INTERNATIONAL
HELLENIC
UNIVERSITY

Innovative Traffic Prediction Techniques under abnormal conditions

Theodorou Traianos-Ioannis

SID: 3301150006

SCHOOL OF SCIENCE & TECHNOLOGY

A thesis submitted for the degree of

Master of Science (MSc) in Information and Communication Systems

JANUARY 2017

THESSALONIKI – GREECE

INTERNATIONAL
HELLENIC
UNIVERSITY

Innovative Traffic Prediction Techniques under abnormal conditions

Theodorou Traianos-Ioannis

SID: 3301150006

Supervisor:	Assist. Prof. Christos Tjortjis
Supervising Committee Members:	Researcher A’ Dimitrios Tzovaras
	Dr. Christos Berberidis

SCHOOL OF SCIENCE & TECHNOLOGY

A thesis submitted for the degree of

Master of Science (MSc) in Information and Communication Systems

JANUARY 2017

THESSALONIKI – GREECE

Acknowledgments

I want to express my deepest gratitude to my supervisor Dr. Christos Tjortjis who was always available and willing to discuss about the dissertation. Additionally, I would like to thanks for the valuable advices and help, Dr. Dimitrios Tzovaras and Dr. Dionisis Kehagias and their research team in traffic prediction, located in Information Technologies Institute of Center of Research and Technology.

I must not fail to mention Mr Athanasios Salamanis, for his assistance in the better understanding of the tools that I used and expand that were previously developed by the research team of CERTH-ITI during the Horizon 2020 European project e-Compass.

Last but not least, I would like to thank my parents and sister, who have been supporting me along my entire life. This dissertation is dedicated to them, for their infinite love.

Abstract

One of the most critical functions of the modern Intelligent Transportation Systems (ITS) is the accurate and real-time short-term traffic prediction. This function becomes even more important under the presence of atypical traffic conditions. In this dissertation, we propose a novel hybrid method for short-term traffic prediction under both typical and atypical conditions.

An Automatic Incident Detection (AID) algorithm that is based on Support Vector Machines (SVM) is utilized to check for the presence of an atypical event (e.g. traffic accident). If one occurs, the k-Nearest Neighbors (k-NN) non-parametric regression model is used to predict traffic. If no such case occurs, the Autoregressive Integrated Moving Average (ARIMA) parametric model is activated.

In order to evaluate the performance of the proposed model, we use open real world traffic data from the Caltrans Performance Measurement System (PeMS). We compare the proposed model with the unitary k-NN and ARIMA models. Preliminary results indicate that the proposed model outperforms its competitors in terms of prediction accuracy under both typical and atypical traffic conditions.

Theodorou Traianos-Ioannis

18-01-2017

Contents

List of Tables

List of Figures

1 Introduction

Nowadays, the interest in developing Intelligent Transportation Systems has grown significantly in respect to the need of providing qualitative transportation services either for individuals or fleets of vehicles. In this context, the ability to accurately predict traffic in various steps ahead of time is of paramount importance.

The main reason for which the traditional traffic prediction models fail to accurately predict traffic in real conditions is the presence of atypical conditions. These conditions may include severe weather conditions, car accidents, road maintenance works and traffic congestion due to special cultural events (e.g. concerts or sport games). These events can cause steep spikes in the traffic time series which are difficult to follow by the standard traffic prediction models, because these models are in most cases fitted with traffic data that include no abnormalities.

Additionaly, these events are difficult to classify into categories because they vary in type, duration, severity, effect on the state of the traffic network etc. Finally, there are cases where incidents may not cause observable effects on the traffic time series or may be spikes that are not necessarily correspond to atypical conditions. These cases make the whole problem of predicting traffic during atypical conditions even more difficult.

Generally, the accurate traffic prediction in a life based condition which contains both typical and atypical conditions is useful in order to avoid car accidents and traffic collisions. The importance of developing Intelligent Transportation Systems can be crucial in many sections. For instance, the reduce of traffic collisions has a positive impact to the environment and the health of citizens in an urban area. Additionally, the quality of life will be improved reducing the CO_2 and the noise pollution level especially, in cities where the traffic problem is increased.

Furthermore, the accurate prediction of traffic in road networks could be vital, for the companies which own a large fleet of trucks. The inefficient daily routing of their trucks could cost vast amount of money and time. More specifically, courier or taxi companies need to be aware of the urban traffic on the following hours, as well as, other transport companies the traffic in highway level.

In this dissertation, a novel pattern transition model is presented for predicting traffic during not only typical, but also (and more importantly) during atypical conditions. We use a SVM-based automatic incident detection model to automatically detect the presence of an atypical situation. If the model detects the presence of an atypical situation, the non-parametric k-NN regression model is fetched to predict traffic. On the contrary, if no such case is detected the ARIMA parametric model is activated.

The following flow chart illustrates the whole procedure of the method we develop. Each row of the dataset contains a time series with speed values, representing the average speed of a road aggregated in 5 minutes' time intervals.

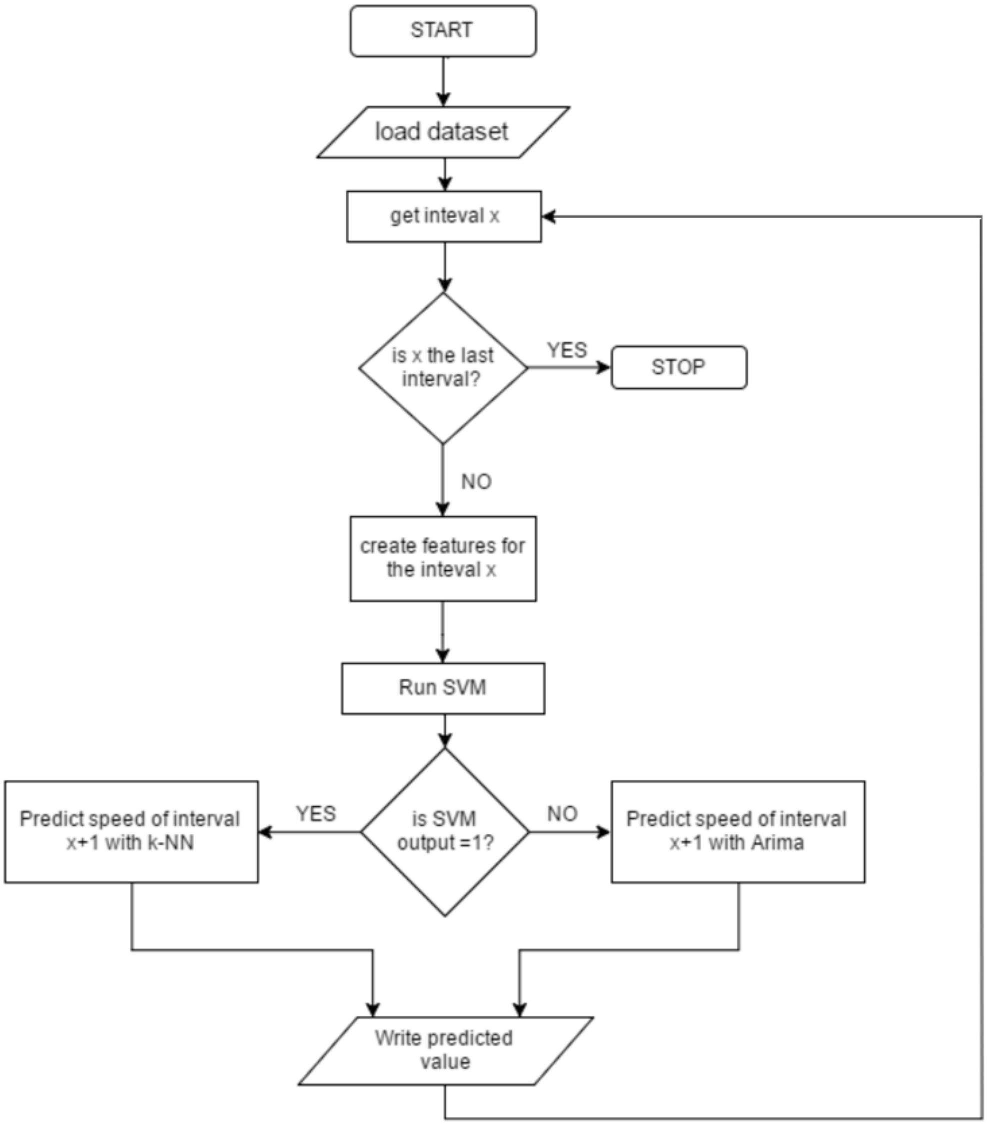

START
load dataset
get inteval x
is x the last interval?
YES
STOP
NO
create features for the inteval x
Run SVM
is SVM output =1?
YES
Predict speed of interval x+1 with k-NN
NO
Predict speed of interval x+1 with Arima
Write predicted value

Figure 1 Flow Chart of the Solution

Initially, the dataset of time series is loaded to the application. For each interval of the testing dataset the features of SVM are created. Then based on the class that the SVM algorithm predicted, the respective prediction algorithm is triggered in order to forecast the value of the next interval.

The evaluation of our model was done using a C++ framewok that taking as input an shapefile with a specific format, that will analyzed further in a next chapter, creates a data structure that represents the road network. During this dissertation, this framework was expanded, adding the map San Jose, California. Moreover, the SVM and the k-NN algorithms was developed and the ARIMA adjusted to the needs of the proposed model. Additionally, during this dissertation the C++ framework enriched with a number of functions in order to calculate a number of metrics for the benchmarking of models. Sensitivity and Specificity was used as metrics for the classification accuracy and SMAPE and RMSE for the of the prediction models.

In summary, the main contributions of this work are the following:

- We propose a novel pattern transition model for short-term traffic prediction under both typical and atypical conditions that automatically recognizes the presence of an atypical situation, and utilizes a different prediction model based on the outcome of an incident detection algorithm.
- The proposed model incorporates the incident information into the fitting process of the predicting models.
- We evaluate the functionality and performance of our model using real world data set which includes both traffic and incident information.

The following dissertation is organized as follow, in the second chapter we describe and debate a number of different approaches for traffic prediction and we will state the most known of them per category. The third chapter contains the whole procedure of the data preprocessing. At first, the initial dataset is described and then we describe the process of the construction of the shapefile and the row dataset with the traffic and incident information. In the fourth chapter, the automated incident detection algorithm is analyzed and its features are described. The fifth chapter, analyze the prediction algorithms that we have used, in order to build the proposed model. The sixth chapter, is dedicated to the evaluation of our solution and the benchmarking against single approaches. Finally, the seventh chapter, contains the conclusion and a description of the future work.

2 Literature Review

The research problem of short-term traffic prediction has been extensively studied during the last ten years. The various techniques presented can be roughly classified into the following four major categories: naïve, parametric, non-parametric and hybrid.

2.1 Naïve Algorithms

Naïve methods are the most cost-effective prediction models and are mainly used as benchmark against which more sophisticated methods can be compared. They are characterized by the absence of any advanced mathematical model for the exploitation of the available traffic data. Some common naïve methods for traffic prediction include the use of the last observed traffic value, the simple moving average of past traffic values with a predefined window T, and the cumulative moving average of past traffic values.

2.2 Parametric Algorithms

Parametric models are those which involve the estimation of predefined parameters using historical traffic data. These methods mainly originate from time series analysis. Most of the works in this class are based on the classic Box & Jenkins Autoregressive Integrated Moving Average model (Box and Jenkins, 1971). In their work, Stathopoulos and Karlaftis presented a multivariate state-space ARIMA approach for modelling and predicting traffic flow, showing that different model specifications are more appropriate for different periods of the day (Stathopoulos and Karlaftis, 2003) the main strength of their state approach is that state space ARIMA could model a wide variety of time series processes which may vary from a rather complicated mixed ARIMA and cross-correlation process to a simpler ARIMA or AR model.

Additionally, Williams et al. proposed a Seasonal ARIMA model that takes into account the fact that traffic data are characterized by periodic cycles (Williams et al., 1998). A seasonal ARIMA model is formed by including additional seasonal terms in the ARIMA models like the form bellow:

$$\text{ARIMA}\quad \underbrace{(p,d,q)}_{\substack{\uparrow \\ \left(\begin{array}{l}\text{Non-seasonal part}\\ \text{of the model}\end{array}\right)}}\quad \underbrace{(P,D,Q)_m}_{\substack{\uparrow \\ \left(\begin{array}{l}\text{Seasonal part}\\ \text{of the model}\end{array}\right)}}$$

Where *m* is the number periods per season. The modelling procedure is almost the same as for non-seasonal data, except that we need to select seasonal AR and MA terms as well as the non-seasonal components of the model.

Moreover, Kamarianakis and Prastacos developed a Space-Time ARIMA (STARIMA) model with robust behavior (Kamarianakis and Prastacos, 2005) which was extended by Min and Wynter who tried to deal with the problem of the supposed stationarity of the process and the constant relationship between the neighbor road segments in a traffic network (Min and Wynter, 2011). STARIMA is an algorithm that takes into account both location and time as a weighed linear combination of previous observations and innovations lagged both in time and space. Finally, Mu et al. proposed a method that utilizes heterogeneous delay embedding (HDE) to extract an informative feature space for regression analysis of traffic data (Mu et al., 2012).The HDE initially categorizes , into different types, the historical and current data of a time-variant measurement, after that, it introduce different delay settings for embedding multiple types of time-series data, and finally removes redundant information and noise from the generated features using orthogonal locality preserving projection. Using this technic HDE aims to achieve accurate prediction for different departure time. There are several other works in this category (Guo and Williams, 2010, Kamarianakis et al., 2012, Ghosh et al., 2009).

2.3 Non-parametric Algorithms

The non-parametric models mainly originate from machine learning and are based on k-NN regression, Artificial Neural Networks (ANN) and Support Vector Regression (SVR) techniques. k-NN for short-term traffic prediction was introduced by Smith and Demetsky who claimed that it outperforms both the naïve method of historical average and the parametric ARIMA model in terms of robustness against variable data sets (Smith and Demetsky, 1996)**.**

Moreover, Clark proposed a model that is a multivariate extension of the k-NN that exploits the three-dimensional nature of the traffic state (Clark, 2003). The k-NN non-parametric regression algorithm was utilized by several other researchers for building accurate traffic prediction models (De Fabritiis et al., 2008, Kindzerske and Ni, 2007, Myung et al., 2012, Zheng and Su, 2014). The fact that differentiate the most of the aforementioned research works was the selection of the distance metric. the most common used distance function in k-nn are cosine, Chi square, and Minkowsky, while Euclidean distance was the one that conclude to more accurate models. Regarding the use of ANNs, Vlahogianni et al. introduced the auto- and cross-correlated effect of the traffic flow time series in a neural network model in the form of external information (Vlahogianni et al., 2003).

Also, Dougherty and Coddet proposed a neural network technique for predicting occupancy, flow and vehicle speed. In their model, they used elasticity testing on the neural network in order to overcome the problem of the multitude of possible input parameters (Dougherty and Coddet, 1997). In the same context, we have the works of Abdulhai et al. and Innamaa (Abdulhai et al., 1997, Innamaa, 2000). Finally, Wu et al. and Hu et al. used the SVR algorithm for accurately predicting traffic (Wu et al., 2003, Hu et al., 2015). The main idea of the Support Vector Regression is the same with the Support Vector Machine (SVM) for classification, to minimize error, individualizing the hyperplane which maximizes the margin, keeping the error in a tolerated level.

Nonetheless, the output of SVR is a real number which has infinite possibilities, therefore it becomes very difficult to predict the information quickly, something necessary in short term traffic prediction. In general the algorithm is more complicated than SVM therefore a lot of things need to be taken in consideration, in order to create a tolerant prediction model.

2.4 Hybrid Algorithms

Noticeably research effort has been given on the development of hybrid traffic prediction techniques that try to exploit the strong characteristics of both parametric and non-parametric techniques. In this context, we have the work of Zhang, who proposed a model that combines ARIMA and ANN processes (Zhang, 2003), where his model consisted of two steps. The first step, included ARIMA to analyze the linear part of the problem and the second step, a neural network model, which modeled residuals of ARIMA. The Gao and Er proposed the Non-linear Autoregressive Moving Average with exogenous inputs (NARMAX) model which combines fuzzy logic and ANN (Gao and Er, 2005). Similarly, Quek et al. presented a special case of fuzzy neural network for short-term traffic prediction which resulted in both high degree of knowledge extraction and generalization of the input and prediction capability (Quek et al., 2006).

2.5 Algorithms under Abnormal Conditions

Despite the multitude of proposed models for short-term traffic prediction in the current literature, very few of them deal with the problem in the context of atypical traffic conditions. Atypical conditions may include severe weather conditions, traffic accidents, road maintenance works, congestion due to special events (e.g. concerts or sport events) etc. These abnormalities lead to traffic conditions that are difficult to be captured by traditional traffic predictions model and thus the number of works in these cases is very limited.

In a noticeable effort, Castro-Neto et al. proposed the Online Support Vector Regression (OL-SVR) model for short-term traffic prediction under both typical and atypical conditions (Castro-Neto et al., 2009). They compared their model with well-known models including Gaussian Maximum Likelihood (GML), Holt exponential smoothing and ANN and have showed that while the GML model performs better than the other models in terms of prediction accuracy under typical traffic conditions, the OL-SVR model performs best under non-recurring atypical traffic conditions. Another instance is the work of Guo et al. who presented a work in which three different configurations of the explanatory traffic variable in conjunction with three different prediction models were used for traffic prediction. The experimental results showed that the k-NN in conjunction with the third configuration of the explanatory variable outperforms the ANN under all conditions (Guo et al., 2010).

Additionaly, the same authors showed that the k-NN and SVR non-parametric regression models have similar prediction accuracy under typical traffic conditions but k-NN outperforms SVR during atypical (Guo et al., 2012). In an extension of the previous two works, the authors presented an enhanced k-NN model which includes data smoothing and de-noising components (Guo et al., 2014). The model compared with five well known machine learning benchmark methods in terms of prediction accuracy and showed that it presents the best results. Another approach is the work of Wu et al. who introduced a hybrid approach called Online Boosting Non-Parametric Regression (OBNR) which consists of two parts: (a) a typical non-parametric regression model used for traffic prediction under typical conditions, and (b) a boosting part activated when atypical traffic conditions have been identified. The boosting part is deactivated when the traffic state turns back to normal. Experiments with real data showed that the OBNR model outperforms the classic non-parametric regression and the SVR models during atypical traffic conditions (Wu et al., 2012).

Finally, an alternative approach was proposed by Ni et al. which in addition to traffic data, uses data from social networks (Twitter) to predict traffic prior to major sport game events. Fusion of both tweet rate and semantic features into the typical prediction model resulted in significant improvements in terms of the achieved prediction accuracy (Ni et al., 2014).

A key characteristic of the aforementioned works is that they do not use traffic data coming from atypical conditions in order to fit their models. They use traffic data coming from typical conditions in order to train their models, and then test their performance on traffic data from both typical and atypical conditions. In this work, we try to incorporate traffic data from atypical cases into the fitting process of our model. This is an essential difference between our work and the aforementioned ones on the problem of short-term traffic prediction under atypical conditions.

3 Preprocessing and Dataset Analysis

In this section, we analyze the data set that was used for the building and the evaluation of the proposed model. Additionally, we describe the whole procedure of the construction both the final dataset and the shapefile that was used.

3.1 General description of initial data

The initial data that was used came from the Caltrans Performance Measurement System. PeMS is an Archived Data User Service that collects over ten years of data for historical analysis. The traffic data are coming from over 39,000 Vehicle Detection Stations (VDS) scattered on the freeway system of all major metropolitan areas of the State of California, USA. The dataset includes flow, occupancy and speed values as well as meta-information about the VDS, e.g. the identification numbers of the district and the freeway in which the VDS is located, the coordinates of the VDS etc. The traffic values are collected every 30 seconds and are aggregated into 5-minute and 1-hour time intervals. The user can select to acquire the data either in their raw or in their aggregated form.

PeMS also provides incident data collected by the California Highway Patrol (CHP). These data contain several information for the incidents occurred on the Caltrans network such as the location of the incident (latitude, longitude), the timestamp, the type (e.g. car accident, road maintenance works etc.), the duration (in minutes) and other. These incidents are reported by the network's users to CHP which logs them. The map of the overall area that provides traffic and incident data in the PeMS system is shown in Figure 2.

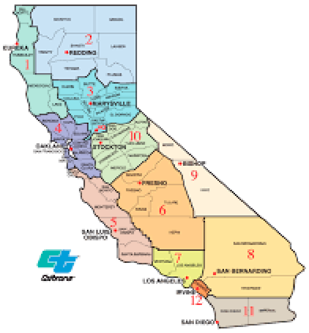


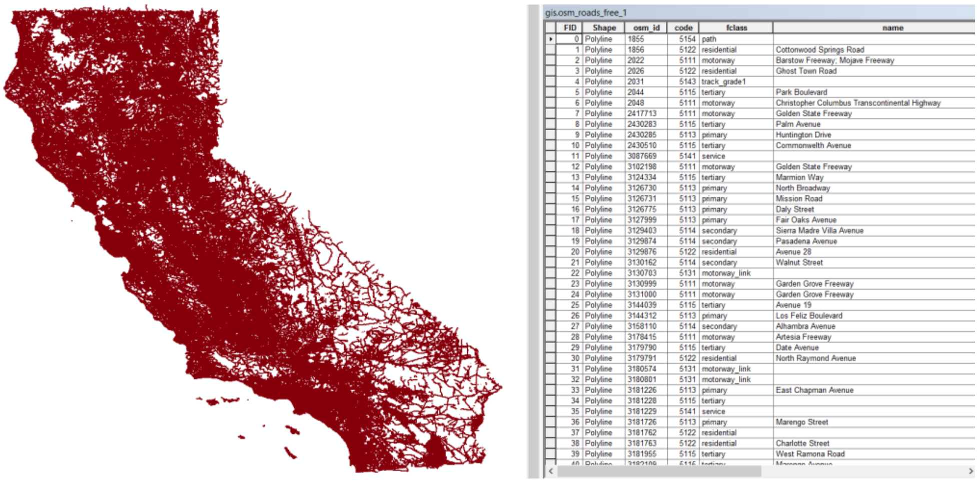
Figure 2 PeMS Caltrans map.

Additionally, for the creation of the road network in terms of road segments and nodes we used a shapefile from the *geofabrik.de* specifically for California DC

In this dissertation, we used a part of the above data set for fitting and evaluating our model. Particularly, the data set used includes traffic data (speed probes) from the area of San Jose, Oakland, California (district 4 in Figure 2) and covers a total time period of 123 days, from May 1 to August 31, 2015. We acquired the data in their aggregated form in 5-minute intervals. Also, the data set contains incidents data for the same area and time period. For the construction of the final dataset a number python scripts were developed combining the traffic and the incident data.

3.2 Preparation of the Shapefile

In order to create the final shapefile of the area of San Jose we use a tool called ArcMap. The shapefile consisted of polylines, where you can combine and filter data, based on spatial information.



gis.osm_roads_free_1

FID	Shape	osm_id	code	fclass	name
0	Polyline	1855	5154	path	
1	Polyline	1856	5122	residential	Cottonwood Springs Road
2	Polyline	2022	5111	motorway	Barstow Freeway; Mojave Freeway
3	Polyline	2026	5122	residential	Ghost Town Road
4	Polyline	2031	5143	track_grade1	
5	Polyline	2044	5115	tertiary	Park Boulevard
6	Polyline	2048	5111	motorway	Christopher Columbus Transcontinental Highway
7	Polyline	2417713	5111	motorway	Golden State Freeway
8	Polyline	2430283	5115	tertiary	Palm Avenue
9	Polyline	2430285	5113	primary	Huntington Drive
10	Polyline	2430510	5115	tertiary	Commonwealth Avenue
11	Polyline	3087669	5141	service	
12	Polyline	3102198	5111	motorway	Golden State Freeway
13	Polyline	3124334	5115	tertiary	Marmion Way
14	Polyline	3126730	5113	primary	North Broadway
15	Polyline	3126731	5113	primary	Mission Road
16	Polyline	3126775	5113	primary	Daly Street
17	Polyline	3127999	5113	primary	Fair Oaks Avenue
18	Polyline	3129403	5114	secondary	Sierra Madre Villa Avenue
19	Polyline	3129874	5114	secondary	Pasadena Avenue
20	Polyline	3129876	5122	residential	Avenue 28
21	Polyline	3130162	5114	secondary	Walnut Street
22	Polyline	3130703	5131	motorway_link	
23	Polyline	3130999	5111	motorway	Garden Grove Freeway
24	Polyline	3131000	5111	motorway	Garden Grove Freeway
25	Polyline	3144039	5115	tertiary	Avenue 19
26	Polyline	3144312	5113	primary	Los Feliz Boulevard
27	Polyline	3158110	5114	secondary	Alhambra Avenue
28	Polyline	3178415	5111	motorway	Artesia Freeway
29	Polyline	3179790	5115	tertiary	Date Avenue
30	Polyline	3179791	5122	residential	North Raymond Avenue
31	Polyline	3180574	5131	motorway_link	
32	Polyline	3180801	5131	motorway_link	
33	Polyline	3181226	5113	primary	East Chapman Avenue
34	Polyline	3181228	5115	tertiary	
35	Polyline	3181229	5141	service	
36	Polyline	3181726	5113	primary	Marengo Street
37	Polyline	3181762	5122	residential	
38	Polyline	3181763	5122	residential	Charlotte Street
39	Polyline	3181955	5115	tertiary	West Ramona Road
40	Polyline	3182109	5115	tertiary	Marengo Avenue

Figure 3 Shapefile of California.

As we previously mentioned, we used only a part of the shapefile that was about San Jose, therefore we create a smaller shapefile keeping only the specific region.

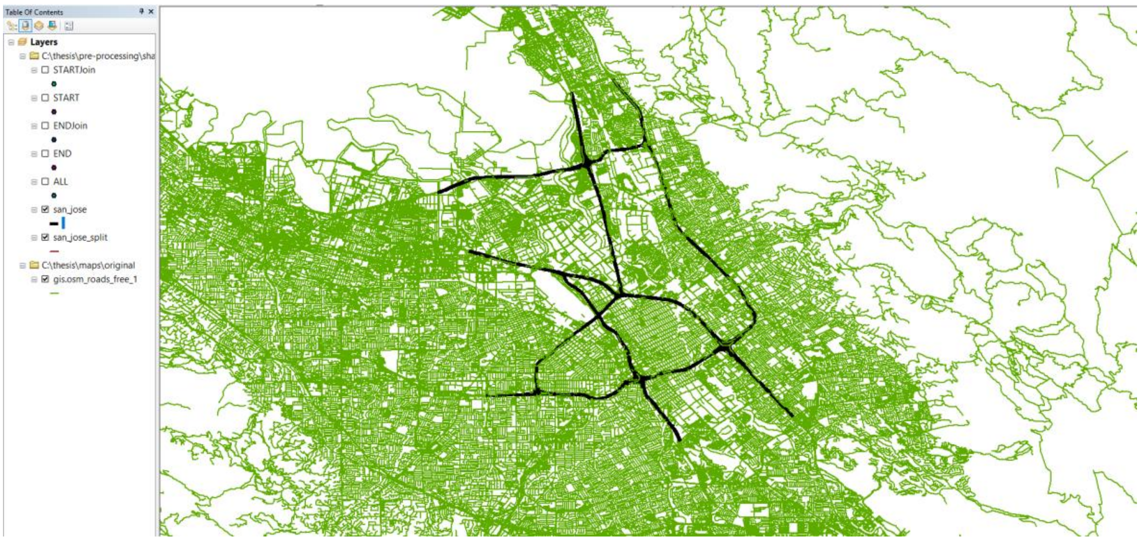
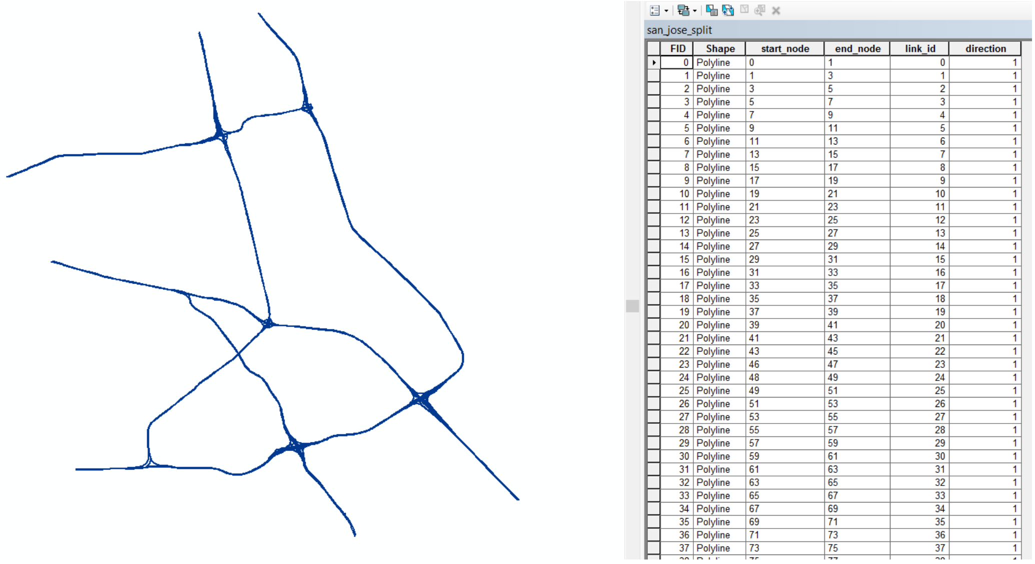


Figure 4 Shapefile of San Jose.

The traffic information that we had, was only from highways. For this reason the final shapefiles included only those road


san_jose_split

FID	Shape	start_node	end_node	link_id	direction
0	Polyline	0	1	0	1
1	Polyline	1	3	1	1
2	Polyline	3	5	2	1
3	Polyline	5	7	3	1
4	Polyline	7	9	4	1
5	Polyline	9	11	5	1
6	Polyline	11	13	6	1
7	Polyline	13	15	7	1
8	Polyline	15	17	8	1
9	Polyline	17	19	9	1
10	Polyline	19	21	10	1
11	Polyline	21	23	11	1
12	Polyline	23	25	12	1
13	Polyline	25	27	13	1
14	Polyline	27	29	14	1
15	Polyline	29	31	15	1
16	Polyline	31	33	16	1
17	Polyline	33	35	17	1
18	Polyline	35	37	18	1
19	Polyline	37	39	19	1
20	Polyline	39	41	20	1
21	Polyline	41	43	21	1
22	Polyline	43	45	22	1
23	Polyline	46	47	23	1
24	Polyline	48	49	24	1
25	Polyline	49	51	25	1
26	Polyline	51	53	26	1
27	Polyline	53	55	27	1
28	Polyline	55	57	28	1
29	Polyline	57	59	29	1
30	Polyline	59	61	30	1
31	Polyline	61	63	31	1
32	Polyline	63	65	32	1
33	Polyline	65	67	33	1
34	Polyline	67	69	34	1
35	Polyline	69	71	35	1
36	Polyline	71	73	36	1
37	Polyline	73	75	37	1

Figure 5 Shapefile with highways of San Jose.

For the creation of the data structure with the spatial information, in terms of road segments and nodes, we used a C++ frameworks that was developed in the context of the European Project e-Compass. This framework uses as input, a shapefile with specific attribute table. This table includes for each road segment an id, the direction which is always one direction in our case because we have data only from highway, the start node id and the end node id. In order to create this format of shapefile we used the geo-processing tools that the ArcMap provides.

More specifically, using the python console of ArcMap and the python library *arcpy* , at first we split the road polylines into links (road segments) polylines using the following script:

```
import arcpy
from arcpy import env
env.workspace = "D:/thesis/sanJoseMap"
arcpy.SplitLine_management("sanJoseMap.shp","D:/thesis/sanJoseMap/sanJoseMap_split.shp"
)
```

Following that we created a node layer for START and END nodes of the links layer, using the following script:

```
import arcpy
from arcpy import env
env.workspace = "D:/thesis/sanJoseMap"
arcpy.FeatureVerticesToPoints_management("sanJoseMap_split.shp",
                                         "D:/thesis/sanJoseMap/ALL.shp",
                                         "ALL")
arcpy.FeatureVerticesToPoints_management("sanJoseMap_split.shp",
                                         "D:/thesis/sanJoseMap/START.shp",
                                         "START")
arcpy.FeatureVerticesToPoints_management("sanJoseMap_split.shp",
                                          "D:/thesis/sanJoseMap/END.shp ",
                                          "END")
```

Running the above script, we had created four new shapefiles the first contains of all the road segments of the map, the second contains all the nodes of the map, the third contains the start nodes of all segments and the forth the end nodes of all segments. In order to combine this information of the four shapefiles we used the spatial join functions of the ArcMap using the following python script we create two new shapefiles where the split shapefile is spatial joined with both start and end nodes.

```
import arcpy
target_features_start = "D:/thesis/sanJoseMap/START.shp"
target_features_end= "D:/thesis/sanJoseMap/END.shp"
join_features = "D:/thesis/sanJoseMap/ALL.shp"
out_feature_class_start = "D:/thesis/sanJose/STARTJoin.shp"
out_feature_class_end = " D:/thesis/sanJose /ENDJoin.shp"
arcpy.SpatialJoin_analysis( target_features_start,
                            join_features,
                            out_feature_class_start,
                            JOIN_ONE_TO_MANY)
arcpy.SpatialJoin_analysis( target_features_end,
                            join_features,
                            out_feature_class_end,
                            JOIN_ONE_TO_MANY)
```

Using simple join, not the spatial one, we join the split shapefile with the results of the above script. Filtering the attributes that created after the join operation in the split shapefile, we created the final format of the shapefile which is suitable for the C++ framework.

3.3 Constuction of the final dataset

As we previously mentioned, the traffic and incident data was acquired from the Caltrans Performance Measurement System. (PeMS). An open government tool from the State of California, which provides in separate datasets information regarding the average speed of the highways of California as well as for the incidents that occurred in a number of different location all over the State.

The average speed of the road was calculated using a number of vehicle detection systems (VDS) which is distributed on the highways of California. The initial dataset of VDS consists of various information, such as the id of VDS, the direction of the road, the district, the county, the city, the coordinates and many else. In our case, we isolated only the VDS_ID, the VDS_LONGITUDE and the VDS_LATITUDE. For each VDS, PEMS provide a file which contains a number of time series with the average speed in mph that this VDS captured. More specifically, the initial speed dataset consists of rows, where each row is a time series which describes the average speed in front of the VDS. Because more than one VDS could correspond to a road, we take the average speed of all VDS of the road, generating the average speed of the road. These time series are separated in 5 minutes' time intervals, which represents the average speed of the road in every single day. So, each row of the dataset consists of 288 time intervals which describe the average speed. PEMS, contains data for more than three thousand VDS, we filter this information using the latitude and longitude, keeping only the VDS of San Jose, which was around 240.

Additionally, the initial incident dataset contains more than 2 million incidents of any type. There are many types of incident such as, traffic hazard, traffic collision, car fire, wrong way driver, animal hit, closure of a road, car accident and many else. Moreover, for each incident, PEMS provides information regarding the longitude and latitude of the incident as well as the timestamp of this incident. We take into consideration any type of incident that PEMS provides and we filtered them based on their longitude and latitude, keeping only the incidents that occurred in the area of San Jose.

Using the C++ framework that builds the network in terms of road segments and nodes we mapped each incident and each VDS to the respective road, using the spatial information that we had from PEMS. The metric that was used for the calculation of the distance between either the incident or the VDS and the center of a road segment was the Euclidian distance. We assumed that an incident that occurred in a radius less than 500 meters from a road, could possible affect the traffic flow of the road. Therefore, using this assumption we attached all the incidents to the respect road.

Finally, knowing how many incidents occurred in a road at a specific timestamp, as well as the average speed of this road in the same timestamp, we construct the final dataset which contains a number of rows where the first attribute was the *ROAD_ID* that the C++ framework generates, the second attribute was the *DAY_ID* (from 0 to 122, which refers to 1 May 2015 to 31 August 2015) and the rest 288 attributes was the 5 minutes time intervals that combines traffic and incident information(for instance: *1;33.4* , where "1" declares that in this interval an incident occurred and the average speed of the road was 33.4 mph)

Table 1 Example of final row data

ROAD_ID	DAY_ID	1	2	3	4	...	286	287	288
118	64	0;69.5	0;69.85	0;69.5	0;69.6	...	0;70.05	0;70.1	0;69.9

This row data was used in order to feed the built network on the C++ framework with traffic and incident information for each road and use this information for the development of the methods that will be discibed in the next sections.

4 Incident Detection and Data Filtering

In order to create our AID model, we used a supervised machine learning algorithm. Specifically, we chose the Support Vector Machines algorithm, which is fairly robust to irrelevant features (Gakis et al. 2014). The basic idea of SVM is to generate a hyperplane that divides the data set into classes. In our case, we have two classes (binary classification problem) which represent the presence or absence of an incident at a specific time interval and road of the traffic network.

In the linear SVM, we are given a training data set with n points of the form (x1, y1), …, (xn, yn) where yi is either 1 or -1 and indicates the class and xi a p-dimensional real vector, called feature vector. In our case we have 5 features so xi is a 5-dimensional feature vector. The objective is to find the maximum-margin hyperplane that divides the group of points xi for which yi = 1 from the group of points for which yi = -1, so that the distance between the hyperplane and the nearest point xi from either group is maximized. Any hyperplane can be written as the set of x satisfying the equation:

$w \cdot x - b = 0$ (1)

where w is the normal vector to the hyperplane and b/|w| a parameter that defines the offset of the hyperplane from the origin along the normal vector w.

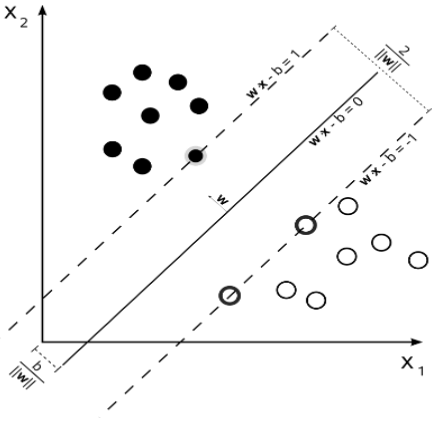


Figure 6 Maximum-margin hyperplane and margins on linear SVM kernel.

In the feature extraction process, we used only speed in order to detect the incidents in a highway. For this reason, we extracted two features based on the speed of the road of interest and its adjacent roads. The speed values used are not only those of the current interval, but also those from a number of intervals prior to the current one.

The first extracted feature F1 is the difference between the speed of the road of interest and the average speed of its adjacent roads, in the direction that the vehicles travel, at the current time interval. This value was normalized by the speed of the road of interest at the current time interval as shown in equation (1).

$$F_1 = \frac{S_{roi,t} - \frac{1}{k}\sum_{j=1}^{k} S_{ar(j),t}}{S_{roi,t}} \qquad (1)$$

S represents speed, ar refers to the adjacent road, roi refers to the road of interest, k is the total number of adjacent roads and t is the current interval.

We also extracted the following three features (based on F1) for three time intervals before the current:

$$F_2 = \frac{S_{roi,t-1} - \frac{1}{k}\sum_{j=1}^{k} S_{ar(j),t-1}}{S_{roi,t-1}} \qquad (2)$$

$$F_3 = \frac{S_{roi,t-2} - \frac{1}{k}\sum_{j=1}^{k} S_{ar(j),t-2}}{S_{roi,t-2}} \qquad (3)$$

$$F_4 = \frac{S_{roi,t-3} - \frac{1}{k}\sum_{j=1}^{k} S_{ar(j),t-3}}{S_{roi,t-3}} \qquad (4)$$

The selection of the optimal number of previous intervals, was made after experimentation with various numbers.

The reason for selecting the above features is the observation that when an incident occurs in a road, the speed of this road and its neighbor roads, in the direction that the vehicles travel, decreases. Nonetheless, if we take into account only the values of these features, the model will be trained to detect only the case that the speed of the current road is low and in particular, much lower that the speed of the next roads. Therefore, the selection of one more feature was necessary. We used as additional feature, the average absolute deviation of the speed of the road of interest in the current interval from its average speed value of all the previous intervals up to current (including the current). The feature's formulation is shown in the following equation:

$$F_5 = \frac{\sum_{j=0}^{p} | S_{roi,t-j} - m(S_{roi,t-j}) |}{p+1},$$
$$m(S_{roi,t-j}) = \frac{\sum_{k=0}^{t-j} S_{roi,t-j-k}}{t-j+1} \qquad (5)$$

where S is the speed of the road of interest, p the number of past intervals that we calculate the average of and m(S) the average value. Using these features we generated a set of feature vectors for each road of interest. Each feature vector is a 5-dimensional vector which values are those of the features F1, …, F5. Using these feature vectors a different SVM-based AID model was built for each road of interest.

Finally, we experimented with various values for the C parameter of the SVM algorithm, using one-out cross validation, in order to estimate those that fit better to our case. Using a grid search on C = 2-5, 2-3, …, 215 with step 2. We concluded that the most suitable value for the C parameter is 1.1.

5 Traffic Prediction

For the task of traffic prediction, we used two models: (a) the ARIMA parametric model to predict traffic under typical conditions and (b) the k-NN model to predict traffic under atypical.

5.1 Autoregressive Integrated Moving Average Model

The Auto Regressive Integrated Moving Average (ARIMA) family of models is the most widely deployed approach for vehicular traffic prediction and for time series prediction in general. ARIMA is a generalisation of the Auto-Regressive Moving Average (ARMA) model, which is applied strictly to stationary time series.

An ARIMA (p, d, q) process is expressed as:

$$\left(1-\sum_{i=1}^{p}\varphi_i L^i\right)\cdot(1-L)^d\cdot X_t=\left(1+\sum_{i=1}^{q}\theta_i L^i\right)\cdot\varepsilon_t \qquad (6)$$

where p is the order of the autoregressive model, d is the degree of differencing and q is the order of the moving average model. In our case, we used an ARIMA (3, 1, 0) model with three previous terms and 1st degree of differencing for reaching stationarity. The resulted model is shown in equation:

$$S'_{roi,t}=\varphi_1\cdot S'_{roi,t-1}+\varphi_2\cdot S'_{roi,t-2}+\varphi_3\cdot S'_{roi,t-3} \qquad (7)$$

where

$$S'_{roi,t}=S_{roi,t}-S_{roi,t-1} \qquad (8)$$

the differenced S’ process which is wide-sense stationary. According to Pfeifer and Deutsch, the best estimate of the f parameters are the maximum likelihood estimates (Pfeifer and Deutsch, 1980). But since without a priori knowledge of their initial values, these estimates cannot be exactly computed, a close approximation via ordinary least squares (OLS) is achieved. In particular, for every training sample an equation of the form of (10), is constructed where f are the unknown parameters. This forms a linear overdetermined system of equations of the form:

$$y = X \cdot \beta \qquad (9)$$

This system using the normal equations is written as:

$$\left(X^T X\right) \cdot \overset{\wedge}{\beta} = X^T \cdot y \qquad (10)$$

Using the OLS method we take the following solution:

$$\overset{\wedge}{\beta} = \left(X^T \cdot X\right)^{-1} \cdot X^T \cdot y \qquad (11)$$

When the model is built (the f parameters have been estimated) the following equation is used for prediction:

$$S'_{t+h} = \varphi_1 \cdot S'_t + \varphi_2 \cdot S'_{t-1} + \varphi_3 \cdot S'_{t-2} \qquad (12)$$

where, h is the prediction horizon which for the case of short-term prediction can be up to one hour.

5.2 k-Nearest Neighbors

For the prediction of the speed values under atypical conditions we have chosen the k-NN regression model. k-NN is a non-parametric algorithm that stores all available cases and predicts the numerical target based on a similarity measure and an averaging scheme. The k-NN algorithm has been used in statistical estimation and pattern recognition already in the beginning of 1970’s as a non-parametric technique.

k-NN prediction is based on the current state vector (at current time interval t) which has the following form:

$$y_{roi,t} = \left[S_{roi,t}, S_{roi,t-1}, S_{roi,t-2}, \ldots, S_{roi,t-p}\right] \qquad (13)$$

where S is the traffic variable (in our case speed) and p the number of past intervals. As shown, the current state vector of a road of interest depends on the values of speed at the current and previous p time intervals. In order to make prediction, the k-NN algorithm creates vectors of the form (15), y1,t, y2,t, … , yN,t for N other roads of the network. When a prediction for the road of interest for h intervals ahead in time is requested, the algorithm compares yroi,t, with y1,t, y2,t, … , yN,t using a distance metric (usually Euclidean distance), and keeps the k vectors with the smallest distances. Then, it calculates the value Sroi,t+h using an averaging scheme on the estimated k neighbors, which in the simplest form is the following:

$$S_{roi,t+h} = \frac{\sum_{i=1}^{k} S_{i,t+h}}{k} \qquad (14)$$

In our implementation, we used the inverse distance weighted average as averaging scheme as follows:

$$S_{roi,t+h} = \frac{\sum_{i=1}^{k} w_i S_{i,t+h}}{k} \qquad (15)$$

where:

$$\begin{aligned} w_i &= \frac{1}{d_i}, \\ d_i &= d_{roi,i} = \sqrt{\sum_{j=0}^{p} \left(S_{roi,t-j} - S_{i,t-j} \right)^2} \end{aligned} \qquad (16)$$

We chose the optimal value for k via one-out cross validation in our data set, based on the prediction accuracy results. By this process, we concluded that the most optimal value of k in our case was 6.

6 Evaluation and Benchmarking

In this chapter, we present the set-up of the evaluation framework, including the construction of the traffic time series and their enrichment with incident information, the choice of a specific part of the Caltrans road network as case study and the separation of the training and test data. Finally, the preliminary evaluation results are presented.

6.1 Constructing Traffic Time Series with Incident Information

In order to build and evaluate our model the first step was to pre-process the initial data (both the traffic and the incident) in order to create traffic time series that will include incident information.

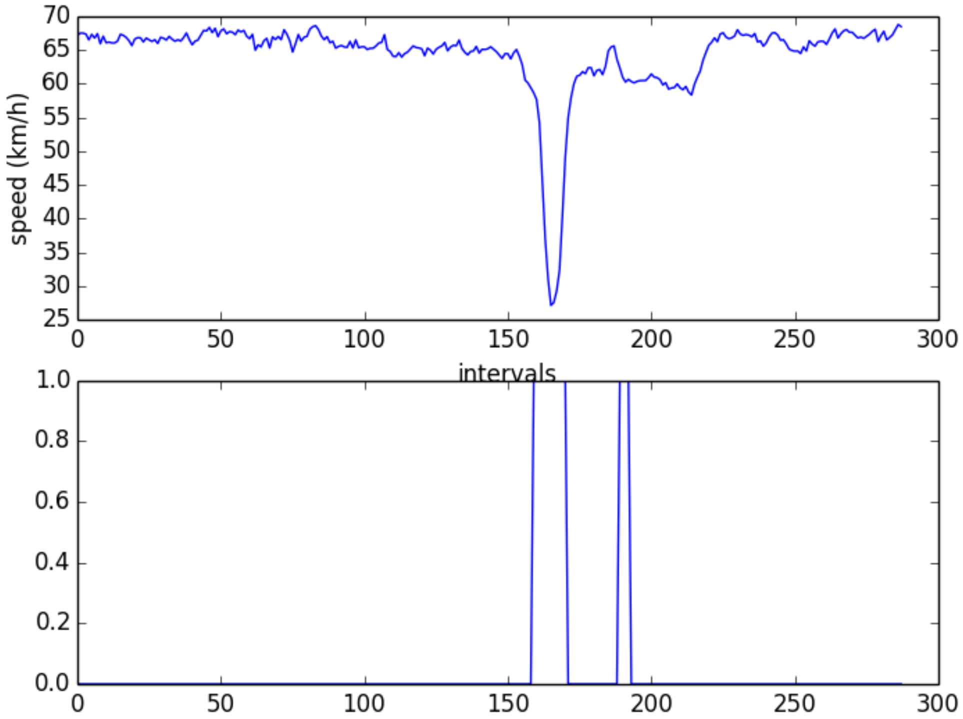


Figure 7 Traffic and Incident data

In the examined area, there are 350 VDS in total, from which 112 were not taken into account because they had traffic data with only zero values. The traffic data from the remaining 238 VDS were matched to road segments of the Caltrans network (based on their coordinates). This process resulted into 55 road segments having traffic data. As we described above, the features of our AID model take into account not only the speed of the road of interest, but also the speed of its adjacent roads. For this reason, we kept only the road segments for which, as well as their spatial neighbors traffic data exist.

Concerning incident data, there were 4193 incidents in the area and time period concerned. These incidents where matched to the aforementioned 55 road segments which have traffic data. For each day of the total examined period and each of these 55 road segments, a speed time series was constructed where each time series member was the speed value for the specific 5-minute interval of this day and road segment. In this way 6,600 (55 road segments times 120 days of traffic data for each road segment) speed were constructed.

These time series except of the traffic information include typical and atypical intervals, indicated by 0 and 1 respectively. The value 1 means presence of an incident in the specific time interval and 0 absence. For instance, for the road segment with identification number 76 on May 21, 2015 on time interval 01:10-01:15 the corresponding value of the speed time series is ‘66.55;0’. This means that the speed was 66.55 miles per hour and no incident situation was present.

6.2 Case study: A Part of US101 Highway

One of the main difficulties when trying to predict traffic under atypical conditions, is that the effect of abnormalities on traffic time series most of the times is not easily observable and interpretable. For instance, there may be an incident with specific characteristics (type, duration, severity, etc.) that caused a steep fall on the traffic time series of a road network, and another incident with exactly the same characteristics that happened on the same road at a different time of the day and had no effect on the traffic time series.

On the other hand, there may be observable discrepancies from the typical pattern of the traffic time series that do not necessarily correspond to the presence of an incident. These situations may confuse the AID model. In order to overcome these difficulties, we had to choose road segments with observable effects on their traffic time series due to occurring incidents. By examining all the 55 road segments, we finally reached one specific road segment which is part of the US 101 freeway (from km to km), and kept its traffic time series in order to build and evaluate our model.

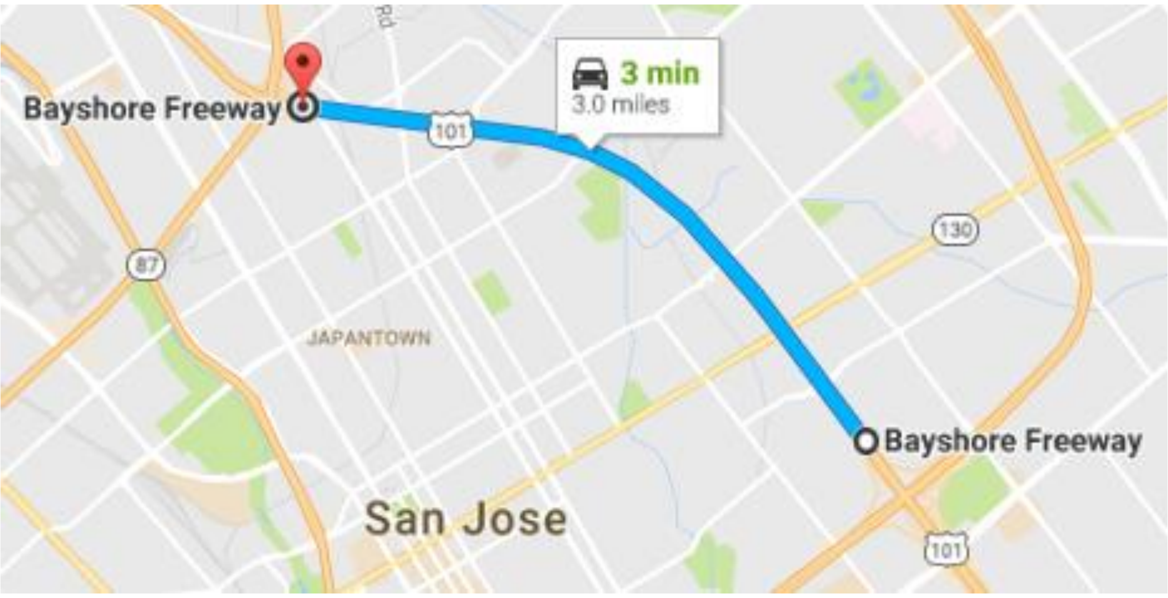


Figure 8 Part of US 101 Highway

6.3 Training Versus Testing Data

The traffic time series of the aforementioned road segment consist the main data set. From this, the one that corresponds to 25 August, 2015 was selected as a test time series, which has both typical and atypical intervals. This time series was selected because it has spikes that corresponds to the occurrence of incidents. Also, is one of the final days of the total examined period, which means that it left us the choice of having more time series for building our model.

The preceding 116-time series formed the training data set. From this, we created three separated data sets in order to fit our model and the benchmarking methods in different traffic conditions. The first data set includes only the ones without atypical intervals (incident-free), whereas the second data set includes those with both typical and atypical (incident). Finally, the third one contains all time series (total).

We trained the AID and the k-NN models using the total training data set, whereas for the ARIMA model the incident-free data set was used. Hence, we incorporate incident information to the fitting process of our model, as opposed to the current related work.

6.4 Benchmarks and Accuracy Metrics

For the evaluation of the AID model we calculated a number of metrics using one-out cross validation in the total training data set. The first metric that we calculated was the accuracy:

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \qquad (17)$$

where TP is the true positive, TN the true negative, FP the false positive and FN the false negative predicted classes. However, accuracy is not really a reliable metric for the real performance of a classifier when the number of samples in different classes vary greatly (unbalanced target) because it will yield misleading results. In our case, from the total number of 288 intervals in the traffic time series, only in 30 or less intervals an incident was occurred. For this reason, in order to evaluate our model accurately, we calculated two additional metrics. The first one was the sensitivity, a measurement of the proportion of positives that are correctly identified, whose formula is shown below:

$$Sensitivity = \frac{TP}{TP + FN} \qquad (18)$$

The second additional metric was specificity, which measures the proportion of negatives that are correctly identified. Its formula is given by the following equation:

$$Specificity = \frac{TN}{TN + FP} \qquad (19)$$

Using the sensitivity and specificity we created the Receiver Operating Characteristic (ROC) curve, which illustrates the performance of our classifier.

For benchmarking we used the unitary ARIMA and k-NN models. These models were initially fitted using only the incident-free training data set, as happens in most of the works on traffic prediction under atypical conditions in current literature, and then using different combinations of all three datasets (incident-free, incident and total). We assessed the resulted accuracies by the means of two metrics: (a) the Root Mean Square Error (RMSE) and (b) the Symmetric Mean Absolute Percentage Error (SMAPE).

RMSE is given by the following formula:

$$RMSE = \sqrt{\frac{\sum_{t=1}^{n}(P_t - A_t)^2}{n}} \qquad (20)$$

where n is the number of predictions, At the actual values and Pt the predicted values.

SMAPE gives a percentage error that has both a lower and an upper bound of 0% and 100% respectively. This makes its values more easily interpretable. The formula of SMAPE is given in the following equation:

$$SMAPE = \frac{100\%}{n} \cdot \sum_{t=1}^{n} \frac{|P_t - A_t|}{|A_t| + |P_t|} \qquad (21)$$

where n is the number of predictions, At the actual values and Pt the predicted values.

6.5 Experimental results

The evaluation results of the AID model are shown in Table 2. As it is shown despite the fact that the accuracy is high, the true positive rate is slightly higher than 60%, while the true negative rate is over 90%. This mean that 6 out of 10 incidents are detected and 1 out of 10 normal conditions are detected as abnormal. The impact of false alarms in our occasion is very low, so we can say that in a future work, we can sacrifice the true negative rate in order to increase the true positive rate. At this point, I need to declare that higher sensitivity does not mean, for sure, better performance of the pattern transition model but only better performance of the automated incident detection model

Table 2: The evaluation results of the AID model.

AID evaluation metrics

Accuracy	0.8986
Sensitivity	0.6364
Specificity	0.9091

Also, the ROC curve of the classifier is shown in Figure 9.

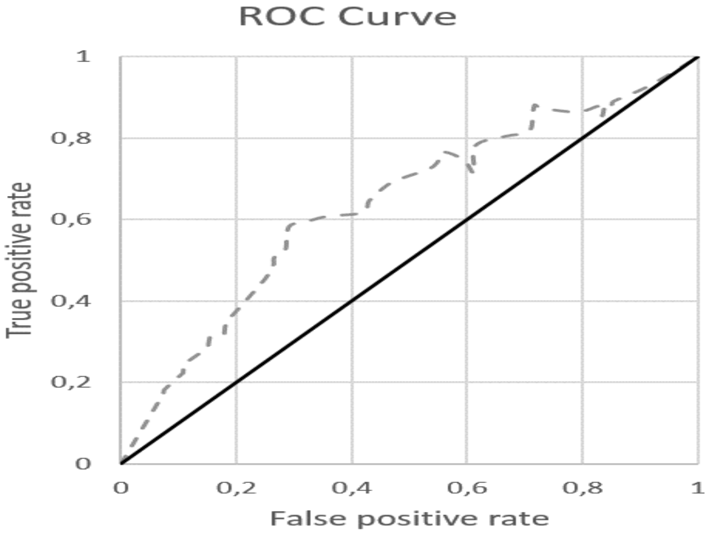


Figure 9 ROC curve of the proposed AID schema.

We can see that although the proposed model is quite above the line of no-discrimination (the diagonal line), it is also quite far from the upper left corner of the ROC space (best possible classification prediction). This mainly happens due to the imbalance of the records of the classification classes (incident, non-incident) in the examined data set. In any case, the curve shows that there is enough room for improvement for the proposed AID model.

The results of the experiments regarding the prediction accuracy of our model are shown in Figure 10 and Figure 11.

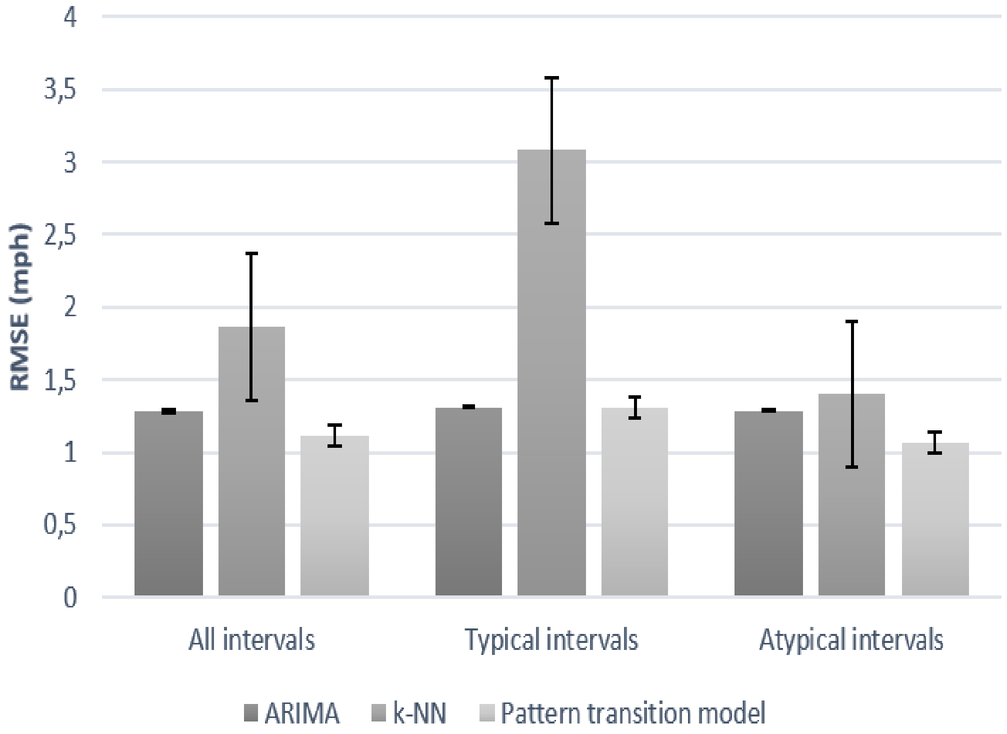


Figure 10 Prediction accuracy results in RMSE.

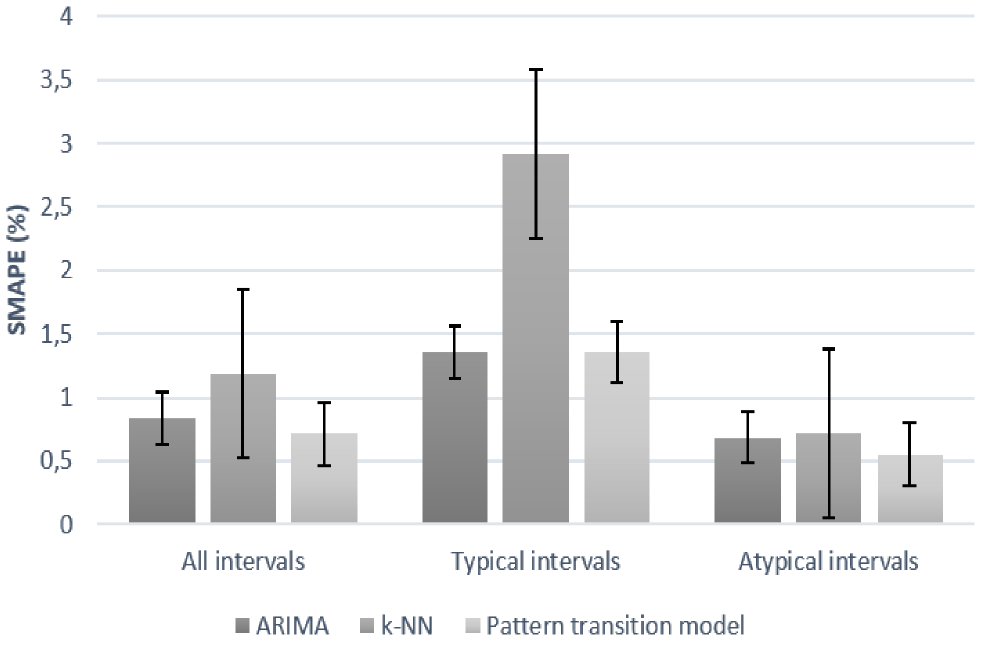


Figure 11 Prediction accuracy results in SMAPE.

As shown, in total the proposed model outperforms its competitors. In particular, our model presents almost similar prediction accuracy with the ARIMA model under typical conditions, but it is superior under atypical conditions.

As already mentioned the benchmarking models were initially fitted with incident-free data. Subsequently, we conducted a series of experiments in which the two unitary benchmarking models were fitted using different combinations of the incident-free, incident and total data sets. In this way, we incorporated the incident information not only in the fitting process of the proposed model, but also in the fitting process of its competitors. As shown in Figure 12 and Figure 13, again the proposed model presents superior accuracy for all intervals.

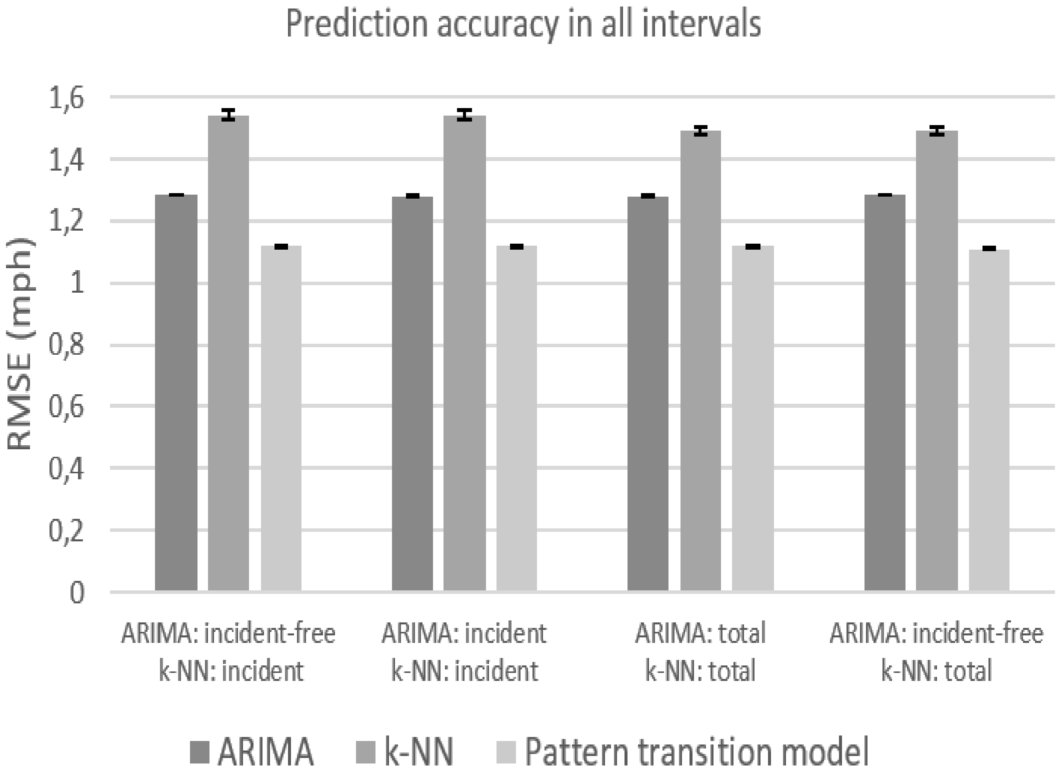


Figure 12 Prediction accuracy results in RMSE, for different benchmarking combinations and for all intervals.

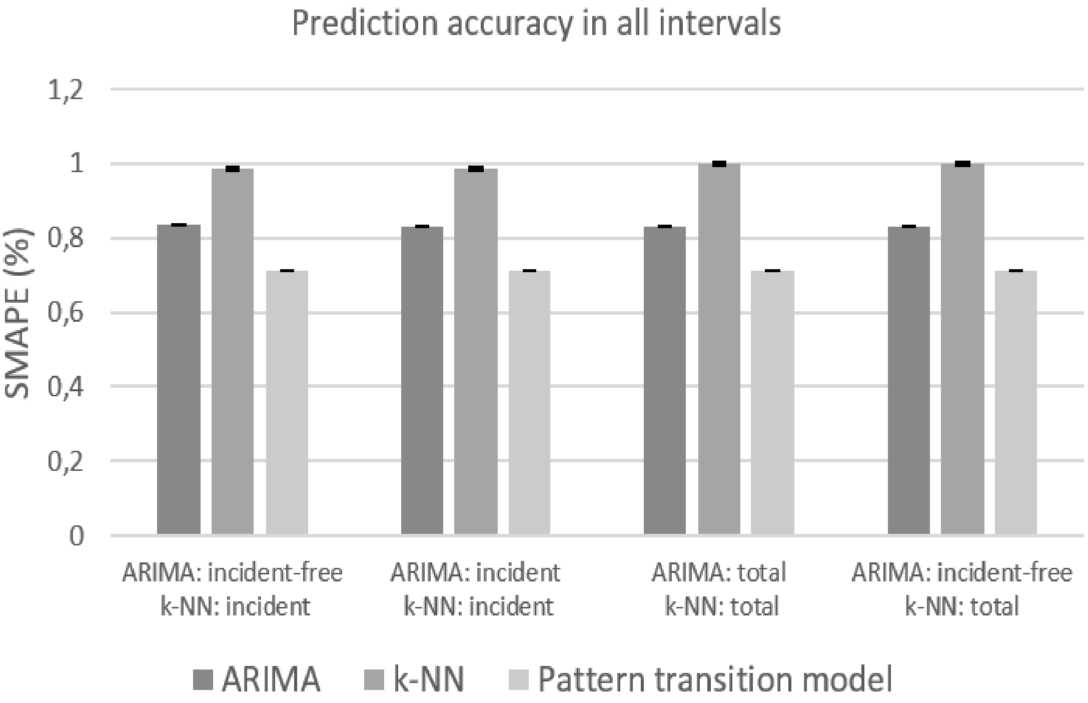


Figure 13 Prediction accuracy results in SMAPE, for different benchmarking combinations and for all intervals.

Finally, we examined the null hypothesis that the proposed model has equal accuracy with the ARIMA model using a statistical t-test. Since there is no indication that the predicted values have normal distributions, we used the Wilcoxon signed-rank test. The test showed that at significance level of 0.05 the null hypothesis could be rejected for all the aforementioned benchmarking cases. Therefore, we can say that the proposed model presents statistically significantly better accuracy from the ARIMA model in all cases.

Taking the aforementioned into consideration, it is obvious that the proposed model adds a value to the current research works. Nonetheless, we must not fail to mentioned that the evaluation of the model is done at a selected road that performs the needed abnormalities in terms of speed fluctuation. These fluctuations are necessary in order to train our model correctly.

7 Conclusion

In this dissertation, we introduced a novel hybrid method for short-term traffic prediction under both typical and atypical traffic conditions. We introduced a SVM-based AID model that identifies the presence of atypical conditions. We use the ARIMA parametric model or the k-NN non-parametric regression model if the AID identifies typical or atypical conditions respectively. We evaluated our model using real open data from the Caltrans PeMS and showed that it outperforms the benchmarking models in terms of prediction accuracy under both typical and atypical conditions.

The main advantage of this approach is that we use the incident information during the training stage and so the model is able to predict more accurate the speed value when an atypical condition occurred. Additionally, the proposed model is able to predict the speed value either the time series contains incidents or not. This mean that our model is a tolerant model that does not affected easily from the testing time series.

On the other hand, the proposed methodology is very sensitive during the training stage. The model must be built using the correct time series which fulfill specific conditions, such as the fluctuation of speed when an incident occurred. Moreover, this restriction means that in some kind roads the proposed model could not be something better than ARIMA. This kind of roads, are roads that there are not many incidents and the incidents that occur does not affect the average speed.

Future work involves experimenting with additional feature extraction techniques for the proposed AID model and comparing it with different incident detection models from the literature. Additionally, other algorithms rather than SVM could be developed and be used for the AID part of this approach. Algorithms such as random forest could be a good idea, taking into account the imbalanced dataset between incident and incident-free intervals.

Furthermore, in this dissertation we used a four months’ dataset of highways of San Jose, California. It is necessary the comparison of the overall proposed model with additional prediction models using larger data sets that will include more test cases. This future work is essentially in order to have further exanimate the usability of the proposed model in more cases. Additionally, the discovery of abnormalities could be done with new technics in addition to the machine learning. For instance, the social media analysis could be useful in order to mine information about an abnormal event that could affect the traffic in a road.

Generally, the pattern transition (regime switching) models are able to perform better, more efficient and more accurate result in traffic prediction because they can be built for more than one conditions, in our case two (typical, atypical). For this reason, the selection of different algorithms in addition to ARIMA and k-NN is something that can be done in the future in order to create the best combination.

8 Bibliography

Abdulhai, B., Porwal H., Recker, W., 1997. Short term freeway traffic flow prediction using genetically optimized time delay based neural net-works. Proc., 78th Transportation Research Board Annual Meeting, Washington D. C., USA.

Box, G. E. P. and Jenkins, G. M., 1971. Time series analysis forecasting and control. Operational Research Quarterly, 22(2), June, pp.199-201.

Castro-Neto, M., Jeong, Y. S., Jeong, M. K., Han, L. D., 2009. Online-SVR for short-term traffic flow prediction under typical and atypical traffic conditions. Expert Systems with Applications, 36(3), pp.6164-6173.

Clark, S., 2003. Traffic Prediction Using Multivariate Nonparametric Regression. Journal of Transportation Engineering, 129(2), pp.161-168.

De Fabritiis, C., Ragona, R., Valenti, G., 2008. Traffic estimation and prediction based on real time floating car data. Proc., IEEE 11th International Conference on Intelligent Transportation Systems, pp.197-203.

Dougherty M. S. and Cobbet M. R., 1997. Short term inter-urban traffic fore-casts using neural networks. International Journal of Forecasting, 13(1), March, pp.21-31.

Gakis E., Kehagias D., Tzovaras D., 2014. Mining Traffic for Road Incidents Detection. IEEE 17th International Conference on Intelligent Transportation Systems, Qingdao, China, pp. 930-935.

Gao, Y. and Er, M., J., 2005. Narmax time series model prediction: feed-forward and recurrent fuzzy neural network approaches. Fuzzy Sets and Systems, 150(2), March, pp.331-350.

Ghosh, B., Basu, B., O’Mahony, M., 2009. Multivariate short-term traffic flow forecasting using time-series analysis. IEEE Transactions on Intelligent Transportation Systems, 10(2), pp.246-254.

Guo, F., Krishnan, R., Polak, J. W., 2012. Short-Term Traffic Prediction Under Normal and Abnormal Traffic Conditions on Urban Roads. 91st Transportation Research Board Annual Meeting, Washington D. C., USA.

Guo, F., Krishnan, R., Polak, J. W., 2014. A novel three-stage framework for short-term travel time prediction under normal and abnormal traffic conditions. 93rd Transportation Research Board Annual Meeting, Washington D. C., USA.

Guo, F., Polak, J. W., Krishnan, R., 2010. Comparison of Modelling Approaches for Short-Term Traffic Prediction under Normal and Abnormal Conditions. IEEE 13th International Conference on Intelligent Transportation Systems, Madeira Island, Portugal.

Guo, J. and Williams, B. M., 2010. Real-time short-term traffic speed level forecasting and uncertainty quantification using layered Kalman filters. Transportation Research Record: Journal of Transportation Research Board, 2175, pp.28-37.

Hu, W., Yan, L., Liu, K., Wang, H., 2015. A Short-term Traffic Flow Forecasting Method Based on the Hybrid PSO-SVR. Neural Processing Letters, pp.1-18.

Innamaa, S., 2000. Short term prediction of traffic situation using MLP-neural networks. Proc., 7th World Congress on Intelligent Systems, Turin, Italy, pp.1-8.

Kamarianakis, Y. and Prastacos, P., 2005. Space-time modelling of traffic flow. Computers & Geosciences, 31(2), pp.119-133.

Kamarianakis, Y., Shen, W., Wynter, L., 2012. Real-time road traffic forecasting using regime-switching space-time models and adaptive LASSO. Applied Stochastic Models in Business Industry, 28(4), pp.297-315.

Kindzerske, M. D. and Ni, D., 2007. Composite nearest neighbour nonparametric regression to improve traffic prediction. Transportation Research Record: Journal of Transportation Research Board. 1993(1), pp.30-35.

Min, W. and Wynter, L., 2011. Real-time traffic prediction with spatiotemporal correlations. Transportation Research Part C: Emerging Technologies, 19(4), August, pp.606-616.

Mu, T., Jiang, J., Wang, Y., 2012. Heterogeneous delay embedding for travel time and energy cost prediction via regression analysis. IEEE Transactions on Intelligent Transportation Systems, 14(1), pp. 214-224.

Myung, J., Kim, D. K., Kho, S. Y., Park, C. H., 2012. Travel Time Prediction Using k-Nearest Neighbour Method with Combined Data from Vehicle Detector System and Automatic Toll Collection System. Transportation Research Record: Journal of Transportation Research Board, 2256, pp.51-59.

Ni, M., He, Q., Gao, J., 2014. Using Social Media to Predict Traffic Flow under Special Event Conditions. 93rd Transportation Research Board Annual Meeting, Washington D. C., USA.

Pfeifer, P. E. and Deutsch, S. J., 1980. A three-stage iterative procedure for space-time modelling. Technometrics, 22(1), February, pp.35-47.

Quek, C., Pasqueir, M. and Lim, B. B. S., 2006. Pop-traffic: a novel fuzzy neural approach to road traffic analysis and prediction. IEEE Transactions on Intelligent Transportation Systems, 7(2), June, pp.133-146.

Smith, B. L. and Demetsky, M. J., 1996. Multiple interval freeway traffic flow prediction. Transportation Research Record: Journal of Transportation Research Board, 155(4), pp.136-141.

Stathopoulos, A. and Karlaftis, M. G., 2003. A multivariate state-space approach for urban traffic flow modelling and prediction. Transportation Research Part C: Emerging Technologies, 11(2), April, pp.121-135.

Vlahogianni, E. I., Karlaftis, M. G., Golias, J. C., 2003. A multivariate neural network predictor for short term traffic prediction in urban signalized arterial. Proc. 10th IFAC Symposium on Control in Transportation Systems, Tokyo, Japan, August.

Williams, B. M., Dursavula, P. K., Brown, D. E., 1998. Urban freeway traffic flow prediction – Application of seasonal autoregressive integrated moving average and exponential smoothing models. Transportation Research Record: Journal of Transportation Research Board, 1644, pp.132-141.

Wu, C. H., Wei, C. C., Su, D. C., Chang, M. H., Ho, J. M., 2003. Travel Time Prediction with Support Vector Regression. IEEE 6th International Conference on Intelligent Transportation Systems, Shanghai, China.

Wu, T., Xie, K., Xinpin, D., Song, G., 2012. A online boosting approach for traffic flow forecasting under abnormal conditions. Proc., 9th International Conference on Fuzzy Systems and Knowledge Discovery, Sichuan, China.

Zhang, G. P., 2003. Time series prediction using a hybrid ARIMA and neural network model. Neurocomputing, 50, January, pp.159-175.

Zheng, Z., Su, D., 2014. Short-term traffic volume forecasting: A k-nearest neighbour approach enhanced by constrained linearly sewing principle component algorithm. Transportation Research Part C: Emerging Technologies, 43, pp.143-157.