WHAT DRIVES PREDICTIONS IN TRAFFIC FORECASTING?
DATA VALUATION FOR DEEP LEARNING ON TIME SERIES

SENIOR THESIS

Dylan V. Chou
Department of Statistics and Data Science
Carnegie Mellon University
Pittsburgh, PA 15213
dvchou@andrew.cmu.edu

Peter Freeman
Department of Statistics and Data Science
Carnegie Mellon University
Pittsburgh, PA 15213
pfreeman@cmu.edu

February 28, 2022

ABSTRACT

Recent research into forecasting highway traffic has explored advanced deep learning frameworks. However, deep learning model predictions lack explainability. The severe consequences of inaccurate traffic predictions on commute time and highway accidents makes explainability important for road traffic models. Explainable AI paradigms such as feature importance or data valuation have been applied to deep learning traffic models that used shapley values and counterfactual explanations, but fewer papers have measured importance over clusters of cyclical data. Our work will be part of a case study on Du et al's long-short term memory (LSTM) encoder-decoder model to fill the gaps in its explainability. To this end, we provide a data valuation framework for the steepest and least steep portions of weekend and workday traffic data over varying lengths of contiguous time observations. Our implementation averages the prediction differences after substituting random samples from an estimated multivariate gaussian over each univariate data portion: traffic flow, speed and journeytime. For multivariate valuation of flow, speed and journeytime, we will adapt the incorporation of indirect causal changes into the shapley value calculation after changing a feature value, which results in sampling from a different multivariate conditional distribution. We will conduct experiments comparing our method's conclusions to kernel SHAP and tree SHAP, opening up to further investigation if any discrepancies in results arise. The functions used to visualize changes in the time series predictions due to random sampling will be stored in a Python package.

Keywords Data Valuation · Deep Learning · Traffic Forecasting

1 Introduction

1.1 Deep Learning in Traffic Forecasting

There has been interest in using deep neural network models to predict traffic information, such as future traffic speeds, and help evaluate traffic efficiency. Incorporation of spatial attributes along with temporal information in graph-based convolutional neural networks became prevalent since [8] when Yu and others presented a neural network framework combining spatial graph representations of roadway traffic connectivity and temporal convolution layers with a specified K_t kernel width to capture adjacent traffic observations in the time series. Additional spatial information collected through sensor stations in the Caltrans Performance Measurement System (PeMS). [31] built on top of the convolutional recurrent neural network paradigm by modeling the stochastic properties of traffic as a diffusion process that's represented by a random walk. Chen and others [4] also implemented a convolutional neural network and used multiple convolutions to model greater time dependencies between traffic observations, and expanded on understanding parameter impact on prediction errors in their deep convolutional neural network. Their network model captures local temporal patterns and global congestion trends of the data, measuring the impact of the number of time slots and number of days available in the training and testing datasets on model forecast errors. Geng and others [12] modeled non-Euclidean correlations between spatially distant regions for ride-hailing services using multi-graph convolutions.

They defined correlation between regions as their connectivity, similarity between their surrounding regions, and their proximity. Regarding diffusion convolutional neural networks, [9] couples temporal attributes with spatial dependencies, which are captured in graph diffusion convolution, or a convolution block constructed from a graphical random walk with a restart probability and a trainable transition matrix. The base paradigm of a convolutional neural network for spatio-temporal data has been used to model stochastic properties of traffic, connectivity between regions in a graph of directed traffic flows, and correlation between those traffic regions.

Frameworks for general traffic forecasting have also been presented as [10] defined a concrete deep learning framework to follow. Yu et al proposed a mixture deep long short term memory (LSTM) network that models both normal traffic and less frequent road accidents. They forecast traffic flow during peak-hour periods with greater variation in truck traffic and vehicle movement patterns. Frameworks have also applied traffic forecasting to solve problems of general travel time prediction. [14] provided a framework to predict travel times along road routes using a spatio-temporal component and an attribute component where sampled vehicle trajectories are sampled and passed through a geo-convolutional layer. Attributes such as driver habits, weather, time and distance traveled along a route are embedded and passed into an attention layer that learns weights for local paths. Ultimately, the model learns a multi-task problem where it learns sampled local paths along with the overall path. [11] implemented a similar framework, but applied the model to accurate public bus transportation.

The application of deep learning frameworks to time-series forecasting has been met with interpretable models and explainability techniques. [13] forecasted multi-step travel times based on roadway traffic predictions through matched spatiotemporal traffic patterns in speed contour plots. Gray-level Co-occurrence matrices are used to identify pairwise speeds that co-occur at various distances along an urban expressway, which can identify similar traffic patterns during forecasts. To forecast traffic patterns, Chen and others [21] draw on an analogy between a single neural network layer and an iteration in the process of reducing noise in data, which allows for greater explainability of their model with the use of an analogy. Other papers were able to make intuitive conclusions about what data their models performed best on as well as the explainability of top performing models. Yang et al [5] trained a recursive neural network (RNN) among other machine learning models on the traffic data provided by the Minnesota Department of Transportation [6], where they found that prediction accuracy improves during the peak hours of traffic in the data. In Yang et al's model and similar deep learning techniques [2, 7], they are able to make intuitive conclusions that predictions further into the future result in larger errors and some models perform better than others based on comparisons of performance metrics. Baric et al [22] benchmarked the explainability of different models on about 10 datasets using confidence based on standard deviation of model parameter estimates. They focus on comparisons of explainability in different models. Ismail and colleagues [23] studied saliency methods in time series by first evaluating importance of time steps before computing the feature importance at a time step. Their analysis remedies issues with feature importance of multiple features over many time steps in a time series. Cho and colleagues [26] visualize temporally activated patterns by identifying network nodes with high activation values in a channel. Visualizations show portions of the traffic time series colored based on convolutional network channels that are activated the most when data during those portions are passed into the model. In turn, the visualization of most activated nodes can explain which portions of the data, such as inflection points, are most valuable during a model's learning process.

As described in [15], new deep learning architectures for short-term traffic prediction add to the already massive pool of papers that do not address the caveats of their approaches, such as the black-box nature of their models or computation costs to train the model. One of the challenges brought up in the paper is the lack of explainable AI perspectives on traffic forecasting. Some papers that do address explainability study ways that upstream and downstream traffic data impact model predictions [16], the features learned by the first layer in an autoencoder [18], and the importance of specific traffic flow time steps on forecasts [17].

1.2 Feature and Data Valuation on Time Series

Feature and data valuation approaches can offer explainability in deep traffic forecasting models. Popular approaches for data valuation on time series include shapley values and counterfactual explanations. Shapley values originated from cooperative game theory where a group of players needs to distribute the gain attained through collective cooperation. Some players may have contributed more than others, which is quantified in the importance of their contribution using shapley values. Shapley values are theoretically quite costly as the shapley value of a feature value is $\phi_j(val) = \sum_{S \subseteq \{x_1,\ldots,x_p\} \setminus \{x_j\}} \frac{|S|!(p-|S|-1)!}{p!}(val(S \cup \{x_j\}) - val(S))$, which requires the sum of all prediction differences between the inclusion of feature x_j and its exclusion over all permutations of subsets of features in a model; the computation is in exponential-time and NP-complete. To circumvent the computation costs, Strumbelj and others [24] approximated shapley values using Monte Carlo simulations to construct new data examples where an average of m randomly sampled differences is computed: $\frac{1}{M} \sum_{m=1}^{M} \widehat{f}(x_{+j}^{m}) - \widehat{f}(x_{-j}^{m})$. $\widehat{f}(x_{+j}^{m})$ is the prediction when a random number of feature values in x are replaced by a randomly sampled vector z except for the j^{th} feature value. $\widehat{f}(x_{-j}^{m})$

takes a randomly sampled vector z that replaces some random number of features in x, but also replaces feature value x_j with z_j. With applications to air quality prediction based on nitrogen dioxide forecasts [32], sales deals prediction based on sales related activities [19], and general time series forecasts for trend cycles [20], shapley values can quantify and compare specific atmospheric factors for air quality prediction, types of tools used during the sales process and frequency of data logging for sales deals prediction, and lagged time series variables for trend-cyclical data, respectively. However, any domain-specific application of shapley values is limited by features with interventional effects and underlying causal dependencies between features [34]. Causal shapley values were then introduced by Heskes and others [37], which measures the total effect of a shapley value to be the sum of the direct effect and the indirect effect:

$$\begin{aligned}\phi_i(\pi) &= \mathbb{E}[f(\mathbf{X}_{\bar{S}}, \mathbf{x}_{\underline{S}\cup i})|do(\mathbf{X}_{\underline{S}\cup i} = \mathbf{x}_{\underline{S}\cup i})] - \mathbb{E}[f(\mathbf{X}_{\bar{S}\cup i}, \mathbf{x}_{\underline{S}})|do(\mathbf{X}_{\underline{S}} = \mathbf{x}_{\underline{S}})] \\ &= \mathbb{E}[f(\mathbf{X}_{\bar{S}}, \mathbf{x}_{\underline{S}\cup i})|do(\mathbf{X}_{\underline{S}} = \mathbf{x}_{\underline{S}})] - \mathbb{E}[f(\mathbf{X}_{\bar{S}\cup i}, \mathbf{x}_{\underline{S}})|do(\mathbf{X}_{\underline{S}} = \mathbf{x}_{\underline{S}})] + \\ &\quad \mathbb{E}[f(\mathbf{X}_{\bar{S}}, \mathbf{x}_{\underline{S}\cup i})|do(\mathbf{X}_{\underline{S}\cup i} = \mathbf{x}_{\underline{S}\cup i})] - \mathbb{E}[f(\mathbf{X}_{\bar{S}}, \mathbf{x}_{\underline{S}\cup i})|do(\mathbf{X}_{\underline{S}} = \mathbf{x}_{\underline{S}})]\end{aligned}$$

The direct effect is the expected change in prediction when feature X_i is replaced with x_i without changes to the other features while indirect effect accounts for distribution changes after the replacement of feature X_i. Provided a causal directed acyclic graph (DAG) of features, one can use the partial ordering of features in a causal graph and sample accordingly from empirical or conditional variable distributions.

Counterfactual explanations for time series model predictions are used to define importance of particular time series observations by replacing an observed value with a generated quantity. [25] used a forward feed counterfactual (FFC) to replace i^{th} time series observations with a generated one. The effect of this replacement is defined as the importance of the i^{th} observation. [27] proposed an instance-based counterfactual explanation technique to find a series of changes to a time series T_q predicted to be in class c that the system would predict to be in class $c^{'}$. Ates and others [28] presented a greedy algorithm to provide counterfactual explanations for multivariate time series, which offers a minimum number of time series substitutions that could maximize the probability of the time series being classified as class c. The counterfactual explanations were not generated, but instead used an x_{dist} would be chosen from the time series training set. However, Keane and others [29] presented three main deficiencies of current counterfactual methods: the lack of user studies or proven preference by end-users, absence of proxies estimating the "psychological" distance of counterfactual explanations, and the open-ended nature in defining how few features a counterfactual explanation should have or "plausibility" of the explanations.

1.3 Our Contribution

Past papers on traffic forecasting are saturated in spatio-temporal models, general deep earning frameworks, comparisons in the explainability of different models, and interpretable traffic models. More broadly, data valuation in time series forecasting primarily concerns minimal changes in features to change prediction outcomes or average changes in prediction output based on randomly generated or sampled feature values as means of measuring value in data. However, among traffic forecasting models, there is a dearth of explainability techniques that evaluate data importance on model predictions while taking into account indirect causal effects, which is a gap that we fill. While the importance of traffic flow time steps have been studied in [17], we undertake an interpretable and methodical approach by considering contributions of historical traffic over defined regions of the data. We examine Du et al's paper that presented a deep LSTM network traffic model trained on cyclical, multivariate – traffic speed, flow, journeytime – data with distinguishable patterns between weekday and weekend traffic. Our contribution will add an explainable AI framework to understand the impact of traffic observations during the weekend or workweek on Du et al's LSTM based encoder-decoder neural network model, and allow practitioners to probe forecast contributions over different portions of cyclical data. To account for an underlying causal structure between traffic observations of traffic speed, flow, and journeytime, we will apply a recent causal discovery algorithm – the structural vector autoregression (SVAR) FCI algorithm based on [35] – that modified the fast causal inference (FCI) algorithm to provide the causal DAG of traffic observations. This work is an addition of data valuation to traffic forecasting literature that tests a hypotheses on periodic data that traffic operators can study. Our probing tool would not be used as a standalone technology, but incorporated in future tests to ascertain whether its addition will increase the accuracy of traffic operators in predicting traffic model behavior. The dearth of evaluation metrics predictive of model explanation efficacy motivates the creation of probing tools. These tools may boost efficacy when tested with the appropriate metrics on users in specific fields such as traffic monitoring [39].

2 Problem Domain

Model probing frameworks can be applied to a variety of types of models trained under different architectures, which requires that our data valuation package be agnostic to how the model is trained or how forecasts are made.

2.1 Objective

Our time series data D is composed of traffic flow, speed, and journeytime observations. We want to answer the question of whether there are parts of the periodic data in the highway dataset that have a substantial contribution to Du et al's [2] model predictions. Namely, we want to answer the following three questions:

- Are highway traffic observations towards the start or end of a day more important in Du et al's model predictions than data in the middle of the day?
- How does the role of traffic observations with larger surrounding changes in weekend traffic flow, speed, and journeytime on Du's model predictions compare to that of workday traffic?
- How does the role of traffic observations with smaller surrounding changes in weekend traffic flow, speed, and journeytime on Du's model predictions compare to that of workday traffic?

Our objective is to identify specific regions in cycles of highway traffic data that drive Du et al's model predictions, which can be helpful in adopting holistic and local model explainability, as defined in a 2018 paper by Lipton [3].

We hypothesize that data with drastic changes, or with large negative or positive consecutive differences between observations, are more important in Du et al's model predictions than traffic that changes less over time. We also claim that the impact of data within a cycle or at the border of two cycles on Du's model predictions depends on the data's rate of change in the time series. We put forth that workday traffic data contributes more to Du et al's model forecasts than weekend traffic due to greater changes in workday traffic. Due to the imbalance between weekend and weekday traffic data, we consider data valuation for the two types of traffic separately.

In general, forecasting traffic information is important in helping traffic operators design strategies to mitigate traffic congestions depending on the traffic flow data. Adding explainability to an existing deep learning model can inform traffic operator decisions. Neural networks make decisions based on passing input values forward through the network, then updating network weights backwards according to a loss function. These models are not intuitive as it's harder to explain why a neural network made a specific decision despite having higher accuracy. Explainability can be presented through the identification of specific characteristics in the data that contribute to deep learning traffic forecasts.

2.2 Datasets

We test our probing tool on the three datasets: Highways England 2013 traffic data, California Department of Transportation Agency Performance Measurement System (PeMS) 04, and PeMS08. The Highways England dataset [41] provides information on average speeds, flows and journey times [1]. There are two types of data in the dataset: Motorway Incident Detection and Automatic Signalling (MIDAS) and Traffic Monitoring Units (TMU) or inductive loops. MIDAS traffic sites use induction loops, and some radar technology, 500 meters apart that detect incidents on the road whereas TMU solely uses induction loops. The PeMS datasets were collected by 3900 sensors throughout the metropolitan areas of California, where PeMS04 is traffic data recorded in the San Francisco Bay area by 3848 sensors on 29 roads from Jan. 1, 2018 to Feb. 28, 2018 and PeMS08 is traffic data in the San Bernardino area detected by 1979 sensors on 8 roads from 7/1/2016 to 8/31/2016. Table 1 summarizes the aforementioned datasets in further detail.

Dataset Name	# Observations	Recording Device(s)	Data Types	Duration
PeMS04	16,992 (307 detectors)	Induction loops	Traffic Flow, Speed, Journey Time	5 minutes
PeMS08	17,856 (170 detectors)	Induction loops	Traffic Flow, Speed, Journey Time	5 minutes
Highways England	37,564	Loops and Radar	Traffic Flow, Speed, Journey Time	15 minutes

Table 1: An overview of the periodic traffic datasets.

With respect to the Highways England Dataset [1], our problem is the following: we are given multivariate data D as a time series of traffic flow, traffic speed, and traffic journeytime. For traffic flow, we have $flow_1, flow_2, ...flow_t$, where the subscript denotes the time step for an observed traffic flow per 15 minutes. The experimental dataset used by Du et al [2] contains 37,564 traffic flow data points from 1/1/2013 to 2/28/2014 in 15 minute intervals for the major road "Site A414" in England. The PeMS datasets also provide the aforementioned multivariate data, but are stored in a 3d numpy array `(number of observations, number of detectors, number of variables recorded (e.g., traffic flow))` rather than a dataframe with each column corresponding to a traffic variable as in the Highways England Dataset.

3 Methods

Our probing tool is intended to be a wrapper for models trained on periodic traffic data, where we would take in the functions used to train and test the traffic models `train, test` and the model object `net`. For instance, in Guo et al [40], we have the function `predict_and_save_results_mstgcn` that takes in a slew of arguments including `net`. `data_loader` is enumerable data that can be modified and `prediction` is the output savable to a numpy array. Agnostic to the training methodology, we train the model and perturb the data (e.g., `data_loader` in Guo et al's code) based on probing by the user and display the`prediction` differences over portions of the forecast ADD HERE. As a first step for setup, the long-short term memory neural network model in Du et al's 2019 paper will be reconstructed using the Encoder, Decoder, and Attention Decoder architectures in Keras, which will be trained on the traffic data recorded every 15 minutes from the highways agency dataset.

We will select k different contiguous data chunk sizes $[s_1, .., s_k]$ to identify regions of the time series with the greatest rates of change:

Algorithm 1 find_greatest_change Algorithm: We estimate the maximum rate of change for data vectors of some length s_i through.

```
procedure FIND_GREATEST_CHANGE($data = [x_1, ..., x_t],\ chunk\_sizes = [s_1, s_2, ..., s_k]$)
    $start\_times \leftarrow []$
    for $s_i$ $in$ $chunk\_sizes$ do
        $max\_change \leftarrow 0$;
        $max\_change\_start\_time \leftarrow 1$;
        $counts \leftarrow \{pos : numPositive(data[1 : s_i + 1]), neg : numNegative(data[1 : s_i + 1])\}$;
        for $start$ $in$ $[1, ..., len(data) - s_i + 1]$ do
            if $counts[pos] > 0$ $and$ $counts[neg] > 0$ then
                $continue$;
            end if
            $e = EuclideanNorm(data[start : start + s_i])$    ▷ Returns the euclidean norm of the pairwise differences in the data vector. Norm is computed on an $(s_i - 1) \times 1$ vector.
            if $e > max\_change$ then
                $max\_change = e$;
                $max\_change\_start\_time = start$;
            end if
            if $start < len(data) - s_i + 1$ then
                $counts[pos]$ += $(1$ $if$ $data[start + s_i + 1] > 0$ $else$ $0) - (1$ $if$ $data[start] > 0$ $else$ $0)$;
                $counts[neg]$ += $(1$ $if$ $data[start + s_i + 1] < 0$ $else$ $0) - (1$ $if$ $data[start] < 0$ $else$ $0)$;
            end if
        end for
        $start\_times.append(max\_change\_start\_time)$;
    end for
    return $start\_times$    ▷ Length of $start\_times$ is $len(chunk\_sizes)$
end procedure
```

For each chunk size s_i, we estimate the changes over a portion of the time series that doesn't have both positive and negative changes between consecutive observations. We are interested in inspecting monotonically increasing or decreasing sequences of traffic flow, speed or journeytime and storing the greatest euclidean norm. The euclidean norm of an $n \times 1$ vector $\mathbf{x}$ is $\sqrt{x_1^2 + ... + x_n^2}$, which can compare consecutive changes in time series vectors. For each chunk size s_i and returned start time j, the portion $[x_j, ..., x_{j+s_i}]$ would have the greatest rate of change. To find the smallest change in traffic flow, speed or journeytime, we record the time series vector with the smallest euclidean norm.

After identifying the region of greatest or smallest rate of change in traffic data $\mathbf{r} = [x_j, ..., x_{j+s_i}]$, we scan over the $data$ and ascertain the portion of each day that matches the most with the region. We compare consecutive differences in traffic flow, speed, or journeytime of the region $\mathbf{d_r} = [d_j, ..., d_{j+s_i-1}]$ and the current portion of the $data$ $\mathbf{d_p} = [d_j^{'}, ..., d_{j+s_i-1}^{'}]$ using cosine similarity: $\frac{\mathbf{d_r} \cdot \mathbf{d_p}}{||\mathbf{d_r}||||\mathbf{d_p}||}$. Figure 1 shows an example of the most similar regions in the green regions of each cycle.

We approximate the value that each traffic observation has on future predictions within regions of large and small consecutive traffic changes. For univariate model explanations, we measure importances at a traffic observation level over multiple different chunk sizes and the attributes flow, speed and journeytime. The importance of the i^{th} traffic observation in the first green region $R_1 = [x_1, ..., x_{chunk\_size}]$ on a workday will be computed by randomly sampling from the estimated multivariate gaussian distribution of $x_1, x_2 - x_1, ..., x_{chunk\_size} - x_{chunk\_size-1}$. The differences from $x_2 - x_1$ to $x_{chunk\_size} - x_{chunk\_size-1}$ can denote $diff_1$,...,$diff_{chunk\_size-1}$. Each random sample would result in a new vector $[z_1, z_1 + diff_1, ..., z_1 + \sum_{p=1}^{chunk\_size-1} diff_p]$. Based on Strumbelj's approximation [24], we can construct two new regions to replace R_1 and compute their difference in predicted traffic observations in the following traffic cycle:

$$[x_1, ..., x_{i-1}, z_1 + \sum_{p=1}^{i-1} diff_p, ..., z_1 + \sum_{p=1}^{chunk\_size-1} diff_p]$$

$$[x_1, ..., x_i, z_1 + \sum_{p=1}^{i} diff_p, ..., z_1 + \sum_{p=1}^{chunk\_size-1} diff_p]$$

We would repeat this over each chunk size, traffic attribute, and cycle length repeatedly for M trials and average them to obtain the direct effect of intervening with the value of the i^{th} traffic observation.

For multivariate model explanations, we would also take into account the causal dependencies between traffic speed, flow and journeytime observations so these relations can be incorporated into the shapley value computation as an indirect effect. We will apply the SVAR FCI causal discovery algorithm in [35] that applies conditional independence tests and identifies the underlying causal relationship between traffic flow, speed and journeytime. After we obtain a general causal ordering, we identify the causal chain graph of strongly connected features, as shown in Figure 2. Referring to Aas and others' approach [36], we will estimate multivariate gaussian distributions to account for dependent features. We will randomly sample observations from the estimated conditional multivariate gaussian to ascertain the value of the three traffic features:

Algorithm 2 Intervention effect incorporated into shapley values from [37], where S is the set of indices with known feature values $\mathbf{x}_S$.

```
procedure ValueFunction($S$)
    for $k \leftarrow 1$ to $N_{samples}$ do
        for all $j \leftarrow 1$ to $|\tau|$ do                    ▷ Run over all chain components in causal order.
            if confounding($\tau_j$) then
                for all $i \in \tau_j \cap \bar{S}$ do
                    Sample $\tilde{x}_i^{(k)} \sim P(X_i|\tilde{\mathbf{x}}^{(k)}_{pa(\tau_j)\cap\bar{S}}, \mathbf{x}_{pa(\tau_j)\cap\bar{S}})$;    ▷ Drawn independently.
                end for
            else
                Sample $\tilde{\mathbf{x}}^{(k)}_{\tau_j\cap\bar{S}} \sim P(\mathbf{X}_{\tau_j\cap\bar{S}}|\tilde{\mathbf{x}}^{(k)}_{pa(\tau_j)\cap\bar{S}}, \mathbf{x}_{pa(\tau_j)\cap\bar{S}}, \mathbf{x}_{\tau_j\cap S})$;    ▷ Gibbs Sampling.
            end if
        end for
    end for
    $v \leftarrow \frac{1}{N_{samples}} \sum_{k=1}^{N_{samples}} f(\mathbf{x}_s, \tilde{\mathbf{x}}^{(k)}_{\bar{S}})$
    return $v$
end procedure
```

We will evaluate our data valuation method based on comparisons with baseline shapley value techniques from [17, 20] that include kernel SHAP and tree SHAP. If our method results in values that contradict those of the baseline, we would investigate further into the hypotheses we posed. Any 'good' sampled observation, or sample that's within distribution and not an outlier, from the multivariate gaussian distribution will be kept based on their local outlier factor (LOF) [38]:

$$reachability\_distance_k(A, B) = max\{k\_distance(B), d(A, B)\}$$

$$lrd_k(A) = \frac{1}{\frac{\sum_{B\in N_k(A)} reachability\_distance_k(A,B)}{|N_k(A)|}}$$

$$LOF_k(A) = \frac{\sum_{B\in N_k(A)} lrd_k(B)}{|N_k(A)| \cdot lrd_k(A)}$$

$N_k(A)$ denotes the k-nearest neighbors to point A, $k\_distance(A)$ finds the distance from A to its k^{th} nearest neighbor, $reachability - distance_k(A, B)$ is the true distance from A to B that's at least the distance from B to its k^{th} nearest

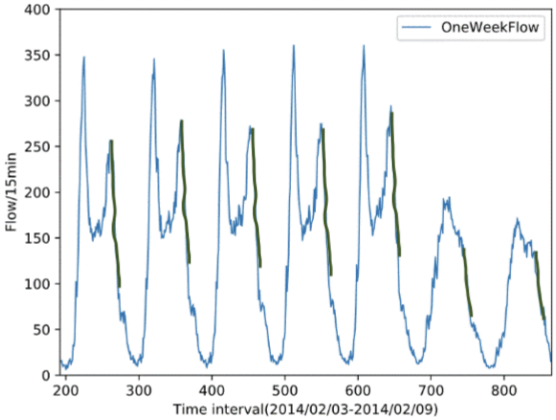

Figure 1: One week traffic flow data from Site A414 of the Highway Agency in England, based on [2]. The highlighted green is one example of regions of the time series that have similarly large changes in traffic flow.

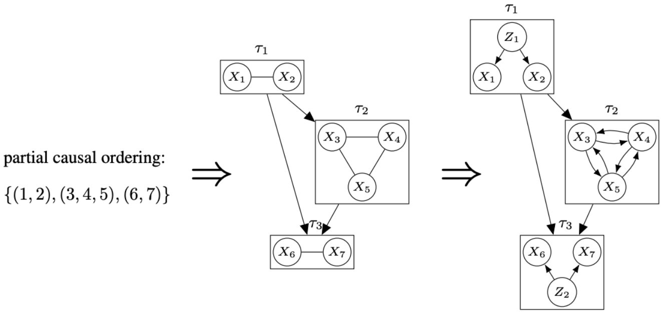

Figure 2: The transition from a partial causal ordering to causal chain graph. The sets of chained components are referred to as $\tau_{\overline{confounding}}$ and the set of variables within the same component is denoted by $\tau_{confounding}$.

neighbor, $lrd_k(A)$ is the local reachability density, and $LOF_k(A)$ is the local outlier factor of A. A could be an observation randomly sampled during our process and nearest neighbors would be observations from the traffic dataset.

Our data valuation approach is limited by the assumptions of equitability between similar regions, where two green regions in workweek traffic are interchangeable. For the project, we assume that such portions of the traffic time series are approximately equivalent and can replace one another during random sampling. Code will be documented in a Github repository and stored in an Python package.

References

[1] Highways Agency network journey time and traffic flow data, 2017. Last Updated 2018. Available: https://data.gov.uk/dataset/dft-eng-srn-routes-journey-times/

[2] Du, Shengdong, et al. “An LSTM based encoder-decoder model for MultiStep traffic flow prediction." 2019 International Joint Conference on Neural Networks (IJCNN). IEEE, 2019.

[3] Lipton, Zachary C. “The Mythos of Model Interpretability: In machine learning, the concept of interpretability is both important and slippery." Queue 16.3 (2018): 31-57.

[4] Chen, Meng, Xiaohui Yu, and Yang Liu. “PCNN: Deep convolutional networks for short-term traffic congestion prediction." IEEE Transactions on Intelligent Transportation Systems 19.11 (2018): 3550-3559.

[5] Yang, Xiaoxue, et al. “Evaluation of short-term freeway speed prediction based on periodic analysis using statistical models and machine learning models." Journal of Advanced Transportation 2020 (2020).

[6] Minnesota Department of Transportation. “Mn/DOT Traffic Data". *Datatools*, 22 March http://data.dot.state.mn.us/datatools/

[7] Tang, Jinjun, et al. “An improved fuzzy neural network for traffic speed prediction considering periodic characteristic." IEEE Transactions on Intelligent Transportation Systems 18.9 (2017): 2340-2350.

[8] Yu, Bing, Haoteng Yin, and Zhanxing Zhu. “Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting." arXiv preprint arXiv:1709.04875 (2017).

[9] Lu, Huakang, et al. “St-trafficnet: A spatial-temporal deep learning network for traffic forecasting." Electronics 9.9 (2020): 1474.

[10] Yu, Rose, et al. “Deep learning: A generic approach for extreme condition traffic forecasting." Proceedings of the 2017 SIAM international Conference on Data Mining. Society for Industrial and Applied Mathematics, 2017.

[11] Tran, Luan, et al. “DeepTRANS: a deep learning system for public bus travel time estimation using traffic forecasting." Proceedings of the VLDB Endowment 13.12 (2020): 2957-2960.

[12] Geng, Xu, et al. “Spatiotemporal multi-graph convolution network for ride-hailing demand forecasting." Proceedings of the AAAI conference on artificial intelligence. Vol. 33. No. 01. 2019.

[13] Zhang, Zhihao, et al. “Probe data-driven travel time forecasting for urban expressways by matching similar spatiotemporal traffic patterns." Transportation Research Part C: Emerging Technologies 85 (2017): 476-493.

[14] Wang, Dong, et al. “When will you arrive? estimating travel time based on deep neural networks." Thirty-Second AAAI Conference on Artificial Intelligence. 2018.

[15] Manibardo, Eric L., Ibai Laña, and Javier Del Ser. “Deep learning for road traffic forecasting: Does it make a difference?." IEEE Transactions on Intelligent Transportation Systems (2021).

[16] Sun, Shiliang, Changshui Zhang, and Guoqiang Yu. “A Bayesian network approach to traffic flow forecasting." IEEE Transactions on intelligent transportation systems 7.1 (2006): 124-132.

[17] Barredo-Arrieta, Alejandro, Ibai Laña, and Javier Del Ser. “What lies beneath: A note on the explainability of black-box machine learning models for road traffic forecasting." 2019 IEEE Intelligent Transportation Systems Conference (ITSC). IEEE, 2019.

[18] Wu, Yuankai, et al. “A hybrid deep learning based traffic flow prediction method and its understanding." Transportation Research Part C: Emerging Technologies 90 (2018): 166-180.

[19] Saluja, Rohit, et al. “Towards a Rigorous Evaluation of Explainability for Multivariate Time Series." arXiv preprint arXiv:2104.04075 (2021).

[20] Selvam, Santhosh Kumar, and Chandrasekharan Rajendran. “tofee-tree: automatic feature engineering framework for modeling trend-cycle in time series forecasting." Neural Computing and Applications (2021): 1-20.

[21] Chen, Siheng, Yonina C. Eldar, and Lingxiao Zhao. “Graph unrolling networks: Interpretable neural networks for graph signal denoising." arXiv preprint arXiv:2006.01301 (2020).

[22] Barić, Domjan, et al. “Benchmarking Attention-Based Interpretability of Deep Learning in Multivariate Time Series Predictions." Entropy 23.2 (2021): 143.

[23] Ismail, Aya Abdelsalam, et al. “Benchmarking Deep Learning Interpretability in Time Series Predictions." arXiv preprint arXiv:2010.13924 (2020).

[24] Štrumbelj, Erik, and Igor Kononenko. “Explaining prediction models and individual predictions with feature contributions." Knowledge and information systems 41.3 (2014): 647-665.

[25] Tonekaboni, Sana, et al. “Explaining time series by counterfactuals." (2019).

[26] Cho, Sohee, et al. “Interpreting Internal Activation Patterns in Deep Temporal Neural Networks by Finding Prototypes." Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining. 2021.

[27] Delaney, Eoin, Derek Greene, and Mark T. Keane. “Instance-based counterfactual explanations for time series classification." International Conference on Case-Based Reasoning. Springer, Cham, 2021.

[28] Ates, Emre, et al. “Counterfactual Explanations for Multivariate Time Series." 2021 International Conference on Applied Artificial Intelligence (ICAPAI). IEEE, 2021.

[29] Keane, Mark T., et al. “If only we had better counterfactual explanations: five key deficits to rectify in the evaluation of counterfactual XAI techniques." arXiv preprint arXiv:2103.01035 (2021).

[30] Parvatharaju, Prathyush S., et al. “Learning Saliency Maps to Explain Deep Time Series Classifiers." (2021).

[31] Li, Yaguang, et al. “Diffusion convolutional recurrent neural network: Data-driven traffic forecasting." arXiv preprint arXiv:1707.01926 (2017).

[32] García, María Vega, and José L. Aznarte. “Shapley additive explanations for NO2 forecasting." Ecological Informatics 56 (2020): 101039.

[33] Saluja, Rohit, et al. “Towards a Rigorous Evaluation of Explainability for Multivariate Time Series." arXiv preprint arXiv:2104.04075 (2021).

[34] Kumar, I. Elizabeth, et al. “Problems with Shapley-value-based explanations as feature importance measures." International Conference on Machine Learning. PMLR, 2020.

[35] Malinsky, Daniel, and Peter Spirtes. “Causal structure learning from multivariate time series in settings with unmeasured confounding." Proceedings of 2018 ACM SIGKDD Workshop on Causal Discovery. PMLR, 2018.

[36] Aas, Kjersti, Martin Jullum, and Anders Løland. “Explaining individual predictions when features are dependent: More accurate approximations to Shapley values." arXiv preprint arXiv:1903.10464 (2019).

[37] Heskes, Tom, et al. “Causal shapley values: Exploiting causal knowledge to explain individual predictions of complex models." arXiv preprint arXiv:2011.01625 (2020).

[38] Breunig, Markus M., et al. “LOF: identifying density-based local outliers." Proceedings of the 2000 ACM SIGMOD international conference on Management of data. 2000.

[39] Hase, Peter, and Mohit Bansal. “Evaluating explainable AI: Which algorithmic explanations help users predict model behavior?." arXiv preprint arXiv:2005.01831 (2020).

[40] Guo, Shengnan, et al. “Attention based spatial-temporal graph convolutional networks for traffic flow forecasting." Proceedings of the AAAI conference on artificial intelligence. Vol. 33. No. 01. 2019.

[41] Highways England - WebTRIS - Faqs, https://webtris.highwaysengland.co.uk/Home/Faqs