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
Thesis Advisor: Professor Peter Freeman

June 24, 2022

Abstract

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 make explainability important for road traffic models. Explainable AI paradigms such as feature importance or individual data observation valuation have been applied to deep learning traffic models, but less work has measured importance over clusters of cyclical data. Our work is part of a case study on the PEMS04 and PEMS08 datasets as well as the A414 Highways England 2013 dataset to fill the gaps in traffic neural network explainability. To this end, we provide a data valuation framework over specified ranges of traffic data. We compare prediction mean absolute error distributions after randomly substituting sampled regions most similar to the specified data portion, which allows us to compare importance estimations of different regions. The estimation method is evaluated on the basis of stability, or that the averages between two MAE distributions from repeated runs are consistent (e.g. the relative ordering between regions is always maintained).

Keywords Data Valuation · Deep Learning · Traffic Flow Forecasting

1 Introduction

1.1 Objective

Traffic congestion is a major problem in highways as it can lead to delays, pollution, and reduce the productivity of drivers. The rise of smart connected vehicle environments (CVEs) enables traffic operators to receive much more detailed information about traffic speeds, journey times of drivers, and overall traffic flow [46]. With traffic flow data collected in CVEs, companies such as Bosch security can train neural models to predict traffic flow, and thereby traffic congestion, and detect traffic bottlenecks. Supplementing intelligent transportation systems (ITS) with a traffic congestion prediction model allows traffic management to send over resources in areas with high predicted congestion beforehand, so congestion may be quickly alleviated. The importance of traffic congestion prediction has led to researchers training more accurate neural network architectures on traffic data from CVEs, such as the Caltrans Performance Measurement System (PeMS) dataset recorded by the California Department of Transportation and the A414 highway traffic dataset recorded by Highways England. Guo et al. [37] and Du et al. [2] have trained a graphical convolutional neural network on the PeMS datasets and a hierarchical convolutional neural network with gated recurrent units on the A414 dataset, respectively. While extensive work by researchers such as Guo and Du have presented architectures that improve traffic flow prediction accuracy, incorrect traffic flow predictions are left unexplained. Explainable AI frameworks on neural networks have studied how much individual time series observations or features contribute to model predictions, but less work has analysed the contribution of a specific set of time series observations on predictions.

Our work builds on the models that Guo and Du trained on the PeMS and A414 datasets, respectively, and investigates which data portions contribute the most to model predictions. Traffic operators can use these model explanations to suggest retraining the model if it focuses too heavily on unreliable traffic flow patterns during prediction. Our approach is to first obtain the baseline prediction errors of their models, then measure the importance that a model weighs region R by identifying similar regions to R based on cross correlation. We substitute R with one of the similar regions, and report the average increase in prediction error of the model on the perturbed data after 32 repeated runs, each having resampled R 25 times, as data importance. We then evaluate the stability of our estimation method, or how consistently the approach ranks one region as more important than another.

For the 2013 Highways England A414 dataset and PeMS datasets, we want to answer the question of whether there are parts of the traffic data that have a substantially greater contribution to model predictions. Namely, we consider two main questions:

- Are highway traffic observations towards the start or end of a day more important in model predictions than data in the middle of the day?
- How does the impact of traffic observations with larger changes in traffic flow on model predictions compare to those with smaller changes?

Our hypothesis is that traffic flow with drastic changes are more important towards Du et al.'s and Guo et al.'s model predictions than smaller changes. We also claim that the impact of data at the start or end of a given day on Du's model predictions depends on the data's rate of change in the time series. In turn, if data at the start of the day only has large positive increases in traffic flow, the importance of "start of day" traffic and regions with large increases in flow are the same.

With regards to testing our hypotheses, suppose we are estimating the importance of region R. Importance is the increase in prediction mean absolute error (MAE) after substituting R with similar regions, where MAE is computed as $\frac{1}{n}\sum_{i=1}^{n}|\widehat{y_i} - y_i|$. In other words, data importance is perceived as the contribution that a set of regions has on traffic flow predictions. y_i is the actual traffic flow observed at time step i, which is the number of vehicles passing over a detector every five (PeMS datasets) or fifteen (Highways England A414 dataset) minutes. $\widehat{y_i}$ is the predicted traffic flow by the model at time step i. In turn, MAE is the absolute difference between the predicted and actual number of vehicles passing over a detector every five or fifteen minutes. It is assumed that a region R with a greater increase in MAE, as a result of masking out regions similar to R, corresponds to greater importance. Our objective is to identify regions of highway traffic data that drive model predictions, which can be helpful in adopting holistic, local model explainability, as defined in a 2018 paper by Lipton [3]. The two aforementioned questions serve to guide our analyses.

The purpose of offering an intuitive comparison of importance between regions of traffic is for laymen, particularly traffic operators, to improve upon existing models. Not only can traffic operators receive predictions from a traffic flow model, but also a data importance report of what the model weighs with greater importance; namely, regions of traffic that led to the greatest losses in performance, or higher MAE, when masked with regions most similar to them are deemed of "greater importance". For instance, if a model relies significantly on one small portion of data to make future traffic flow predictions, traffic operators can call into question the robustness of the model and bring attention to retraining the model. 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 operators about real-time decision-making (e.g., managing vehicle operations or optimal route planning) alongside ITS. Not only can explainability frameworks help improve traffic prediction models by companies operating ITS, such as Bosch [45], but also improve models that GPS systems use to compute optimal routes amid dynamic traffic conditions.

1.2 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 Yu et al. [8] 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). Li et al. [31] built on top of the convolutional recurrent neural network paradigm by modeling the stochastic properties of traffic as a diffusion process that is 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, Lu et al. [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 Yu et al. [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. Wang et al. [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 travel times for sampled local paths along with the overall path. Tran et al. [11] implemented a similar framework, but applied the model to accurate travel time prediction for public bus transportation.

The application of deep learning frameworks to time-series forecasting has been met with interpretable models and explainability techniques. Zhang et al. [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 a given time step before computing the feature importance at a time step. Their analysis remedies issues with feature importance of multiple features at individual 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 the 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]. More recently, research is trending towards understandable systems [39, 40], where decision models are equipped with visualizations of Local Interpretable Model-Agnostic Explanations (LIME) and SHap Additive exPlanations (SHAP) values.

1.3 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 [35], which measures the total effect of a shapley value to be the sum of the direct effect and the indirect effect:

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

$$=\mathbb{E}[f(\mathbf{X}_{\bar{S}},\mathbf{x}_{\underline{S}\cup i})|do(\mathbf{X}_{\underline{S}}=\mathbf{x}_{\underline{S}})]-\mathbb{E}[f(\mathbf{X}_{\bar{S}\cup i},\mathbf{x}_{\underline{S}})|do(\mathbf{X}_{\underline{S}}=\mathbf{x}_{\underline{S}})]+$$

$$\mathbb{E}[f(\mathbf{X}_{\bar{S}},\mathbf{x}_{\underline{S}\cup i})|do(\mathbf{X}_{\underline{S}\cup i}=\mathbf{x}_{\underline{S}\cup i})]-\mathbb{E}[f(\mathbf{X}_{\bar{S}},\mathbf{x}_{\underline{S}\cup i})|do(\mathbf{X}_{\underline{S}}=\mathbf{x}_{\underline{S}})]$$

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. Tonekaboni et al [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. Delaney et al [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 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.

1.4 Our Contribution

Past papers on traffic forecasting are saturated with spatio-temporal models, general deep learning frameworks, comparisons in the explainability of different models, and interpretable traffic models. Data valuation in time series forecasting has primarily concerned minimal changes in variable features required to change prediction outcomes or average changes in prediction error after randomly sampling feature values. Existing data valuation methods, such as in [17], only measure the impact of individual traffic observations on model predictions and have gone in the direction of accounting for underlying causal relationships when measuring the importance of multiple time series variables. However, for the univariate traffic forecasting problem, there is a dearth of explainability techniques that evaluate data importance on model predictions over contiguous time series observations, which is a gap that we fill. While the importance of traffic flow data has been studied in [17], we undertake a methodical and interpretable approach by considering contributions of regions of historical traffic defined through automated top-k searches of the steepest and least steep traffic flow changes and manual labeling of start of the day and end of the day traffic. We examine three cyclical time series datasets – PEMS04, PEMS08, and Highways England A414 2013 datasets – that aim to forecast traffic flow. Our explainable AI framework estimates the impact of masking out traffic observations during the start of the day, end of the day, increasing, and decreasing portions of PeMS and A414 data on Guo et al's and Du et al.'s model prediction errors, respectively, and allow practitioners to probe forecast contributions over different portions of cyclical data. Our work can extend current data valuation methods in traffic forecasting as we may flexibly test hypotheses about data importance on traffic flow patterns that traffic operators can understand. The framework may be 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 [36].

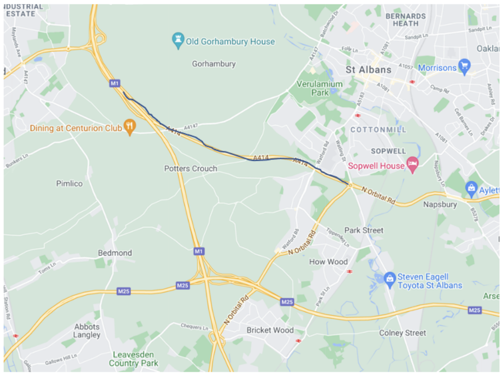


Figure 1: Region of Highway A414 in England between junction M1-J7 and A405 where average speed and journey time as well as traffic flow are being measured, which is shown as a black line. Journey time refers to the duration of travel of a vehicle over the black line. Traffic flow refers to the number of vehicles per 15 minutes that pass over the outlined portion of Highway A414.

2 Datasets

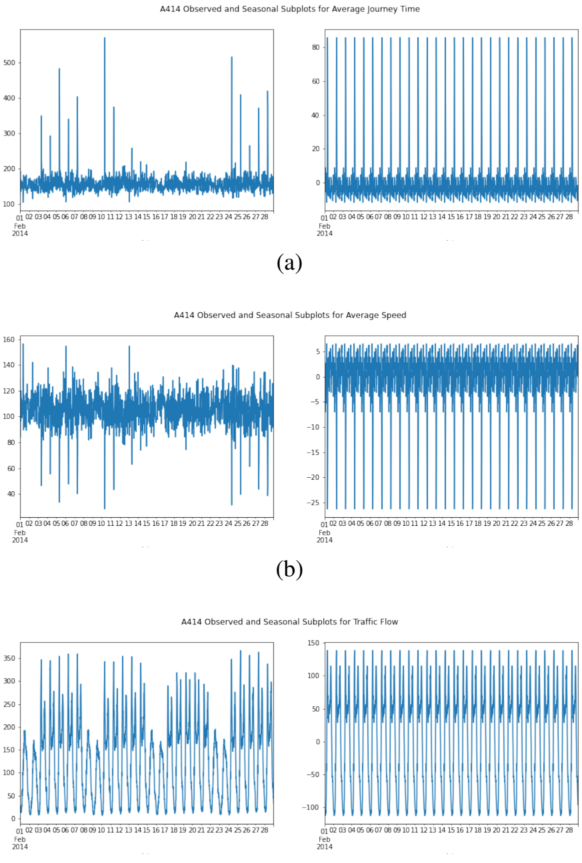
The datasets section describes the three traffic flow prediction datasets that the estimation method was run over. We break down the dataset sizes, what they were recorded on, features that were recorded, and the time that each traffic observation spans. Three seasonality graphs are shown in Figure 2 as evidence of cyclic behavior in the traffic flow data to motivate perturbations primarily on traffic flow. It is preferable to have concrete and similar cycles in the perturbed data because the approach described in the *Methodology* section relies on substituting the data regions with those most similar to it.

Overview The probing tool was tested on three datasets: Highways England 2013 A414 traffic data, California Department of Transportation Agency Performance Measurement System (PeMS) 04, and PeMS08. The Highways England dataset [38] 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, Occupancy	5 minutes
PeMS08	17,856 (170 detectors)	Induction loops	Traffic Flow, Speed, Occupancy	5 minutes
A414	37,564	Loops and Radar	Traffic Flow, Speed, Journey Time	15 minutes

Table 1: An overview of the periodic traffic datasets. The number of observations includes the training, validation, and testing datasets. Data types are specific features that are recorded in each of the datasets. Duration corresponds to the time span that is aggregated into a single observation in the dataset. Durations and number of observations are shown to provide a comparison of the datasets before explanation of the estimation method is discussed and experimental results are presented.

Highways England A414 dataset and Du et al. With respect to the Highways England Dataset [1], our problem is the following: given multivariate data D as a time series of traffic flow (number of vehicles passing through the portion of A414 drawn in Figure 1 over a fifteen-minute interval), traffic speed, and traffic journeytime (average time of a vehicle to make the journey along A414 over 15 minutes), predict future traffic flow by some number of k timesteps. For instance, traffic flow has observations $flow_1, flow_2, ...flow_t$ denoting a traffic flow observation every 15 minutes. The seasonality of the three variables is shown in Figure 2, where it is evident that traffic flow has concrete periodicity that will allow us to identify sufficient similar regions to resample in our estimation procedure.



(a)

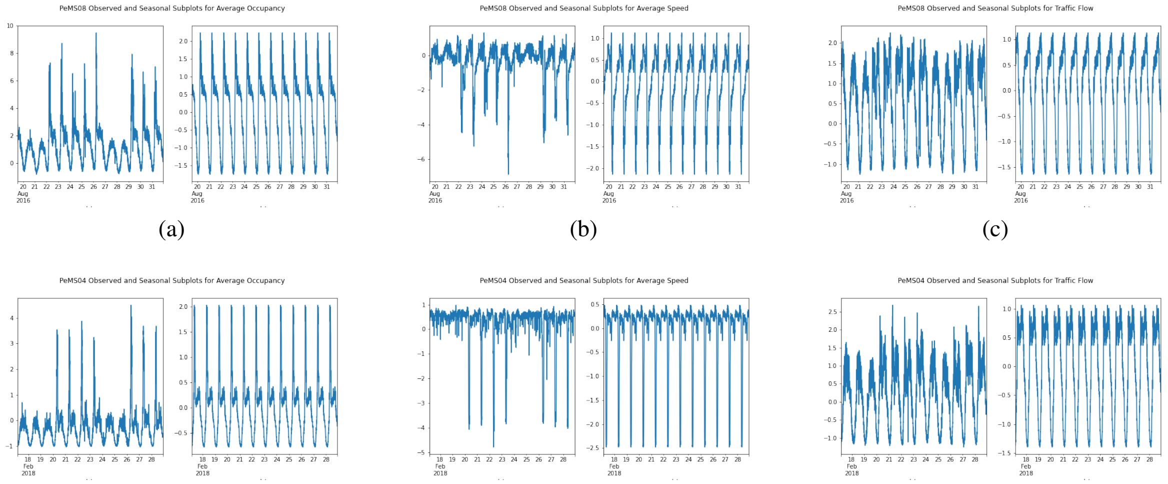
(b)

(c)

Figure 2: (a) Seasonality Plot for Average Journey Time (2/1/2014-2/28/2014) (b) Seasonality Plot for Average Speed (c) Seasonality Plot for Traffic Flow. The trend plots, particularly the second graph from the top in (c), illustrate how traffic flow is the only data that contains definitive and repeated regions of data preferable in the resampling similar regions portion of our estimation approach. The seasonal component in average journey time on the A414 highway has anomalous spikes and the average speed has anomalous dips. The trend in traffic flow is most clearly cyclic and contains similar regions to sample from, devoid of irregular spikes or dips that may impact resampling.

In Figure 2, the seasonal component of traffic flow is most evidently cyclic with a relatively concrete period. This allows for sufficiently many similar data portions to be resampled during perturbations as there aren't anomalous spikes or dips in the data such as in average speed or journey time. In turn, focus is put on perturbing the importance of traffic flow over other variables for the A414 dataset. The experimental dataset used by Du et al. [2] contains 37,564 traffic flow data points from 1/1/2013 to 12/31/2013 and 2/1/2014 to 2/28/2014 in fifteen-minute intervals for the major road “Site A414" in England. The data is split 80%/20% into training and validation data, respectively, over the 2013 data for Du et al.'s model and the 2014 data is used for testing. The A414 dataset is a dataframe that contains the feature columns of “AverageJT" (average vehicle journey time along highway A414 between A405 and junction M1 J7), “AverageSpeed" (in kilometers per hour, the average speed measured at the start and end of the A414 road), and Flow (the number of vehicles expected to be detected by the National Traffic Information Service (NTIS) link to road A414 in the span of 15 minutes). The geographical portion of highway A414 that all data was recorded across is illustrated in the map in Figure 1.

PeMS datasets and Guo et al. The PeMS datasets provide average speed (average speeds of vehicles every 5 minutes), occupancy (fraction of a 5 minute interval where a vehicle is over a detector), and traffic flow (number of vehicles passing over detector in 5 minute interval), but are stored in a 3d array `(# observations, # detectors, # variables recorded, # points ahead of current)` rather than a dataframe as in the Highways England Dataset. The PeMS04 dataset have 307 detectors and have up to 11 data points following each observation i per input dimension (average speed, occupancy, and traffic flow). PeMS04 is split into a training (1/1/2018-2/5/2018) and testing dataset (2/5/2018-2/28/2018). The three dimensions of the PeMS04



(a) (b) (c)

(d) (e) (f)

Figure 3: (a) PeMS08: Plot of Average Occupancy Input Tensor Values at Detector 0 (b) PeMS08: Plot of Average Speed Input Tensor Values at Detector 0 (c) PeMS08: Plot of Traffic Flow Input Tensor Values at Detector 0 (d) PeMS04 Seasonality of Average Occupancy. (e) PeMS04 Seasonality for Average Vehicle Speed (f) PeMS04 Seasonality for Traffic Flow. The three graphs along the first row are shown as visual evidence for resampling similar regions of data from cyclic traffic flow in PeMS08. The seasonal portion of the bottom three graphs illustrate the nearly identical cycles of traffic flow that may be resampled in our estimation method in PeMS04. Other modalities of traffic data do not provide as many non-irregular similar regions in cycles as traffic flow.

input training data and single traffic flow training output are illustrated along the second row of Figure 3. During our estimation method's perturbations, focus is put on estimating the importances of traffic flow in PeMS04 over other variables as there are more similar regions across the cycles in PeMS04's test data. Traffic flow in PeMS04's test data has fewer spikes or dips such as those seen in average occupancy or speed. When identifying data regions of greatest and least changes, our estimation method may have group together the anomalous spikes or dips with the periodic rise or fall in speed and occupancy. The PeMS08 dataset has the same number of data points following each observation, but instead has 170 detector sites recording data on flow, speed, and occupancy, or how long vehicle spends over a detector. As shown in the first row of Figure 3, traffic flow in PeMS08 test data is similar to PeMS04's as both have regular periodic rise and falls without anomalous spikes that may lead to inaccurate estimation of a data region; our estimation method may group together anomalies with periodic behavior when perturbing average speed and occupancy. In turn, we isolate perturbations to PeMS08's traffic flow test data.

3 Methodology

In this section, the problem formulation is presented on how the importance of contiguous traffic data "R" is estimated, followed by the procedure: first identifying the appropriate dataset that the user should call perturbations on, then running the perturbations that substitute out "R" with an independent block of data most similar to R. The act of masking out region R is to estimate how much the prediction error rises when the model is not exposed to R. The resulting changes in mean average error (MAE) are grouped by 25 observations, whose distributions are used during evaluation to ensure the estimates are stable. We have a sample size of 25 observations because we chose a 95% confidence level ($\alpha = 0.05$) for the MAEs produced on the PeMS datasets and 90% confidence level for the A414 datasets. Namely, for PeMS08, a sufficient sample size to attain 95% confidence in the returned MAEs with a one-sided margin of 0.01625 random sampling error in the MAEs of our sample of size n is as follows: $M_{error}/2 = 1.96 * \frac{\sigma}{\sqrt{n}} \implies n = \frac{1.96^2\sigma^2}{(0.0325)^2} \implies n = \frac{1.96^2(0.0410)^2 4}{(0.0325)^2} = 24.46 \approx 25$, where $\sigma = 0.0410$ is the standard deviation among 800 computations of MAE increases for the PeMS08 dataset. Similarly, for PeMS04, $n = \frac{1.96^2(0.035)^2 4}{0.0325^2} = 17.821 \approx 18$, so 25 is a sufficient sample size for the MAEs. 25 is a sufficient sample size for the A414 dataset at the 90% confidence level with an error margin of 0.965 in MAE units: $n = \frac{1.645^2(1.459)^2 4}{(0.965)^2} = 24.743 \approx 25$. For the economic use of virtual machine resources, we adhere to the 25 sample size when testing confidence intervals at the $\alpha = 0.0015625$ level and will test our estimation method with larger samples in future work. After repeated runs of the estimation method,

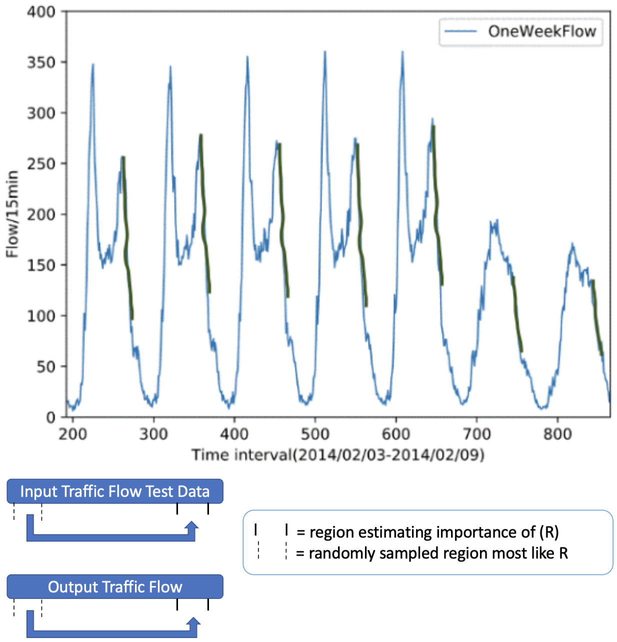


Figure 4: The blue graph shows one week of traffic flow data from Site A414 of the Highway Agency in England, based on [2]. The green portions of data are examples of regions with similarly drastic decreases in traffic flow, which are substituted out for one another during perturbations. The perturbation method between two similar green regions is also illustrated as one would substitute a similar past region with the region we are estimating the importance of (R).

the consistency of the conclusions over the runs (i.e. whether one region is deemed more important than another) is evaluated. Consistency is measured as the proportion of runs, out of the 32 total runs, that result in the same statistically significant outcome. For each run, a 2-sample bootstrap t-test between two regions with 25 observations corrected for multiple testing using the Bonferroni correction is run.

3.1 Problem Formulation

Given a sequence of n time series observations to perturb $D = [x_1, ..., x_n]$ along with a region $R = D[i : j]$ we are estimating the importance of, we substitute R with its top $k = 15$ most similar regions: $[R_1, ..., R_{15}]$. In turn, we would either substitute out a region that follows R to avoid look-ahead bias (i.e., when the model is exposed to batch data that is has not been available yet) or substitute R with a preceding region. The paradigm used to classify the importance of features most similar to R is illustrated in Figure 4. The region of importance R' is effectively substituted out with data from another region R', presumed to be equivalent to R. We assume R and R' to fall under the same “concept" or feature along the traffic flow time series in the time series D, which we resample from to estimate its importance. Wang et al. [43]’s overarching definition of concept classes falling into separate subspaces explains that a “concept" is associated with a label that is mapped to a set of features. Namely, for traffic flow prediction, we can map an abstract label, such as “rapid alleviation of congestion", to regions of traffic defined to have the greatest rates of increase in traffic flow. We apply similar mappings for the smallest rates of increase in traffic flow and greatest rates of decrease in traffic flow. The importance of the region R is measured as the increase in prediction mean absolute error after masking out regions with identical features as R.

3.2 Defining the Regions

To identify whether data at the start or end of a day in test data is more significant in making predictions, we measure the importance the first and last four hours per day in the input test data of the A414 dataset and first and last four hours and 10 minutes of the PeMS datasets. To answer whether greatest or least consecutive changes in traffic flow contribute more to model forecasts, we define regions with a minimum threshold of points that need to be consecutively increasing or decreasing that have the greatest or lowest euclidean norm, or $norm = \sqrt{x_1^2 + x_2^2 + ...x_n^2}$ for a sequence of traffic flow observations $x_1, x_2, ..., x_n$. Table 2 summarizes the parameters used to define the regions of largest decreases, largest increases, and smallest increases in traffic flow.

Dataset Region Name	Top K	% Consecutive	# Neighbors in Smoothing	# Observations
PeMS04 [Largest Increases]	50	70%	15	50
PeMS04 [Smallest Increases]	50	70%	5	50
PeMS04 [Largest Decreases]	40	60%	5	50
PeMS08 [Smallest Increases]	10	80%	10	50
PeMS08 [Largest Increases]	15	70%	5	50
PeMS08 [Largest Decreases]	12	60%	5	50
A414 [Largest Decreases]	10	90%	2	16
A414 [Largest Increases]	25	60%	2	16
A414 [Smallest Increases]	20	30%	2	16

Table 2: The name of the data region specifies the dataset and type of region. The top K column is the number of data regions to maintain in the heap. The % consecutive column is the proportion of observations among "# Observations" that are consecutively increasing or decreasing. "# Neighbors in Gaussian Smoothing" is the size of the neighborhood that would have a weighted average taken to denoise the traffic flow time series. This is to cancel out alternating consecutive increases and decreases that may exclude data regions that do not meet the minimum threshold "% Consecutive", but are among the top K regions with the largest or smallest traffic flow changes. The "# Observations" column is the fixed number of observations in each data region among the top K. Table parameters are provided to allow for replicability by the user.

Our experiments run Algorithm 1 shown in Appendix section A, that takes in a minimum threshold of consecutive points that need to be monotonically increasing or decreasing. In Algorithm 1, we scan over the time series with a fixed block size s and identify regions of low and high rates of change in traffic flow data.

The top k regions satisfy the threshold constraint and have either the least or greatest euclidean norm (i.e. the former being the regions with lowest rate of change and latter being highest). k is maximized so we may end up with the maximum number of regions that have the highest and lowest changes in traffic flow. After the k regions have been chosen, some may be overlapping, which would not be independent with each other. In turn, we run a dynamic programming algorithm that maintains a table `OPT` of time series intervals and maximizes the number of non-overlapping intervals we choose among the k regions, which is Algorithm 2 shown in Appendix section A.

We define the "start" and "end of the day" regions as the first and last four hours and ten minutes of the day, respectively, for the PeMS datasets. We selected these durations because the number of observations (50) allows us to compare the resampled regions with other abstract concepts mapped to discrete regions with a specified feature; namely, highest or lowest rates of increase. Similarly, for the A414 dataset, we define the start and end of the day as the first and last four hours as the definition of the start and end of the day remains relatively consistent between the A414 and PeMS datasets despite A414 being reported every fifteen minutes and PeMS being reported every five minutes. Because substituting out 16 observations in the A414 dataset is too few to result in a noticeable increase in model prediction MAE, we substitute out 3 similar regions, a total of 48 observations, with our preceding reference region.

3.3 Perturbation-Data Compatibility

Our probing tool is intended to be a wrapper for models trained on periodic traffic data, which also requires perturbations to be applied to the correct data. Based on Strumbelj's Monte-Carlo method, we perturb a given testing sample that we want to predict the output for. The testing sample in the case of the A414 and PeMS datasets is the input tensor for testing data, which we would predict the traffic flow for. The A414 dataset is a dataframe with a 1D-column corresponding to the traffic flow. Both of the PeMS datasets are composed of four dimensions, which have been specified in the "Datasets" section 2. As a result, for the A414 dataset, we perturb the traffic flow column `A414df['Flow']`. For the PeMS datasets, we extract the `test_x` entry from the data dictionary and per traffic detector, we perturb the traffic flow observations as well as the 11 traffic flow data points following the last entry in `test_x`. We hold the average occupancy and speed of the vehicles constant as region R is effectively substituted out from the traffic flow time series, remeasured with the information after R is masked out. In turn, the test input data for PeMS and A414 – `pems08_data['test_x'][:,detector,0,0]` and `pems04_data['test_x'][:,detector,0,0]` for traffic flow data recorded at detector "detector" and `A414df["Flow"]` – are perturbed, respectively.

3.4 Resampling from Similar Regions

To select regions similar to $R = D[i : j]$ with length $l_R = |R|$, we set a threshold for the number of most similar regions to examine. Our case study examines 15 non-overlapping regions most similar to R. The top 15 most similar regions to

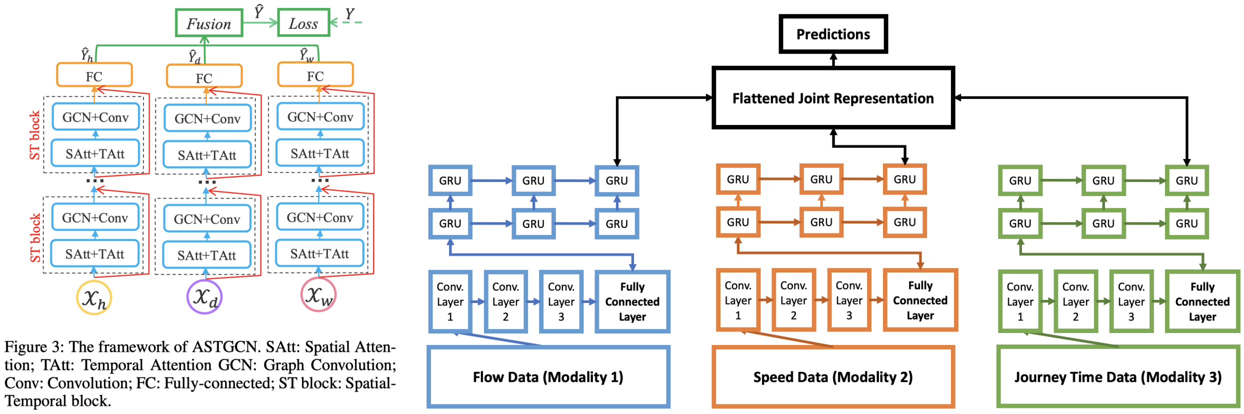


Figure 3: The framework of ASTGCN. SAtt: Spatial Attention; TAtt: Temporal Attention GCN: Graph Convolution; Conv: Convolution; FC: Fully-connected; ST block: Spatial-Temporal block.

Figure 5: The model architectures for the two models of interest in our case study. The left displays the ASTGCN model architecture as a series of graph convolution layers. The right shows Du et al.'s multimodal hierarchical model, which was implemented on the basis of what was described in their paper. Between every two consecutive layers in Du et al.'s model implementation, we apply a batch normalization and set a dropout rate of 30%, as specified in the paper. Each modality of data has 3 convolution layers, followed by a fully connected layer, followed by 6 Gating Recurrent Units (GRUs). A GRU is equivalent to a LSTM with a forget gate [44]. Namely, the forget gate excludes output vector values, after passing through the sigmoid function $\sigma(x) = \frac{1}{1+e^{-x}}$, closer to 0 in future computations and includes those closer to 1.

R are stored in a max-heap data structure, which is ordered by cross-correlation. Cross correlation is used to ascertain similarity between two non-overlapping regions of a time series, which is defined as a similarity metric between two functions that considers the amount one has to shift relative to the other to attain maximum cross correlation. Cross correlation does not have a unique definition, but for vectors v_1, v_2, we consider the sum over n entries where n =length of v_2. Each entry in the sum would be the product between a shifted version of v_1 by some input k and the conjugate function of v_2. Namely, $c_{av}(k) = \sum_{i=1}^{n} a[n+k]conj(v[n])$, where $conj(y) = sup_{x \in dom\ f}(y^T x - f(x))$ [42].

A stationary block-bootstrap-type of resampling is applied: from one below the lower bound of R, or $i-1$, contiguous blocks of traffic flow are generated such that their lengths are sampled from a geometric distribution with a success probability of $p = \frac{1}{window\_size}$, where the $window\_size$ is the expected length ($l_R = |R|$) of a contiguous block that's being resampled from the time series. The same is done for non-overlapping contiguous blocks that start from one above the upper bound of R, or j. The similar regions of the time series that will be resampled is illustrated in Figure 4. Two of these blocks that are presumed to be equivalent “concepts" or “features" of the time series are randomly chosen and the later block is substituted with earlier traffic flow data to avoid look-ahead bias. After effectively masking out the information from R or a data region with similar cross correlation to R, we run the trained model over the data, obtain the prediction output in batches, and retrieve the new prediction MAEs that would have increased from the original data.

3.5 Evaluation

For the PEMS04 and PEMS08 datasets, we examine how consistent the estimation method performs on the spatial-temporal graph convolution network implemented by Guo et al. [37] and on the hierarchical GRU model by Du et al. [2]. The model architectures are described in Figure 5.

For a region with high rate of change defined in Algorithm 1 R_h and another with low rate of change R_l, we run the estimation method for these regions in batches of 25 observations. Each batch is averaged and each respective region, R_h and R_l, contains 32 of these means. The distribution of these means are compared, where a bootstrapped 2-sample test is done to identify whether to reject the hypothesis that reruns on average result in consistently higher or lower mean MAEs. Our evaluation of our estimation method comes from *stability*, which is the consistency of our estimation method to produce MAEs of perturbed data where, on average, masking out information from one region always results in worse average prediction error than the other.

4 Results

We present the results from two models: Gu et al.'s graph convolution neural network and our implemented version of Du et al.'s hierarchical convolution-GRU model. The results are broken down to the baseline prediction errors of

Dataset (3394 observations)	**MAE**	**RMSE**	**MAPE**
Original ASTGCN pytorch model	26.65	41.37	0.17

Figure 6: The baseline prediction errors of the ASTGCN graphical convolution model on the PeMS04 test dataset after being trained and validated on the predefined train/validation datasets. The MAE of the original ASTGCN PeMS04 model is subtracted from the increased prediction MAE of the model on the perturbed PeMS04 traffic flow data.

Dataset (3567 observations)	**MAE**	**RMSE**	**MAPE**
Original ASTGCN pytorch model	18.28	28.21	0.12

Figure 7: The baseline prediction errors are displayed for PeMS08's test dataset after being trained on PeMS08 training/validation datasets. The baseline MAE is subtracted from the increased traffic flow prediction MAE of the model on the perturbed PeMS08 traffic flow data.

Gu's ASTGCN model run over PeMS04 and PeMS08 and Du's hierarchical model run over the A414 dataset as well as the distribution of the estimated changes in prediction MAEs subsequent to masking out regions similar to the data region we are measuring the importance of (R). The prediction error is measured with three metrics: mean absolute error (MAE: $\frac{1}{n}\sum_{i=1}^{n}|\widehat{y}_i - y_i|$), root-mean-square error (RMSE: $\sqrt{\sum_{i=1}^{n}\frac{(\widehat{y}_i-y_i)^2}{n}}$), and mean absolute percentage error (MAPE: $\frac{1}{n}\sum_{i=1}^{n}\frac{|\widehat{y}_i-y_i|}{y_i}$), where $\widehat{y}_i$ is the predicted traffic flow at observation i and y_i is the actual traffic flow at observation i.

4.1 Baseline

The ground truth after running 65 epochs, that uses early stopping, with prediction over 11 points for the PEMS04 dataset is shown in Figure 6.

After running 61 epochs due to early stopping for the PEMS08 dataset, the prediction error over 11 data points is shown in Figure 7.

Having recreated the following neural architecture for the A414 highway traffic dataset, we have the best weights after 35 epochs with early stopping. Prediction over the test data (2/1/2014-2/28/2014) is shown graphically in Figure 8, where the baseline MAE without perturbations is 13.677.

4.2 Overall Estimations and Evaluation

This subsection reports the results from the 32 runs of 25 repeated perturbations on each of five regions: R_1 is the region with the greatest consecutive increases in traffic flow. R_1 contains mostly steep upward spikes in traffic flow. R_2 has the smallest consecutive increases in traffic flow. This is also viewed as the regions with the smallest changes in traffic flow. R_2 contains crests and troughs in the traffic flow time series. R_3 contains the greatest decreases in traffic flow. R_4 contains observations from the first ˜four hours of the day and R_5 is roughly the last four hours of the day. We run a 2-sample bootstrap t-test without the assumption that the prediction MAEs are normally distributed. To correct for multiple testing, we will use the conservative adjustment to error rate of Bonferroni correction as we do not have ground truths for whether the importance of one region should be greater than the other. The correction applied will reduce the alpha level of each of the 32 hypothesis tests by $\frac{1}{32}$. In turn, we test at the $\alpha/32 = \frac{0.05}{32} = 0.0015625$ level after the Bonferroni correction.

We found that the region pairs R_1 and R_4 as well as R_3 and R_5 do not have significantly distinct averages in prediction MAE increases, or importances. This can be attributed to increasing traffic flow at the start of each day and decreasing flow at the end. In turn, our original hypothesis was correct in the importance at the start and end of the day may be similar to R_1, R_2, R_3 depending on the change in traffic flow at the start and end of the day. We also found that data with smaller changes in traffic flow do not always have an estimated importance greater than larger changes in traffic flow. The ASTGCN graph convolutional model appears to weigh regions with larger increases in the PeMS04 dataset with greater importance than smaller increases overall. For the PeMS08 dataset, the ASTGCN model appears to weigh smaller changes in traffic flow with slightly more importance than larger changes during traffic flow predictions. Our implemented hierarchical model from Du et al.'s paper appears to weigh larger increases in traffic flow with similar importance as smaller increases.

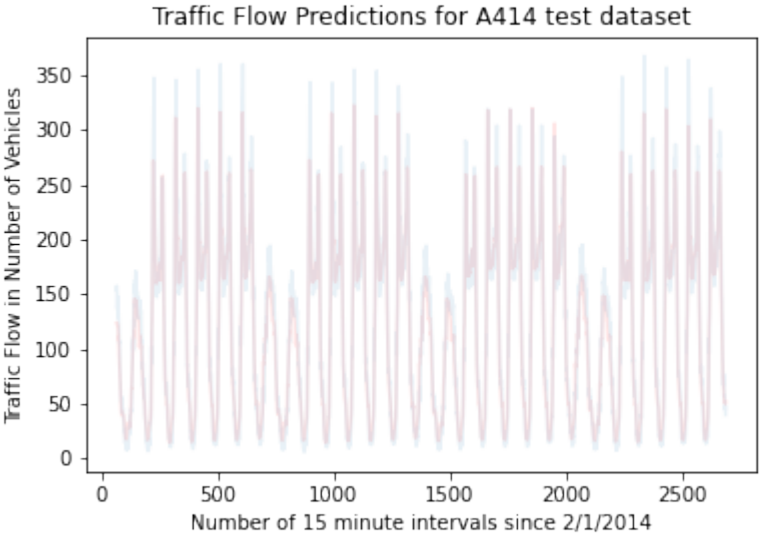


Figure 8: Traffic flow predictions for the A414 test dataset: after 35 epochs, the best weights were chosen and predictions were output from the resulting hierarchical neural model. The blue graph is the ground-truth traffic flow data in the A414 test set. The red graph illustrates the predictions from the network after the model was trained on the training dataset spanning from 1/1/2013 to 12/31/2013. The plot is intended to illustrate the accuracy of the model predictions on A414 traffic flows, where our implementation of Du et al's model underperform on drastic changes in traffic flow (the upward spikes in the plot).

4.2.1 PeMS datasets

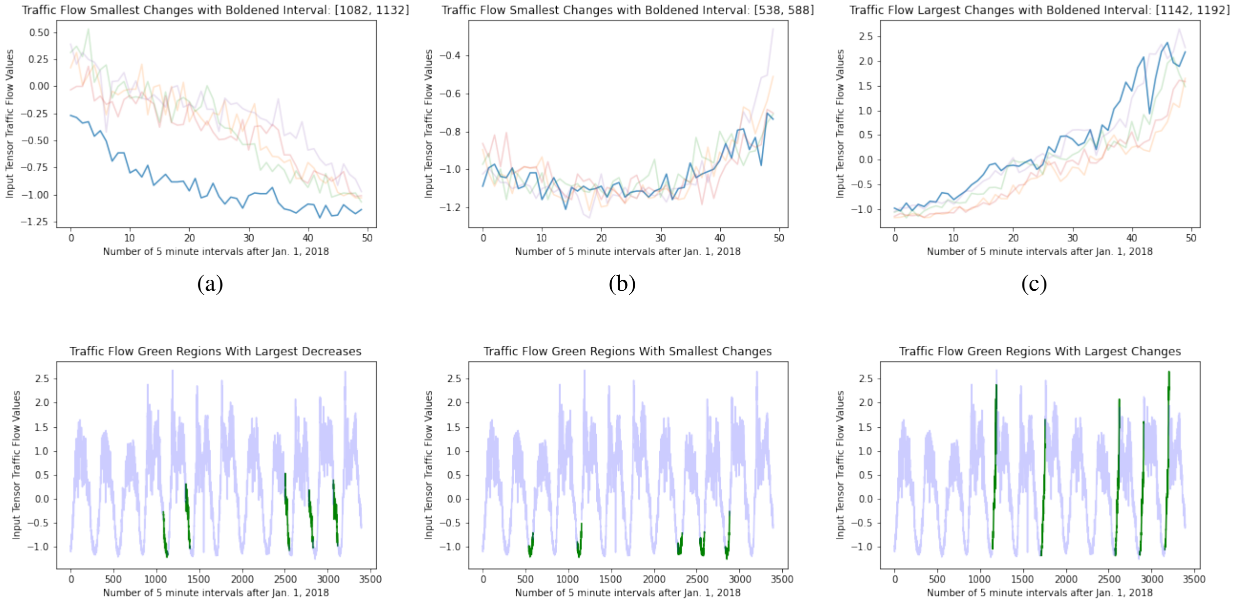
We empirically found $k = 50$ to be the minimum value for k before including additional data intervals in the top k would not lead to sufficiently similar data regions. After running the top $k = 50$ regions for PeMS04 dataset, we obtain the top 5 regions with the greatest and lowest euclidean norms that have at least 70% of consecutive positive differences in traffic flow. The smallest consecutive positive changes in PeMS04 are shown in Figure 9(b).

The largest consecutive positive changes are shown in Figure 9(c). All PeMS regions defined by abstract feature concepts such as “rate of change" or the “start or end of the day" are 50 observations long as a typical interval of a crest trough, upward, or downward changes in traffic flow span 50 observations. Similarly, in keeping with roughly 4 hours at the start and end of the day, 48 observations are masked during perturbations for the A414 dataset as this length captures crests, troughs, upward, and downward trends in the traffic flow.

In order to maintain the same number of regions to randomly resample (5), we relax the percentage of consecutive decreasing traffic flows by 10% and maintaining the top $k = 40$ regions that satisfy the top 10% with the highest euclidean norms shown in Figure 9(a). The euclidean norms for decreasing regions of data are computed over the negative entries in a data region, where positive values are zeroed out.

The estimated increases in prediction MAE for Guo et al.'s ASTGCN model on the PeMS04 dataset are shown in the first row of histograms in Figure 10. There is an apparent visual difference, on average, between the prediction MAEs of each region type (largest increases, largest decreases, and smallest increases). Running two hypothesis tests in separate trials – null hypotheses $H_0 : \mu_{R_1} = \mu_{R_2}$ and $H_0 : \mu_{R_2} = \mu_{R_3}$ – yields the 95% confidence intervals [-0.0212,-0.0126] and [-0.0091,-0.0012] of the differences $R_2 - R_1$ and $R_3 - R_2$, respectively. We use $\alpha = 0.05$ as there is no multiple testing in the trials. This is consistent with the plot of the three histograms in Figure 10(a) as the largest increases, on average, contributes the most to traffic flow forecasts, followed by smaller increases, which is followed by larger decreases.

While it is evident that the estimated increases in prediction MAEs on PeMS04 overall had differences in average prediction MAEs, or importances, between region types, the thirty-two 25-observation samples of each region type revealed primarily a difference between the largest increases and largest decreases in traffic flow. After running a bootstrap of 20,000 resampled samples of size 3394, or the number of observations in the PeMS04 dataset, we found that for 22/32 of the confidence intervals from the 32 repeated runs of the null hypothesis $H_0 : \mu_{R_1} = \mu_{R_3}$, we reject that there is no difference between the average increases in prediction MAEs of the largest decreasing and increasing



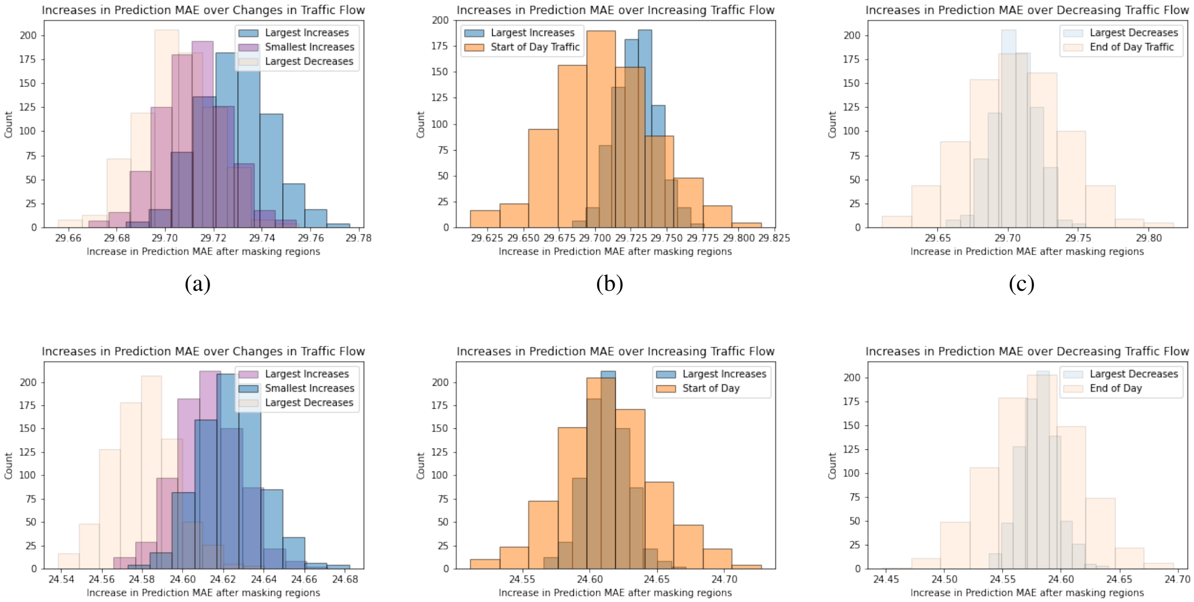
(a) (b) (c)

(d) (e) (f)

Figure 9: (a) The graph is an example of the types of regions classified as having the greatest decreases in traffic flow. (b) An example plot shows the top 5 smallest changes in traffic flow. (c) This graph illustrates five portions of the PeMS04 test dataset with the largest positive consecutive changes in traffic flow.
(d) The graph displays five regions with decreasing traffic flow in green, which illustrates the results from running Algorithm 1. (e) This graph illustrates the regions with the smallest changes in traffic flow on a macro-scale over the entire test dataset of PeMS04, which is used as an example for the equivalent regions in the PeMS08 and A414 datasets. The curves are used to visualize the results of Algorithm 1 in Appendix section A for a PeMS dataset. (f) The graph displays the regions on the scale of the PeMS04 dataset in its entirety, which are examples of regions with the highest increases in traffic flow for a PeMS dataset. The plots suggest that regions with the largest changes in traffic flow also have the greatest absolute slope whereas those with the smallest changes correspond to crests and troughs in traffic data, which implies maintained levels of traffic.

traffic flow regions and consider further testing to qualify the difference. On the other hand, when testing the null hypothesis $H_0 : \mu_{R_1} = \mu_{R_5}$ in a separate trial, 29/32 of the runs resulted in rejection that there is no difference between average increases in prediction MAEs of the largest increasing traffic flow and end of the day traffic regions. Regions with steeper increases in traffic flow seem to be associated with greater importance, or increases in prediction MAE, than those towards the end of the day. The confidence intervals of the difference in prediction MAE increases between increasing traffic flow and end of the day regions for the PeMS04 dataset are shown in Figure 11 on the left side graph. For a given confidence interval in Figure 11(a) not crossing 0, we are 99.844% confident that the average difference in prediction MAEs between regions R_1 and R_5 falls in the set of plausible negative differences that excludes 0.

For the PeMS08 dataset, Figure 9 displays five example green regions of the regions with greatest rate of decrease, increase, and smallest rates of increase from left to right. The intervals with the smallest increasing used a heap structure that maintained the top $k = 10$, a threshold of at least 80% of the traffic flow observations with consecutive positive changes and 15 neighbors during smoothing. Estimations of the five regions with the smallest or largest euclidean norm satisfying the criteria in Table 2 are shown in the second row of Figure 10. After testing the null hypothesis $H_0 : \mu_{R_2} = \mu_{R_1}$ in a 2-sample bootstrap t-test on the regions over all keys, we have that p<0.0001, [-0.0130,-0.0099] and reject that the average prediction MAEs of R_2 and R_1 are equal, suggesting that regions with less increases in traffic flow may contribute more to flow forecasts than those with larger increases. Similarly, we obtain p<0.0001,[-0.0333,-0.0301] over the difference in regions $R_1 - R_3$, so large increases in traffic flow may have greater importance towards predictions than large decreases. Running thirty-two 20,000 bootstrapped samples at the $\alpha = 0.0015625$ level results in 16/32 of the runs having significantly different data importances between end of the day traffic and the largest increases in traffic flow. However, the runs do not have strong stability as the hypothesis tests do not consistently suggest that region R_5 has less importance than R_1, so we may not be able to conclude relative importances. Testing the null hypothesis $H_0 : \mu_{R_1} = \mu_{R_3}$ in a separate trial resulted in the 32 runs plotted in the middle



(a) (b) (c)

(d) (e) (f)

Figure 10: The first row of histograms are prediction MAEs of different traffic flow changes, increasing regions of traffic, and decreasing regions of traffic from left to right for the PeMS04 dataset. The second row displays the aforementioned for the PeMS08 dataset. The orange plots display the start and end of a given day, which have a greater prediction MAE variance than the largest increasing and decreasing regions as we averaged the blue regions over the top 5 regions. The plots are used as a supplement to the hypothesis test results we reports. They are the exploratory data analyses that suggest a relative data importance ordering between the three regions (largest increasing, largest decreasing, smallest increasing). The orange-blue paired plots shows that start of day traffic and regions with the largest increases in traffic flow have similar importances and end of day traffic and regions with the largest decreases in traffic flow have roughly equivalent importances.

graph in Figure 11. 27/32 of the runs were statistically significant (p<0.0015625) where the confidence interval did not cross 0 (significant difference in prediction MAEs). The stability in the estimation of the difference in importances between regions of largest decreases and largest increases in traffic flow is better than that of end of day traffic and largest increases in traffic flow, but further testing would need to be done to conclude whether largest decreases in traffic flow are most important than largest increases towards model predictions. The graph on the right in Figure 11 displays the most stable data importance estimation between the increase in predicted MAE of region R_3 and R_2, where all 32 runs do not have confidence intervals covering 0. In Figure 11(c), we are 99.844% confident that the average difference in increases of prediction MAE between regions R_3 and R_2 falls in a set of plausible negative differences excluding 0.

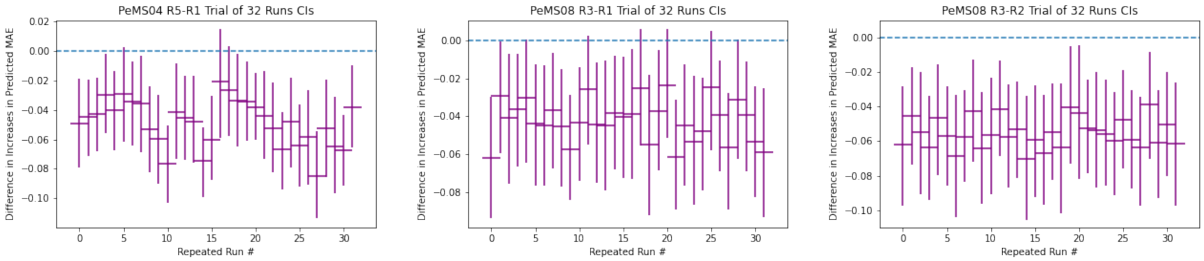
4.2.2 A414 dataset

In Figure 12, we plot the regions that we define as having the largest and smallest rates of change. Similar to the plots of the PeMS dataset regions, we split the “largest changes" region type into increasing and decreasing traffic flow.

The increase in Du et al.'s model prediction MAEs on the A414 dataset shown in Figure 13 are averaged over the five regions: largest increases, smallest increases, largest decreases, start of the day, and end of the day. We observe that the overall increase in prediction MAEs is similar among the three region types and did not result in statistically significant observations at the $\alpha = 0.0015625$ level. We may attribute this to our implemented model weighing all regions from past data with the same importance when making forecasts.

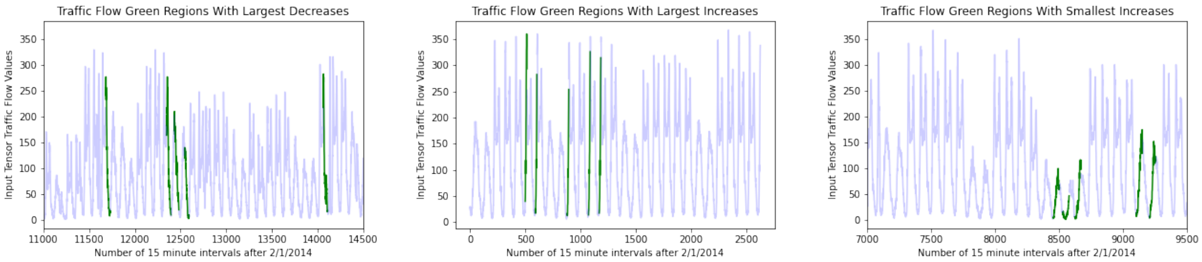
5 Conclusion

Contingent on our estimation method, our hypothesis about larger changes in traffic flow contributing more to traffic predictions than smaller changes holds for the PeMS08 dataset, but not for the PeMS04 or A414 datasets. We also



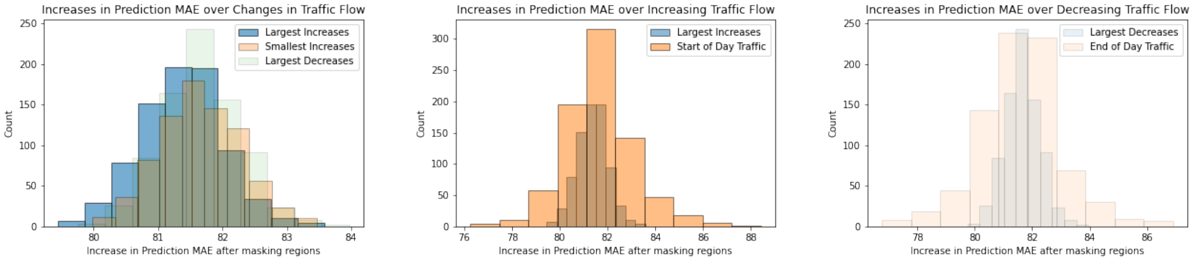
(a) (b) (c)

Figure 11: The graph on the left displays the confidence intervals (CIs) from 32 runs on the PeMS04 dataset, where each run had 20,000 bootstrapped sampled difference in average prediction MAEs. The graph in the middle contains the CIs from 32 runs on the PeMS08 dataset of the difference in prediction MAE between the regions defined by the largest decreases and increases in traffic flow. The plot on the right displays confidence intervals across the 32 runs of the difference in prediction MAEs: $R_3 - R_2$. The confidence intervals are provided as results of the repeated runs and the stability of the estimation method.



(a) (b) (c)

Figure 12: The green regions in the three graphs correspond to the top 5 regions of traffic flow data with the largest euclidean norms that have the greatest rate of decreases, rate of increases, and smallest consecutive increases in traffic flow from left to right. The plots are displayed as part of the results for the "Defining the Regions" portion of our estimation methodology.



(a) (b) (c)

Figure 13: The three histograms are averaged over 800 repeated runs of the 5 regions per region type with the highest (in "largest" region types) or lowest (in the "smallest increases" region type) euclidean norms. The average increases in prediction MAE among the five regions are not significantly different from one another, where confidence intervals of all pairs among R_1, R_2, R_3, R_4, and R_5 exclude 0.

claim that the impact of data at the start or end of the day on traffic predictions depends on how traffic flow changes during those times. This holds for both of the PeMS datasets, where “start of day" traffic flow is always increasing and “end of the day" traffic always decreases; “start of day" traffic has similar importance towards model predictions as increasing traffic flow regions and the same with “end of day" traffic. In turn, because increases in traffic flow contribute more to model predictions than decreases, the rise in traffic flow at the start of the day appears to have greater data importance than the decrease at the end of the day.

The estimated increases in prediction MAEs after masking out defined regions for the PeMS04 dataset appeared to be greatest for regions with the largest increases in traffic flow. This corresponds to the model deriving its forecasts more heavily on shifts in traffic flow from an increase in the number of drivers than maintenance of existing traffic, which may be from the PeMS04 dataset having less variability (i.e. less drops in traffic flow due to congestion during busy hours of the day) in data portions with smaller changes in traffic flow. This may prompt the model to have variable steep shifts in traffic flow drive its predictions. For the PeMS08 dataset, we found the region with the greatest average increase in prediction MAE after masking out similar regions were those that had the smallest changes in traffic flow, which is evident in our initial analysis comparing the three regions (largest increases, smallest increases, and largest decreases) and the confidence intervals in Figure 11. Guo et al.'s model may weigh regions with smaller changes more heavily because of greater variability in crests and troughs in the PeMS08 dataset than the PeMS04 dataset. More variability in traffic flow during crests would suggest greater drops in traffic flow during the busiest hours of the day (as crests can span from 9 AM to 5 PM), which would suggest traffic congestion amidst the highest volume of vehicles. Comparisons between the importance of regions with the largest increases in traffic flow over time and those with the smallest may have been insignificant in the PeMS04 and PeMS08 datasets because both regions have similar contributions to model predictions.

The increase in prediction MAEs of our implemented model based on the parameters specified in Du et al.'s paper were overall similar across regions with different characteristics in traffic flow changes, whose differences were not statistically significant at the $\alpha = 0.0015625$ level. The Bonferroni correction was made because we tested the null hypotheses $H_0 : \mu_{R_1} = \mu_{R_2}, H_0 : \mu_{R_2} = \mu_{R_3}, H_0 : \mu_{R_1} = \mu_{R_3}$ 32 times each in three separate trials.

While the resulting confidence intervals for $R_5 - R_1$ in the PeMS04 dataset and $R_3 - R_1$ in the PeMS08 dataset were the most stable comparisons of data region importance, there were a few intervals that crossed 0 and it could not be concluded whether the two regions had distinct average increases in prediction MAEs. The most stable data importance comparison between regions R_3 and R_2 in Figure 11(c) suggests that regions with the smallest increases in traffic flow may contribute more to traffic flow forecasts than those with the largest decreases, but further experiments must be conducted with a larger sample size and greater power to ascertain whether the estimations are truly consistent.

6 Limitations and Future Work

Beyond the three datasets that we have applied our estimation method to, we can explore other traffic forecasting datasets collected from more recently recorded traffic sites and further test the stability of our estimation method. Particularly with the wide adoption of connected vehicle environments, more fine-grained data may also be collected about traffic congestion on highways and intersections.

Also, when we define the regions that we intend to perturb, we may consider not just traffic flow, but multiple variables and account for underlying causal relationships by summing indirect and direct effects in causal shapley values, as presented by Heskes et al. In addition to the regions that we perturbed, we could also increase the number of reruns of the estimation algorithm on the datasets given that time permits. This can result in lower variability and may give us a better idea whether the mean of the average change in prediction errors for one region is always in the same ordering relative to the other regions of a time series.

Our methodology makes the assumption that resampled regions from traffic flow that are most similar to some region R are virtually equivalent, may be treated as the same “feature" in a periodic traffic time series, and can be substituted with one another. However, similarity may not imply that these regions can replace one another as a means to “mask" out information from the input data for the model. The assumption enables flexible preliminary analysis of our estimation method as we can test a variety of hypotheses about importance of any region of the time series. However, we may afford further correctness by restricting the data to individual time steps $t_1, ..., t_n$ (i.e. reducing the flexibility of our estimation method) that are each considered a feature and implement a hypothesis testing framework that can ascertain the number of perturbations required to guarantee stability, as in Zhou et al. [41].

Future work would also involve testing additional hypotheses under our estimation framework or alternative data importance approaches, which can allow us to further verify the stability of our estimation method and see whether adding estimation methods to offer an idea of data importance can help laymen better understand the behavior of black-box models.

While we evaluated the *consistency* of the data importance results, additional evaluation should measure the improvement in prediction error (MAE, RMSE, MAPE) after traffic operators observe the data importance from our estimation method and provide a criteria of ground-truth regions in the traffic time series (traffic flow, speed, journey time, or average occupancy) that would drive future traffic flow predictions. The engineers would then retrain the model to fit the criteria and we would deem the decrease in prediction MAE as our evaluation metric of our estimation method's *efficacy*. Our data importance approach should also have its *interpretability* evaluated using counterfactual simulatability [36]. Namely, we would recruit subjects, or traffic operators, who predict data importance of edited historical traffic flow data, and measure the mean absolute differences between their predictions and the true data importance changes in prediction MAE of the model after masking out specific regions of data.

7 Acknowledgements

This work was done for my Senior Honors Thesis under the supervision of Professor Freeman from the CMU Statistics Department, who I would like to thank for guiding me through the process of my thesis research.

8 Appendix

A Algorithms

Algorithm 1 We estimate the regions with maximum and minimum rate of change with length s. v is the time series sequence $x_1, ..., x_t$ that we are finding the maximum or minimum rate of changes in. L is the number of time series observations per time series interval we examine the rate of change of, which has been ascertained empirically to be 150. k is a maximized number of intervals over v to return with the highest or lowest rates of change. $threshold$ is the minimum number of consecutive differences that may be positive or negative. $neighbors$ is the range of the points that gaussian smoothing averages. The argument is used to denoise regions of data that may truly be increasing or decreasing, but would not satisfy the threshold of consecutive changes all being negative or positive. $smallest$ is a boolean specifying whether we are finding the regions of data with the least or greatest rates of change (i.e. the least or greatest euclidean norm of consecutive positive or negative differences in traffic flow).

```
procedure FIND_OPTIM_CHANGE($v, L, k, threshold, neighbors, smallest$)
    $optim\_changes \leftarrow heap([])$
    for $i\ in\ range(L, len(v))$ do
        $diffs \leftarrow deltas(v[(i-L):i]);$
        $smoothed\_points \leftarrow linear\_smoother(diffs, neighbors);$
        if $get\_consecutive\_diffs(threshold, smoothed\_points)$ then
            $diffs \leftarrow list(filter(lambda\ x : x >= 0, diffs));$
            $euclidean\_norm \leftarrow (-1\ if\ smallest\ else\ 1) * \sqrt{\sum_{i=1}^{n} diffs[i]^2});$
            if $len(optim\_changes) < k$ then
                Push $[i-L, i, euclidean\_norm])$ onto $optim\_changes;$
            else
                $smallest\_change \leftarrow$ euclidean norm of smallest change in $optim\_changes;$
                if $(euclidean\_norm > smallest\_change)$ then
                    Remove $low\_val$ from $optim\_changes;$
                    Push $[i-L, i, euclidean\_norm]$ onto $optim\_changes;$
                end if
            end if
        end if
    end for
    $optim\_intervals \leftarrow map$ each element $[start, end, norm]$ to $[start, end]$ in $optim\_changes$
    $optim\_intervals \leftarrow MAX\_NUM\_INTERVALS(optim\_intervals)$        ▷ Calls Algorithm 2
end procedure
```

Algorithm 2 The first procedure obtains the preceding non-overlapping interval of the j^{th} interval in the list of intervals. The second procedure applies dynamic programming, where the maximum number of intervals and set of intervals up to the $(j-1)^{th}$ ordered interval is stored in table OPT. The entry $OPT[j-1]$ computed based on previous table entries.

```
procedure GET_PREC($intervals,\ j$)
    $prec \leftarrow None$;
    $curr\_idx \leftarrow j - 1$;
    while $curr\_idx \geq 0$ do
        $curr\_interval \leftarrow intervals[curr\_idx]$
        if !$are\_overlapping(curr\_interval, intervals[j])$ then
            $prec \leftarrow curr\_idx$;
            $break$;
        end if
        $curr\_idx - = 1$;
    end while
    return $prec$
end procedure

procedure MAX_NUM_INTERVALS($intervals$)
    $intervals.sort()$;
    $OPT \leftarrow [0] * len(intervals)$;
    for $j\ in\ range(len(intervals))$ do
        $prec\_idx \leftarrow get\_prec(intervals, j)$
        if $prec\_idx$ is None then
            $OPT[j] \leftarrow (1, [intervals[j]])$;
        else
            $OPT[j] \leftarrow ((OPT[prec\_idx][0] + 1), OPT[prec\_idx][1] + [intervals[j]])$
                    $if (OPT[prec\_idx][0] + 1) > OPT[j-1][0]$
                    $else\ OPT[j-1]$
        end if
    end for
    return $OPT[-1]$
end procedure
```

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

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

[37] 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.

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

[39] Anderková, Viera, and František Babič. "Better understandability of prediction models: a case study for data-based road safety management system." 2021 IEEE 21st International Symposium on Computational Intelligence and Informatics (CINTI). IEEE, 2021.

[40] Yuan, Chen, et al. “Application of explainable machine learning for real-time safety analysis toward a connected vehicle environment." Accident Analysis & Prevention 171 (2022): 106681.

[41] Zhou, Zhengze, Giles Hooker, and Fei Wang. “S-lime: Stabilized-lime for model explanation." Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining. 2021.

[42] Zhang, Jun, and Ting-Kam Leonard Wong. “Lambda-Deformed probability families with subtractive and divisive normalizations." Handbook of Statistics. Vol. 45. Elsevier, 2021. 187-215.

[43] Wang, Jiaqi, et al. “Interpretable image recognition by constructing transparent embedding space." Proceedings of the IEEE/CVF International Conference on Computer Vision. 2021.

[44] Gers, Felix A., Jürgen Schmidhuber, and Fred Cummins. “Learning to forget: Continual prediction with LSTM." Neural computation 12.10 (2000): 2451-2471.

[45] Global, Bosch. “Intelligent Transportation Systems (ITS)." Bosch Security and Safety Systems I North America, 15 June 2022, https://www.boschsecurity.com/us/en/industries/intelligent-transportation-systems-its/.

[46] Yao, Zhihong, et al. “A dynamic optimization method for adaptive signal control in a connected vehicle environment." Journal of Intelligent Transportation Systems 24.2 (2020): 184-200.