# An Analysis of Distributional Shifts in Automated Driving Functions in Highway Scenarios

Oliver De Candido, Xinyang Li, and Wolfgang Utschick

2022 IEEE 95th Vehicular Technology Conference: (VTC2022-Spring)
June 19–22, 2022
Helsinki, Finland

© 2022 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.

# An Analysis of Distributional Shifts in Automated Driving Functions in Highway Scenarios

Oliver De Candido, Xinyang Li, and Wolfgang Utschick

Abstract—We investigate the distributional shifts between datasets which pose a challenge to validate safety critical driving functions which incorporate Machine Learning (ML)-based algorithms. First, we describe the possible distributional shifts which can occur in highway driving datasets. Following this, we analyze—both qualitatively and quantitatively—the distributional shifts between two publicly available, and widely used, highway driving datasets. We demonstrate that a safety critical driving function, e.g., a lane change maneuver prediction, trained on one dataset will not generalize as expected to the other dataset in the presence of these distributional shifts. This highlights the impact which distributional shifts can have on safety critical driving functions. We suggest that an analysis of the datasets used to train ML-based algorithms incorporated in safety critical driving functions plays an important role in building a safety-argument for validation.

## I. Introduction and Motivation

Machine Learning (ML)-based algorithms are becoming ever more popular to solve the wide-range of challenges which Autonomous Vehicles (AVs) may face. They have the ability to learn representations from a few examples, alleviating the burden of manually modelling every possible scenario an AV might encounter. However, this lack of explicit specification makes the validation of driving functions which incorporate ML-based algorithms challenging. It has been shown that the current functional safety standards, e.g., the ISO 26262 standard, are not directly applicable when ML-based algorithms are incorporated in the driving function [1]. To this end, a safety-argument can be created to validate the driving functions which incorporate ML-based algorithms, see, e.g., [2; 3].

In this paper, we analyze the data used to train ML-based driving functions. An analysis of the data should be one part in the overarching safety-argument since this is an important aspect of ML-based algorithms to validate. If the data which an ML algorithm is trained on does not contain generalizable features, then the driving functions which incorporate this ML-based algorithm may not perform as expected. A related challenge is the possible miss-match between the distribution of the training data and the distribution of the test data or the distribution of the real-world data during deployment. This mismatch between distributions is known as a distributional shift. Examples of the different types of distributional shifts will be introduced in Section II.

We consider the case where a safety critical driving function is learned via supervised ML algorithms. We analyze two publicly available highway driving datasets: the highD dataset [4] and the Next Generation SIMulation (NGSIM) dataset [5; 6]. Numerous publications propose ML-based driving algorithms and demonstrate them using these datasets, see, e.g., [7] and references therein. Our analysis indicates that there are significant distributional shifts between these datasets.

Following this analysis, we consider the effect of these distributional shifts on a safety critical driving function. The driving function we consider is a lane change maneuver prediction, which will be introduced in Section III-B. We train various ML-based algorithms to solve this problem. Ultimately, we show that they all suffer under the distributional shift, i.e., when the models are trained on one dataset and tested on the other dataset, there is a drop in performance (between 36% and 40% in accuracy). This highlights the importance of considering distributional shifts when building a safety-argument for validation.

Amodei et al. [8] were the first to summarize the challenge of robustifying real-world ML systems to distributional shifts. In the context of ML-based algorithms in AVs, the authors of [9; 10] were the first to discuss the challenge of distributional shifts. However, in these publications, the authors only state the hypothetical challenge of dealing with distributional shifts [9]. Here, we take this one step further, and explicitly discuss and analyse the types of distributional shifts which can occur in highway driving scenarios.

Due to the abundance of image data and the relative ease with which researchers can collect new datasets, distributional shifts between image datasets have been more thoroughly studied. Torralba and Efros [11] study the implicit biases which image datasets contain, which is a closely related problem to distributional shifts. They show that a classifier can easily classify which dataset images with the same class label belong to. Moreover, they show that when training an ML algorithm on one dataset and testing on another, the performance drops by roughly 48%. This happens despite the fact that each of the standard image datasets claims to be representative.

Recht et al. [12] show that ML-based algorithms trained on a popular benchmark image dataset cannot generalize onto a novel test dataset. The authors create a new dataset by following the same pre-processing steps used to collect the original dataset. They show that all of the state-of-the-

This research was funded by Audi AG.

Department of Electrical and Computer Engineering, Technical University of Munich, 80333 Munich, Germany; Email: oliver.de-candido@tum.de.

art ML-based classification algorithms perform 10% worse in terms of accuracy on this new dataset. This indicates that there is a distributional shift between the original and the newly created datasets.

Recently, Koh et al. [13] introduce seven new datasets with real-world distributional shifts to train and to test ML algorithms on. Additionally, the Shifts dataset [14] was introduced with the same motivation; it also contains vehicle motion prediction data. These datasets were created with domain experts to represent true distributional shifts which can occur during deployment.

In this paper, we analyse the distributional shifts which can occur in realistic highway driving data. This analysis, to the best of our knowledge, has not been done before. We identify and quantify the distributional shifts, and show the effect they have on ML-based safety critical automated driving functions. This analysis can be used as one statement in the overall safety-argument.

## II. Distributional Shifts in Highway Driving Data

A distributional shift or dataset shift [15] describes the phenomenon when the distribution of the data which were used to train the ML algorithms does not match that of the test data. Moreover, the data are assumed to be independent and identically distributed (i.i.d.), which generally does not hold in the real-world.

In supervised ML tasks, we train the algorithms on a training dataset $\mathcal{S} = \left\{(\boldsymbol{X}^{(i)}, y^{(i)})\right\}_{i=1}^{N}$, where the inputs $\boldsymbol{X} \in \mathcal{X}$ and labels $y \in \mathcal{Y}$ are i.i.d. samples of an unknown joint distribution, $p(\boldsymbol{X}, y)$. Then, we test on a test dataset. In general, a distributional shift describes the case where $p_A(\boldsymbol{X}, y) \neq p_B(\boldsymbol{X}, y)$, where $p_A$ and $p_B$ are the unknown joint distributions of two datasets $\mathcal{S}_A$ and $\mathcal{S}_B$.

### A. Covariate Shift

This form of shift occurs when the covariate distribution $p(\boldsymbol{X})$, e.g., the distribution of the velocities, differs between datasets, but the posterior distribution $p(y|\boldsymbol{X})$ remains the same [16]. In highway driving data, this can be observed when recording data at different locations, or in countries where the driving rules and drivers' behavior might differ. For example, the average driving velocity on a German highway will be higher than on a US highway; or the traffic density on different highway sections might differ. A covariate shift can lead to the ML algorithm to fit the training data well, but the model can be misspecified on the test data. One can use importance sampling to reweigh the samples' contribution to the estimation error and compensate for a covariate shift [16].

### B. Prior Probability Shift

A prior probability shift occurs when the prior distribution $p(y)$, e.g., the probability of a lane change, differs but the likelihood, $p(\boldsymbol{X}|y)$, remains the same [15]. Since the prior probability is usually estimated on one dataset, this estimate might not be valid for other datasets. For example, when extracting driving maneuvers from a highway dataset, there are far more samples with the label "lane keeping" than there are with the label "lane change"; moreover the label "emergency stop" is rare. Thus, it is possible to use a priori knowledge of the problem at hand to estimate which prior probabilities are more or less likely. Since we are trying to predict the label $y$, it is unsurprising that this form of distributional shift will affect the performance of an ML algorithm.

### C. Concept Shift

A concept shift, or concept drift [17], can be defined as the case where either the posterior distribution $p(y|\boldsymbol{X})$, e.g., the probability of a lane change for a given scenario, or the likelihood $p(\boldsymbol{X}|y)$, e.g., the probability that a given scenario describes a lane change, differs between datasets [18]. Such a shift can result from a change in the causal relationship between the input data and the corresponding labels. In highway driving data, this form of shift can easily occur. For example, since the implicit driving rules are baked into highway driving datasets certain driving maneuvers will occur in different datasets, e.g., on German highways, vehicles should not overtake on the right-hand side, making the probability for this maneuver low. This is the most difficult form of distributional shift for ML algorithms to deal with [19].

### D. Sources of Distributional Shifts

1) Sample Selection Bias: This can occur whenever data are collected in the real-world under self-imposed constraints. Thus, the data which are collected might not accurately represent the true underlying distribution [15]. In highway driving datasets a sample selection bias could occur due to, e.g., the time of collection (during rush-hour), the location on the highway (just after an on-ramp), or the weather (filming during bad weather). Additionally, since datasets are usually collected by different teams, the post-processing techniques can differ.

2) Imbalanced Dataset: In multi-class classification tasks, one class could be rarer than others. To overcome this, researchers can randomly sub-sample the over-represented classes and artificially "balance" the dataset such that each class is represented by the same number of samples. In highway driving, lane changes are much rarer than lane keeping maneuvers, so researchers balance datasets before training ML algorithms on them. This is a sample selection bias with a known selection bias.

3) Domain Shift: As defined in [15], this occurs when the interpretation of the measurements, e.g., the measurement units, varies for different datasets. For example, highway driving datasets can be collected by teams in countries which use the metric measurement units, e.g., kilometers, whereas other teams might use the imperial measurement units, e.g., miles. This can be compensated for by mapping the representations from one measurement unit to another. Another example could be recording a

| Covariate | Unit | Description |
|---|---|---|
| $v_{\text{lat.}}$ | m/s | Lateral velocity |
| $v_{\text{long.}}$ | m/s | Longitudinal velocity |
| $d_{\text{ahead}}$ | m | Dist. to vehicle ahead in the current lane |
| $d_{\text{behind}}$ | m | Dist. to vehicle behind in the current lane |
| $d_{\text{l, ahead}}$ | m | Dist. to vehicle ahead in the left lane |
| $d_{\text{l, behind}}$ | m | Dist. to vehicle behind in the left lane |
| $d_{\text{r, ahead}}$ | m | Dist. to vehicle ahead in the right lane |
| $d_{\text{r, behind}}$ | m | Dist. to vehicle behind in the right lane |

TABLE I: The tracked covariates for each vehicle.

| | Left | Right | Keep | Total |
|---|---|---|---|---|
| highD (unbalanced) | 2274<br>(5.88%) | 2587<br>(6.69%) | 33822<br>(87.43%) | 38683<br>(100%) |
| highD (balanced) | 2274 | 2274 | 2274 | 6822 |
| NGSIM (unbalanced) | 1555<br>(2.53%) | 500<br>(0.81%) | 59382<br>(96.66%) | 61437<br>(100%) |
| NGSIM (balanced) | 500 | 500 | 500 | 1500 |

TABLE II: The number of lane change (left and right) and lane keeping (keep) scenarios in each dataset, before and after dataset balancing.

highway scenario from a bird's-eye view or from a vehicle driving in traffic—depending on the vantage point the same scenario can have various interpretations, e.g., due to occlusions.

4) Source Component Shift: This occurs when the data are sampled from a number of different sources, e.g., sensors, or if different sub-populations are measured. For example, the type of camera used to record highway driving data can affect the dataset. Different vehicles will also show different covariate distributions, e.g., trucks drive slower than cars. Here, the source of measurement directly causes different covariates and targets to be captured in the datasets [15].

## III. Problem Formulation

### A. Datasets

We investigate the distributional shifts between two publicly available, and widely used, realistic highway driving datasets. The first dataset was collected by the NGSIM program on US highways. The researchers filmed the trajectories of vehicles for a period of time on both the I-80 freeway [6] and the US highway 101 [5]. These datasets were recorded from multiple cameras fixed on top of skyscrapers close to the highways. In total, 1.5 hours of driving data were collected; they include a few hundred lane changes each. Since both datasets were collected within the scope of the same project, we combine them into a single NGSIM dataset. The second dataset is summarized in the highD dataset [4]. These data were recorded by flying a drone above various German highway segments. Similar to the NGSIM dataset, the recordings from different locations are combined into a single dataset. In total, more than 16 hours of highway driving data were collected which contain thousands of lane changes.

We extract covariates of each tracked vehicle from both dataset (see Tab. I). If a tracked vehicle is in the outer most left (or right) lane, the lateral distance to vehicles to the left (or right) of this vehicle is set to zero since there are no vehicles further left (or right) than it. Since the NGSIM dataset is recorded in imperial units, we convert all measurements into metric units for easier comparison. This can be noted as an example of a domain shift (cf. Subsec. II-D3). Furthermore, the NGSIM dataset is recorded at a sampling rate of 10 Hz, whereas the highD dataset is recorded at a sampling rate of 25 Hz. Thus, we sub-sample the highD dataset to have comparable datasets. This is an example of a source component shift (cf. Subsec. II-D4).

In the end, we summarize the tracked covariates in a multi-variate time series signal

$$\boldsymbol{X} = [\boldsymbol{x}[1], \boldsymbol{x}[2], \ldots, \boldsymbol{x}[N]] \in \mathbb{R}^{\Gamma \times N}, \tag{1}$$

where at each time-stamp $n \in \{1, \ldots, N\}$, we summarize the $\Gamma$ covariates in a vector $\boldsymbol{x}[n] = [x_1[n], x_2[n], \ldots, x_\Gamma[n]]^T \in \mathbb{R}^\Gamma$. We take scenarios where each vehicle is tracked for at least $N$ time-stamps, and the event, i.e., the lane change, we want to classify occurs within this time duration. In our experiments, we consider 8 s prior to the lane change, i.e., $N = 80$.

### B. Lane Change Maneuver Prediction

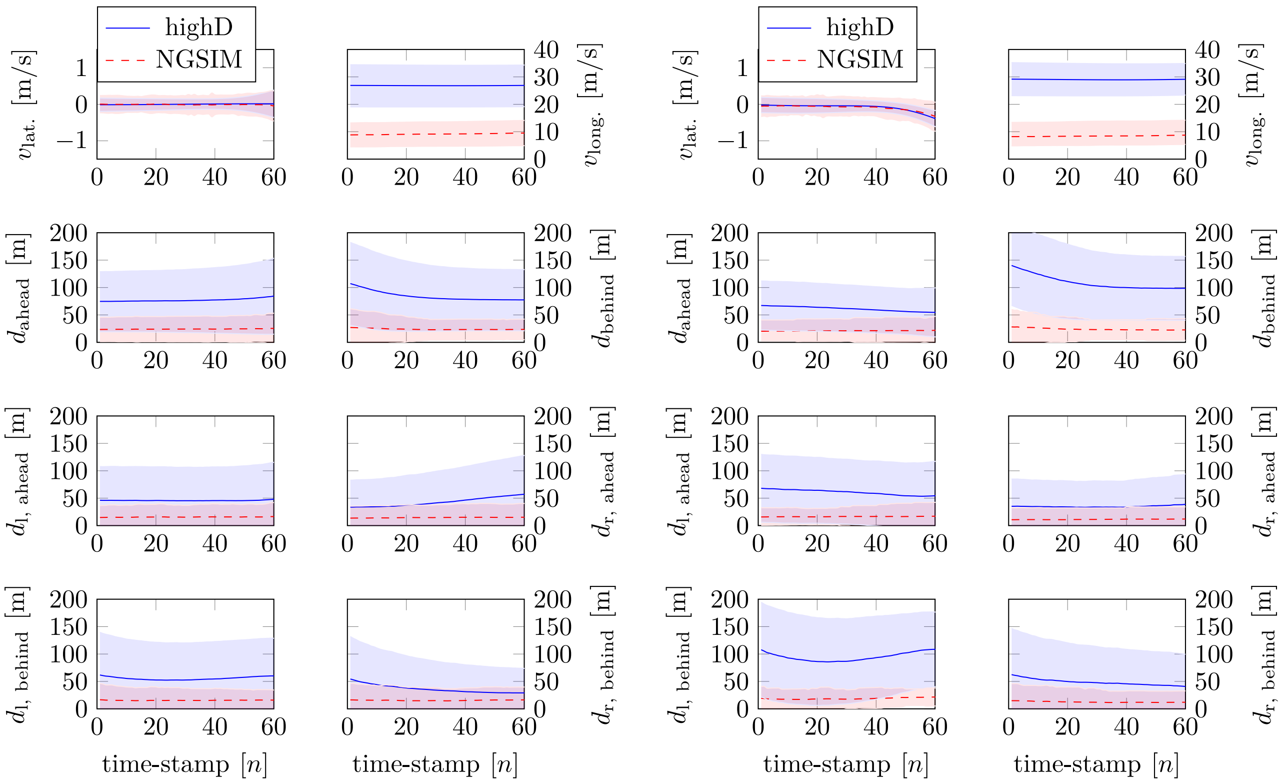
We predict if surrounding vehicles are going to change lanes on a highway, i.e., we track the vehicles surrounding the ego vehicle over time (see (1)), and we label each scenario with the label $y \in \mathcal{Y} = \{-1, 0, +1\}$, representing left lane changes, lane keeping, and right lane changes, respectively. The label corresponds to the driving maneuver which occurs at time-stamp $N$, i.e., for lane changes the center of mass of the vehicle crosses the lane marking at time-stamp $N$. Furthermore, we assume a fixed prediction horizon of 2 s (20 time-stamps), i.e., the classifiers are trained on inputs with $N' = N-20$ time-stamps to predict the driving maneuver which occurs in 2 s.

In the end, the training dataset is summarized as

$$\mathcal{S}_l = \left\{(\boldsymbol{X}^{(1)}, y^{(1)}), \ldots, (\boldsymbol{X}^{(M_l)}, y^{(M_l)})\right\}, \tag{2}$$

where each input is a multi-variate time-series datum $\boldsymbol{X} \in \mathbb{R}^{\Gamma \times N'}$ and each class label is $y \in \mathcal{Y}$. In our simulations, we consider an input of $N' = 60$ time-stamps. The index $l$ represents either the highD or the NGSIM dataset; $M_l$ is the number of samples in dataset $l$.

Since both datasets have a different number of lane-change and lane keeping scenarios, it is important that we perform dataset balancing to train the ML-based classifiers (cf. Subsec. II-D2). We uniformly random sample scenarios from each class to ensure that the number of samples per class is equal. The total number of scenarios before and after dataset balancing can be seen in Tab. II. There are many more lane keeping scenarios than lane changes before re-balancing the datasets. This imbalance is unsurprising



(a) Average covariate values for all driving maneuvers.

(b) Average covariate values for the left lane changes.

Fig. 1: Qualitative analysis of the datasets by visualizing the average covariate values at each time-stamp for each dataset. The shaded areas represent one standard deviation away from the mean.

since lane changes are relatively rare driving maneuvers when compared to lane keeping.

## IV. Detecting Distributional Shifts in Highway Data

### A. Qualitative Analysis

First, we plot the average value of each covariate from the datasets introduced in Subsec. III-A. In Fig. 1a, we plot the average covariate value at each time-stamp–this represents the average covariate distribution for each covariate at each time-stamp. We observe that the average longitudinal velocity (upper right sub-plot) is much higher in the highD dataset. This can be expected, since this dataset was collected on a German highway. Moreover, we observe that the distances (lower six sub-plots) to other vehicles is much smaller in the NGSIM dataset, which implies that the traffic density is higher. The average trend of some of the signals is also different between the two datasets, e.g., the covariate $d_{\text{r, ahead}}$ grows on average in the highD dataset and stays relatively constant in the NGSIM dataset. This could be explained by different driving styles between the two countries. The variance of the covariates in the highD dataset is also larger than in the NGSIM dataset. This could affect an ML-based algorithm trained on this dataset. Thus, we can qualitatively conclude that there is a covariate shift (cf. Subsec. II-A) between the highD and the NGSIM dataset.

Another qualitative analysis is to visualize the likelihood distribution for the different driving maneuvers. To this end, we plot the average value of the covariates corresponding to a left lane change from both datasets in Fig. 1b, i.e., the mean of $p(\boldsymbol{X}|\{y=-1\})$. We observe that the average lateral velocity for left lane changes is almost identical for both datasets. However, the means of the other covariates differ significantly, especially the longitudinal velocity. We can see that the trend of the distance covariates is different between the datasets. This qualitative analysis indicates that there is also a concept shift (cf. Subsec. II-C) between the two datasets. Similar conclusions can be drawn when investigating the distribution of the other driving maneuvers.

### B. Quantitative Analysis

Next, we employ a two-sample statistical hypothesis test to verify whether or not the samples in two datasets originate from the same underlying probability distribution.

1) Statistical Hypothesis Tests: Suppose we have two datasets, $\mathcal{S}_A = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_M\}$ and $\mathcal{S}_B = \{\boldsymbol{x}'_1, \ldots, \boldsymbol{x}'_{M'}\}$ where the samples $\boldsymbol{x} \sim P$ and $\boldsymbol{x}' \sim Q$ are i.i.d. sampled from the probability distributions $P$ and $Q$, respectively.

We assume $\boldsymbol{x}, \boldsymbol{x}' \in \mathcal{X}$. We are interested in distinguishing between two hypotheses: the null hypothesis, $H_0 : P = Q$, and the alternative hypothesis, $H_\mathrm{A} : P \neq Q$. To this end, a test statistic $T : \mathcal{X}^M \times \mathcal{X}^{M'} \rightarrow \mathbb{R}$ is constructed based on the samples in both datasets. To deal with the situation where the distribution of the test statistic is unknown, we use the bootstrapping method [20], i.e., we uniformly resample the union of the datasets $\mathcal{S}_A \cup \mathcal{S}_B$ without replacement, to estimate the empirical distribution of the test statistic under the null hypothesis $H_0$.

After deriving the test statistic, we can calculate a $p$-value of the test statistic to estimate the statistical significance. This is the probability of the two-sample test returning a test statistic as large as the test statistic calculated on the datasets $\mathcal{S}_A$ and $\mathcal{S}_B$ (before bootstrap resampling) when $H_0$ is true. Thus, the null hypothesis is rejected if the $p$-value lies under a pre-defined significance value $\alpha$ and accepted otherwise; $\alpha$ is usually set to 0.01.

2) Kernel Test Statistic: A popular choice of a non-parametric test statistic used for two-sample tests is based on the Maximum Mean Discrepancy (MMD) measure, see, e.g., [21] for more details. To define the MMD, we first define a positive definite kernel $k : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$ in a Reproducing Kernel Hilbert Space (RKHS) $\mathcal{H}$, where $k(\boldsymbol{x}, \boldsymbol{x}') = \langle \phi(\boldsymbol{x}), \phi(\boldsymbol{x}') \rangle_\mathcal{H}$ with a corresponding feature mapping $\phi \in \mathcal{H}$. The MMD measure between two probability distributions can be defined as [21, Lemma 4],

$$\mathrm{MMD}^2(P, Q) = \|\boldsymbol{\mu}_P - \boldsymbol{\mu}_Q\|^2_\mathcal{H}, \tag{3}$$

with the mean embedding of the distribution $P/Q$ defined as $\boldsymbol{\mu}_{P/Q} \in \mathcal{H}$. Thus, the MMD measure compares all moments of the distributions in the RKHS. The MMD measure is equal to 0 if and only if $P = Q$ [21, Lemma 5].

If we have two datasets $\mathcal{S}_A$ and $\mathcal{S}_B$ whose samples are assumed to be drawn from $P$ and $Q$, respectively, we can state an unbiased empirical estimate of the squared MMD measure as

$$\widehat{\mathrm{MMD}}^2(\mathcal{S}_A, \mathcal{S}_B) = -\frac{2}{MM'} \sum_{i=1}^{M} \sum_{j=1}^{M'} k(\boldsymbol{x}_i, \boldsymbol{x}'_j) \tag{4}$$

$$+ \frac{1}{M(M-1)} \sum_{\substack{i=1 \\ j \neq i}}^{M} k(\boldsymbol{x}_i, \boldsymbol{x}_j) + \frac{1}{M'(M'-1)} \sum_{\substack{i=1 \\ j \neq i}}^{M'} k(\boldsymbol{x}'_i, \boldsymbol{x}'_j).$$

In our experiments, we employ the Gaussian kernel, $k(\boldsymbol{x}, \boldsymbol{x}') = \exp(-\gamma \|\boldsymbol{x} - \boldsymbol{x}'\|^2_2)$, where $\boldsymbol{x}, \boldsymbol{x}' \in \mathcal{X}$, and the bandwidth parameter $\gamma = 1/(D\sigma^2)$ with the input dimension $D$ and where we use $\sigma^2$ as the variance of all samples in the datasets $\mathcal{S}_A$ and $\mathcal{S}_B$.

3) Classifier Two-Sample Test: Alternatively, we can train a neural network to detect whether the samples come from the same distribution–we use the Classifier Two-Sample Test (C2ST) [22]. First, we create a training dataset $\mathcal{D}$ by taking an equal number of samples from both datasets, i.e.,

$$\mathcal{D} = \{(\boldsymbol{x}_i, y_i = 1)\}_{i=1}^{M_\mathcal{D}} \cup \{(\boldsymbol{x}'_j, y_j = 0)\}_{j=1}^{M_\mathcal{D}}, \tag{5}$$

where the samples $\boldsymbol{x} \in \mathcal{S}_A$ and $\boldsymbol{x}' \in \mathcal{S}_B$, and we take $M_\mathcal{D} \leq \min(M, M')$. Furthermore, we split the dataset $\mathcal{D}$ into $\mathcal{D}_\text{train}$ and $\mathcal{D}_\text{test}$, with $M_{\mathcal{D},\text{train}}$ and $M_{\mathcal{D},\text{test}}$ samples, respectively

| | MMD-Test [$p$-value] | $T_\text{C2ST}$ [Acc. %] |
|---|---|---|
| $\mathcal{S}_\text{highD}$ vs $\mathcal{S}_\text{NGSIM}$ | 0.0005 | 92.17% |
| $\mathcal{S}^\text{keep}_\text{highD}$ vs $\mathcal{S}^\text{keep}_\text{NGSIM}$ | 0.0005 | 88.00% |
| $\mathcal{S}^\text{left}_\text{highD}$ vs $\mathcal{S}^\text{left}_\text{NGSIM}$ | 0.0005 | 96.50% |
| $\mathcal{S}^\text{right}_\text{highD}$ vs $\mathcal{S}^\text{right}_\text{NGSIM}$ | 0.0005 | 96.00% |

(a) Hypothesis tests which reject the null hypothesis, indicating that there is a covariate shift and concept shifts. All MMD-test $p$-values are $< 0.01$ and $T_\text{C2ST}$ accuracies are $> 85\%$.

| | MMD-Test [$p$-value] | $T_\text{C2ST}$ [Acc. %] |
|---|---|---|
| $\mathcal{S}_\text{highD}$ vs $\mathcal{S}_\text{highD}$ | 0.5570 | 47.69% |
| $\mathcal{S}^\text{left}_\text{highD}$ vs $\mathcal{S}^\text{left}_\text{highD}$ | 0.5625 | 46.59% |
| $\mathcal{S}^\text{right}_\text{highD}$ vs $\mathcal{S}^\text{right}_\text{highD}$ | 0.3240 | 47.69% |
| $\mathcal{S}^\text{keep}_\text{highD}$ vs $\mathcal{S}^\text{keep}_\text{highD}$ | 0.4300 | 48.13% |
| $\mathcal{S}_\text{NGSIM}$ vs $\mathcal{S}_\text{NGSIM}$ | 0.5655 | 52.67% |
| $\mathcal{S}^\text{left}_\text{NGSIM}$ vs $\mathcal{S}^\text{left}_\text{NGSIM}$ | 0.2535 | 50.00% |
| $\mathcal{S}^\text{right}_\text{NGSIM}$ vs $\mathcal{S}^\text{right}_\text{NGSIM}$ | 0.6955 | 55.00% |
| $\mathcal{S}^\text{keep}_\text{NGSIM}$ vs $\mathcal{S}^\text{keep}_\text{NGSIM}$ | 0.5440 | 49.00% |

(b) Hypothesis tests which accept the null hypothesis, indicating no distributional shifts. All MMD-test $p$-values are $> 0.01$ and $T_\text{C2ST}$ accuracies are around 50%.

TABLE III: Results indicating a distributional shift between the NGSIM dataset and the highD dataset; no distributional shift was detected within either dataset.

Thus, we train a classification function $g : \mathcal{X} \rightarrow [0, 1]$, which approximates the posterior distribution $p(y_k = 1|\boldsymbol{x})$, when employing a sigmoid activation function at the output of the classifier.

A hypothesis test can now be performed by analyzing the accuracy of the classifier on the test dataset, i.e.,

$$T_\text{C2ST} = \frac{1}{M_{\mathcal{D},\text{test}}} \sum_{k=1}^{M_{\mathcal{D},\text{test}}} \mathbb{I}(\text{round}(g(\boldsymbol{x}_k)) = y_k), \tag{6}$$

with the indicator function $\mathbb{I}$. The test statistic $T_\text{C2ST}$ is Gaussian distributed with $T_\text{C2ST} \sim \mathcal{N}(1/2, 1/(4M_{\mathcal{D},\text{test}}))$ [22]. Thus, the null hypothesis is accepted when the test statistic $T_\text{C2ST}$ (the accuracy of the classifier) is around 50% and rejected when a pre-defined threshold is exceeded, e.g., $T_\text{C2ST} \geq 85\%$.

4) Quantitative Results: We estimate the $p$-value of the statistical test based on the MMD test statistic by resampling the datasets 2000 times. The C2ST results are depicted as the classification accuracy on $\mathcal{D}_\text{test}$. The C2ST network architecture is taken from [23, Sec. 2.2.2], with one output neuron.

We observe in Tab. IIIa that both the statistical hypothesis tests based on the MMD test statistic and the C2ST reject the null hypothesis for the whole dataset. The null hypothesis is rejected because the estimated $p$-value is lower than a threshold $\alpha = 0.01$ for the test based on the MMD statistic, and the accuracy of the C2ST is larger than 85%. The bottom three rows indicate that there is a concept shift (cf. Section II-C) between the two datasets for all driving maneuvers.

In Tab. IIIb, we show the results of the statistical hypothesis tests with samples only from one dataset, i.e., we split each dataset into two disjoint sets, and test whether the two sub-sets come from the same distribution. We observe that the hypothesis tests indicate that samples from within each dataset stem from the same distribution. Moreover, the driving maneuvers within each dataset also come from the same underlying distribution. These results indicate that combining the two NGSIM datasets into one dataset, as done in this analysis, was meaningful.

## V. Effects of Distributional Shifts on Learned Models

### A. Simulation Setup

We train ML-based classifiers to perform the lane change maneuver prediction task (cf. Section III-B). In total, we train three models: (i) a Recurrent Neural Network (RNN)-based algorithm with Long Term Short Term Memory (LSTM) cells as proposed by [24]; (ii) an RNN-based algorithm with Gated Recurrent Unit (GRU) cells as proposed by [25]; and (iii) a Convolutional Neural Network (CNN)-based algorithm with 1-D filters due to CNNs' good performance in time-series classification tasks, see, e.g., [23; 26]. We use the LSTM [24] and the GRU [25] architectures as proposed in the publications. The CNN architecture is taken from [23, Sec. 2.2.2].

We use the the cross entropy loss as a training loss,

$$\mathcal{L}(\boldsymbol{\Theta}) = - \sum_{m=1}^{M_{\text{train},l}} \sum_{k=1}^{K} t_k^m \ln(z_k^L), \qquad (7)$$

where $\boldsymbol{\Theta}$ are the network parameters, the one-hot-encoded true label $t_k^m \in \{0, 1\}$ for all of the training data from dataset $l$ with a total of $M_{\text{train},l}$ samples, and the output class probabilities $z_k^L$ defined as the output of the ML algorithm after the softmax activation function.

We split both balanced datasets into disjoint sets. We separate the highD dataset with a 80%/20%-split of the total number of samples for training and testing, respectively. The whole NGSIM dataset is used for testing the trained ML algorithms. Thus, we create the datasets $\mathcal{S}_{\text{highD, train}}$, $\mathcal{S}_{\text{highD, test}}$, and $\mathcal{S}_{\text{NGSIM, test}}$, respectively.

We train each algorithm using 5-fold cross-validation [27, Ch. 7.2]. Thus, we obtain 5 different classifiers for each model type. We train each algorithm for 50 training epochs using the Adam optimizer [28] with mini-batches of size 32 and an initial learning rate of $\beta = 0.001$. The algorithms are trained using $\mathcal{S}_{\text{highD, train}}$.

### B. Effect on Classifier Performance

To visualize the effect which the distributional shifts have on trained ML algorithms, we plot the accuracy of each classifier (we have 5 classifiers per model type) on the test sets from both datasets. This visualization method was introduced in [12].

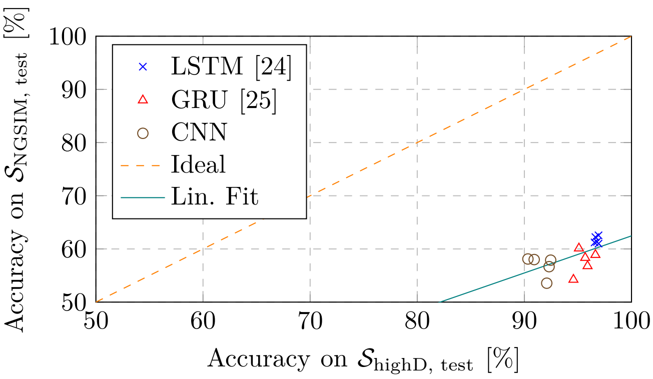


Fig. 2: The accuracy of the learned ML algorithms, trained on $\mathcal{S}_{\text{highD, train}}$, and tested on both test sets. All models perform better on the distribution they were trained on.

We observe in Fig. 2 that the accuracy on $\mathcal{S}_{\text{highD, test}}$ is strictly larger than on $\mathcal{S}_{\text{NGSIM, test}}$. All ML models achieve an accuracy of over 90% on the $\mathcal{S}_{\text{highD, test}}$. However, on $\mathcal{S}_{\text{NGSIM, test}}$, the models achieve an accuracy below 65%. We observe that the performance of the CNN-based classifiers drop by an average of 38%. The performance of the LSTM-based and the GRU-based classifiers dropped by an average of between 36% and 40%. This significant drop in classification performance highlights the problem of distributional shifts on safety critical driving functions.

The line labeled "Ideal" in Fig. 2 shows the ideal transferability between two datasets if there was no distributional shift. Moreover, the "Lin. Fit" line shows a linear function fit through the accuracy pairs. Its slope is $\approx 0.70$. This indicates that for every percentage point of accuracy improvement on highD, the model will gain less than one percentage point on NGSIM.

## VI. Conclusions

Motivated by the challenge of validating ML-based algorithms in safety critical driving functions, we turn our attention to the data which is used to train these driving functions. First, we recapitulate the different types of distributional shifts which can occur in highway driving data, and we discuss when these shifts may occur. Subsequently, we demonstrate that a distributional shift exists between two widely used, public highway driving datasets. We provide both a qualitative and a quantitative analysis of the distributional shifts between the datasets. Furthermore, we show that these shifts impact the performance of ML-based algorithms trained on the datasets.

We argue that an initial step in creating a safety-argument to validate ML-based driving functions is to analyze the distribution of the data which is used to train them. Using the analysis proposed in this paper, an engineer can begin to investigate the distributional shifts between various training and test datasets. If a distributional shift is detected, the challenge remains to detect when a vehicle leaves the distribution it was trained on, or to robustify the ML-based algorithm against such distributional shifts. Thus, to extend this research, one could investigate out-of-distribution detection, continual learning methods, or transfer learning methods.

## References

[1] R. Salay, R. Queiroz, and K. Czarnecki, "An Analysis of ISO 26262: Using Machine Learning Safely in Automotive Software," 2017. [Online]. Available: http://arxiv.org/abs/1709.02435

[2] P. Koopman and M. Wagner, "Toward a Framework for Highly Automated Vehicle Safety Validation," SAE Tech. Papers, 2018.

[3] G. Schwalbe and M. Schels, "A Survey on Methods for the Safety Assurance of Machine Learning Based Systems," in 10th Eur. Congr. on Embedded Real Time Softw. and Syst. (ERTS), 2020.

[4] R. Krajewski, J. Bock, L. Kloeker, and L. Eckstein, "The highd dataset: A drone dataset of naturalistic vehicle trajectories on german highways for validation of highly automated driving systems," in Intell. Transp. Syst. Conf.. (ITSC). IEEE, 2018.

[5] J. Colyar and J. Halkias, "NGSIM - US Highway 101 Dataset," 2006. [Online]. Available: https://www.fhwa.dot.gov/publications/research/operations/07030/index.cfm

[6] ——, "NGSIM - Interstate 80 Freeway Dataset," 2006. [Online]. Available: https://www.fhwa.dot.gov/publications/research/operations/06137/index.cfm

[7] A. Rudenko et al., "Human Motion Trajectory Prediction: A Survey," The Int. J. of Robot. Res., vol. 39, no. 8, 2020.

[8] D. Amodei et al., "Concrete Problems in AI Safety," 2016. [Online]. Available: http://arxiv.org/abs/1606.06565

[9] S. Burton, L. Gauerhof, and C. Heinzemann, "Making the Case for Safety of Machine Learning in Highly Automated Driving," in Comput. Safety, Rel., and Secur. (SAFECOMP). Lecture Notes in Comput. Sci., vol. 10489. Springer, Cham, 2017.

[10] S. Shafaei, S. Kugele, M. H. Osman, and A. Knoll, "Uncertainty in machine learning: A safety perspective on autonomous driving," in Comput. Safety, Rel., and Secur. (SAFECOMP). Lecture Notes in Comput. Sci., vol. 11094. Springer, Cham, 2018.

[11] A. Torralba and A. A. Efros, "Unbiased Look at Dataset Bias," in Comput. Vision and Pattern Recognit. (CVPR). IEEE, 2011.

[12] B. Recht, R. Roelofs, L. Schmidt, and V. Shankar, "Do ImageNet classifiers generalize to ImageNet?" in Int. Conf. on Mach. Learn. (ICML). PMLR, 2019.

[13] P. W. Koh et al., "WILDS: A Benchmark of in-the-Wild Distribution Shifts," in Int. Conf. on Mach. Learn. (ICML). PMLR, 2021.

[14] A. Malinin and Others, "Shifts: A Dataset of Real Distributional Shift Across Multiple Large-Scale Tasks," arXiv, 2021. [Online]. Available: http://arxiv.org/abs/2107.07455

[15] J. Quinonero-Candela, M. Sugiyama, N. D. Lawrence, and A. Schwaighofer, Dataset Shift in Machine Learning. MIT Press, 2009.

[16] H. Shimodaira, "Improving predictive inference under covariate shift by weighting the log-likelihood function," J. of Statistical Planning and Inference, vol. 90, no. 2, 2000.

[17] G. I. Webb, R. Hyde, H. Cao, H. L. Nguyen, and F. Petitjean, "Characterizing concept drift," Data Mining and Knowl. Discovery, vol. 30, no. 4, 2016.

[18] V. M. Souza, D. M. dos Reis, A. G. Maletzke, and G. E. Batista, "Challenges in benchmarking stream learning algorithms with real-world data," Data Mining and Knowl. Discovery, vol. 34, no. 6, 2020.

[19] J. G. Moreno-Torres, T. Raeder, R. Alaiz-Rodríguez, N. V. Chawla, and F. Herrera, "A unifying view on dataset shift in classification," Pattern Recognit., vol. 45, no. 1, 2012.

[20] G. J. Szekely and M. L. Rizzo, "Testing for equal distributions in High Dimension," InterStat, vol. 5, no. 16.10, 2004.

[21] A. Gretton, K. M. Borgwardt, M. J. Rasch, B. Schölkopf, and A. Smola, "A Kernel Two-Sample Test," J. of Mach. Learn. Res., vol. 13, 2012.

[22] D. Lopez-Paz and M. Oquab, "Revisiting classifier two-sample tests," Int. Conf. on Learn. Representations (ICLR), 2017.

[23] H. Ismail Fawaz, G. Forestier, J. Weber, L. Idoumghar, and P. A. Muller, "Deep learning for time series classification: a review," Data Mining and Knowl. Discovery, vol. 33, no. 4, 2019.

[24] H. Q. Dang, J. Fürnkranz, A. Biedermann, and M. Hoepfl, "Time-to-lane-change prediction with deep learning," in Intell. Transp. Syst. Conf. (ITSC). IEEE, 2017.

[25] Z. Yan et al., "Time to lane change and completion prediction based on Gated Recurrent Unit Network," in Intell. Vehicles Symp. (IV). IEEE, 2019.

[26] O. De Candido et al., "Towards Feature Validation in Time to Lane Change Classification using Deep Neural Networks," in Intell. Transp. Syst. Conf. (ITSC). IEEE, 2020.

[27] T. Hastie, R. Tibshirani, and J. Friedman, The Elements of Statistical Learning - Data Mining, Inference, and Prediction. Springer, 2009.

[28] D. P. Kingma and J. L. Ba, "Adam: A method for stochastic optimization," in Int. Conf. on Learn. Representations (ICLR), 2015.