 3) the set of possible TMF factor matrices is a superset of all possible BMF factor matrices. We showed therefore that the applications of TMF implicitly include all the applications of BMF (e.g. the students-courses example from Section 3.1, or the Role-Based Access Control problem *without* missing values [KSS03]).

FasTer Introduced to Solve TMF: We presented FasTer, a heuristic-based algorithm for approximating the solution to instances of the TMF problem in $\mathcal{O}(k^2nm)$ time. At a high level the approach taken by FasTer is a "leapfrog" one: it solves first for U whilst keeping B fixed, then solves for B whilst keeping U fixed. This process is iterated so long as the error continues to reduce. Termination is guaranteed because the TMF objective function only has a finite number of possible values.

FasTer is non-deterministic because it begins with a stochastic initialization of B. That is, a single row vector from D is selected at random to form the first basis vector in B, after which $k-1$ of the most "diverse" observations with respect to it are selected from D for the remaining initial basis patterns in B. We noted that rigorous arguments supporting this kind of initialization approach are given in [ÇM09; BMD09; TKB12].

We presented the pseudocode for FasTer and made an optimized, documented C++ implementation available for download and re-use. The implementation has been successfully used in one independent study to date [Vav+].

FasTer Shown to be More Effective than the State-Of-The-Art: Our empirical analysis of FasTer involved comparing it to the state-of-the-art algorithms for discrete matrix factorizations (Asso, Asso$^{\text{MV}}$, PaNDa and GreEss) on TMF, BMF and MVBMF problems. For reference, we also compared to the linear-algebra ("real-valued") techniques Singular Value Decomposition, Non-negative Matrix Factorization and Maximum-Margin Matrix Factorization (MMMF, see Section 3.2.2). Our generative model included parameters that enabled us to systematically vary the number of "ground truth" source patterns k, their density ($\rho_{\mathfrak{t}}$ and $\rho_{\mathfrak{u}}$), the density α of the matrix U the percentage η of randomly-injected noise. Inspired by the original work on BMF [Mie09], we selected sensible defaults for each of these parameters.

The results showed that FasTer outperformed Asso, Asso$^{\text{MV}}$, PaNDa and GreEss nearly consistently on these data. In the case of TMF, the quantitative improvement that FasTer offered over Asso and PaNDa was substantial. One reason for this is that Asso and PaNDa cannot directly solve TMF problems. That is, to compare to Asso and PaNDa, we first needed to encode the ternary data matrix into binary format. To achieve this, we mapped each ternary value to a binary triple: $\mathfrak{f} \mapsto (0,0,1)$, $\mathfrak{u} \mapsto (0,1,0)$, $\mathfrak{t} \mapsto (1,0,0)$, thereby tripling the matrix dimension m. Although at first seeming to be a sensible choice, this encoding is in fact unfair to both Asso and PaNDa. Specifically, it results in non-equivalence between the BMF and TMF optimization goals (see **Paper B**).

The TMF empirical analysis in **Paper A** therefore evaluated Asso and PaNDa using an objective function that was different to the one that Asso and PaNDa believed they were solving. A fair encoding was later detailed in papers **Papers B and C**. The experiments in **Papers B and C** in turn made use of this fair encoding (the results still showed that FasTer is superior on such data).

In the case of MVBMF, we observed that FasTer consistently outperformed Asso$^{\text{MV}}$ on data sets with up to 99% missing values. Like Asso, the Asso$^{\text{MV}}$ variant responds negatively to an increase in the density ρ_t of the "ground truth" basis patterns. As is generally the case, all algorithms produced poorer-quality results for an increasing decomposition rank k. Informally, the decompositions clearly become "more difficult" for increasing k. More formally, we see that the approximation factor increases with respect to k for the greedy algorithm that solves the ±PSC covering problem (see Section 3.2.1, noting that FasTer solves similar problems in a greedy way during each of its iterations).

In the case of BMF, we observed that Asso is the closest competitor and produced FasTer-similar results for varying "rank" k, noise η and usage-matrix density λ. Asso and PaNDa, however, both yielded poorer results when the density of the "ground truth" patterns increased, whereas FasTer remained relatively stable in this respect. In the case of Asso, the reason for this sub-optimal behavior is given in [Mie+08, p. 1354]: "If for all i so that $b_{pi} = 1$ there is q so that also $b_{qi} = 1$, then we cannot find row $b_{p\cdot}$ from A". In our context, as we increased the density of the "ground truth" patterns, the probability of basis vectors *sharing* t values for any given column increased. The Asso heuristic, based on computing the association confidences between columns (Section 3.2), is less effective in such cases.

The real-valued techniques SVD and NMF consistently outperformed all discrete techniques on the BMF problems with respect to the objective function (Equation (3.2)). This was expected, and echoes the results seen in [Mie09]. Even though SVD and NMF use the arithmetic mixing operators (normal addition and multiplication), they have a higher degree of freedom for selecting values in their factor matrices. In the case of SVD, the algorithm can also solve optimally with respect to its objective function. For the MVBMF experiments, the results from the MMMF method painted a similar picture: MMMF is likewise based on a real-valued decomposition and uses the classical matrix product. Of course, SVD, NMF and MMMF are not a fair comparison in our context because their factors have clear interpretation weaknesses in the context of the domain (see Section 3.1). The error with respect to the objective function therefore fails to tell the whole story.

To get a better idea of the whole story, we compared FasTer to Asso and PaNDa on three real-world ternary data sets. In each case, FasTer outperformed Asso and PaNDa in terms of the reproduction error. Additionally, we *qualitatively* analyzed FasTer's results in the context of the domain, offering intuitive interpretations for the discovered knowledge. Additionally, we quantitatively compared FasTer to Asso$^{\text{MV}}$ on ten real-world Boolean data sets (on which we injected up to 99% missing values). FasTer consistently outperformed Asso$^{\text{MV}}$ on these data.

Finally, we note that FasTer often outperformed NMF and SVD on the TMF problems. In this case, two arbitrary schemes were selected to map the ternary values in the data to real values for use with NMF and SVD. Clearly it is not possible to represent the results of the ternary truth tables using classical arithmetic operations on these values, so the results for both NMF and SVD can be attributed to their difficulties in trying to separate the ternary sources by using linear mixing mechanisms only.

FasTer Shown to be more *Efficient* than the State-Of-The-Art: Our theoretical time-complexity analysis of the FasTer algorithm showed that it has a worst-case run-time complexity in $\mathcal{O}(nmk^2)$. This result was based on the assumption that the number of high-level "refine-and-alternate" iterations is independent of n, m and k. We verified this assumption empirically. We also noted that the theoretical run-time dependency on k would in fact have been cubic, had we not exploited bitwise operations to reduce a critical calculation from $\mathcal{O}(k)$ to $\mathcal{O}(1)$ operations.

We executed experiments to compare the run-time of FasTer against the run-time of Asso, PaNDa and GreEss. These experiments confirmed the run-time growth of FasTer to be in $\mathcal{O}(nmk^2)$. Both Asso and PaNDa have linear run-time complexity in n, however have quadratic time complexity in m (in agreement with the respective theoretical analyses). In summary, FasTer's run-time grows linearly with the size of the data; that of Asso and PaNDa is super-linear. GreEss is the most expensive (quadratic growth in n, cubic growth in m). Finally, we presented results showing how the global (user-supplied) randomization-round-count affects the quality of the results on problems of varying size (n, m and k).

Paper B: Ternary Matrix Factorization: Problem Definitions and Algorithms

Approximation Factor Proven for TMF Sub-Problem Under Certain Conditions: A primary contribution of **Paper B** was the investigation of the ±PSC problem in the context of TMF. Specifically, we observed that the Ternary Usage Problem (TUP – the problem that solves for each row of U during any given FasTer iteration when B is fixed) exhibits similarities to the ±PSC problem. We posed the question: Can we reduce ±PSC to TUP? If this could be shown to be possible, we could express each TUP problem as a ±PSC problem and use the algorithm from [Mie08a] with a known approximation factor (see Section 3.2.1).

We proved in this paper that the answer depends on the form of our contrast function

$\oslash_{\mathcal{T}}$, which in turn depends on how the semantics of the ternary logical propositions are interpreted for a given application. For one concrete case, where the ternary propositions are understood to be ordinal in nature (like a Likert item) and the proposition of $\mathfrak{u}$ is "between" $\mathfrak{f}$ and $\mathfrak{t}$, the reduction succeeds. For the other common case, where the ternary propositions are understood to be equally different from one another, we provided a proof that a reduction is not possible (Theorem 3 in **Paper B**).

FasTer$_{\pm\text{PSC}}$ Introduced for Solving TMF: Based on the successful reduction from ±PSC to TUP in one case, we introduced FasTer$_{\pm\text{PSC}}$ as a FasTer variant. Instead of the original greedy heuristic, FasTer$_{\pm\text{PSC}}$ uses the ±PSC algorithm from [Mie08a] to solve each TUP instance. The provable approximation factor suggested that the effectiveness of FasTer$_{\pm\text{PSC}}$ would be greater than that of vanilla FasTer.

The use of the ±PSC algorithm, however, came at the cost of scalability. Specifically, our theoretical analysis of the FasTer$_{\pm\text{PSC}}$ run-time showed that its worst-case run-time is $\mathcal{O}(k^2nm(m+k)\log(m+k))$, compared to the worst-case run-time complexity of vanilla FasTer ($\mathcal{O}(k^2nm)$).

FasTer$_{\pm\text{PSC}}$ Shown to Outperform FasTer (Effectiveness) Under Certain Conditions: We showed that FasTer$_{\pm\text{PSC}}$ can give very competitive results, significantly outperforming vanilla FasTer, particularly for small-to-moderate values of k. For $k = 12, 14, 16$, for example, the reconstruction error was very close to the error attributable to the noise. Said differently, FasTer$_{\pm\text{PSC}}$ achieved near-optimal decompositions for the default parameter values.

Unfortunately, FasTer$_{\pm\text{PSC}}$ yielded to vanilla FasTer for large k and large $\rho_\mathfrak{t}$ (the density of *true* values in the "ground-truth" basis matrix B). This behavior was attributed to the use of the ±PSC algorithm, the approximation factor for which is known to grow with both k and $\rho_\mathfrak{t}$ (see Section 3.2.1).

Parallellizing FasTer Delivered Promising Speedup Results: Another primary contribution of **Paper B** was the implementation of a parallel version of FasTer. Using empirical evidence of "diminishing returns", we first argued that the use of parallelization for improving the TMF solution *accuracy* (through more initialization rounds) is inefficient. We also found memory usage to be of secondary concern (the storage of the problem data structures can exploit sparsity, and FasTer requires neither floating-point arithmetic nor Asso-style large intermediate storage). It was thus decided to exploit the shared-memory, multiprocessor programming OpenMP [Boa08] API and focus on strong- and weak-scaling scenarios.

After identifying the elements of FasTer that were good candidates for parallelization, we discussed the advantages and disadvantages of the various scheduling strategies *static*, *dynamic* and *guided*. We presented speedup and efficiency results in strong- and weak-scaling scenarios. To this end, both the processor (virtual core) count and scheduling strategy were varied. Speedup and efficiency was shown to be near-ideal for up to approximately eight processors. At 22 processors the speedup reached a maximum of approximately 13. Overall, we did not witness a large variation in the speedup and efficiency between the scheduling strategies. We did, however, witness *super-linear* speedup for one case of static scheduling. This result was not particularly surprising – we subsequently clarified the low-level cache-based mechanisms that can lead to such an effect when static scheduling is in place.

Paper C: Factorizing Complex Discrete Data "with Finesse"

BMF, TMF, OMF and Beyond (MDFS as a Unifying Framework): Perhaps the primary contribution of **Paper C** was the development of a general matrix factorization problem that subsumes BMF, TMF and OMF, and provides a formal structure for additional decompositions of this type (helping to address **Challenge 1**).

After reviewing the problem definitions of BMF, TMF and OMF, we highlighted their similarities and began pursuing a generalization. One question in particular remained a hurdle: what should be the structure and purpose of the *usage* matrix? At this stage it is warranted to explain the intuition behind our eventual decision to recommend a Boolean usage matrix for the general case.

In Figure 3.2 (Section 3.1) the interpretation of the BMF usage matrix is clear – a row indicates which basis patterns are "mixed" (disjunction) to explain the corresponding data-set observation. The OMF example from Figure 3.3 is also readily understood – the largest scale value (■) is the indicator value in this case.

In the seminal OMF article [BK13] it is implicitly argued that the OMF usage matrix (like that in Figure 3.3) should *not* restrict entries to the two extremes (in a binary fashion), but rather allow values from the full ordinal scale. The OMF multiplication operator $\otimes_{\mathcal{L}}$ is used to this end. For completeness we intended to support this kind of relaxation in our generalization, however it is worthwhile to briefly discuss it here from an interpretability perspective.

Consider a synthetic data set over the five-element ordinal scale □░▒▓■. Such a scale could correspond to the format of a typical Likert item in a questionnaire (e.g. □ = strongly disagree, ░ = disagree, ▒ = neutral, ▓ = agree, ■ = strongly agree). For illustrative purposes we assume the data set has n observations and four attributes. The four elements in a row represent a respondent's answers to four Likert items. We assume that the data set has an exact OMF decomposition of rank $k = 3$. The ordinal usage row vectors $u_{1\cdot}, \dots, u_{n\cdot}$ contain entries from the same five-element ordinal scale and prescribe the recipe for "mixing" the ordinal basis row vectors $b_{1\cdot}$, $b_{2\cdot}$ and $b_{3\cdot}$ to form each data set observation $d_{1\cdot}, \ldots, d_{n\cdot}$:

$$\begin{array}{rcccc} d_{1\cdot} & 1/4 & 1/4 & 2/4 & 3/4 \\ & & \vdots & & \\ d_{n\cdot} & 0 & 0 & 0 & 0 \end{array} = \begin{array}{rccc} u_{1\cdot} & 1 & 3/4 & 1/4 \\ & & \vdots & \\ u_{n\cdot} & 1/4 & 2/4 & 3/4 \end{array} \odot \begin{array}{ccccl} 0 & 1/4 & 2/4 & 3/4 & b_{1\cdot} \\ 2/4 & 1/4 & 2/4 & 0 & b_{2\cdot} \\ 1/4 & 0 & 1/4 & 1/4 & b_{3\cdot} \end{array}$$

Our focus here is the last data observation: how does the decomposition explain $d_{n\cdot}$? From the last usage vector $u_{n\cdot}$ we see it explained by "a lower agreement with $b_{1\cdot}$ mixed with a moderate agreement with $b_{2\cdot}$ and mixed with a higher agreement with $b_{3\cdot}$.". Based on the multiplication operator suggested in [BK13], specifically Łukasiewicz's strong conjunction connective $(a \otimes b) = \max(\square, a + b - \blacksquare)$, we *mathematically* understand the "strongly disagree" result ($d_{n\cdot} = \square\square\square\square$).

Although we stress that this is a pathological example, it does raise the question of whether or not this interplay of *two* non-linear operations (one for "mixing" and one for "weighting") is perhaps too esoteric or unintuitive for practitioners to grasp. In this work we hence focused primarily on the OMF variant which restricts the usage matrix entries to be binary. The ternary case (TMF) also works along these lines: the problem definition stipulates that the usage matrix should be binary, based on the justification that "unknown" propositions in the usage matrix may be difficult to comprehend (see **Paper A**). Finally, such a restriction also offered us a bonus: it enabled us to **handle heterogeneous data sets** (**Challenge 4**), which is another key advantage of our framework.

With all this in mind, the pieces were in place to introduce our framework, named Matrix Factorizations over Discrete Finite Sets (MDFS). We presented the formal requirements for the types of data that are supported by the framework. Specifically, a feature is supported by our framework if it satisfies our definition of being a "mixable" feature. A "mixable" feature must be one that is measured over a finite set of values and one that admits a particular algebraic structure. This algebraic structure must have a binary, closed "mixing" operator (the analog to arithmetical addition) and a contrast function for measuring (dis)similarity between members of the set. Finally, each feature must include an indicator for whether or not it supports an analog to multiplication (e.g. Boolean conjunction), provide that binary function if so, and also provide a mechanism for generating new candidate values during the search for patterns (in the trivial case this could enumerate all items of the corresponding set).

Additional Features Deduced for the MDFS Framework: MDFS is a unifying framework for Blind-Source Separation on "mixable" discrete data. Compared to BSS techniques from linear algebra, MDFS uses factorizations employing a specialized matrix product that respects the "mixing" semantics of each data type. MDFS subsumes BMF, TMF and OMF, which each use a specialized matrix product for their respective target data types.

Having induced a general framework, we were then able to ask the question: *Can we deduce further data types that satisfy our requirements for a "mixable" set?* A further contribution from this paper was the investigation of a new feature type that lends itself well to this kind of decomposition. The feature could be called "itemsets over an ontology", or simply a "tree" feature.

We find such features in many modern-day information systems. In our paper we presented the real-world example of recipe data from *yummly.com*. For any given recipe, Yummly measures information over various measurement scales. The "flavor" flags, for example, are Boolean, and the "rating" measurement is an ordinal one. Additionally, each recipe includes a set of *ingredients*, and these ingredients are nodes in Yummly's ingredients ontology (Figure 5.1). This is a concrete example of an "itemsets over an ontology" or "tree" feature.

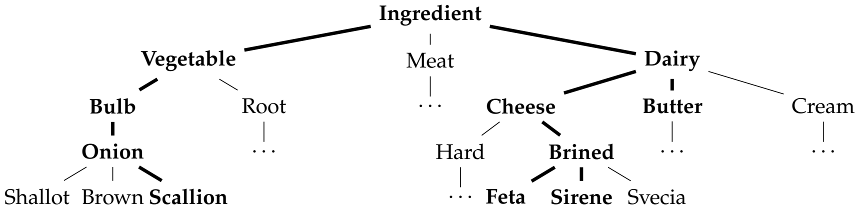


Figure 5.1: A basic culinary ingredients ontology. The **bold** sub-tree represents (the recipe with) the ingredients list scallions, feta, sirene and butter.

By factorizing this heterogeneous data matrix using MDFS, we were able to expose latent sources that led to useful information in the domain context. For example, the basis matrix included associations of the form: "Recipes like pies and quiches that include meat *and* dough products are often associated with a high rating and a long preparation time", or "recipes with chili are often associated with a piquant flavor".

Finesse Introduced to Solve MDFS: We presented Finesse, a heuristic-based algorithm for approximating the solution to instances of the MDFS problem in $\mathcal{O}(k^2 nm\,|\mathcal{F}_l|^2)$ time (where $\mathcal{F}_l$ is the discrete feature set having the largest cardinality). At a high level the algorithmic paradigm is similar to that taken by FasTer ("leapfrog" or "alternate and iterate"). However, Finesse no longer dictates a strict data type and strict mixing semantics. It handles features in an abstract way, enabling concrete extensions to additional data types that support our formal requirements for a "mixable" set. It thus supports input data sets with features measured over a variety of discrete measurements scales. To this end, the relevant software interfaces can be implemented. Our publicly-available implementation includes out-of-the-box support for Boolean, ternary, ordinal and tree features.

Finesse Shown to Find Intuitive Patterns on Large-Scale, Heterogeneous Data: In addition to the aforementioned Yummly example, we further illustrate FINESSE's real-world applicability, and in particular its *scalability*, through a novel investigation of the popular *imdb.com* (Internet Movie Database) movie data here. In particular, we are curious about film *shooting locations*, and ask how they associate with *genres*, *ratings* and a technical *widescreen* attribute.

Whilst geographical places (suburbs, political states, countries and so on) are related through a hierarchy, the IMDB data files only contain simple text strings for each film's shooting locations. *The Shawshank Redemption (1994)*, for example, is listed as having had one scene shot in *St. Croix, U.S. Virgin Islands*. To prepare the data for MDFS, we used Google's *Places* API[1] for extracting the hierarchical geographical components associated with these strings. We generated the 82646-node "filming locations ontology" from this information. Note that one could argue that the requirement of an ontology is a disadvantage of our technique, however it is also clear that many applications have the ontology for their data ready-made (e.g. Yummly uses an ingredients ontology for improving the user's search experience on their website).

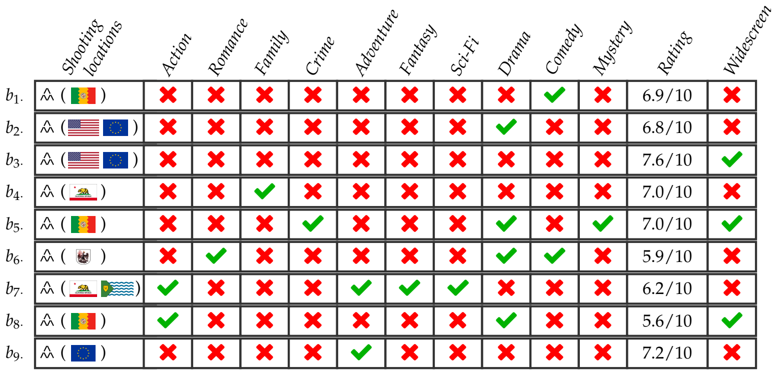
Each row vector in the IMDB data matrix represents a single movie over a number of features, the first being an ontology-itemset feature for the shooting locations. Analogously to the aforementioned ingredients example, a matrix value in the first column is a subtree of the 82646-node locations ontology (for example, the subtree encapsulating various neighborhood shooting locations in Paris and New York). A number of Boolean features follow, recording the presence or absence of *genre* tags like "Action" and "Romance". Rating information is next – here IMDB measures on a scale of 1.0 to 10.0 with increments of 0.1, so this feature is ordinal with $|\mathcal{L}| = 91$. Finally, a ternary feature is used for the *widescreen* attribute as it contains missing (*unknown*) values which we wish to impute. This heterogeneous data set (together with the filming locations ontology) is in the public repository for reuse.

The repository also contains the results for $3 \leq k \leq 16$. Figure 5.2 shows the rank-nine ($k = 9$) decomposition, again selected for interpretation because it corresponds to the "elbow" point on the error-k curve (Figure 5.13). This decomposition of the $n = 14690$ titles[2] for which rating, location and genre information was available required approximately two hours running on a single virtual core (Intel Xeon X5690 3.46 GHz). Despite the data set being larger than the Yummly example, the reconstruction error was lower (6%) due to the relative sparsity of the genre features. This IMDB example also

[1] developers.google.com/places

[2] Titles including "(TV)" were explicitly excluded.

illustrates that it is not uncommon to find large real-world ontologies (82646 nodes here) where Finesse's scalable approach is more tractable than the use of binary encoding and a BMF technique (simple encoding here would give $n \cdot m > 1 \times 10^9$, making techniques like Asso and PaNDa+ intractable for use).



	Shooting locations	*Action*	*Romance*	*Family*	*Crime*	*Adventure*	*Fantasy*	*Sci-Fi*	*Drama*	*Comedy*	*Mystery*	*Rating*	*Widescreen*
$b_{1\cdot}$	( )	✗	✗	✗	✗	✗	✗	✗	✗	✓	✗	6.9/10	✗
$b_{2\cdot}$	(🇺🇸 🇪🇺)	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗	6.8/10	✗
$b_{3\cdot}$	(🇺🇸 🇪🇺)	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	7.6/10	✓
$b_{4\cdot}$	( )	✗	✗	✓	✗	✗	✗	✗	✗	✗	✗	7.0/10	✗
$b_{5\cdot}$	( )	✗	✗	✗	✓	✗	✗	✗	✓	✗	✓	7.0/10	✓
$b_{6\cdot}$	( )	✗	✓	✗	✗	✗	✗	✗	✓	✓	✗	5.9/10	✗
$b_{7\cdot}$	( )	✓	✗	✗	✗	✓	✓	✓	✗	✗	✗	6.2/10	✗
$b_{8\cdot}$	( )	✓	✗	✗	✗	✗	✗	✗	✓	✗	✗	5.6/10	✓
$b_{9\cdot}$	(🇪🇺)	✗	✗	✗	✗	✓	✗	✗	✗	✗	✗	7.2/10	✗

Figure 5.2: The basis matrix corresponding to the rank nine ($k = 9$) decomposition of the IMDB data.

We offer a brief interpretation. The American motion-picture industry is unsurprisingly very present, with movie shooting locations often found directly in Los Angeles ( ) as well as in California ( ) and nationally in general (🇺🇸). Movies like *Mob City (2013)*, having elements of crime, drama and mystery and shooting locations in Los Angeles ($b_{5\cdot}$), have slightly better overall ratings than comedies like *Fired Up! (2009)* from that area ($b_{1\cdot}$). These in turn are generally associated with better ratings than action-drama films ($b_{8\cdot}$) like *Blue Thunder (1983)*. Perhaps owing to its panoply of geoclimatic regions, Canada and its cities (e.g. Vancouver) are identified in $b_{7\cdot}$ as frequently being the set for action-adventure films like *Journey to the Center of the Earth (2008)*. The intuitive latent association of Buenos Aires ( ) with *romance* is visible in $b_{6\cdot}$, mapped by the usage matrix to titles like *Verónica: El rostro del amor (1982)* and *El amor tiene cara de mujer (1964)*. Interestingly, widescreen films having shooting locations in America *and* Europe (🇪🇺) frequently enjoy higher ratings ($b_{3\cdot}$). *Earth Story (1998)* is such an example – this film was shot in widescreen but this information is not included in the IMBD database (Finesse was able to impute it using the ternary feature). Of course, many titles are explained by a *mixture* of patterns. *Napoléon (2002)*, for example, is an adventure-drama filmed in widescreen on the European and North-American continents with a rating of 7.3. It is explained in the usage matrix by $b_{2\cdot} \oplus b_{3\cdot} \oplus b_{9\cdot}$. Finally we note that the rating distribution is in line with the average film rating of 6.9 on IMDB.

Finesse Shown to be More *Effective* Than the State-Of-The-Art: Our results showed that Finesse outperforms the state-of-the-art techniques Asso, GreEss and PaNDa+ (note that PaNDa+ [LOP14] is an improvement over the PaNDa algorithm investigated in **Papers A and B**) almost consistently.

Our first focus in this paper was on ordinal data sets, like those found in survey research. Again we used a generative model inspired by real-world data. Specifically, we harvested and analyzed the distribution of entries in ten studies with Likert items (See Table 4.1). Analysis of these questionnaires showed that they had an approximately-uniform distribution of response values, so we designed our generative model such that the default parameters generated data matrices with an approximately uniform distribution of response values.

Varying each parameter systematically, we found that Finesse outperformed both Asso, GreEss and PaNDa+ in terms of reconstruction error. To compare to Asso and PaNDa+, we encoded the ordinal data in Boolean form. We again used an encoding scheme that ensured equivalence between the objective functions being solved by each algorithm. GreEss was the only other technique that could work with an ordinal data matrix directly, so no encoding was required. Unfortunately, however, it was only tractable to compare to the computationally-intensive GreEss on small data (e.g. $n = m = 50$).

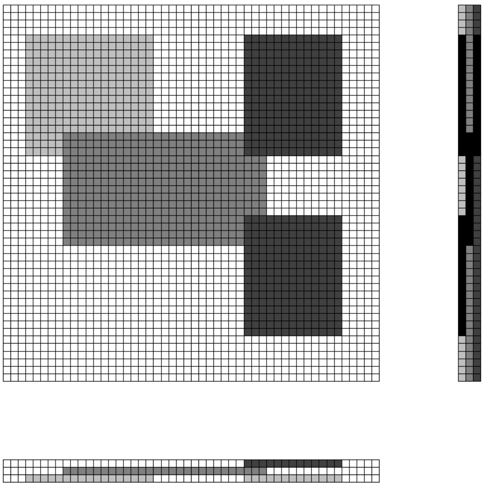


Figure 5.3: GreEss achieves the optimal decomposition on this synthetic, no-noise ordinal data. The basis matrix is shown below, and the usage matrix to the right. It correctly identifies the three ground-truth patterns ($k = 3$).

Although GreEss yields near-optimal results for the case of zero noise (and thus outperformed Finesse), our results showed that its performance significantly degraded as realistic levels of noise are added (it yields to Finesse at less than 10% noise). To understand why, we consider the GreEss decomposition for a simplified ordinal data set with $\eta = 0\%$ (no noise) in Figure 5.3.

We see from Figure 5.3 that the decomposition is exact. Without the need for a parameter k, GreEss impressively finds the three latent patterns (bottom), various mixtures of which (right) precisely reproduce each row in the data set. We now make the problem more realistic by lightly "salting" our data matrix: one element from each of the 42 non-zero columns is set to zero (less than 2% noise). GreEss' decomposition after this change is in Figure 5.4 (this time displayed canonically).

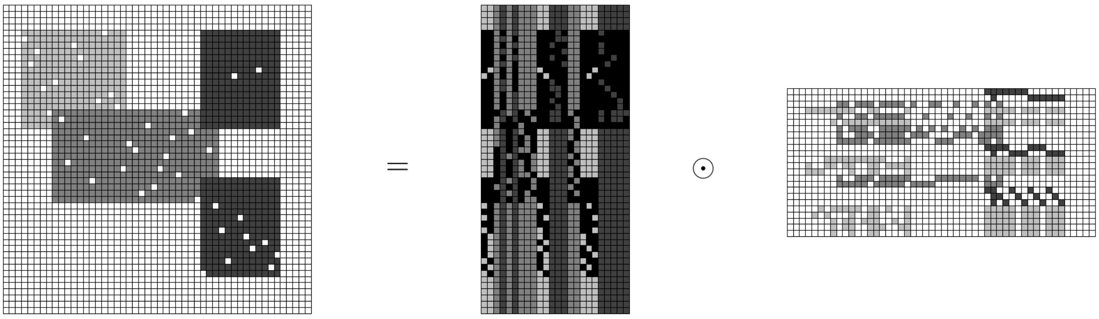


Figure 5.4: GreEss requires 24 basis vectors to exactly reconstruct this synthetic, noisy data set.

We see from Figure 5.4 that GreEss now needs to generate 24 basis vectors to exactly reconstruct the data set. The three ground-truth basis vectors found in the zero-noise case (Figure 5.3) are now seen fragmented. The reason for this is that GreEss precisely models the signal *and* the noise, using a *from-below* approach where over-coverage is prohibited [BK13]. This practically means that small quantities of "salt" noise are sufficient to cause the fragmentation of the true signal in the decomposition. Asso, Finesse and PaNDa+ are more liberal in this respect, tolerating patterns which may somewhat over- or under-cover the data in the hope of *separating* the signal from the noise. Such examples show that this balanced approach, the basis of which is formulated in the Positive-Negative Partial Set-Cover problem [Mie08a], is more appropriate for realistic levels of noise.

We also compared Finesse to both Asso and PaNDa+ on the ten aforementioned real-world survey data sets. Again, we were unable to compare to GreEss here due to the size of the data. The "ground-truth" model-order k was not known for these data, so we ran experiments for both $k = 10, 20$. In all cases (20 experiments in total), Finesse outperformed both Asso and PaNDa+.

Finally, we also repeated the BMF experiments with the new PaNDa+ algorithm. Compared to the experiments in **Papers A and B**, these experiments used a different generative model. Specifically, the default parameters generated a data matrix with an equal distribution of Boolean $\mathfrak{f}$ and $\mathfrak{t}$ values. The results are shown in Figure 5.5. Finesse outperformed both Asso and PaNDa+ in all cases.

Finesse More *Efficient* Than the State-Of-The-Art: The theoretical worst-case run-time complexity of Finesse is linear in the size of the data, whereas Asso and PaNDa+ (like PaNDa) are both quadratic in the number m of features. GreEss is yet more complex: its run-time is quadratic in the number n of objects and cubic in the number m of features. This result was confirmed in our practical experiments on run-time/scalability.

In addition to our Java version, we investigated two performance-optimizations for Finesse. The first step was to rewrite the algorithm in C++ and utilize SSE4 intrinsics (Streaming SIMD Extensions 4) for vectorization. Specifically, Finesse can exploit bitwise `AND`, bitwise `OR`, bitwise `XOR` and `POPCNT`[3]. Recent advances in SIMD instruction sets on general-purpose CPUs process up to 512 bits at a time in this fashion (e.g. AVX-2 and AVX-512), significantly reducing Finesse's absolute response time on a single CPU. Our run-time experiments included "proof of concept" results for the SSE4 (128-bit) case, which showed that the absolute response time of the optimized C++ version was approximately $\frac{1}{20}$ that of the Java version. This version outperformed the PaNDa+ algorithm, which has likewise been optimized in C++. The second optimization of Finesse – the use of OpenMP for parallelization of Finesse's inner loops on ten processors – yielded in a further speedup factor between 8 and 9 (analogous to the results seen for FasTer parallelization in **Paper B**).

Paper D: Skinny-dip: Clustering in a Sea of Noise

Motivation: Data Sets with "Extreme" Clutter Expose Significant Limitations in Existing Clustering Paradigms: We considered as a case-study the synthetic two-dimensional data set in Figure 5.6. 80% of the objects are sampled from a global, uniform "clutter" distribution. In the figure, the raw data is accompanied by attempts to cluster it using the popular Expectation-Maximization (EM), k-Means and Mclust-EM algorithms. The quality of their results leaves much to be desired, particularly their treatment of the global "clutter" or "noise". EM and k-Means are examples of well-known concrete algorithms in the partition-based clustering paradigm. In this paradigm, the separation of large volumes of clutter is often not a fundamental consideration. Centroid-based

[3] "Population count" – efficient calculation of the number of 1s in a bit string.

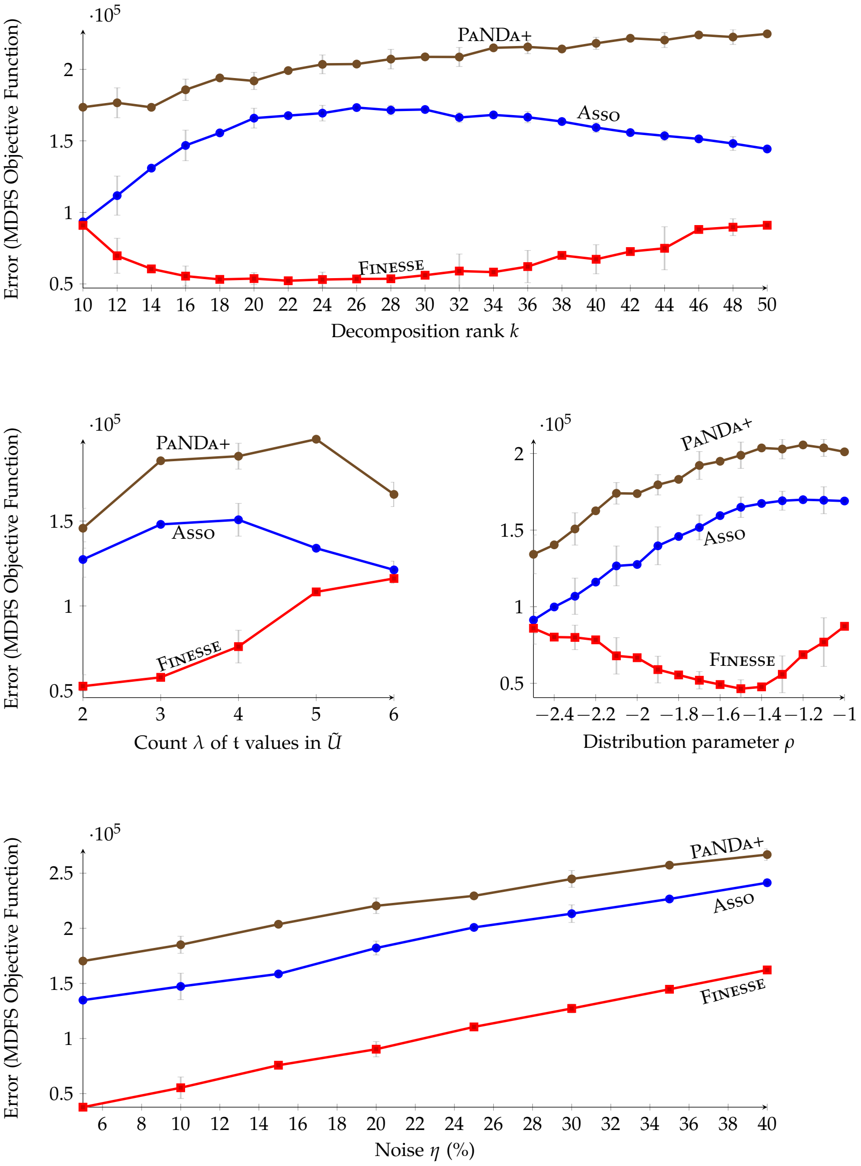


Figure 5.5: Boolean Matrix Factorization for synthetic data with varying data-generation parameters k, λ, ρ and η.

techniques often fail to consider noise whatsoever, dictating that each point be assigned to its nearest centroid.

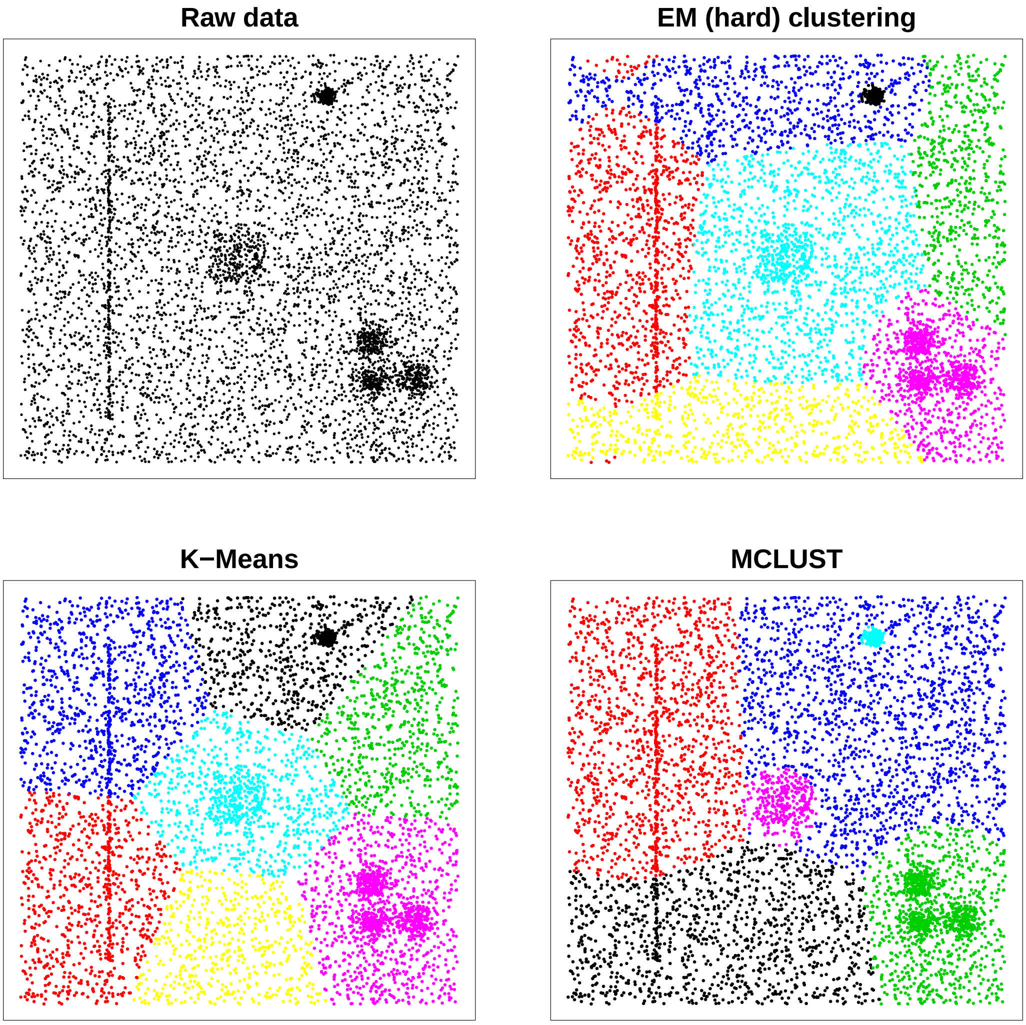


Figure 5.6: Results from further clustering techniques on the "running example" data in **Paper D**. MCLUST is the algorithm used in [DR98] (discussed in Section 3.4.1).

In **Paper D** we demonstrated analogous results using more specialized methods from other clustering paradigms, like the density- and spectral-based DBSCAN and Self-Tuning Spectral Clustering [ZP05] algorithms. In addition to our "running example", we further motivated our work by showing *real-world* data sets which exhibit the characteristics on which we focus: high clutter and heterogeneous cluster shapes.

Formal "Dip Test" Identified as a Useful Tool for Data Mining: Given the sub-optimal results from existing clustering techniques on such data, we considered alternative approaches to the problem. We investigated the intuitive notion that equates clusters with the modes, or "modal regions", of a multivariate distribution. Intuitively, this concept was a strong fit: modes are generally invariant to globally-added "clutter", and can have a range of spatial extents. We noted that the idea of "hunting" for the modes of a multivariate distribution had already been studied in the statistical literature [BP09; Ooi12], however had received somewhat less attention in the data-mining community. Specifically, "mode-hunting" clustering techniques had often been based on computing non-parametric density estimates of the data using methods like Kernel Density-Estimation (KDE) [Ros+56; Par62]. In [LRL07] it is explained that it is computationally-expensive to perform clustering based on KDE (quadratic-time in the data set size n) and that KDE additionally requires the tuning of a "bandwidth" parameter.

In Hartigan's *dip test* we identified existing statistical work that enabled the detection of modal regions in a non-parametric, linear-time and parameter-free way. We thus began our novel contributions by investigating the dip in the context of extracting heterogeneous clusters in cluttered environments. The *dip* is a statistic that measures the *departure from unimodality* of a given (univariate) empirical sample [HH85]. The dip *test* is a statistical hypothesis test that uses the *dip* to address the null hypothesis "the given data was sampled from a unimodal distribution". In addition to helping reject or accept this null hypothesis, the dip test yields a further piece of useful information from its calculations, namely the *interval* corresponding to the most primary mode. We observed that the test has a list of advantages that mirror much of what we wish to see in a general clustering technique. These include 1) no strict assumptions about the form of a "cluster" (i.e. non-parametric), 2) no required algorithm parameters, 3) highly robust to global clutter and outliers, 4) deterministic, 5) shift- and scale-invariant, and 6) linear run-time complexity in the size of the sample.

We also noted, however, that the test has two fundamental limitations: 1) it is applicable to univariate samples only, and 2) it identifies only one (primary) modal interval. Despite its limitations, we argued that its advantages make a dip-based clustering approach worthwhile pursuing. The remainder of our contributions in this work were based on overcoming these limitations to produce a general clustering technique.

UniDip Introduced to Clutter Univariate Data: We took the first step in the direction of our goal by introducing UniDip, a recursive dip-based algorithm for clustering univariate data. The algorithm operates by exploiting the two fundamental questions answered "out-of-the-box" by the dip, namely: 1) *Is the distribution from which the sample was taken multimodal?*, and 2) *Where is the interval corresponding to the most primary mode?*

Based only on this information, UniDip follows a recursive "divide-and-conquer" approach to extract the locations of all modal intervals in a univariate sample. It is non-parametric and requires no algorithm parameters (other than a threshold α for statistical significance, which is set to 0.05 by default). In Figure 5.7 we see a demonstration of UniDip's ability to extract modal intervals formed from various distribution types. Its result on a busier sample is included in **Paper D**.

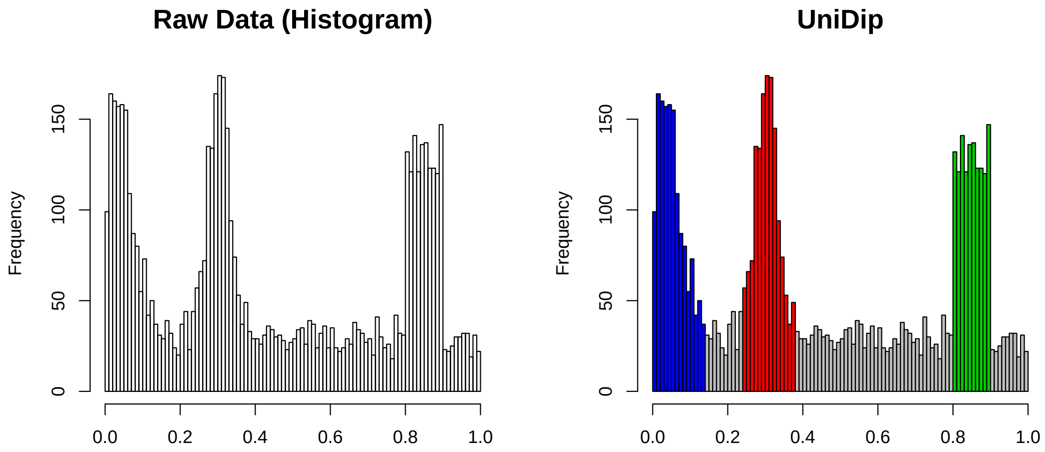


Figure 5.7: The UniDip result on a univariate sample (from left: a Beta distribution, a Gaussian distribution, a uniform distribution, and 50% "global" clutter over the interval $[0, 1]$).

UniDip inherits all of the core advantages of the dip mentioned in the previous section. It is a standalone technique, and can be used for finding all the modes of a continuous univariate distribution. To the best of our knowledge, this non-parametric, noise-robust, linear-time and parameter-free technique for finding such intervals is a novel contribution.

SkinnyDip Algorithm Presented and its Properties Investigated: UniDip solves our need to identify all modal intervals in a sample. However, it only does so for univariate data. The next step in the direction of our goal was to extend this approach to the multivariate case. The result, SkinnyDip, is a recursive dip-based algorithm for clustering continuous *multivariate* data. At a high level, one can describe SkinnyDip as a technique that recursively partitions the data, dimension by dimension, based only on the results of the univariate application of the *dip*. To do this, SkinnyDip applies UniDip on filtered univariate projections of the data in a systematic way. SkinnyDip yields "modal hyperintervals" (e.g. rectangles in two dimensions) corresponding to each detected cluster. Data falling within one of these areas is assigned the corresponding cluster label. Other data is labeled as noise/clutter.

Compared to other clustering techniques, SkinnyDip has a unique combination of properties. Its practical run-time is linear in the number of objects and dimensions. It requires no parameters, other than a threshold for statistical significance (in our work we always use the standard value of $\alpha = 0.05$). It is deterministic and highly robust to clutter and outliers. Finally, and perhaps most interestingly, it requires no multivariate measure of distance on the space. This allows SkinnyDip to extract clusters with different spatial extents, like the long, thin rectangle in Figure 5.6. To the best of our knowledge, SkinnyDip is the first general clustering technique for vector data that does not perform multivariate distance computations between objects in the data. That is, in line with our motto in Section 1.3.1, it curiously questions a fundamental assumption made by existing clustering approaches, namely that a multivariate measure for the distance between objects is required on the space.

SparseDip Algorithm Presented for Optimal-Subspace Pursuit: We presented SparseDip as an additional layer on top of SkinnyDip. The goal of SparseDip is to enable clustering on high-dimensional spatial data sets by searching for a subspace in which SkinnyDip can be applied. SparseDip does this by searching for directions in the data that are *maximally multimodal* with respect to the dip statistic. Considering that multimodality implies separation, we argued that focusing on such directions is a useful heuristic for the task of clustering.

By exploiting results relating to the continuity of the dip [KL05], we were able to make use of gradient ascent in the search for these maximum-dip directions. Unfortunately, the relevant function is non-convex, containing many local optima. For this reason, the choice of the starting point for gradient ascent became an important factor.

To select candidate starting points for gradient ascent, we evaluated two options. The first, a naïve approach, involved simple random-sampling of candidate starting points. The second involved using the nodes of a sparse grid. A sparse grid is an efficient data structure that can be used to represent functions in high-dimensional spaces. Sparse grids are often used in the field of numerical simulation in order to make tractable the computation of approximate solutions to partial differential equations (PDEs) in high dimensions [BG04]. In our case the end-goal was not to solve a PDE, nor to accurately represent or interpolate the function. We decided to evaluate sparse grids simply as a mechanism to find candidate starting points for gradient ascent. We empirically compared the sampling from sparse grids with simple random sampling and found that the sparse grids approach worked more favorably on a number of optimization benchmark functions (one example is shown in Figure 5.8). Additionally, the use of sparse grids led to higher-quality results on all ten of our real-world data sets.

SparseDip builds an orthogonal basis using the maximum-dip directions that are found. The search is terminated when no more significant directions are found (with respect to a significance threshold α). The projection of the data onto this basis is then passed to SkinnyDip, which performs the clustering in that subspace.

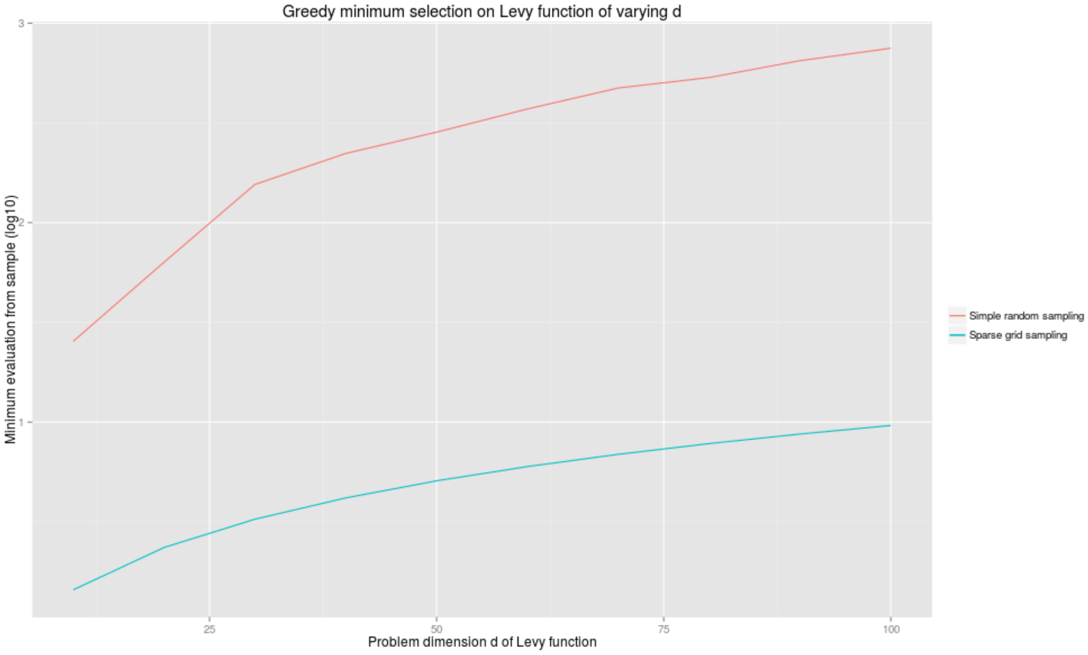


Figure 5.8: Using a constant sample size of the Levy function, the curves show the minimum value found by varying dimensionality when using 1) simple random sampling and 2) sparse-grid-based sampling.

SkinnyDip Shown to be Robust Against "Extreme" Levels of Noise and Clutter: We presented results showing that SkinnyDip is able to extract intuitive clusters embedded in high levels of noise and clutter (see Figure 5.9). These experiments suggested that the performance of techniques from the remaining clustering paradigms was degraded at this level of noise. In addition to experiments on *synthetic* data with varying amounts of clutter, we presented a *real-world* case study using the North Jutland (Denmark) Road

Network data from the UCI Machine-Learning Repository. This data is highly noisy, with insignificant road-segments spaced more or less evenly through the countryside. On this data we showed that SkinnyDip is able to extract intuitive clusters corresponding to high-settlement areas (cities). These clusters have by no means a Gaussian spatial form. The spatial form of the city of Frederikshavn, for example, is constrained by its proximity to the coast. SkinnyDip's properties of being non-parametric and not relying on a particular measure of distance in the multivariate space was of noticeable advantage here.

SkinnyDip Shown to Outperform the State-Of-The-Art in Terms of Effectiveness: We empirically evaluated SkinnyDip against six state-of-the-art automated clustering algorithms from different paradigms (partition-based, density-based, spectral-based). The first part of our evaluation was completed in a controlled way on synthetically-generated data. The model we selected for synthetically-generating data was based on the complexities we were attempting to address: high levels of clutter and heterogeneous (albeit still convex) cluster shapes (**Challenge 4**). The generative model included parameters that gave us control over 1) the number of clusters, 2) the number of dimensions, and 3) the level of clutter.

Using Adjusted Mutual Information as our evaluation metric, we found that SkinnyDip was able to outperform all comparison techniques consistently on our generated data when working with problems of dimensionality up to $m = 8$. Above this level, we found that the SkinnyDip result was not useful. Indeed, as the dimensionality of the problem increased, the limitations of SkinnyDip's "dimension by dimension" heuristic became significant. However, we argued that it is uncommon to find a real-world clustering that exists in a non-redundant way in this many dimensions. We suggested the use of SparseDip as a pre-processing option for practical data sets of non-trivial dimensionality. As discussed in Section 4.6.1, we additionally evaluated a portion of the clustering results using the Normalized Mutual Information metric, which suggested that there is little deviation between the two measurements (Figure 5.9).

We further evaluated SkinnyDip on ten real-world data sets from public repositories (see Section 4.4). The data varied in size and dimensionality (up to $n = 7494$ and $m = 16$). In seven cases out of ten, SkinnyDip outperformed the suite of comparison techniques. In two of the remaining cases, it ranked second. Additionally, the use of SparseDip identified a number of interpretable and useful "maximally-modal" directions in the data that yielded intuitive insights into the domain.

SkinnyDip Shown to Outperform the State-Of-The-Art in Terms of Efficiency: As a full-space clustering technique, we showed that SkinnyDip's theoretical worst-case runtime complexity is $\mathcal{O}(n \cdot \log(n) \cdot m + n \cdot m \cdot k)$, where n is the number of data objects,

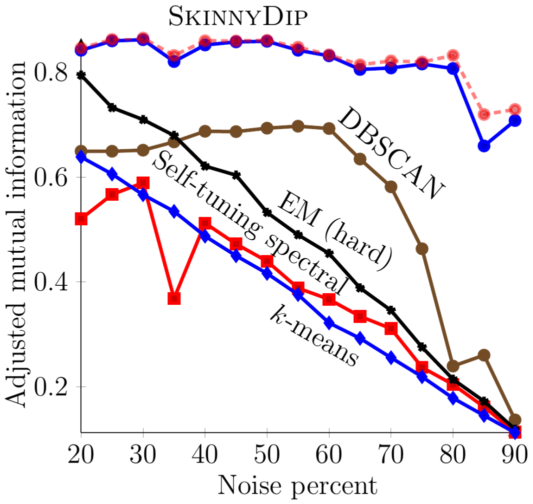


Figure 5.9: A reproduction of Figure 2 in **Paper D**, with the addition of SkinnyDip's Normalized Mutual Information values (the NMI is the top-most dotted red line series, seen slightly above the blue AMI measurement for SkinnyDip).

m is the dimension of the space, and k is the maximum number of clusters that is found from a call to UniDip (by definition not greater than the number of clusters found in total). The logarithmic factor is related only to the requirement of sorting the data (computation of the *dip* requires a sorted sample). Using aggressively-optimized R sorting routines, we were unable to detect this logarithmic factor. Practically, our run-time experiments even showed that SkinnyDip scales *sub-linearly* with the size of the data, which is attributable to the fact that data objects cease to be considered by SkinnyDip's recursive calls after they have been labeled as noise. Practically then, linear complexity is a conservative statement about SkinnyDip's run-time. As a comparison, the standard implementation of DBSCAN is quadratic[4] in the number of objects n.

In terms of *absolute* response time, SkinnyDip outperformed all six comparison techniques. The closest competitor was DBSCAN. The slowest of the competition was Self-Tuning Spectral Clustering; the computation of its locally-scaled affinity matrix is very expensive and we were not able to complete all of its run-time experiments within a reasonable time frame.

[4] Optimized implementations of DBSCAN can reduce this factor to $n \cdot \log(n)$.

Paper E: Let's see your Digits: Anomalous-State Detection using Benford's Law

Benford's Law Holds for Data From Many Online Services: We showed empirically that Benford's Law (BL) holds for data from the online services Twitter, Wikipedia, YouTube and GitHub. Benford's original manuscript [Ben38] had already shown empirically that many real-life sets of numerical data obey the law. These data included population numbers, lengths of rivers and a list of physical and mathematical constants. Since the original manuscript, it has also become well-known that BL holds for economic data [Nig99; DHP04]. To the best of our knowledge, however, our work is the first to investigate conformance for Wikipedia, YouTube and GitHub data (conformity of Twitter data was already shown in [Gol15]). We argued that this is a convenient property, particularly because data from such online services is often considered "noisy", and because the Benford distribution requires no parameterization.

Benford's Law is a "Natural" Tool for Anomaly-Detection in Time-Series Data: We reflected on Benford's original manuscript [Ben38] which argued that, despite mathematicians having named the sequence $1, 2, 3, \ldots$ the "natural" numbers, *Nature* is found to count $e^0, e^x, e^{2x}, e^{3x}, \ldots$ much more often. Numerous examples of natural and man-made processes that follow geometric or logarithmic progressions are given in Benford's manuscript. Given a set of numbers that was generated by such an underlying process and that span a large range, random samples over that set will yield a collection of numbers that follow Benford's law. We argued that conformity to BL is therefore a sensible measure for its "naturalness", and that deviations from this conformity can be understood as resulting from some "unnatural" influence. We further argued that significant deviations from the Benfordness property in a window of data that is *streamed* with high-bandwidth in real-time (**Challenge 2**) is an intuitive indication for "unnatural" or "anomalous" system behavior, and that this mechanism may prove useful as a "red-flagging" approach for many applications.

Statistical Hypothesis Test Presented for Benford's-Law Conformity: In order to objectively evaluate the BL-conformity of arbitrary univariate numerical samples, we presented a statistical hypothesis test based on the formal Kolmogorov-Smirnov one-sample test. The test exploits the fact that the mantissae of a Benford set are uniformly distributed in $[0, 1)$ [NW12].

We compared our hypothesis test with seven other state-of-the-art BL-conformity tests. These tests were either based on testing single digits at once, testing all digits at once, or investigating the mantissae. We used two ideal Benford sequences of the same length for evaluating the tests. Six of the seven tests, based on considering the *discrete* digit distributions, failed to conclude that the second sequence was Benford due to identified discretization errors. The seventh test, the so-called Mantissa Arc test, correctly identified both sequences as Benford. We showed using a simple example, however, that the Mantissa Arc test is not sufficient: it concludes Benfordness on a numerical sample that is clearly not Benford. Our test was thus used to form the basis for our anomaly-detection technique BenFound.

BenFound Introduced as a Novel Anomaly-Detection Technique for Time-Series Data: We presented BenFound as a technically-uncomplicated anomaly-detection technique that can be used to monitor "Benford-style" data, that is, data generated by multiplicative or exponential growth phenomena. We explained how BenFound only depends on the choice of a window size w and a statistical threshold α, and discussed how these values can sensibly be set. To the best of our knowledge, this is a novel approach to the general task of anomaly-detection in time-series data. All of the other approaches that we know of consider the behavior and evolution of the *absolute* values of the measurements, or statistics or moments based directly thereon. We are not aware of a general anomaly-detection technique for time-series data that considers only the leading digits and/or the mantissae.

BenFound Shown to Detect "Unnatural" Behavior in Online Services: We presented three case studies showing how BenFound can detect "unnatural" behavior in online services. The first case-study was related to Wikipedia data. We investigated a large set of change-size deltas (in bytes) for page-edit events that were labeled by Wikipedia as "non-bot". Analyzing the leading two-digit distribution exposed significant "spikes" which, upon further analysis, were found to corresponded to bot behavior (autonomous agents that were mass-editing Wikipedia content). Wikipedia's labeling system had failed to label the edits as "bot" edits.

The second case-study involved listening to the Twitter tweets against the **#FathersDay** hashtag. Approximately one week before Father's Day, the Benfordness of the signal was significantly rejected. It turns out that the hashtag had been "hijacked" by a number of spam and advertising accounts "unnaturally" trying to generate a profit from the event. Finally, our third case-study involved listenting to the Twitter tweets against the **#PokemonGO** hashtag. A sharp drop was seen on July 16 and corresponded to unnatural tweet behavior because of a confirmed denial-of-service attack on the Pokemon Go game servers.

BenFound Shown to Outperform Ten State-Of-The-Art Change-Point and Anomaly-Detection Techniques on Synthetic Data: We generated synthetic time-series data and performed an experiment comparing BenFound to ten state-of-the-art anomaly-detection techniques from the statistics and data-mining literature. The synthetic data simulated a change in the underlying process from Benford to non-Benford, and then its regression from non-Benford to Benford thereafter. BenFound was the only technique able to identify both changes to the underlying generative process. Many of the other techniques generated an indigestible number of false positives.

5.2 Implications for Research

This thesis has three main implications for research:

Implication 1: Knowledge can be extracted in the complex context of incomplete, heterogeneous and discrete data measured over non-ratio scales.

Our contributions imply that Blind-Source Separation can be performed over heterogeneous data sets containing certain kinds of discrete data types (**Challenge 4**). Through the introduction of the Ternary Matrix Factorization (TMF) problem, for example, we have provided a basis for performing Blind-Source Separation on data measured on the scale of ternary logic. This logical structure additionally implies a useful perspective on the problem of incomplete Boolean measurements (MVBMF) and has applications in collaborative filtering and the mining of access roles.

With the Matrix Factorizations over Discrete Finite Sets (MDFS) framework, we extend this concept to arbitrary discrete finite sets having a sensible notion of "mixing". The MDFS framework implies progress towards a unified theory of data mining (**Challenge 1**) because it subsumes a number of existing data mining techniques (BMF, TMF and OMF). The MDFS framework also implies that researchers can deduce additional applications by identifying further features with applicability.

Implication 2: Knowledge can be found despite the complexities of "extreme" global clutter and high dimensionality.

Our contributions imply that object-groupings (clusters) can be found despite their presence in a sea of up to 90% noise (**Challenge 4**) and despite their location in an arbitrarily-oriented, low-dimensional subspace (**Challenge 2**). Through the introduction of our clustering algorithm SkinnyDip and the subspace-search algorithm SparseDip, we have introduced a novel take on the problem of extracting knowledge through clustering. The SkinnyDip result implies, for example, that a multivariate distance or "similarity" measure is *not* a strict requirement for a vector-based clustering method. It also implies that the relatively-unknown *dip* test is an effective tool for data mining – it can efficiently and non-parametrically quantify a notion of "separation" in continuous univariate data, *and* locate the primary "modal" region in such data. It does this in linear-time without parameters.

Implication 3: Knowledge can be extracted from the complex context of high-bandwidth time-series data without needing a parameterized model.

Our contributions imply an intriguing new approach for analyzing high-bandwidth numerical time-series data (**Challenge 3**). The approach does not involve the use of a parameterized model for explaining the distribution of the *absolute* measurement values. Specifically, our anomaly-detection approach BenFound implies that the availability of the *absolute* values of the time-series data measurements is not strictly necessary. By monitoring the conformance of the *leading digits* or *mantissae* to Benford's Law, we obtain an elegant mechanism for determining whether a system has deviated from its natural state. Our experiments on real-world data imply that this approach is by no means limited to data from social-media streams like Twitter. Indeed, our results on Wikipedia, GitHub and YouTube data suggest that many systems measure phenomena that obey Benford's Law. To the best of our knowledge, Benford's Law has not yet been investigated or exploited by the data-mining community.

5.3 Implications for Practice

Practitioners will benefit from the results of this thesis in the following four ways.

Implication 1: Our proposed techniques outperform the state-of-the-art with respect to solution quality on many practical problems.

The effectiveness of a given algorithm (with respect to a sensible objective function) is a key consideration for practitioners looking for high-quality solutions. For the complex problems we investigate, it is of course difficult to compare the quantitative performance between algorithms over the *complete* set of problem instances. For this reason, we took care in the design of our empirical evaluation to ensure that our generative models reflected the kind of data seen in practice. On such data, we showed that our algorithms outperformed the state-of-the-art in the majority of cases. For our work on discrete matrix factorizations, for example, one generative model was inspired by patterns found in survey research. For our work on clustering, the generative model was inspired by the levels of clutter often seen in minefield-detection and the clustering of galaxies. For our work on anomaly-detection in time series data, our generative model was inspired by dynamic observations of the "Benfordness" of system metrics, and how non-conformance often relates to events of interest in the system under investigation.

In addition to synthetic data, we showed the effectiveness of our techniques on a number of real-world examples. Practitioners almost always deal with real-world data, and such data is seldom "well-behaved". In many cases, our quantitative evaluations were complemented by qualitative ones that showed our techniques yield results which are more *interpretable* and *intuitive* than the state-of-the-art (helping to address **Challenge 4**).

Perhaps the most concrete evidence of the practical impact of our work is found in a recent independent study of TMF [Vav+]. In this study, the authors cite estimates that the adoption of role-based access control (RBAC) saved American organizations $USD1.8 billion in the year 2009 [OL10]. The most significant expense in the transition from permission-based access control to RBAC was noted to be in the role engineering and mapping stage. Our TMF work assists in this migration process by effectively solving the role-based mining with missing-values problem at scale. In our work, we showed empirically that our algorithm FasTer outperforms existing techniques that are able to solve this problem. This conclusion was independently verified in this study [Vav+].

Implication 2: Our proposed techniques exhibit practically *linear* run-time behavior in the size of the data.

An algorithm that yields the globally-optimal result to a problem that needed to be solved *yesterday* is of little use. Often more so than researchers, practitioners and the "production" applications on which they work need to produce results on short time scales. To directly address this challenge, the approaches we have presented yield highly-competitive results at linear run-time complexity in the size of the data. In many cases, our algorithms outperform the state-of-the-art in terms of *both* effectiveness and efficiency, helping to address **Challenge 2** and simplifying the practitioner's job of algorithm-selection.

Implication 3: Our proposed techniques require no "obscure" parameters.

A Utopian system of unsupervised learning would promptly addresses the request *Here is my data, tell me what I want to know* without requiring any further information. Practically, algorithms need some information about the nature of the patterns which should be sought and presented. In many cases, this information is embedded as an assumption in the technique itself (e.g. Gaussian clusters); in others, it may be provided in the form of an algorithm parameter (e.g. number of patterns to find). Some algorithm parameters are not intuitively connected with the form of the solution and are instead used "internally" for managing a heuristic (e.g. a rounding threshold for an intermediate matrix). We might call such parameters "obscure", because a practitioner may find it difficult to know how the nature of the solution varies when varying the parameter. The iterative nature of data mining and KDD becomes more difficult when such parameters are involved, because the user may be less certain about how to proceed. Obscure parameters are hence best avoided where possible.

Fortunately, the algorithms we present in this thesis do not require parameters of this kind. FasTer requires only the number k of source patterns to find, as does Finesse. SkinnyDip is even more lean: its default parameter value for the statistical-significance threshold is equal to the typically-used threshold in most of statistics ($\alpha = 0.05$). This value seldom needs to be changed in practice (indeed, we did not change it for all our work). BenFound requires an analogous significance threshold α and the width of the sliding window, which the user can appreciate as representing a trade-off between statistical power and the ability to capture system dynamics. From the perspective of algorithm parameters, our algorithms are therefore an attractive choice for practitioners.

Implication 4: Our proposed techniques have documented implementations available in publicly-accessible locations online. All results are reproducible.

Replication is an important criteria by which scientific claims are judged [Pen11]. The ability to replicate results can also be of use for practitioners who wish to get started on applying our techniques to their own problems. In each of our research articles, we have provided references (URLs) to supplementary material that includes a downloadable research prototype of the proposed method. Each repository includes a detailed description and verbose code examples for getting started with the algorithm.

In the most recent contribution (**Paper E**) we also adopted and advocated the use of Knitr [Xie15]. Knitr is a LaTeX preprocessor that enables transparent and reproducible research by embedding the source code required for reproducing all results, figures and tables directly alongside the source for the report.

5.4 Limitations

Like all research, there are limitations to the contributions of this thesis. In the following sections we discuss these limitations in detail.

5.4.1 Non-Optimality

Perhaps the most obvious (albeit important) limitation of the methods presented in this thesis is the non-optimality of their solutions.

The field of data mining is rife with NP-Hard problems, including those treated in this thesis. For example, applying our FasTer algorithm to the Boolean Matrix Factorization problem involves an iterative heuristic that, at each step, needs to solve a set of optimization instances of the ±PSC problem. The ±PSC problem itself is NP-Hard, which currently means that globally-optimal solutions are generally not attainable for it in a computationally-tractable manner. Assuming we want "efficient" algorithms, the limitation of non-optimal solutions will therefore continue for some time (at least until the field sees revolutionary and constructive contributions in the direction of $P = NP$ and similar conjectures, or alternate computing machines like practical quantum computers become available).

Another important limitation is that "optimality" in the sense of a problem's objective function may not coincide with the "optimal" solution from the perspective of the practitioner or domain expert. SkinnyDip, for example, involves a hard-partitioning of the data points based on which points are determined to belong to the "modes" of the distribution. Practically, it uses mathematical and statistical notions for where modes terminate. A practitioner may well subjectively argue that these boundaries are not optimal. To some extent then, the notion of "optimality" extends to the choice of the objective function, which in the context of exploratory and unsupervised learning may be a subjective one.

5.4.2 Limited Approximability Results

Although our work includes theoretical results regarding the worst-case *run-time complexity* of our algorithmic contributions, it is somewhat more deficient of theoretical results regarding *effectiveness*.

For the FasTer algorithm, we were able to show that we can successfully reduce ±PSC to one of the sub-problems (Ternary Usage Row) under certain conditions. For this case it was possible to present an approximation ratio (at the cost of increased run-time complexity). We were not, however, able to present approximation ratios for the higher-level TMF and MDFS problems.

Good approximation algorithms have escaped researchers for a wide variety of NP-Hard problems. In [Aro98] it is argued that achieving certain reasonable approximation ratios is no easier than computing optimal solutions. That is, a number of negative results suggest that approximation itself can be NP-Hard for many problems.

For some algorithms, the convergence to a *local* minimum can be shown (e.g. k-Means [SI84]). We were also unable to prove such convergence for the FasTer and Finesse algorithms. Indeed, for the TMF and MDFS problems addressed in this thesis, discussions of the mathematical notions of "convexity", "differentiability", "intervals" and "locality" are not sensible because the domain of the objective functions is finite and non-continuous.

Without proofs for approximation ratios and "local" convergence, we resorted to 1) proofs that show algorithm-termination upon reaching a minimum during their search strategy, and 2) empirical evaluation. These approaches, particularly empirical evaluation, are common in the data-mining community. Strictly speaking, however, we concede that such evaluations cannot be used as the basis for conclusive statements about an algorithm's general performance.

5.4.3 Limited Empirical Evaluation

All of the methods presented in this thesis are evaluated empirically on real-world and synthetically-generated data sets.

Experiments on synthetically-generated data involve selecting a parameterized generative model. Such a generative model should ideally be motivated by real-world applications. It is important to realize, however, that the set of problem instances generated by all parameter combinations of such a model is typically only a small subset of the complete population of problem instances. In our case, it was not even possible to evaluate all parameter combinations of our generative models due to the *curse of dimensionality*. Our empirical evaluation was hence limited to the selection of sensible "default" generative parameters, from which individual parameters were varied over a range.

In many cases the proposed algorithms outperformed the state-of-the-art, however this was not always the case. Our empirical evaluation identified a number of generative parameters to which our algorithms reacted negatively. In the case of FasTer and Finesse, increasing the density of the usage matrix (i.e. the number of mixed patterns) caused in some cases poorer performance compared to the Asso algorithm. In the case of SkinnyDip being used as a full-space clustering approach, the clustering results degraded quickly with increasing spatial dimensionality.

5.4.4 Algorithm Parameters

Although we have noted in this thesis that our algorithms do not rely on "obscure" parameters, we cannot claim that they are entirely parameter-free. The following parameters are required by our methods. The most appropriate value for each parameter may not be known a priori, potentially causing confusion for practitioners.

1. TMF and MDFS (and their algorithms FasTer and Finesse) require the specification of the parameter k (the number of latent/source patterns to find). Additionally, these methods require the user to choose a number of "randomization rounds"

(defaulting to 20). At the cost of longer running time, we showed that a larger number of randomization rounds yields a higher-quality end result.

2. Our proposed "itemsets over an ontology" ("tree") feature for MDFS assumes that the tree structure is known a priori and provided as an input. Although we concede that this makes it more difficult to prepare the data for analysis, we argue that such a tree structure would likely already exist in many applications. The website *yummly.com*, for example, exploits its existing proprietary ingredients ontology for optimizing the user-search experience on their website.

3. Our clustering technique SkinnyDip relies on a threshold for statistical significance α, which can optionally be varied as a parameter.

4. SparseDip, our subspace-search technique, requires the specification of the number of grid points to evaluate in the search for a gradient-ascent starting point. Here there is likewise a trade-off between run-time and result quality.

5. BenFound requires the parameters w and α, representing the width of the sliding time-window and the threshold for statistical significance respectively. We discuss the setting of these parameters in **Paper E**.

5.4.5 Non-Determinism

Only a subset of our algorithms are deterministic (SkinnyDip and BenFound). The algorithms FasTer and Finesse are non-deterministic due to the use of a non-deterministic initialization heuristic. Specifically, the initialization procedure of both techniques involves a random selection. No other initialization strategies were evaluated, and we did not investigate mechanisms to make FasTer and Finesse deterministic. We note that Asso, a competitor to these methods, is deterministic.

5.4.6 Performance Trade-Offs

We introduced practical limitations to some of our methods in order to improve their run-time performance. For example, the FasTer algorithm was presented to be used for $k \leq 32$. This decision was made in order to reduce the run-time complexity from $\mathcal{O}(nmk^3)$ to $\mathcal{O}(nmk^2)$ through the exploitation of bitwise instructions on 64-bit architectures. As each ternary value requires two bits, the maximum permissible length of a row vector in the usage matrix or a column vector in the basis matrix is 32. This limitation could be mitigated by exploiting larger-width SSE instructions (as in Finesse), or accepting the higher run-time complexity $\mathcal{O}(nmk^3)$. We also note a limitation of our work on evaluating FasTer in high-performance environments. In this part of our work, we only considered a shared-memory environment, so we are unable to draw conclusions about the performance of FasTer on distributed-memory architectures or GPUs.

5.4.7 Side-Effects of our Complexity-Reducing Assumptions

As with most data-mining techniques, our methods rely on a number of assumptions for reducing complexity. Our techniques may yield sub-optimal results when these assumptions are violated. In the case of TMF and FasTer, we note that our work on Missing-Value Boolean Matrix Factorization (MVBMF) assumed the *missing completely at random* case (MCAR). We did not perform synthetic experiments for the *missing at random* (MAR) case, nor the *missing not at random* (MNAR) case, so we are unable to draw conclusions about the performance of FasTer in such cases.

Our framework MDFS demands that each admissible “mixable” feature fulfill a number of formal requirements. In particular, we assume the existence of a sensible contrast function $\oslash$, which may be difficult to objectively formulate.

One of SkinnyDip’s most intriguing strengths (no multivariate distance computations) is related to one of its main complexity-reducing assumptions. That is, SkinnyDip assumes that the cluster structure can still be reconstructed by taking systematic univariate projections. We can construct pathological examples which violate this assumption. Consider the relatively simple clustering problem in Figure 5.10. Two rectangular clusters exist in a “sea” of approximately 70% noise. SkinnyDip is able to detect the full clusters, but also includes a number of obvious noise points off to the side in each case. The problem here is that SkinnyDip first projects onto the horizontal axis. The “shadows” of the two-dimensional clusters appear as one when looking up from this axis, so SkinnyDip can only tell them apart after it has recursed into the second dimension.

A second effect of this complexity-reducing SkinnyDip assumption is that it models clusters as “hyper-intervals”, which may be sub-optimal. In two dimensions, for example, the clusters are modeled using rectangles. Rectangles cannot precisely model “rounded” cluster structures, like the Gaussian clusters in Figure 5.11.

Our method BenFound relies on the assumption that “unnatural” or “fraudulent” behavior causes a violation of Benford’s Law. If this assumption is violated, we see that BenFound is not immune to false negatives. That is, if true anomalies are created by agents that are aware of Benford’s Law, those agents may tune their anomalies such as to remain undetected by the law. However, the success of BL to date (particularly in forensic accounting [NW12]) suggests that many real-life fraudsters are not aware of the law.

Finally, BenFound relies on the assumption that the system metrics in focus follow BL in the “natural” state. We concede that many metrics do not follow BL. For example,

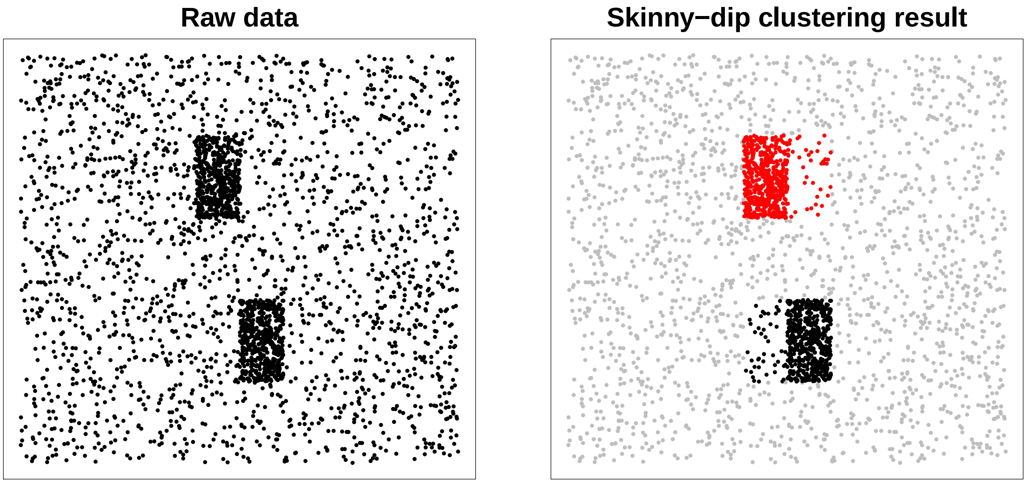


Figure 5.10: An example data set on which SkinnyDip achieves a less-than-optimal result (univariate projections limitation).

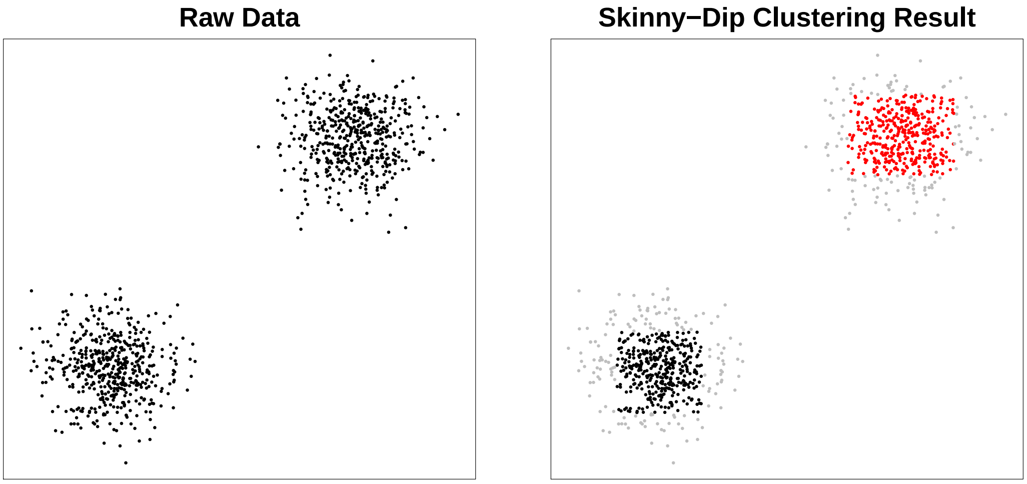


Figure 5.11: An example data set on which SkinnyDip achieves a less-than-optimal result ("boxed" or "rectangular" cluster-models limitation).

consider the signal obtained when monitoring the server response time of a popular website over time (Figure 5.12). Such data is normally not associated with any natural growth processes, so the digit distribution is by no means Benford. BenFound would be in a constant "red flag" state on such data (not useful).

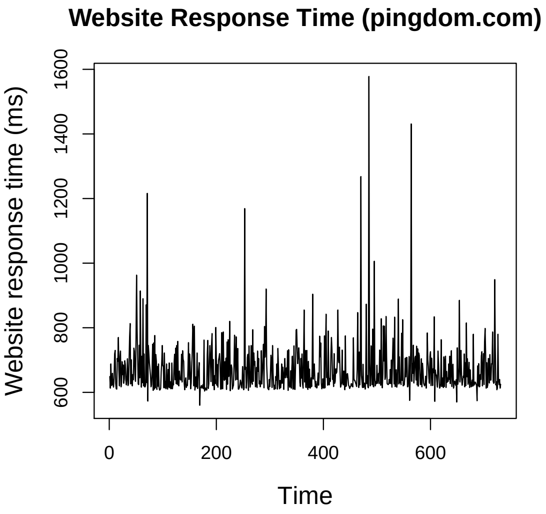


Figure 5.12: An example of non-Benford data: server response times for a popular website.

5.4.8 The Multiple Comparisons Problem

A subset of our proposed methods apply multiple statistical hypothesis tests in an automated fashion. This raises the question as to whether the multiple-comparisons problem is applicable [Sal10]. SkinnyDip, for example, may apply the *dip* test multiple times during its execution. In statistics, the application of multiple simultaneous hypothesis tests to a given sample should be accompanied by changing the significance level α to a more conservative (lower) value. Controlling procedures like Bonferroni correction [Dun61] can be used to this end. Bonferroni correction compensates for the fact that, as the number of tests increases, it becomes more likely that at least one test will report a significant result simply by chance.

In the case of SkinnyDip, we note the following with respect to the multiple comparisons problem:

- The dip test is used in a heuristic manner, and it is not known how many tests will be applied a priori. That is, the UniDip algorithm on which SkinnyDip is based will perform k tests, where k is the number of modal intervals (clusters) found at "runtime".

- UniDip ceases to apply the dip as soon as it finds a non-significant result.
- The size of the sample changes for each application of the dip (each is a subset of the full data set), meaning that the test theoretically loses power as UniDip progresses.

These observations complicate the situation, making it difficult to determine whether or not the multiple comparisons problem is a valid concern in SkinnyDip, and how correction should be performed if so. Although we believe that SkinnyDip has more important limitations that should be addressed first (e.g. the reliance on univariate projections), we concede that there may be a limitation due to this multiple comparisons effect as well. A future investigation into the multiple comparisons problem in the context of SkinnyDip may yield a more robust algorithm.

A statistical hypothesis test is also applied in an automated fashion in our method BenFound. In this case, the sample changes for each application of the test, so again the situation is more complex and not directly equivalent to the classical multiple-comparisons scenario.

5.5 Future Research

Our MDFS framework is one which leaves room for additional research. Perhaps the most interesting question is: *What other data types are admissible to MDFS?* We have identified a particular kind of graph structure in which it makes sense to *mix* instances ("itemsets over an ontology"). Other kinds of graph-based structures could also be investigated. A multitree, for example, can be used to represent multiple overlapping taxonomies over the same ground set. They are a class of directed acyclic graph with easily-identifyable substructures that are trees. Such structures are found in various applications, like a family tree which contains multiple inter-family marriages but no blood-relative marriages, or a corpus of academic material that is structured into a syllabus in different hierarchical ways by different university professors [FZ94]. List- or set-based objects may also be appropriate. For example, we can imagine a classical "itemsets" feature for which the measurement is a set of objects from some catalog (as in market-basket analysis). The "mixing" mechanism for such a feature would be the set union operator.

Perhaps the next most obvious improvement for TMF and the MDFS framework would be automated model-order selection. That is, the complete automation of the algorithm can be achieved by removing the need for the user to provide the parameter k. Automatic model-order selection is a difficult problem, particularly considering that algorithms like FasTer can only *approximately* calculate the solution for any given k. In our work on MDFS, we used the "elbow" heuristic to estimate a sensible model order k before moving to interpret the data (Figure 5.13). Some approaches for model-order selection try to automate the detection of such a point. More sophisticated approaches are based on various ways of interpreting *Occam's Razor*, which suggests that "among competing hypotheses, the one with the fewest assumptions should be selected". The Minimum Description Length principle [Ris78; Grü07] or the Bayesian Information Criterion [Sch+78] are often used in this light, helping to find a balance between a model's complexity and its performance. An investigation into automatic model-order selection for TMF and MDFS would ideally compare a number of such approaches on a variety of synthetic and real-world data, similar to the approach used in [MV14].

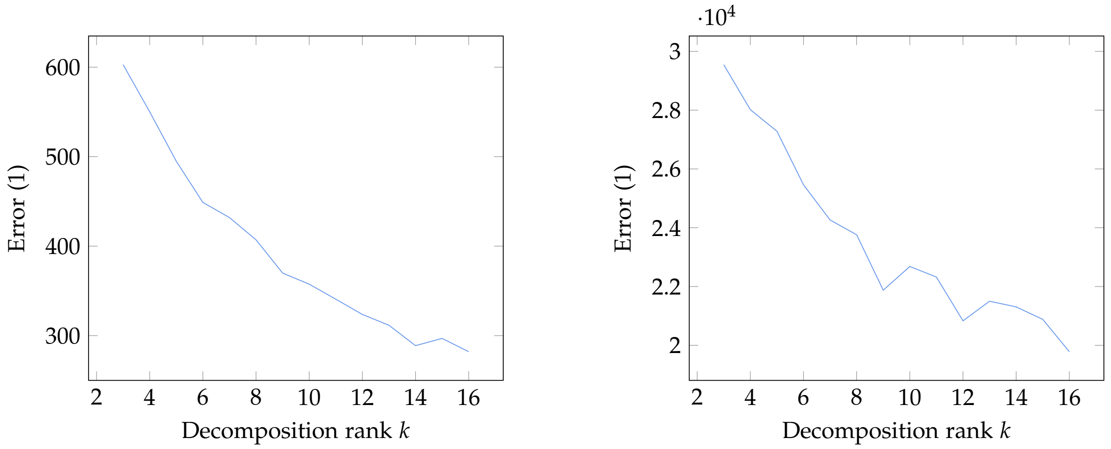


Figure 5.13: Model-order selection for Yummly (left) and IMDB (right) data. The "elbow" points are approximately $k = 6$ and $k = 9$ respectively.

Another direction for further research on our proposed matrix decompositions relates to the investigation of additional constraints. Like BMF and OMF, our formulation of the TMF and MDFS problems impose no strict constraints on the nature of the factors in the basis matrix (other than that they should contain entries from the finite set in question, of course). The objective function is based solely on the reconstruction error. In techniques from linear algebra, the factors *are* typically constrained. In PCA we see the constraint of orthogonality; in ICA we see the constraint of statistical-independence; in NMF we see the constraint of non-negativity. Interestingly, there already exist BMF variants that dictate similar constraints. For example, the Boolean CX (BCX) Decomposition problem [Mie08b] is similar to BMF, however the set of basis vectors in BCX must be a subset of the original (observed) vectors. The Boolean CUR problem is a further variant with similar constraints [Mie08b]. Unfortunately, the practical usefulness of these decompositions is not entirely clear, as the original article (and, to the best of our knowledge, future work) does not show real-world applications which serve to motivate the choice of these constrained techniques over vanilla BMF. Despite this, and even if only for theoretical purposes, one could investigate analogous constrained versions of TMF and MDFS. Future applications may find such variants useful.

Our clustering algorithm SkinnyDip is the first to consider the use of the dip test. Although elegant, the dip is only usable on univariate data, so important information is sometimes lost when SkinnyDip performs the necessary univariate projections from multivariate data. To improve on this, SkinnyDip could be modified to perform multiple "projection rounds" on the data (perhaps heuristic-based), and the results aggregated in an ensemble-like manner.

Overcoming the "box clusters" limitation of SkinnyDip may be a comparatively easier problem to solve. One approach might work as follows. Each detected SkinnyDip cluster could have its mean density computed. This density, along with the detected cluster points, could serve as a "seed" for "elaborating" on the cluster in a local density-based fashion. The propagation algorithm might be similar to that of DBSCAN, for example [Est+96]. Another approach might be to create multiple overlapping "boxes" for the same cluster from different projection angles, again using the results to "fine-tune" the shape of each cluster.

With respect to our work on exploiting Benford's Law (BL), it is difficult not to be intrigued by this "gem of statistical folklore" [BH11]. Indeed, it remains somewhat of a mystery, having not yet been fully explained [BH11]. To the best of our knowledge, our contribution is the first that exploits the law as the basis of a data-mining technique. We anticipate that BenFound, our non-parametric anomaly-detection technique based on BL, will consequently generate some interest in the data-mining community. Instead of being used as the basis for a "red-flagging" anomaly-detection technique, one might use BL as the basis for *ranking* the "authenticity" of a set of phenomena. Ranking Twitter hashtags, YouTube channels or sets of networks or graphs are some concrete ideas that could be investigated in this respect.

Finally, there is room in all our work for even more aggressive hardware optimization. Although we have optimized some of our techniques for shared-memory parallel environments, we did not consider the performance gains that are possible through graphics cards. The performance of FasTer, for example, is dependent on the ability to efficiently evaluate bitwise instructions. A modern CPU core can typically execute four 32-bit instructions per clock (using a 128-bit SSE instruction), whereas many modern GPUs can execute thousands of such instructions per clock (having many more ALUs or "shaders"). Although other factors would of course need to be considered ("executive work" like branching logic), we expect that a graphic-card implementation of FasTer would yield significant performance gains.

6 Conclusion

Data mining is the application of specific algorithms to prepared data for the purpose of either prediction or description. The scope of this thesis was restricted to exploratory, unsupervised, descriptive data mining. We have advanced the state-of-the-art by presenting novel methods which address a number of complexities in modern information systems: 1) heterogeneous data types and measurement scales, 2) missing information, 3) clutter and noise, 4) high dimensionality and 5) high-bandwidth time-series data. We designed linear-time algorithms for our problems, discussed their properties, and empirically evaluated their performance on 1) synthetically-generated data, 2) real-world data, and 3) run-time behavior.

From a research perspective this thesis offers a number of results. To help address the complex challenges of heterogeneous data types, heterogeneous measurement scales and missing values, we presented Ternary Matrix Factorization and the Matrix Factorizations over Discrete Finite Sets framework. To help address the complex challenges of clutter and high-dimensionality in clustering, we presented the novel clustering paradigm SkinnyDip. To help address the complex challenge of anomaly-detection in high-bandwidth time-series data, we presented the insightful method BenFound based on the exploitation of Benford's Law. We thus achieved our goals by demonstrating how these kinds of complexities can be addressed in variants of the most fundamental kinds of data mining problems (finding associations, clustering objects and detecting anomalies).

The results of this thesis are beneficial to practitioners from a number of perspectives. From the perspective of effectiveness, we showed empirically that each algorithm can outperform the state-of-the-art on synthetic and real-world data with respect to appropriate quality measures. From the perspective of efficiency, we showed that each algorithm exhibits practically-linear run-time growth in the size of the data. From the perspective of usability, we showed that none of our techniques require "obscure" parameters. From the perspective of interpretability, we showed that the majority of our algorithms yield directly-interpretable results in the context of the domain (only BenFound requires some level of post-processing). Finally, we discussed how all implementations are provided in publicly-accessible repositories online, and how all results are reproducible.

We identified a number of limitations. Our methods are mostly based on heuristics and are unable to solve optimally in the general case (“hard” computational complexity). We provide only a single approximation guarantee for a sub-problem of one method. Our evaluation was mainly empirical and did not cover all possible problem instances. FasTer and Finesse require the provision of the parameter k, the number of patterns to find. SkinnyDip may yield sub-optimal results as a result of univariate projections and “blocky” cluster bounding. BenFound is not immune to false-negatives and can only be applied to systems with metrics that follow Benford’s Law in their “natural” state.

We identified a number of future research directions. Further concrete types of “mixable” finite sets could be identified and implemented for MDFS, thereby widening its applicability. Enhancements to SkinnyDip could be made with the aim of overcoming some of its limitations, particularly its tendency to produce “rectangular” clusters and its dependency on univariate projections. Finally, the real-time concepts in BenFound might be applied to data contexts other than univariate time series, like monitoring the flow of information between nodes of a network.

Bibliography

[AIS93] R. Agrawal, T. Imieliński, and A. Swami. “Mining association rules between sets of items in large databases.” In: *Acm sigmod record*. Vol. 22. 2. ACM. 1993, pp. 207–216.

[AKZ08] E. Achtert, H.-P. Kriegel, and A. Zimek. “ELKI: a software system for evaluation of subspace clustering algorithms.” In: *International Conference on Scientific and Statistical Database Management*. Springer. 2008, pp. 580–585.

[Alb+16] F. D. Albareti, C. A. Prieto, A. Almeida, F. Anders, S. Anderson, B. H. Andrews, A. Aragon-Salamanca, M. Argudo-Fernandez, E. Armengaud, E. Aubourg, et al. “The Thirteenth Data Release of the Sloan Digital Sky Survey: First Spectroscopic Data from the SDSS-IV Survey MApping Nearby Galaxies at Apache Point Observatory.” In: *arXiv preprint arXiv:1608.02013* (2016).

[Ank+99] M. Ankerst, M. M. Breunig, H.-P. Kriegel, and J. Sander. “OPTICS: ordering points to identify the clustering structure.” In: *ACM Sigmod Record*. Vol. 28. 2. ACM. 1999, pp. 49–60.

[Aro98] S. Arora. “The approximability of NP-hard problems.” In: *Proceedings of the thirtieth annual ACM symposium on Theory of computing*. ACM. 1998, pp. 337–348.

[BC57] R. Bellman and R. Corporation. *Dynamic Programming*. Rand Corporation research study. Princeton University Press, 1957. ISBN: 9780691079516.

[Ben+10] F. Benevenuto, G. Magno, T. Rodrigues, and V. Almeida. “Detecting spammers on twitter.” In: *Collaboration, electronic messaging, anti-abuse and spam conference (CEAS)*. Vol. 6. 2010, p. 12.

[Ben38] F. Benford. “The law of anomalous numbers.” In: *Proceedings of the American Philosophical Society* (1938), pp. 551–572.

[BG04] H.-J. Bungartz and M. Griebel. “Sparse grids.” In: *Acta numerica* 13 (2004), pp. 147–269.

[BH11] A. Berger and T. P. Hill. “Benford’s law strikes back: no simple explanation in sight for mathematical gem.” In: *The Mathematical Intelligencer* 33.1 (2011), pp. 85–91.

[Bis81] J. Biskup. "A formal approach to null values in database relations." In: *Advances in Data Base Theory*. Springer, 1981, pp. 299–341.

[BJ08] K. Bernhard and V. Jens. *Combinatorial optimization: Theory and algorithms*. 2008.

[BK13] R. Belohlavek and M. Krmelova. "Beyond Boolean matrix decompositions: toward factor analysis and dimensionality reduction of ordinal data." In: *2013 IEEE 13th International Conference on Data Mining*. IEEE. 2013, pp. 961–966.

[BMD09] C. Boutsidis, M. W. Mahoney, and P. Drineas. "An improved approximation algorithm for the column subset selection problem." In: *Proceedings of the twentieth Annual ACM-SIAM Symposium on Discrete Algorithms*. Society for Industrial and Applied Mathematics. 2009, pp. 968–977.

[Boa08] O. Board. "OpenMP application program interface version 3.0." In: *The OpenMP Forum, Tech. Rep.* 2008.

[Böh+06] C. Böhm, C. Faloutsos, J.-Y. Pan, and C. Plant. "Robust information-theoretic clustering." In: *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM. 2006, pp. 65–75.

[BP09] P. Burman and W. Polonik. "Multivariate mode hunting: Data analytic tools with measures of significance." In: *Journal of Multivariate Analysis* 100.6 (2009), pp. 1198–1218.

[Bro00] A. W. Bronkhorst. "The cocktail party phenomenon: A review of research on speech intelligibility in multiple-talker conditions." In: *Acta Acustica united with Acustica* 86.1 (2000), pp. 117–128.

[Buh+13] H. U. Buhl, M. Röglinger, F. Moser, J. Heidemann, et al. "Big data." In: *Business & Information Systems Engineering* 5.2 (2013), pp. 65–69.

[BY09] R. S. Baker and K. Yacef. "The state of educational data mining in 2009: A review and future visions." In: *JEDM-Journal of Educational Data Mining* 1.1 (2009), pp. 3–17.

[Car+00] R. D. Carr, S. Doddi, G. Konjevod, and M. V. Marathe. "On the red-blue set cover problem." In: *Symposium on Discrete Algorithms: Proceedings of the eleventh annual ACM-SIAM symposium on Discrete algorithms*. Vol. 9. 11. Citeseer. 2000, pp. 345–353.

[CFF00] A. Cuevas, M. Febrero, and R. Fraiman. "Estimating the number of clusters." In: *Canadian Journal of Statistics* 28.2 (2000), pp. 367–382.

[CM02] K. J. Cios and G. W. Moore. "Uniqueness of medical data mining." In: *Artificial intelligence in medicine* 26.1 (2002), pp. 1–24.

[ÇM09] A. Çivril and M. Magdon-Ismail. "On selecting a maximum volume submatrix of a matrix and related problems." In: *Theoretical Computer Science* 410.47 (2009), pp. 4801–4811.

[Dha13] V. Dhar. "Data science and prediction." In: *Communications of the ACM* 56.12 (2013), pp. 64–73.

[DHP04] C. Durtschi, W. Hillison, and C. Pacini. "The effective use of Benford's law to assist in detecting fraud in accounting data." In: *Journal of forensic accounting* 5.1 (2004), pp. 17–34.

[DLR77] A. P. Dempster, N. M. Laird, and D. B. Rubin. "Maximum likelihood from incomplete data via the EM algorithm." In: *Journal of the royal statistical society. Series B (methodological)* (1977), pp. 1–38.

[DR98] A. Dasgupta and A. E. Raftery. "Detecting features in spatial point processes with clutter via model-based clustering." In: *Journal of the American Statistical Association* 93.441 (1998), pp. 294–302.

[Dun61] O. J. Dunn. "Multiple comparisons among means." In: *Journal of the American Statistical Association* 56.293 (1961), pp. 52–64.

[Eis+11] D. J. Eisenstein, D. H. Weinberg, E. Agol, H. Aihara, C. A. Prieto, S. F. Anderson, J. A. Arns, É. Aubourg, S. Bailey, E. Balbinot, et al. "SDSS-III: Massive spectroscopic surveys of the distant universe, the Milky Way, and extra-solar planetary systems." In: *The Astronomical Journal* 142.3 (2011), p. 72.

[Est+96] M. Ester, H.-P. Kriegel, J. Sander, X. Xu, et al. "A density-based algorithm for discovering clusters in large spatial databases with noise." In: *Kdd*. Vol. 96. 34. 1996, pp. 226–231.

[FB75] J. D. Francis and L. Busch. "What We Now Know about "I Don't Knows"." In: *The Public Opinion Quarterly* 39.2 (1975), pp. 207–218.

[FPS96] U. Fayyad, G. Piatetsky-Shapiro, and P. Smyth. "From data mining to knowledge discovery in databases." In: *AI magazine* 17.3 (1996), p. 37.

[FZ94] G. W. Furnas and J. Zacks. "Multitrees: enriching and reusing hierarchical structure." In: *Proceedings of the SIGCHI conference on Human factors in computing systems*. ACM. 1994, pp. 330–336.

[Gol15] J. Golbeck. "Benford's Law Applies to Online Social Networks." In: *PloS one* 10.8 (2015), e0135169.

[Grü07] P. D. Grünwald. *The minimum description length principle*. MIT press, 2007.

[Hal+09] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten. "The WEKA data mining software: an update." In: *ACM SIGKDD explorations newsletter* 11.1 (2009), pp. 10–18.

[HH85] J. A. Hartigan and P. Hartigan. "The dip test of unimodality." In: *The Annals of Statistics* (1985), pp. 70–84.

[HKO04] A. Hyvärinen, J. Karhunen, and E. Oja. *Independent Component Analysis*. Adaptive and Cognitive Dynamic Systems: Signal Processing, Learning, Communications and Control. Wiley, 2004. ISBN: 9780471464198.

[HL04] M. Hu and B. Liu. "Mining and summarizing customer reviews." In: *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM. 2004, pp. 168–177.

[Hor12] K. Hornik. "The comprehensive R archive network." In: *Wiley Interdisciplinary Reviews: Computational Statistics* 4.4 (2012), pp. 394–398.

[HPP00] B. Heisele, T. Poggio, and M. Pontil. *Face Detection in Still Gray Images*. A.I. memo 1687. Cambridge, MA: Center for Biological and Computational Learning, MIT, 2000.

[HTK15] K. Hayashi, M. Takanori, and K.-i. Kawarabayashi. "Real-time topic detection on twitter with topic hijack filtering." In: *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM. 2015, pp. 417–426.

[IM98] P. Indyk and R. Motwani. "Approximate nearest neighbors: towards removing the curse of dimensionality." In: *Proceedings of the thirtieth annual ACM symposium on Theory of computing*. ACM. 1998, pp. 604–613.

[JKM14] N. A. James, A. Kejariwal, and D. S. Matteson. "Leveraging Cloud Data to Mitigate User Experience from "Breaking Bad"." In: *arXiv preprint arXiv:1411.7955* (2014).

[K+11] H. C. Koh, G. Tan, et al. "Data mining applications in healthcare." In: *Journal of healthcare information management* 19.2 (2011), p. 65.

[Kan+02] R. V. Kantety, M. La Rota, D. E. Matthews, and M. E. Sorrells. "Data mining for simple sequence repeats in expressed sequence tags from barley, maize, rice, sorghum and wheat." In: *Plant molecular biology* 48.5-6 (2002), pp. 501–510.

[Kar+04] A. Karatzoglou, A. Smola, K. Hornik, and A. Zeileis. "kernlab-an S4 package for kernel methods in R." In: (2004).

[Kar72] R. M. Karp. "Reducibility among combinatorial problems." In: *Complexity of computer computations*. Springer, 1972, pp. 85–103.

[KB11] D. Kumar and D. Bhardwaj. “Rise of data mining: current and future application areas.” In: *IJCSI International Journal of Computer Science Issues* 8.5 (2011).

[KKZ09] H.-P. Kriegel, P. Kröger, and A. Zimek. “Clustering high-dimensional data: A survey on subspace clustering, pattern-based clustering, and correlation clustering.” In: *ACM Transactions on Knowledge Discovery from Data (TKDD)* 3.1 (2009), p. 1.

[KL05] A. Krause and V. Liebscher. “Multimodal projection pursuit using the dip statistic.” In: *Preprint-Reihe Mathematik* 13 (2005).

[Kri+07] H.-P. Kriegel, K. M. Borgwardt, P. Kröger, A. Pryakhin, M. Schubert, and A. Zimek. “Future trends in data mining.” In: *Data Mining and Knowledge Discovery* 15.1 (2007), pp. 87–97.

[KSS03] M. Kuhlmann, D. Shohat, and G. Schimpf. “Role mining-revealing business roles for security administration using data mining technology.” In: *Proceedings of the eighth ACM symposium on Access control models and technologies*. ACM. 2003, pp. 179–186.

[LAF15] N. Laptev, S. Amizadeh, and I. Flint. “Generic and scalable framework for automated time-series anomaly detection.” In: *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM. 2015, pp. 1939–1947.

[LK07] D. Liben-Nowell and J. Kleinberg. “The link-prediction problem for social networks.” In: *Journal of the American society for information science and technology* 58.7 (2007), pp. 1019–1031.

[LOP14] C. Lucchese, S. Orlando, and R. Perego. “A Unifying Framework for Mining Approximate Top-Binary Patterns.” In: *IEEE Transactions on Knowledge and Data Engineering* 26.12 (2014), pp. 2900–2913.

[LRL07] J. Li, S. Ray, and B. G. Lindsay. “A nonparametric statistical approach to clustering via mode identification.” In: *Journal of Machine Learning Research* 8.Aug (2007), pp. 1687–1723.

[LS99] D. D. Lee and H. S. Seung. “Learning the parts of objects by non-negative matrix factorization.” In: *Nature* 401.6755 (1999), pp. 788–791.

[Mac+67] J. MacQueen et al. “Some methods for classification and analysis of multivariate observations.” In: *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*. Vol. 1. 14. Oakland, CA, USA. 1967, pp. 281–297.

[McG+02] A. McGrail, E. Gulski, E. Groot, D. Allan, D. Birtwhistle, and T. Blackburn. "Datamining techniques to assess the condition of high voltage electrical plant." In: *CIGRE Paris WG15* 11 (2002).

[MHR10] J. Marchini, C. Heaton, and B. Ripley. *fastICA: FastICA Algorithms to perform ICA and Projection Pursuit. R package version 1.1-13.* 2010.

[Mie+08] P. Miettinen, T. Mielikäinen, A. Gionis, G. Das, and H. Mannila. "The discrete basis problem." In: *IEEE Transactions on Knowledge and Data Engineering* 20.10 (2008), pp. 1348–1362.

[Mie08a] P. Miettinen. "On the positive–negative partial set cover problem." In: *Information Processing Letters* 108.4 (2008), pp. 219–221.

[Mie08b] P. Miettinen. "The boolean column and column-row matrix decompositions." In: *Data Mining and Knowledge Discovery* 17.1 (2008), pp. 39–56.

[Mie09] P. Miettinen. "Matrix decomposition methods for data mining: Computational complexity and algorithms." PhD thesis. University of Helsinki, 2009.

[Mil56] G. A. Miller. "The magical number seven, plus or minus two: Some limits on our capacity for processing information." In: *Psychological review* 63.2 (1956), p. 81.

[MJ14] D. S. Matteson and N. A. James. "A nonparametric approach for multiple change point analysis of multivariate data." In: *Journal of the American Statistical Association* 109.505 (2014), pp. 334–345.

[MP14] S. Maurus and C. Plant. "Ternary matrix factorization." In: *©2014 IEEE. Reprinted, with permission, from Maurus, Samuel and Plant, Claudia, Ternary matrix factorization, 2014 IEEE International Conference on Data Mining.* 2014, pp. 400–409.

[MP16a] S. Maurus and C. Plant. "Factorizing Complex Discrete Data with Finesse." In: *©2016 IEEE. Reprinted, with permission, from Maurus, Samuel and Plant, Claudia, Factorizing Complex Discrete Data with Finesse, 2016 IEEE International Conference on Data Mining.* 2016.

[MP16b] S. Maurus and C. Plant. "Skinny-dip: Clustering in a Sea of Noise." In: *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining* (2016), pp. 1055–1064.

[MP16c] S. Maurus and C. Plant. "Ternary Matrix Factorization: problem definitions and algorithms." In: *Knowledge and Information Systems* 46.1 (2016), pp. 1–31.

[MP17] S. Maurus and C. Plant. “Let’s see your Digits: Anomalous-State Detection using Benford’s Law.” In: *Submitted to the Research Track of the 2017 SIAM International Conference on Data Mining* (2017).

[MV14] P. Miettinen and J. Vreeken. “mdl4bmf: Minimum description length for Boolean matrix factorization.” In: *ACM Transactions on Knowledge Discovery from Data (TKDD)* 8.4 (2014), p. 18.

[Nig99] J. Nigrini Mark. “I’ve got your number: How a mathematical phenomenon can help CPAs uncover fraud and other irregularities.” In: *Journal of Accountancy* 187.5 (1999).

[NW12] M. Nigrini and J. Wells. *Benford’s Law: Applications for Forensic Accounting, Auditing, and Fraud Detection*. Wiley Corporate F&A. Wiley, 2012. ISBN: 9781118152850.

[OL10] A. C. O’Connor and R. J. Loomis. “2010 Economic Analysis of Role-Based Access Control.” In: *NIST, Gaithersburg, MD* 20899 (2010).

[Ooi12] H. Ooi. “Density visualization and mode hunting using trees.” In: *Journal of Computational and Graphical Statistics* (2012).

[Par62] E. Parzen. “On estimation of a probability density function and mode.” In: *The annals of mathematical statistics* 33.3 (1962), pp. 1065–1076.

[Pen11] R. D. Peng. “Reproducible research in computational science.” In: *Science* 334.6060 (2011), pp. 1226–1227.

[Poe+88] G. S. Poe, I. Seeman, J. McLaughlin, E. Mehl, and M. Dietz. “"Don’t Know" Boxes in Factual Questions in a Mail Questionnaire Effects on Level and Quality of Response.” In: *Public Opinion Quarterly* 52.2 (1988), pp. 212–222.

[Rig10] G. Rigaill. *Pruned dynamic programming for optimal multiple change-point detection*. 2010.

[Ris78] J. Rissanen. “Modeling by shortest data description.” In: *Automatica* 14.5 (1978), pp. 465–471.

[Rit+11] A. Ritter, S. Clark, Mausam, and O. Etzioni. “Named Entity Recognition in Tweets: An Experimental Study.” In: *EMNLP*. 2011.

[Rit+12] A. Ritter, Mausam, O. Etzioni, and S. Clark. “Open Domain Event Extraction from Twitter.” In: *KDD*. 2012.

[Ros+56] M. Rosenblatt et al. “Remarks on some nonparametric estimates of a density function.” In: *The Annals of Mathematical Statistics* 27.3 (1956), pp. 832–837.

[Sal10] N. Salkind. *Encyclopedia of Research Design*. SAGE Publications, 2010. ISBN: 9781506319315.

[Sch+78] G. Schwarz et al. "Estimating the dimension of a model." In: *The annals of statistics* 6.2 (1978), pp. 461–464.

[SI84] S. Z. Selim and M. A. Ismail. "K-means-type algorithms: a generalized convergence theorem and characterization of local optimality." In: *IEEE Transactions on pattern analysis and machine intelligence* 1 (1984), pp. 81–87.

[SK09] X. Su and T. M. Khoshgoftaar. "A survey of collaborative filtering techniques." In: *Advances in artificial intelligence* 2009 (2009), p. 4.

[Sla96] P. Slavík. "A tight analysis of the greedy algorithm for set cover." In: *Proceedings of the twenty-eighth annual ACM symposium on Theory of computing*. ACM. 1996, pp. 435–441.

[SRJ04] N. Srebro, J. Rennie, and T. S. Jaakkola. "Maximum-margin matrix factorization." In: *Advances in neural information processing systems*. 2004, pp. 1329–1336.

[SS13] S. Sagiroglu and D. Sinanc. "Big data: A review." In: *Collaboration Technologies and Systems (CTS), 2013 International Conference on*. IEEE. 2013, pp. 42–47.

[Ste46] S. S. Stevens. *On the theory of scales of measurement*. 1946.

[SVM06] J.-F. Superby, J. Vandamme, and N. Meskens. "Determination of factors influencing the achievement of the first-year university students using data mining methods." In: *Workshop on Educational Data Mining*. Vol. 32. Citeseer. 2006, p. 234.

[Tip+15] S. Tippmann et al. "Programming tools: Adventures with R." In: *Nature* 517.7532 (2015), pp. 109–110.

[TKB12] C. Thurau, K. Kersting, and C. Bauckhage. "Deterministic CUR for Improved Large-Scale Data Analysis: An Empirical Study." In: *SDM*. SIAM. 2012, pp. 684–695.

[TL10] L. Tang and H. Liu. "Community detection and mining in social media." In: *Synthesis Lectures on Data Mining and Knowledge Discovery* 2.1 (2010), pp. 1–137.

[Vav+] S. Vavilis, A. I. Egner, M. Petkovic, and N. Zannone. *Role Mining with Missing Values*. Tech. rep. Eindhoven University of Technology (Security Group), and Philips Research Europe.

[VEB09] N. X. Vinh, J. Epps, and J. Bailey. "Information theoretic measures for clusterings comparison: is a correction for chance necessary?" In: *Proceedings of the 26th Annual International Conference on Machine Learning*. ACM. 2009, pp. 1073–1080.

[VHK14] O. Vallis, J. Hochenbaum, and A. Kejariwal. “A novel technique for long-term anomaly detection in the cloud.” In: *6th USENIX Workshop on Hot Topics in Cloud Computing (HotCloud 14)*. 2014.

[VS08] J. Vreeken and A. Siebes. “Filling in the blanks-Krimp minimisation for missing data.” In: *2008 Eighth IEEE International Conference on Data Mining*. IEEE. 2008, pp. 1067–1072.

[Wan+05] J. T. Wang, M. J. Zaki, H. T. Toivonen, and D. Shasha. “Introduction to Data Mining in Bioinformatics.” In: *Data Mining in Bioinformatics*. Springer, 2005, pp. 3–8.

[Wil45] F. Wilcoxon. “Individual comparisons by ranking methods.” In: *Biometrics bulletin* 1.6 (1945), pp. 80–83.

[WM02] W.-K. Wong and A. Moore. “Efficient algorithms for non-parametric clustering with clutter.” In: *Proceedings of the 34th Interface Symposium*. 2002.

[Xie15] Y. Xie. *Dynamic Documents with R and knitr*. Vol. 29. CRC Press, 2015.

[Yao03] Y. Yao. “Information-theoretic measures for knowledge discovery and data mining.” In: *Entropy Measures, Maximum Entropy Principle and Emerging Applications*. Springer, 2003, pp. 115–136.

[YHX14] X. Yu, D. Hu, and J. Xu. *Blind Source Separation: Theory and Applications*. Wiley, 2014. ISBN: 9781118679845.

[YM12] P. Yadava and P. Miettinen. “BMF with missing values.” In: *Master's Thesis, University of Saarland* (2012).

[YW06] Q. Yang and X. Wu. “10 challenging problems in data mining research.” In: *International Journal of Information Technology & Decision Making* 5.04 (2006), pp. 597–604.

[YWB11] X. Ying, X. Wu, and D. Barbará. “Spectrum based fraud detection in social networks.” In: *2011 IEEE 27th International Conference on Data Engineering*. IEEE. 2011, pp. 912–923.

[Zan82] C. Zaniolo. “Database relations with null values.” In: *Proceedings of the 1st ACM SIGACT-SIGMOD symposium on Principles of database systems*. ACM. 1982, pp. 27–33.

[ZP05] L. Zelnik-Manor and P. Perona. “Self-tuning spectral clustering.” In: *Advances in Neural Information Processing Systems 17* (2005).

[ZSK12] A. Zimek, E. Schubert, and H.-P. Kriegel. “A survey on unsupervised outlier detection in high-dimensional numerical data.” In: *Statistical Analysis and Data Mining* 5.5 (2012), pp. 363–387.

List of Figures

List of Tables