Compression, estimation and prediction models for large road networks

Muhammad Tayyab Asif

2016

Muhammad Tayyab Asif. (2016). Compression, estimation and prediction models for large road networks. Doctoral thesis, Nanyang Technological University, Singapore.

<https://hdl.handle.net/10356/65985>

<https://doi.org/10.32657/10356/65985>

Compression, estimation and prediction models for large road networks

Muhammad Tayyab Asif

School of Electrical & Electronic Engineering

A thesis submitted to the Nanyang Technological University
in partial fulfilment of the requirements for the degree of
Doctor of Philosophy

2016

Statement of Originality

I hereby certify that the work embodied in this thesis is the result of original research and has not been submitted for a higher degree to any other University or Institution.

.....

Date

.....

Muhammad Tayyab Asif

Acknowledgements

To my family and folks.

I would like to express my sincere gratitude to my supervisor, Prof. Justin Dauwels for his consistent support and guidance since I joined Nanyang Technological University (NTU). I am indebted to him for teaching me how to translate abstract ideas into useful research problems. He has been a constant source of inspiration for me throughout this endeavour.

I also owe my sincere appreciation to our collaborator Prof. Patrick Jaillet for the ever insightful discussions.

I would also like to express my gratefulness towards my friends and lab mates, whose constant and never ending support has helped me complete this endeavor successfully!

Abstract

Compression, estimation and prediction models for large road networks

by

Muhammad Tayyab Asif¹

Supervisor: Asst. Prof. Justin Dauwels¹

¹ Nanyang Technological University, Singapore

Developing models for large road networks is not a trivial task. We propose data-driven models that can help intelligent transportation systems (ITS) in dealing with issues such as managing large amount of traffic information and incomplete data sets. Furthermore, we also discuss the problem of traffic prediction in the context of large-scale networks.

Our main objective is to develop generic models that can be applied to practical city-scale networks which are typically composed of different types of roads. Let us start with the problem of data management. Intelligent transportation systems require data with high spatial and temporal resolution. This leads to the issue of efficiently storing and managing large amount of traffic information. For efficient data storage and management, we develop a two-step near lossless compression algorithm for data obtained from large road networks. In the first step, we obtain the common global traffic patterns by applying various subspace methods. Then, we apply Huffman coding to keep the maximum loss of traffic information (due to compression) below a pre-specified threshold.

Naturally, the traffic information obtained from on ground sensors is usually incomplete. Consequently, missing data is a common problem faced by ITS. We propose

and analyze performances of various matrix and tensor based methods to impute missing information in traffic data sets. These methods try to estimate missing traffic information by extracting dominant traffic patterns from incomplete data sets. We analyze the performance of these methods for different road categories and during different days of the week. We also study the impact of the choice of latent traffic patterns on the estimation accuracy of various matrix and tensor completion methods.

Finally, we study the problem of large-scale traffic prediction. Information about future traffic conditions can potentially improve the performance of ITS. However, the performance of prediction algorithms may not remain the same across the network and during different time instances. To help ITS deal with this uncertainty, we propose various data-mining methods that can extract patterns related to these variations in prediction performance across large networks.

The results show that data-driven models can efficiently extract global traffic trends from large and heterogeneous road networks. These models can prove useful in solving the issue of data management as well as recovering missing traffic information. Furthermore, data-driven models can also help predictive ITS applications in hedging the inherent uncertainty associated with predicted traffic conditions.

Author's publications

Journal Papers:

- [J1] N. Mitrovic, A. Narayanan, M. T. Asif, A. Rauf, J. Dauwels, P. Jaillet, **On Centralized and Decentralized Architectures for Traffic Applications**, IEEE Transactions on Intelligent Transportation Systems, 2016 (*Accepted for publication*).

- [J2] M. T. Asif, N. Mitrovic, J. Dauwels, and P. Jaillet, **Matrix and tensor based methods for missing data estimation in large traffic networks**, IEEE Transactions on Intelligent Transportation Systems, 2015 (*Accepted for publication*).

- [J3] N. Mitrovic, M.T. Asif, J. Dauwels, and P. Jaillet, **Low-dimensional Models for Compressed Sensing, and Prediction of Large-Scale Traffic Data**, IEEE Transactions on Intelligent Transportation Systems, vol. 16, no. 5 (2015): 2949-2954.

- [J4] M.T. Asif, S. Kannan, N. Mitrovic, J. Dauwels, and P. Jaillet, **Near-Lossless Compression for Large Traffic Networks**, IEEE Transactions on Intelligent Transportation Systems, vol. 16, no. 4 (2015): 1817-1826.

- [J5] M. T. Asif, J. Dauwels, C. Y. Goh, A. Oran, E. Fathi, M. Xu, M. Dhanya, N. Mitrovic, and P. Jaillet, **Spatiotemporal patterns in large-scale traffic**

speed prediction, IEEE Transactions on Intelligent Transportation Systems, vol. 15, no. 2 (2014): 794-804.

Conference Papers:

- [C1] Narayanan, A., N. Mitrovic, M.T. Asif, J. Dauwels and P. Jaillet, **Travel Time Estimation using Speed Predictions**, 18th International IEEE Conference on Intelligent Transportation Systems (ITSC), 2015 (*accepted for publication*).
- [C2] W. L. Leong, K. Lee, K. Swapnil, X. Li, H. Y. Tong Victor, N. Mitrovic, M. T. Asif, J. Dauwels, and P. Jaillet, **Improving traffic prediction by including rainfall data**, 14th Intelligent Transportation Systems Asia-Pacific Forum, 2015 (*Top Ten Best Paper Award*).
- [C3] Jere, S., J. Dauwels, M.T. Asif, N. Mitrovic, A. Cichocki, and P. Jaillet, **Extracting Commuting Patterns in Railway Networks through Matrix Decompositions**, 13th IEEE International Conference on Control, Automation, Robotics and Vision (ICARCV), 2014.
- [C4] Dauwels, J., A. Aslam, M.T. Asif, X. Zhao, N. Mitrovic, A. Cichocki, and P. Jaillet, **Predicting Traffic Speed in Urban Transportation Subnetworks for Multiple Horizons**, 13th IEEE International Conference on Control, Automation, Robotics and Vision (ICARCV), 2014.
- [C5] Ansar, R., P. Sarampakhul, S. Ghosh, N. Mitrovic, M.T. Asif, J. Dauwels, and P. Jaillet, **Evaluation of Smart-Phone Performance for Real-Time Traffic Prediction**, 17th IEEE International Conference on Intelligent Transportation Systems (ITSC), pp. 3010-3015. IEEE, 2014.
- [C6] Mohan, D.M., M.T. Asif, N. Mitrovic, J. Dauwels, and P. Jaillet, **Wavelets on Graphs with Application to Transportation Networks**, 17th IEEE

-
- International Conference on Intelligent Transportation Systems (ITSC), pp. 1707-1712. IEEE, 2014.
- [C7] N. Mitrovic, M. T. Asif, J. Dauwels, P. Jaillet, **Compressed Prediction of Large-Scale Urban Traffic**, 39th IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), pp. 5984-5988. IEEE, 2014.
- [C8] G. Gopi, J. Dauwels, M. T. Asif, S. Ashwin, N. Mitrovic, U. Rasheed and P. Jaillet, **Bayesian Support Vector Regression for Traffic Speed Prediction with Error Bars**, 16th IEEE Intelligent Transportation Systems Conference (ITSC), 2013, pp. 136-141. IEEE, 2013.
- [C9] N. Mitrovic, M. T. Asif, U. Rasheed, J. Dauwels and P. Jaillet, **CUR Decomposition for Compression and Compressed Sensing of Large-Scale Traffic Data**, 16th IEEE Intelligent Transportation Systems Conference (ITSC), 2013, pp.1475-1480. IEEE, 2013.
- [C10] M. T. Asif, K. Srinivasan, J. Dauwels and P. Jaillet, **Data compression techniques for urban traffic data**, IEEE Symposium on Computational Intelligence in Vehicles and Transportation Systems (CIVTS), 2013, pp. 44-49. IEEE, 2013.
- [C11] M. T. Asif, N. Mitrovic, L. Garg, J. Dauwels, and P. Jaillet, **Low-dimensional models for missing data imputation in road networks**, 38th IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2013, pp. 3527-3531. IEEE, 2013.
- [C12] M. T. Asif, J. Dauwels, C. Y. Goh, A. Oran, E. Fathi, M. Xu, M. M. Dhanya, N. Mitrovic and P. Jaillet, **Unsupervised Learning Based Performance Analysis of n-Support Vector Regression for Speed Prediction of a Large Road Network**, 15th IEEE Intelligent Transportation Systems Conference (ITSC), pp. 983-988. IEEE, 2012.

- [C13] C.Y. Goh, J. Dauwels, N. Mitrovic, M. T. Asif, A. Oran, and P. Jaillet, **Online map-matching based on Hidden Markov model for real-time traffic sensing applications**, 15th IEEE Intelligent Transportation Systems Conference (ITSC), pp. 776-781. IEEE, 2012.

Patent:

- [P1] J. Dauwels, M. T. Asif, N. Mitrovic, and P. Jaillet, **Compression, Compressed Sensing and Compressed Prediction of Urban Traffic Data**, Singapore patent application no. 10201406502P.

Contents

Acknowledgements	i
Abstract	iii
Author's publications	v
List of contents	ix
List of figures	xiii
List of tables	xvii
List of abbreviations	xix
1 Introduction	1
1.1 Motivation	1
1.2 Related work	4
1.2.1 Traffic data compression	4
1.2.2 Traffic estimation (missing data imputation)	6
1.2.3 Traffic prediction	8
1.3 Contributions and outline of the thesis	14

2	Near loss-less compression of traffic data	19
2.1	Introduction	19
2.2	Data set and performance measures	20
2.2.1	Data collection and quality	21
2.2.2	Data set for experiments	23
2.2.3	Performance measures	23
2.3	Low-dimensional models	25
2.3.1	Singular value decomposition	25
2.3.2	2D Discrete cosine transform	27
2.3.3	Wavelets	28
2.3.4	Non-negative matrix factorization	29
2.3.5	Tensor method for traffic data compression	31
2.3.6	Performance comparison	31
2.4	Near-lossless data compression	35
2.4.1	Encoder	35
2.4.2	Decoder	37
2.5	Discussion	38
3	Missing data estimation for road networks	43
3.1	Introduction	43
3.2	Traffic data set and performance measures	44
3.2.1	Data set	44
3.2.2	Performance measures	45

3.3	Missing data estimation	49
3.3.1	Least squares method (LS)	49
3.3.2	Variational bayesian principal component analysis (VBPCA) .	50
3.3.3	Fixed point continuation with approximate singular value de- composition (FPCA)	51
3.3.4	Tensor decomposition	53
3.3.5	Mean substitution	54
3.3.6	Latent factors	60
3.4	Discussion	61
4	Spatial and temporal patterns in large-scale prediction	71
4.1	Introduction	71
4.2	Large-scale prediction	72
4.2.1	Traffic prediction	72
4.2.2	Training and test data for supervised learning	74
4.3	Traffic prediction algorithms	74
4.3.1	Support vector regression	75
4.3.2	Artificial neural networks	76
4.3.3	Exponential smoothing	77
4.4	Data set and performance measures	78
4.5	Comparison of different prediction algorithms	80
4.6	Spatial and temporal patterns in prediction performance	85
4.6.1	Analysis of spatial prediction patterns	87
4.6.2	Analysis of temporal prediction patterns	89
4.7	Discussion	92

5	Conclusion	103
5.1	Summary	103
5.2	Conclusion and future directions	105
5.2.1	Low-dimensional traffic models	105
5.2.2	Combining data-driven prediction with microscopic simulation	106
5.2.3	Models for non-nominal conditions (incidents / extreme weather conditions)	107
5.2.4	Connected vehicles	108
	Bibliography	109

List of Figures

2.1	Test network consisting of 18,101 road segments from the country wide road network of Singapore.	21
2.2	Missing Data percentages: The study was conducted for South East part of Singapore. The test network consisted of more than 15000 road segments (data: August, 2011).	22
2.3	Actual and reconstructed speed profiles of a subset of road segments from $\mathbf{A}^T$ and $\hat{\mathbf{A}}^T$, respectively. The colorbars represent the colors corresponding to different speed values. A row in the image represents the speed values (actual $\mathbf{a}_i^T$ and reconstructed $\hat{\mathbf{a}}_i^T$) of an individual link for 1 st and 2 nd August, 2011.	26
2.4	Reconstruction efficiency of different wavelet types.	29
2.5	Construction of a 3-way tensor for traffic data sets.	30
2.6	Compression efficiency of HO-SVD in terms of element ratio and relative error.	32
2.7	Comparison of the reconstruction efficiency of different low-dimensional models.	33
2.8	Reconstructed speed profile of a typical road segment using different low-dimensional models. The units for speed values are km/h. The speed data in this figure is taken from 1 st and 2 nd August, 2011. . . .	34

2.9	Schematic block diagram of the near-lossless compression of traffic data obtained from large road networks. $Q(\cdot)$ refers to the quantization scheme used for residuals (see (2.16)).	35
2.10	Performance of near-lossless compression methods based on different low-dimensional models (SVD/DCT/wavelets/NMF).	40
3.1	Incomplete and reconstructed speed profiles (from incomplete data) of a subset of road segments from $\mathbf{D}^T$ and $\hat{\mathbf{A}}^T$, respectively. The colorbars represent the colors corresponding to different speed values. The maroon (brownish-red) slots in the top image represent the missing values. A row in the image represents the speed values (incomplete $\mathbf{d}_i^T$ and reconstructed $\hat{\mathbf{a}}_i^T$) of an individual link for 3 rd and 4 th August, 2011.	46
3.2	Actual and reconstructed speed profiles (from incomplete data) of a subset of road segments from $\mathbf{A}^T$ and $\hat{\mathbf{A}}^T$, respectively. The colorbars represent the colors corresponding to different speed values. A row in the image represents the speed values (actual $\mathbf{a}_i^T$ and reconstructed $\hat{\mathbf{a}}_i^T$) of an individual link for 3 rd and 4 th August, 2011.	47
3.3	Network profile tensor and the corresponding unfolded matrix form $\mathbf{U}$. The matrices $\{\mathbf{A}\}_{i=1}^q$ each contain one day of speed data from the test network.	52
3.4	Impact of restricting the maximum number of factors on the imputation accuracy of VBPCA. The WRE is calculated for the case when 90% of speed data is missing.	54
3.5	Weighted relative error by considering different number of latent factors (rank) for different percentages of missing data. The test network is composed of expressways (CATA).	55
3.6	Weighted relative error by considering different number of latent factors (rank) for different percentages of missing data. The test network is composed of major arterial roads (CATB).	56

3.7	Weighted relative error by considering different number of latent factors (rank) for different percentages of missing data. The test network is composed of minor arterial roads (CATC).	57
3.8	Weighted relative error by considering different number of latent factors (rank) for different percentages of missing data. The test network is composed of slip roads.	58
3.9	Weighted relative error by considering different number of latent factors (rank) for different percentages of missing data. The test network is composed of access roads.	59
3.10	Weighted relative error of the proposed algorithms for different percentages of missing data and various road networks. Mean sub. refers to the substitution performed by applying (3.17).	62
3.11	Weighted relative error of the proposed algorithms during different days of the week for various road networks. The reconstruction error is for the case when 50% of data is missing.	63
3.12	Weighted relative error of the proposed algorithms during different days of the week for various road networks. The reconstruction error is for the case when 70% of data is missing.	64
3.13	Weighted relative error of the proposed algorithms during different days of the week for various road networks. The reconstruction error is for the case when 90% of data is missing.	65
4.1	The map of region for large scale prediction.	77
4.2	Different road categories in the network.	78
4.3	Performance comparison of prediction algorithms for different prediction horizons.	81
4.4	Error distribution of algorithms for different prediction horizons. . . .	82

4.5	Road category wise performance comparison.	83
4.6	Standard deviation of error within each road category.	84
4.7	An example of variations in prediction error across the network. . . .	86
4.8	Error distribution for different clusters.	94
4.9	Properties of spatial clusters. In Fig. 4.9d, blue links (s_b, s_d, s_g) and red links $(s_a, s_c, s_e, s_f, s_h)$ correspond to spatial clusters ω_2 and ω_3 respectively.	95
4.10	Road segments belonging to different spatial clusters.	96
4.11	Daily performance patterns for road segment s_1 (consistent link). Hexagons represent the clusters and the entry within each hexagon denotes the number of days in that cluster. Color bar represents cluster similarity in terms of Euclidean distance.	101
4.12	Daily performance patterns for road segment s_2 (inconsistent link). Hexagons represent the clusters and the entry within each hexagon denotes the number of days in that cluster. Color bar represents cluster similarity in terms of Euclidean distance.	102
5.1	Concept map showing the effect of external factors (weather, incidents) on network-wide traffic conditions. Incorporating information about these factors can potentially improve the effectiveness of low-dimensional models.	107

List of Tables

2.1	Categories of road segments. The test network comprises of 18,101 road segments.	20
2.2	Compression ratios of different methods for $\delta = 0$. HC refers to the case, in which Huffman coding is directly applied to perform lossless compression.	37
2.3	Near-lossless compression performance of low-dimensional models for different tolerance levels $\delta = \{1 \text{ km/h}, \dots, 15 \text{ km/h}\}$	38
2.4	PSNR for different road categories.	39
2.5	PSNR for different methods during weekdays and weekends for the month of August, 2011.	39
3.1	Size of different test networks. Each test network is composed of roads from a specific category. These roads belong to the city-state network of Singapore. Primary and local access roads are referred as “Other Roads” in the table.	44
3.2	Variance of the imputed speed data for different road types. The units for variance are km^2/hr^2 . The values in the brackets represent the variance of imputed speed data w.r.t the actual speed data. . . .	66
3.3	Bias of the imputed speed data for different road types. The units for bias are km/hr	67

4.1	Categories of road segments.	79
4.2	Spatial cluster validation	92
4.3	Cluster centers for temporal clusters.	93
4.4	Distribution of links in temporal clusters.	93
4.5	Performance for different spatial clusters. The MAPE values are in percentage	93

List of abbreviations

ANN	ARTIFICIAL NEURAL NETWORKS
BPCA	BAYESIAN PRINCIPAL COMPONENT ANALYSIS
CP	CANONICAL POLYADIC DECOMPOSITION
CP-WOPT	CANONICAL POLYADIC WEIGHTED OPTIMIZATION
C-WSVM	CHAOS-WAVELET ANALYSIS-SUPPORT VECTOR MACHINE MODEL
CR	COMPRESSION RATIO
DCT	DISCRETE COSINE TRANSFORM
ER	ELEMENT RATIO
FPCA	FIXED POINT CONTINUATION WITH APPROXIMATE SINGULAR VALUE DECOMPOSITION
GAA	GREATER ATHENS AREA
HO-SVD	HIGHER ORDER SINGULAR VECTOR DECOMPOSITION
HC	HUFFMAN CODING
ICA	INDEPENDENT COMPONENT ANALYSIS
ITS	INTELLIGENT TRANSPORTATION SYSTEMS
LTA	LAND TRANSPORTATION AUTHORITY
LS	LEAST SQUARES
LS-SVM	LEAST SQUARES SUPPORT VECTOR MACHINES
MLN	MARKOV LOGIC NETWORKS
MAPPCA	MAXIMUM A POSTERIORI PCA
MAE	MAXIMUM ABSOLUTE ERROR
MAPE	MEAN ABSOLUTE PERCENTAGE ERROR
MSE	MEAN SQUARE ERROR
MS	MEAN SUBSTITUTION

NMF	NON-NEGATIVE MATRIX FACTORIZATION
PSNR	PEAK SIGNAL-TO-NOISE RATIO
PeMS	PERFORMANCE MEASUREMENT SYSTEM
PCA	PRINCIPAL COMPONENT ANALYSIS
PPCA	PROBABILISTIC PRINCIPAL COMPONENT ANALYSIS
POPFNN-TVR	PSEUDO OUTER-PRODUCT FUZZY NEURAL NETWORK USING THE TRUTH-VALUE-RESTRICTION
RBF	RADIAL BASIS FUNCTION
RMSE	ROOT MEAN SQUARE ERROR
SARIMA	SEASONAL AUTOREGRESSIVE INTEGRATED MOVING AVERAGE
SSVRCIA	SEASONAL SUPPORT VECTOR REGRESSION MODEL WITH CHAOTIC IMMUNE ALGORITHM
SSVRCSA	SEASONAL SUPPORT VECTOR REGRESSION MODEL WITH CHAOTIC SIMULATED ANNEALING ALGORITHM
SOM	SELF ORGANIZING MAPS
SVD	SINGULAR VALUE DECOMPOSITION
SVR	SUPPORT VECTOR REGRESSION
TTI	TRAVEL TIME INDEX
VBPCA	VARIATIONAL BAYESIAN PRINCIPAL COMPONENT ANALYSIS
WRE	WEIGHTED RELATIVE ERROR

Chapter 1

Introduction

1.1 Motivation

Transportation and mobility issues lie at the core of challenges faced by expanding cities. Intelligent Transportation Systems (ITS) have the potential to address these challenges. With the development of inexpensive sensor technologies, traffic management systems can now collect information about traffic states (flow, speed), abnormal events (accidents, road works) and the environment (weather) in real time. This information can potentially improve the mobility conditions for commuters and drivers [1]. In this thesis, we will focus on three key issues faced by ITS, which are: (i) Compressing large volume of sensor data (ii) estimating complete network-wide traffic information from incomplete traffic data (iii) handling uncertainty in predicted traffic conditions by data-driven algorithms. To this end, we develop methods that can deal with large and heterogenous road networks.

Let us start with the problem of management of traffic data. A typical intelligent transportation system would require data with high spatial and temporal resolution to model the traffic behavior of the network. However, the size of data coupled with scale of the urban networks presents its own set of challenges. Road networks of large metropolitan areas usually contain thousands of road segments. Secondly,

temporal resolution of 5 minutes is typically used for data acquisition. Naturally, the large data size puts serious constraints on the traffic management systems. It also limits the scalability of many data driven applications in the case of large networks. To address the issue of data management, we propose various subspace methods to obtain low-dimensional models for large and diverse road networks. We then utilize these low-dimensional models to store only the dominant traffic trends across the network, instead of actual data. In this way, we can substantially reduce the storage requirements for traffic data sets. Although, these subspace methods can provide high compression efficiency, they suffer from a major disadvantage. They can only perform lossy compression. For instance, consider $x(t)$ to be the traffic speed on a certain road at time t . Let us denote the reconstructed speed value, as a result of lossy compression, by $\hat{x}(t)$. Lossy compression schemes provide no guarantee that the absolute reconstruction error $|x(t) - \hat{x}(t)|$ will remain below a certain threshold δ . To limit the maximum reconstruction error below a certain threshold δ , we propose a two-step compression algorithm. We first perform lossy compression by developing low-dimensional model for the network. Then we encode the residuals by applying Huffman coding and keeping quantization error of residuals below the specified tolerance limit. The resulting compression algorithm guarantees that the maximum loss of information at any time instance and for any road segment will remain below a certain tolerance level.

So far, we have considered a scenario where the sensors can provide complete traffic information across the network during every sampling interval. In practical systems, this may not be the case. The sensors also have their own limitations. For instance, loop detectors suffer due to sparse spatial coverage, whereas the information provided by GPS probes such as taxis has highly irregular spatial and temporal resolution. These issues make the problem of missing data unavoidable for these systems. For ITS, this is a critical issue. Data-driven methods rely heavily upon historical and real-time data for calibration and validation purposes. Incomplete information about the past and current traffic states can seriously hamper their efficiency. In this study, we consider the problem of missing data in large and diverse

road networks. We propose various matrix and tensor based methods to estimate these missing values by extracting common traffic patterns in large road networks.

The issues of missing data estimation and data compression are related to the data management aspects of ITS. Let us now look at the application side of intelligent transportation systems. Nowadays, ITS are increasingly incorporating prediction engines to improve their operational performance [2]. The ability to accurately predict traffic speed in a large and heterogeneous road network has many useful applications related to traffic management and operations. Consequently, traffic prediction is a commonly studied problem in transportation research [2]. In principle, data-driven methods, such as support vector regression (SVR), can predict traffic with high accuracy because traffic tends to exhibit regular patterns over time. However, the predicted traffic conditions are prone to uncertainty. Prediction accuracy might vary for different test conditions. The actual prediction error can only be calculated once field data becomes available. For instance, consider 15 minutes prediction horizon. In this case, we can only calculate prediction error after 15 minutes. Consequently, the applications which use prediction data remain vulnerable to variations in prediction error. Nonetheless, insights into the spatio-temporal variations in prediction performance can help reduce this uncertainty. We propose unsupervised learning methods, such as k-means clustering, principal component analysis, and self-organizing maps, to mine spatiotemporal performance trends at the network level and for individual links. Such tools can prove highly useful for predictive ITS applications.

We aim to develop methods for traffic prediction, estimation and compression which can deal with large and diverse road networks. Moreover, for practical considerations the problems of estimation, compression and prediction are inter-related. A system performing traffic prediction inevitably will have to deal with the problem of missing data (estimation) and large data sets (compression).

1.2 Related work

In this section we briefly review the literature related to traffic compression, estimation (missing data imputation) and traffic prediction. The existing studies mostly deal with limited scenarios such as a few intersections or highway portions. Practical networks are much more diverse. Consequently, the performance of proposed methods has not been analyzed for large and diverse road networks. Dealing with large-scale networks brings out its own set of challenges and the method developed for specific scenarios are typically not suited to solve those problems. This is a major shortcoming in previous studies. Let us start with the area of traffic data compression.

1.2.1 Traffic data compression

Road networks of large metropolitan areas usually contain thousands of road segments. A temporal resolution of 5 minutes is typically used for data acquisition. This provides ITS with detailed information about the state of the network and helps to improve their performance [1]. The large size of data also presents its own set of constraints. The data size puts serious constraints on the data management systems [3]. It also limits the scalability of many data driven applications in case of large networks [4]. These issues can limit the usefulness of ITS for large networks.

Previously this problem has been studied in context of a few intersections [3, 5] and Origin Destination (OD) pairs [6]. Tsekeris et al. applied principal component analysis (PCA) to model the variability in traffic flow data. For this purpose, they collected one month of traffic flow information from 140 locations in Greater Athens Area (GAA). They used aggregation interval of 15 minutes. They found that traffic conditions in the test network can be efficiently represented by a small number of principal components (or eigenflows) [7]. Qu et al. applied PCA to compress data obtained from a small road network comprising of around 50 links and 17 intersections. The test network was located between the 2nd and 3rd ring road in the

east zone of Beijing's UTC_SCOOT control system. They analyzed the compression efficiency of PCA for traffic flow data [3]. However, they did not provide any comparison with other low-rank approximation techniques. Xing et al. considered robust PCA to extract dominant patterns in traffic flow data obtained from Anhui province in China. The data was collected from 120 entrance/exit toll stations with sampling interval of 15 minutes [8]. Djukic et al. analyzed a small freeway network comprising of 52 road segments for OD pair data compression [6]. Zhao et al. compared the performance of PCA and independent component analysis (ICA) for compressing traffic flow data [9]. They analyzed data obtained from Beijing urban street (around 10 detectors) and Washington freeway system (several detectors) and concluded that PCA slightly outperforms ICA. They also reported that the choice of sampling interval does not significantly affect the underlying distribution of data.

Xiao et al. compressed flow data obtained from individual links using 2D wavelet transform [5]. They did not provide any numerical analysis regarding the compression efficiency of the method. For analysis, they collected volume, speed and occupancy information for 65 sections in Beijing for a duration of one week. Shi et al. also performed compression of data obtained from a single loop detector located on an expressway site in Shanghai using wavelets [10].

Hofleitner et al. proposed non-negative matrix factorization (NMF) to obtain a low-dimensional representation of a large network consisting of around 2,626 road segments [11]. They used 184 days of data (01-05-2010 to 31-10-2010) from the mobile millennium platform. They applied NMF to extract spatial and global patterns in the network. However, no analysis in terms of reconstruction efficiency of NMF was provided.

In a recent study, Li et al. applied Granger causality to determine dependencies/relationships between traffic conditions on a test road network in California, USA. To this end, they considered data from performance measurement system (PeMS). They collected data from 1000 sites [12].

In summary, we can state that previous studies related to traffic data compression

have focused on data collected from either individual roads or small networks [5, 3, 10, 13]. These studies typically do not compare the reconstruction efficiency of various low-dimensional models for a given road network [11, 3, 5, 14]. Furthermore, the proposed methods do not provide any upper bound on the maximum loss of information.

We aim to analyze the compressibility of large and diverse networks. We also propose a two-step algorithm to perform near-lossless compression for traffic data sets.

1.2.2 Traffic estimation (missing data imputation)

Field traffic data is usually obtained from sensors such as loop detectors and GPS probes. Loop detectors are expensive to install. Hence, they only provide sparse spatial coverage. GPS probes are cheaper. They can be easily installed on taxis and public transportation. Consequently, GPS probes provide better spatial coverage. However, these probes offer another set of challenges [15]. The data obtained from GPS sensors often has highly erratic spatial and temporal resolution. This can result in data sets where a high percentage of data is missing. Different studies in this regard have reported missing data percentages from 4% to more than 90% [16, 17, 18].

The methods proposed to solve the problem of missing data can be broadly divided into two categories: function estimation and matrix/tensor completion. In the first case, it is typically assumed that the problem of missing data is localized to certain links and time intervals. In this way, the historical data can be used to obtain the relationship function between the target road and its neighbors or past states of that road. For instance, Chen et al. [19] developed relationship models between neighboring loop detectors using historical data. This relationship function was then used to impute missing values for faulty detectors. Zhong et al. [20] trained neural networks and used temporal features to estimate missing values. Zhang et al. [21] also used a similar approach and applied least squares support vector machines to

estimate missing values. The function estimation techniques require complete historical data to learn the relationship models. Hence, these methods will not work if historical data has missing values. In practical scenarios, uncorrupted historical data may not be available. On the other hand, matrix and tensor completion methods do not require training data to perform imputation [22]. Consequently, these methods have garnered considerable interest in the field of transportation studies [22, 23, 24, 25, 26, 27, 16].

Traffic states across neighboring roads tend to be strongly correlated [28]. These relationships imply that road networks can be represented by low-dimensional models. Matrix and tensor completion methods utilize these patterns to estimate the missing values by obtaining a suitable low-rank approximation of the incomplete tensor/matrix. However, previous studies involving matrix/tensor completion methods for traffic data sets mostly focused on data obtained from a few roads or intersections. For instance, Li Qu et al. [22, 23] used Bayesian principal component analysis (BPCA) to perform imputation for traffic flow data. They analyzed a small network consisting of around 50 road segments. Li Li et al. [24] used data from four detectors for analysis. Chang et al. [25] compared the performance of various matrix completion methods on a test network of around 50 links. Tan et al. [16] performed missing data imputation by tensor decomposition methods. For analysis, they considered four road segments and represented their data obtained from each road as a 3-way tensor.

Traffic conditions across city-scale networks also tend to have certain common global patterns [29, 30, 11, 31]. Some studies [26] have considered the problem of missing data in large networks, albeit in a limited manner. These studies did not analyze the performance of imputation algorithms for different road types (expressways, arterial roads, access roads) and during different days of the week. Furthermore, they did not analyze the bias and variance of the imputed traffic data.

In summary, function estimation methods for imputation have limited application for large networks due to their dependency on uncorrupted historical data [19, 20,

21, 32]. Secondly, the previous studies which applied matrix and tensor completion methods mostly considered data from a single link or a few intersections [16, 22, 23, 24, 25]. These studies typically do not analyze the performance of imputation methods for different road types and during different days of the week [26, 22, 23]. Furthermore, analysis in terms of variance, bias, and the impact of the rank of the estimated low-dimensional model on the imputation performance also needs to be considered.

It is important to point out that the problem of traffic estimation (missing data imputation) is slightly different from that of prediction. In case of prediction, it is assumed that historical data (of the road and neighbors) is available. In case of estimation, the historic data might also suffer from the same problem [22, 23]. The spatial and temporal distribution of missing data points is usually highly erratic [33]. This limits the utility of imputation methods, which require complete historical or current information from neighbors.

In this thesis, we address the aforementioned limitations in previous studies by performing missing data imputation for large road networks comprising of expressways, arterial roads, access roads as well as slip roads. We propose various matrix and tensor based methods that can extract global traffic patterns from incomplete data.

1.2.3 Traffic prediction

Accurate information about near-future traffic conditions can potentially improve the performance of many ITS applications including (not limited to) dynamic management of toll costs, improving fuel efficiency, improving operational efficiency of freight operations, driver experience and mobility on demand systems amongst other applications [34, 35]. Consequently, short-term traffic forecasting is one of the most commonly studied topics in transportation studies [2, 36]. Recently machine learning methods such as artificial neural networks (ANN) and support vector regression (SVR) have found considerable interest in the area of traffic forecasting. These algorithms tend to provide better prediction results than competing methods

[37, 38, 4, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49]. In the following, we briefly discuss some of these works.

Wu et al. applied SVR to perform travel time prediction on three origin-destination pairs. The data was obtained from Taiwan area national freeway bureau (TANFB). They showed that SVR provides better accuracy than other standard methods [38]. Wang et al. developed wavelets based kernel function to perform traffic speed prediction. They referred to their model as chaos-wavelet analysis-support vector machine model (C-WSVM). They obtained data from dual loop detectors installed on a 13 km stretch (with detectors separated by a distance of around 330 m) of Yanan freeway in Shanghai, China [50]. Zhang et al. proposed least squares support vector machines (LS-SVM) to predict travel time index (TTI) for data obtained from performance measurement system (PeMS) in California, USA [44]. Gopi et al. applied Bayesian SVR to perform speed prediction on a test network consisting of around 50 road segments [51]. Castro-Neto et al. proposed online-SVR for predicting traffic flow. Their proposed method has the advantage of iterative learning, rather than typical batch learning configuration. For analysis, they considered data from 7 freeway locations in California (PeMS dataset) [52]. Su et al. also considered incremental SVR to perform traffic flow prediction. For analysis, they used aggregated data obtained from I-880 freeway in California [53]. Wei-Chiang Hong proposed two variants of SVR for traffic flow forecasting. These variants were termed as seasonal support vector regression model with chaotic simulated annealing algorithm (SSVRCSA) [54] and seasonal support vector regression model with chaotic immune algorithm (SSVR-CIA) [55]. The analysis was conducted on data collected from 3 motor-way sites in Panchiao City, Taiwan.

Quek et al. proposed a method termed as pseudo outer-product fuzzy neural network using the truth-value-restriction (POPFNN-TVR) to perform traffic flow prediction. They collected data on two locations in Singapore. The data was collected in conjunction with land transportation authority (LTA) of Singapore [49]. Park et al. proposed an ensemble of neural networks to perform speed prediction using PeMS data set. For analysis, they collected data from 52 sensors [48]. Eleni Vla-

hogianni applied neural networks to perform traffic flow prediction for signalized arterial roads. Their test network consisted of 140 arterial roads in Athens, Greece [56]. Li et al. also applied neural networks to predict travel time. This study was conducted on a 36.1 km stretch of National Freeway No. 1 in Taiwan. To this end, they collected data from 22 dual loop detectors. They also incorporated additional information about weather and accidents in their prediction model [57]. Dunne et al. developed a multivariate ANN based prediction structure with three adaptive learning strategies [58]. This method takes traffic speed and flow information to predict future speed and flow values on a road segment. For analysis, they collected data on two motor-way sites in United Kingdom.

Other traffic prediction methods include Markov chains [59], Negative Binomial Additive Models [60], stochastic differential equations [61], Lasso based regression [62] and Kalman filtering technique [63, 64].

Xie et al. combined Kalman filter and wavelet transform to predict traffic flow along four locations on interstate 5 (I-5), Interstate 90 (I-90), and Interstate 405 (I-405) in Seattle, USA [63]. In a recent study, Lippi et al. compared the performance of various methods including Kalman filter with Seasonal Autoregressive Integrated Moving Average (SARIMA) [65], SVR and ANN [66]. For comparison, they considered PeMS data set and trained prediction models using traffic flow information from 16 stations. They reported that SVR can provide comparable performance to SARIMA and does not suffer from high computation time.

We would also like to briefly discuss an alternate approach to perform multi-horizon prediction termed as unified prediction. These models are typically developed to predict multiple traffic parameters simultaneously. Most of the previously mentioned studies focus on the development of separate prediction models for each road segment and prediction horizon. Traffic parameters such as speed, flow and travel time tend to be influenced by traffic conditions on the neighboring roads [67, 68]. These relationships are sometimes incorporated in forecasting models to improve the prediction performance [2]. Developing unified models for network-wide multi-

horizon prediction can potentially lead to improved prediction performance. Some recent studies [67, 28, 42] have explored the problem of unified models for traffic forecasting. The predictors proposed in [28] and [42] are capable of performing multi-horizon prediction using a single model. However, these techniques still require separate models for individual links. Lippi et. al. proposed Markov logic networks (MLN) to perform simultaneous prediction of traffic states on multiple roads and for multiple prediction horizons [67]. Dauwels et al. proposed various partial least squares methods to simultaneously forecast traffic conditions for multiple road segments and prediction-horizons [69]. The methods proposed in [69] and [28] have been shown to perform prediction for moderately size networks (consisting of 250 – 500 road segments) in real time. However, the key challenge faced by such predictions models [69, 28] is that the computational complexity increases quadratically i.e. $O(p^2)$ with the network size p . Similarly the memory requirements also increase by $O(p^2)$. Hence, they are not suitable for large-scale networks. Consequently, in this study, we will develop individual models for each road segment and prediction horizon.

In summary, traffic prediction has garnered considerable interest in the area of transportation systems. These methods have been applied to different traffic parameters such as traffic speed, flow, and travel time. The algorithms try to extract relationships between given traffic features and future trends. Typical traffic features include past traffic trends of the road [38, 49] and data from neighboring roads [43, 28]. Different studies consider sampling intervals ranging from 2 minutes [47], 5 minutes [48, 28, 49] to 15 minutes [40, 43]. Naturally, large sampling interval yields lower prediction error [28] as it tends to suppress more volatile components in the time series [70]. Nonetheless, the scope of these studies remains limited to special cases. For instance, the studies [38, 39, 48, 40, 66] consider highways or motorways only. Some other studies [42, 49] do consider more general scenarios, albeit for some small regions. Consequently, the performance patterns in large heterogeneous networks have not been investigated. Min et al. considered a moderate sized road network consisting of about 500 road segments [28]. However, they developed a

custom model for the test area, which is not available. This limits any meaningful comparison with their proposed method. They analyzed the performance by taking into account different road categories.

In this thesis, our focus is not on the problem of traffic prediction, rather we would like to model the uncertainty that is inherently attached to any predictive modeling application. For scenarios involving a few roads or intersections, such uncertainty can be minimized by carefully tuning the model parameters, incorporating additional features and developing models for specific cases. Van Hinsbergen et al. adopted this approach to train a committee of neural networks to perform travel time prediction [39]. Such an approach is clearly not scalable for a road network consisting of thousands of road segments. Let us consider a test case of a road network consisting of $p = 5000$ links and the goal is to perform multi-horizon prediction up till 60 minutes in real-time. Furthermore, let us consider that the sampling interval Δt is 5 minutes. Hence, we would need to train prediction models for $K = 12$ future horizons. To model such a system, we would require $Kp = 60,000$ predictors. Naturally, fine-tuning model parameters, performing feature selection for each road, training ensemble models or incorporating complex algorithms would not work in such a scenario. Traffic prediction for large networks requires modular and easily scalable algorithms with minimum number of parameters and generic configurations that can work for roads with different speed limits, capacities and covering different areas (urban, rural, downtown). To this end, we consider commonly employed configurations of SVR and ANN. This still leaves us with the problem of modeling uncertainty in prediction. Forecasting methods may provide different prediction performance for different roads. Moreover, prediction accuracy may also depend upon the time and the day (weekdays/weekends). These spatial and temporal performance trends contain useful information about predictability of networks. ITS applications can utilize this information, to provide more robust and accurate solutions. Traffic prediction studies usually analyze performance by applying point estimation methods such as Mean Absolute Percentage Error (MAPE) [37, 28, 38, 4, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49].

These point estimation measures work well for overall performance comparison. However, they fail to provide any insight into underlying spatial and temporal prediction patterns. Consider a large road network. Intuitively, traffic on certain roads will be easier to predict than others. An average measure in this regard will not provide detailed information. As the previous studies usually focused on small regions or highways, so these performance trends have not been studied in detail [37, 38, 4, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49]. Our goal is to model and quantify uncertainty in traffic prediction for large heterogeneous road networks. We propose data mining approach to deal with this issue. To extract spatial and temporal prediction patterns for large networks, we propose unsupervised learning methods such as k-means clustering and principal component analysis (PCA). For link level temporal prediction patterns, we use self organizing maps (SOM). We apply these data mining algorithms to extract performance trends in SVR prediction data.

Although the focus of this research is on macroscopic traffic prediction, it is important to mention the progress in the area of microscopic simulation/prediction [71]. Some common micro-simulation traffic softwares include AIMSUN [72], MITISM [73], PARAMICS [74], OLSIM [75], SUMO [76] and VISSIM [77, 78]. The macroscopic prediction is suitable for managing traffic at the network level and providing guidance to drivers. On the other hand, microscopic prediction is highly useful in modeling additional factors such as driver behavior, pedestrian movement, fuel consumption of vehicles and emissions amongst other applications. In driver behavior modeling, the simulators incorporate car following and lane changing aspects associated with drivers. One promising application for microscopic prediction is in the area of V2X applications. A key aspect in this regard is the integrated modeling of vehicular motion and communication networks. Micro-simulation softwares can effectively model the motion of individual vehicles and can incorporate driver behavior as well. Integrating these simulators with communication simulations such as NS3 in the form of close loop system can allow for real-time data exchange between the simulators. The behavior of individual vehicles would naturally impact the channel properties of communication network. This would prove highly invaluable in mod-

eling driver response and the impact on traffic on the road in the presence of ad-hoc or cellular communication networks.

1.3 Contributions and outline of the thesis

The thesis is structured as follows:

- In chapter 2, we consider the problem data compression for large-scale networks. The related studies have mostly focused on the problem of lossy compression of traffic data-sets. Furthermore, these works only considered small test networks. We also present a near-lossless compression algorithm for traffic data sets. The proposed provides a guarantee that the maximum loss of information after decompression will remain below a pre-specified threshold. The results also show that large-scale road networks can also be modeled as low-rank matrices/tensors.

Publications:

- M.T. Asif, S. Kannan, N. Mitrovic, J. Dauwels, and P. Jaillet, **Near-Lossless Compression for Large Traffic Networks**, IEEE Transactions on Intelligent Transportation Systems, vol. 16, no. 4 (2015): 1817-1826.
- M. T. Asif, K. Srinivasan, J. Dauwels and P. Jaillet, **Data compression techniques for urban traffic data**, IEEE Symposium on Computational Intelligence in Vehicles and Transportation Systems (CIVTS), 2013, pp. 44-49. IEEE, 2013.
- In chapter 3, we discuss the problem of traffic estimation. The problem of missing data imputation (estimation) in transportation studies is mostly studied in the context of a few intersections whereas, in practical scenarios, ITS have to deal with this issue in the context of city-scale road networks. To this end, we

propose methods that can infer dominant traffic patterns in road networks with diverse composition. We consider *2-way* (matrix) and *3-way* (tensor) structures for traffic data and propose different methods to infer low-dimensional models from these data structures from incomplete traffic information. We analyze the performance of these models for different road categories. We also study the effect of the choice of rank on the estimation accuracy of different low-dimensional models. Our main contributions are as follows:

In transportation systems, the problem of missing data is typically handled by applying variants of the un-constrained weighted least square approach [25]. We propose the nuclear norm minimization based approach to solve the problem of missing data in large-scale transportation systems. We apply fixed point continuation with approximate singular value decomposition (FPCA) [79] to obtain a low-rank representation for large-scale road networks in presence of missing data. The results show that its performance is less sensitive to daily variations in traffic data. Furthermore, the method also provides better or similar performance (in terms of weighted relative error (WRE), variance and bias) in comparison other algorithms for different road categories.

Probabilistic PCA based methods have been previously used to estimate missing traffic information from incomplete data sets [24, 22]. However, these studies only considered small networks and did not evaluate the performance of probabilistic methods for different road categories and during different days of the week. Furthermore, analysis in terms of induced bias and variance in the recovered speed data also needs to be considered. Moreover, BPCA formulations used in these studies [22] do not scale well for large-scale systems. In this study, we consider a variant of variational Bayesian principal component analysis (VBPCA)[80], which is suitable for large-scale road networks.

We compare the performance of above mentioned methods with baseline matrix and tensor decomposition methods such as weighted least squares and canonical polyadic (CP) decomposition. We analyze the performance of these

methods for different road categories and for days of the week. We analyze the variance and bias induced by these methods in the imputed speed data. Furthermore, we discuss the impact of rank selection on the performance of different methods.

Publications:

- M. T. Asif, N. Mitrovic, J. Dauwels, and P. Jaillet, **Matrix and tensor based methods for missing data estimation in large traffic networks**, *submitted to IEEE Transactions on Intelligent Transportation Systems, 2015*.
- M. T. Asif, N. Mitrovic, L. Garg, J. Dauwels, and P. Jaillet, **Low-dimensional models for missing data imputation in road networks**, 38th IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2013, pp. 3527-3531. IEEE, 2013.
- In chapter 4, we discuss the problem of large scale prediction for traffic networks. Unlike previous works, our focus is on modeling uncertainty when predicting traffic conditions for large and heterogenous road networks such as city-scale network of Singapore. As previous works show it is indeed possible to forecast future traffic conditions with high accuracy. However, as these works mostly focused on small test networks, they did not analyze the performance of such predictors for large-scale networks consisting of diverse set of roads. Naturally, fine-tuning a prediction algorithm to achieve high accuracy for each road segment in a city-scale network is an intractable problem. Furthermore, traffic conditions on certain roads/time periods might be inherently un-predictable. We address these questions by applying unsupervised clustering techniques to provide quantitative analysis on the spatial and temporal predictability of traffic conditions in large networks.

Publications:

- M. T. Asif, J. Dauwels, C. Y. Goh, A. Oran, E. Fathi, M. Xu, M. Dhanya,

- N. Mitrovic, and P. Jaillet, **Spatiotemporal patterns in large-scale traffic speed prediction**, IEEE Transactions on Intelligent Transportation Systems, vol. 15, no. 2 (2014): 794-804.
- M. T. Asif, J. Dauwels, C. Y. Goh, A. Oran, E. Fathi, M. Xu, M. M. Dhanya, N. Mitrovic and P. Jaillet, **Unsupervised Learning Based Performance Analysis of n-Support Vector Regression for Speed Prediction of a Large Road Network**, 15th IEEE Intelligent Transportation Systems Conference (ITSC), pp. 983-988. IEEE, 2012.
- We conclude the thesis by discussing potential extensions related to our current work in chapter 5.
 - In addition to above mentioned works, the author also contributed to the works published in [81, 82, 83, 84, 69, 85, 86, 87, 51, 82] and [88].

Chapter 2

Near loss-less compression of traffic data

2.1 Introduction

Advancements in sensor technologies have enabled intelligent transportation systems (ITS) to collect traffic information with high spatial and temporal resolution [1, 89, 11]. This has led to the development of data driven algorithms for many traffic related applications such as sensing, traffic prediction, estimation, and control [27, 90, 91, 28, 92, 81, 26, 93, 22, 14, 94]. However, the huge size of collected data poses another set of challenges. These challenges include storage and transmission of these large data sets in an efficient manner. Moreover, nowadays many ITS applications need to transfer information to remote mobile devices [95, 96]. These devices usually have limited storage capacity and may also have limited bandwidth. In such scenarios, efficient data compression can be considered as a major requirement for optimal system operation.

We propose to utilize spatial and temporal patterns for efficient compression of traffic data sets. Traffic parameters tend to exhibit strong spatial and temporal correlations [28, 30]. These spatial and temporal relationships can help to obtain

Table 2.1: Categories of road segments. The test network comprises of 18,101 road segments.

CATA	CATB	CATC	Slip Roads	Other
2,167	8,674	2,078	1,832	3,350

a low-dimensional representation for a given network. The low-dimensional models have been previously considered for applications related to feature selection [28, 97, 30], missing data imputation [22, 27] and estimation [11, 94]. Low-dimensional models provide compressed state for a given network. Hence, they are naturally suited for data compression.

Previous studies related to traffic data compression have mostly focused on data from a single road or small subnetworks [5, 3, 10, 13]. Furthermore, These studies related to low-dimensional models such as in [13, 98, 5, 11, 3, 99, 14, 100, 101] only consider lossy compression. In lossy compression, there is no bound on the maximum absolute error (MAE). In terms of traffic data sets, lossy compression does not provide any bound on the loss of information during individual time instances.

In this study, we perform near-lossless compression of speed data obtained from a large test network comprising of around 18,000 road segments in Singapore. The network consists of a diverse set of roads such as expressways, region around the Changi Airport, industrial, residential, arterial roads etc. We consider different subspace methods such as singular value decomposition (SVD), discrete cosine transform (DCT), wavelet transform and NMF for low-dimensional representation. We compare their reconstruction efficiency for a large and diverse road network. We analyze the performance of the proposed algorithm for different road categories as well as during weekdays and weekends.

2.2 Data set and performance measures

In this section, we explain the data set considered in this study. We also provide different performance measures that we will use for analysis in later sections.

Figure 2.1: Test network consisting of 18,101 road segments from the country wide road network of Singapore.

2.2.1 Data collection and quality

Let us also briefly discuss the data collection process undertaken by LTA, Singapore and practical issues such as missing data. LTA utilizes a range of sensors to estimate traffic conditions on the roads. These sources include junction eyes (J-Eyes), webcams, loop detectors and GPS probes [102, 103]. Other sources include gantries installed in the downtown area of Singapore, which are used to collect tolls from cars entering the downtown area. They have divided the road network of the country into around 50,000 road segments thus providing high spatial resolution. The length of each road segment varies from around 50 – 200 meters. The speed data used for the experiments in this thesis was provided by LTA by fusing traffic information from various on-ground sensors. To improve the quality and resolution of data, the aggregation was performed off-line. The main sources of data were GPS trackers on taxis and loop detectors. To pick suitable roads, we validated the quality of speed data on each road by amount of data missing (in the data set obtained from LTA) for that particular road. For illustration, let us consider the example of south east part of Singapore (see Fig. 2.2). By picking roads with less than 3% missing data,

Figure 2.2: Missing Data percentages: The study was conducted for South East part of Singapore. The test network consisted of more than 15000 road segments (data: August, 2011).

we can be reasonably certain that adequate amount of probe sensors were available on those roads for most of the time. Hence, for all the experiments we only picked those roads for which the amount of missing data was low.

2.2.2 Data set for experiments

We represent the road network shown in Fig. 2.1 by a directed graph $G = (N, E)$, where the set E consists of p road segments such that $E = \{s_i\}_{i=1}^p$. Here s_i represents a road segment. The set N contains the list of nodes in the graph. For this study, we consider average speed data. Let $z(s_i, t_j)$ represent the average speed on the road segment s_i during the time interval $(t_j - \Delta t, t_j)$ where $\Delta t = 5$ minutes. For each road s_i , we create speed profile $\mathbf{a}_i \in \mathbb{R}^n$ where $\mathbf{a}_i = [z(s_i, t_1) \dots z(s_i, t_n)]^T$. These speed profiles are stacked together to obtain the network profile $\mathbf{A} \in \mathbb{R}^{n \times p}$ where $\mathbf{A} = [\mathbf{a}_1 \dots \mathbf{a}_p]$. Fig. 2.3 shows the example of a low-dimensional representation $\hat{\mathbf{A}}$ of the network profile matrix $\mathbf{A}$. For the analysis, we consider continuous speed data from the month of August, 2011.

The test network is composed of a diverse set of road segments belonging to roads from different categories as well as different regions: residential, industrial, downtown, around the airport etc. Table 2.1 shows the number of roads belonging to each category as defined by LTA. Expressways are assigned to category A (CATA), where as major and minor arterial roads belong to CATB and CATC, respectively. The primary access and local access roads are referred as “Other” in Table 2.1.

2.2.3 Performance measures

Let us now consider different performance measures. We calculate the relative error between the actual network profile $\mathbf{A}$ and the estimated network profile $\hat{\mathbf{A}}$ as:

$$\text{Relative Error} = \frac{\|\mathbf{A} - \hat{\mathbf{A}}\|_F}{\|\mathbf{A}\|_F}, \quad (2.1)$$

where $\|\mathbf{A}\|_F$ is the Frobenius norm of matrix $\mathbf{A}$, which is calculated as:

$$\|\mathbf{A}\|_F = \left(\sum_{i=1}^n \sum_{j=1}^p a_{ij}^2 \right)^{1/2}. \quad (2.2)$$

The relative error provides a measure of loss of signal energy $\mathbf{A} - \hat{\mathbf{A}}$ as compared to the original network profile $\mathbf{A}$. We also calculate mean absolute percentage error (MAPE) as follows:

$$\text{MAPE} = \frac{1}{np} \sum_{i=1}^n \sum_{j=1}^p \left| \frac{a_{ij} - \hat{a}_{ij}}{a_{ij}} \right| \times 100\%, \quad (2.3)$$

where $\hat{a}_{ij}$ is the reconstructed speed value $\hat{z}(s_j, t_i)$ for link s_j at time t_i . We will use these measures to compare the reconstruction efficiency of different subspace methods.

We consider the following measures to analyze the efficiency of the proposed near-lossless compression algorithm. We calculate the maximum absolute error (MAE) for the reconstructed network profile $\hat{\mathbf{A}}$ as:

$$\text{MAE}(\mathbf{A}, \hat{\mathbf{A}}) = \max_{i,j} |a_{ij} - \hat{a}_{ij}|. \quad (2.4)$$

We also calculate the peak signal-to-noise ratio (PSNR) as:

$$\text{PSNR} = 20 \cdot \log_{10} \left(\frac{2^B - 1}{\sqrt{\text{MSE}}} \right), \quad (2.5)$$

where B is the resolution of the data set in bits. The mean square error (MSE) is calculated as:

$$\text{MSE} = \frac{1}{np} \|\mathbf{A} - \hat{\mathbf{A}}\|_F^2. \quad (2.6)$$

PSNR is commonly used in the domain of image processing to evaluate the performance of compression algorithms [104].

2.3 Low-dimensional models

In this section, we briefly discuss various subspace methods to obtain low-dimensional representations for large road networks. To this end, we will consider the following methods: singular value decomposition (SVD), 2D discrete cosine transform (DCT), 2D wavelet transform and non-negative matrix factorization (NMF). We compare their reconstruction efficiency in terms of the number of elements Θ required to reconstruct a particular low-dimensional representation $\hat{\mathbf{A}}$. We define the element ratio as:

$$\text{Element Ratio(ER)} = \frac{np}{\Theta}, \quad (2.7)$$

where np represents the total number of elements in the network profile matrix $\mathbf{A}$.

2.3.1 Singular value decomposition

Singular value decomposition based methods have found applications in many ITS applications including missing data imputation [22, 27] and estimation [14]. By applying SVD, network profile matrix $\mathbf{A}$ can be represented as $\mathbf{A} = \mathbf{U}\mathbf{S}\mathbf{V}^T$, where the columns of matrix $\mathbf{U} \in \mathbb{R}^{n \times n}$ and matrix $\mathbf{V} \in \mathbb{R}^{p \times p}$ are called the left singular vectors and the right singular vectors of $\mathbf{A}$ respectively. The matrix $\mathbf{S} \in \mathbb{R}^{n \times p}$ is a diagonal matrix containing $\min(n, p)$ singular values of the network profile matrix $\mathbf{A}$.

The left singular vectors can be obtained by performing eigenvalue decomposition of $\mathbf{A}\mathbf{A}^T$ such that $\mathbf{A}\mathbf{A}^T = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^T$. The matrix $\mathbf{\Lambda}$ contains eigenvalues of $\mathbf{A}\mathbf{A}^T$ where $\mathbf{U}^T\mathbf{U} = \mathbf{I}$. Similarly the right singular vectors can be obtained by performing eigenvalue decomposition of $\mathbf{A}^T\mathbf{A}$ such that $\mathbf{A}^T\mathbf{A} = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^T$. The singular values $\{\sigma_i\}_{i=1}^{\min(n,p)}$ are calculated as $\{\sigma_i = \sqrt{\lambda_i}\}_{i=1}^{\min(n,p)}$, where λ_i is the i^{th} diagonal entry of $\mathbf{\Lambda}$. Furthermore, we can obtain the rank- r ($r \leq \min(n, p)$) approximation of $\mathbf{A}$ as:

$$\hat{\mathbf{A}} = \sum_{i=1}^r \sigma_i \mathbf{u}_i \otimes \mathbf{v}_i, \quad (2.8)$$

Figure 2.3: Actual and reconstructed speed profiles of a subset of road segments from $\mathbf{A}^T$ and $\hat{\mathbf{A}}^T$, respectively. The colorbars represent the colors corresponding to different speed values. A row in the image represents the speed values (actual $\mathbf{a}_i^T$ and reconstructed $\hat{\mathbf{a}}_i^T$) of an individual link for 1st and 2nd August, 2011.

where $\mathbf{u}_i \otimes \mathbf{v}_i = \mathbf{u}_i \mathbf{v}_i^T$ and σ_i is the i^{th} diagonal entry of $\mathbf{S}$. The vectors $\mathbf{u}_i$ and $\mathbf{v}_i$ are the columns of matrices $\mathbf{U}$ and $\mathbf{V}$ respectively.

For traffic related applications, the matrix $\mathbf{A}^T \mathbf{A}$ can be interpreted as the covariance matrix for the road segments $\{s_i\}_i^p$ in the network G . Consequently, if the traffic patterns $\{\mathbf{a}_i\}_{i=1}^p$ between the road segments $\{s_i\}_{i=1}^p$ are highly correlated then the network profile $\mathbf{A}$ can be compressed with high efficiency. We perform lossy compression, by storing an appropriate low-rank approximation obtained from (2.8). To this end, we need to store r columns each from the matrices $\mathbf{U}$ and $\mathbf{V}$ and r elements from the matrix $\mathbf{S}$. Hence, the total number of stored elements will be $\Theta = (n + p + 1)r$.

2.3.2 2D Discrete cosine transform

In SVD based decomposition, we obtain the basis vectors $\{\mathbf{v}_i\}_{i=1}^p$ from the covariance matrix $\mathbf{A}^T \mathbf{A}$ of road segments in the network. Consequently, we need to store the matrices $\mathbf{U}$ and $\mathbf{V}$ along with the singular values $\{\sigma_i\}_{i=1}^r$ for decompression. In 2D DCT, we consider the cosine family as the basis set and use these basis functions to transform the network profile $\mathbf{A}$ into the so called frequency domain with matrix $\mathbf{Y} \in \mathbb{R}^{n \times p}$ containing the frequency coefficients. For the transformation, let us represent the speed $z(s_{j+1}, t_{i+1})$ for link s_{j+1} at time t_{i+1} as m_{ij} such that $m_{ij} = z(s_{j+1}, t_{i+1})$. We can then calculate the transformed coefficients $y_{k_1 k_2}$ as:

$$y_{k_1 k_2} = \alpha_{k_1} \alpha_{k_2} \sum_{i=0}^{n-1} \sum_{j=0}^{p-1} m_{ij} \cos\left(\frac{k_1(2i+1)\pi}{2n}\right) \cos\left(\frac{k_2(2j+1)\pi}{2p}\right), \quad (2.9)$$

where $0 \leq k_1 < n$, $0 \leq k_2 < p$. The factors α_{k_1} and α_{k_2} are defined as:

$$\alpha_{k_1} = \begin{cases} \sqrt{\frac{1}{n}} & k_1 = 0 \\ \sqrt{\frac{2}{n}} & k_1 = 1, \dots, n-1, \end{cases} \quad (2.10)$$

$$\alpha_{k_2} = \begin{cases} \sqrt{\frac{1}{p}} k_2 = 0 \\ \sqrt{\frac{2}{p}} k_2 = 1, \dots, p-1. \end{cases} \quad (2.11)$$

As the basis functions are orthonormal, the inverse transform can be easily calculated as:

$$\hat{m}_{ij} = \sum_{k_1=0}^{n-1} \sum_{k_2=0}^{p-1} \alpha_{k_1} \alpha_{k_2} y_{k_1 k_2} \cos\left(\frac{k_1(2i+1)\pi}{2n}\right) \cos\left(\frac{k_2(2j+1)\pi}{2p}\right), \quad (2.12)$$

where $\hat{m}_{ij}$ is the estimated speed value $\hat{z}(s_{j+1}, t_{i+1})$ for link s_{j+1} at time t_{i+1} . Traffic parameters such as speed and flow tend to be highly correlated across the network [22, 6]. Therefore, we expect that most of the information contained in the network profile $\mathbf{A}$ can be represented by considering a small number of frequency components Θ and we can discard the rest of the frequency components $\{y_{k_1 k_2} = 0\}_{k_1 k_2 \notin \Omega}$. Here Ω is the set of the indices of the frequency components used for reconstruction of traffic data [99]. As the basis functions are pre-specified, we only need to store the elements belonging to the set Ω to reconstruct the network profile $\hat{\mathbf{A}}$.

2.3.3 Wavelets

Wavelet transforms have been widely used in compression related applications including images [105, 106] and medical data sets such as electroencephalogram (EEG) [107, 108]. Similar to DCT, wavelets also perform compression using a pre-specified basis set. Wavelet based methods have also been applied for compression of traffic related data sets [13, 10, 100, 5]. However, these studies only analyze data obtained from either individual links or small networks [5, 13]. Moreover, these studies have not compared the performance of wavelets with other subspace methods for traffic data sets. In this study, we apply 2D wavelet transform to compress speed data obtained from a large road network.

To choose an appropriate wavelet type, we consider 5 commonly used wavelets and compare their reconstruction efficiency for a large road network. The different

Figure 2.4: Reconstruction efficiency of different wavelet types.

wavelet types we consider are *Near Symmetric*, *Bi-orthogonal 3/5*, *Discrete Meyer*, *Daubechies* and *Coiflets* wavelets. We calculate the element ratio for the wavelet transform by taking the ratio of the total number of elements np in the network profile $\mathbf{A}$ and the number of wavelet coefficients Θ used for the reconstruction.

Fig. 2.4 shows the reconstruction performance of different wavelet types. For our analysis we will consider *Near Symmetric* wavelet as it provides the best reconstruction efficiency.

2.3.4 Non-negative matrix factorization

The non-negative matrix factorization (NMF) provides low-rank approximation of a given matrix by constraining the factors to be non-negative. It has found applications in many fields including text mining [109] and transportation systems [11, 30]. In the field of transportation systems, NMF has been applied to applications related to estimation as well as prediction [11, 30]. In this study, we will focus on the reconstruction efficiency of NMF for large-scale networks. For the network profile $\mathbf{A}$,

Figure 2.5: Construction of a 3-way tensor for traffic data sets.

the NMF optimization problem is defined as:

$$\begin{aligned}
 \min \quad & f(\mathbf{W}, \mathbf{H}) = \frac{1}{2} \|\mathbf{A} - \mathbf{WH}\|_F^2, \\
 \text{s.t.} \quad & w_{ij} \geq 0 \quad \forall i, j, \\
 & h_{ij} \geq 0 \quad \forall i, j,
 \end{aligned} \tag{2.13}$$

where w_{ij} and h_{ij} are the elements of matrices $\mathbf{W} \in \mathbb{R}^{n \times r}$ and $\mathbf{H} \in \mathbb{R}^{r \times p}$ respectively. Furthermore, matrices $\mathbf{H}$ and $\mathbf{W}$ both have rank r , where $r \leq \min(n, p)$. Since traffic parameters such as speed, flow and travel time take on non-negative values, NMF may prove suitable for obtaining row-rank approximations for traffic data sets. The non-negative nature of factors $\mathbf{W}$ and $\mathbf{H}$ can potentially provide better interpretability for the underlying model $\hat{\mathbf{A}} = \mathbf{WH}$ [110, 11]. The optimization problem in (2.13) is typically solved using the multiplicative update algorithm [11]. To reconstruct a given compressed state, we need to store $\Theta = (n + p)r$ elements.

2.3.5 Tensor method for traffic data compression

Tensor based methods have been shown to achieve high compression efficiency for biomedical data sets such as electroencephalogram (EEG) recordings [107, 108]. In this section, we assess the compression efficiency of 3-way tensor representation for traffic data sets. Tensor decomposition algorithms require large computational resources. Therefore, we will consider data from a subset of roads $\{\sigma_i\}_{i=1}^{p_t} \subset \{s_i\}_{i=1}^p$, where $p_t = 6,024$. To create the 3-way tensor, we first create daily profile matrices $\{\mathbf{A}_i^t \mid \mathbf{A}_i^t \in \mathbb{R}^{d_t \times p_t}\}_{i=1}^m$ for the month of August. Each matrix $\mathbf{A}_i^t$ contains one day of speed data for p_t links. We then stack these daily profile matrices to obtain a 3-way tensor $\underline{\mathbf{A}} \in \mathbb{R}^{d_t \times p_t \times m}$ (see Fig. 2.5). To perform rank $(r_1 - r_2 - r_3)$ tensor decomposition, we apply higher order singular vector decomposition (HO-SVD) [111, 112]. HO-SVD will decompose the 3-way tensor into following components:

$$\underline{\mathbf{A}} = \underline{\mathbf{C}} \times_1 \mathbf{W}^{(1)} \times_2 \mathbf{W}^{(2)} \times_3 \mathbf{W}^{(3)}, \quad (2.14)$$

where $\underline{\mathbf{C}} \in \mathbb{R}^{r_1 \times r_2 \times r_3}$ is the core tensor, and the matrices $\mathbf{W}^{(1)} \in \mathbb{R}^{d_t \times r_1}$, $\mathbf{W}^{(2)} \in \mathbb{R}^{p_t \times r_2}$ and $\mathbf{W}^{(3)} \in \mathbb{R}^{m \times r_3}$ are termed as factor matrices. In (2.14), $\times_n$ represents the mode- n multiplication [113]. We obtain the rank $(r_1 - r_2 - r_3)$ HO-SVD by following the procedure outlined in [113]. To reconstruct a given compressed state, we need to store $(r_1 \times r_2 \times r_3 + d_t \times r_1 + p_t \times r_2 + m \times r_3)$ elements.

Fig. 2.6 shows the compression efficiency of HO-SVD. It seems that 3-way tensor representation may not be useful for traffic data compression. However, we need to consider two issues. The optimal selection of ranks r_1 , r_2 and r_3 is a non-trivial problem. Secondly, the core tensor in HO-SVD is not diagonal, which results in a large storage overhead.

2.3.6 Performance comparison

In this section, we compare the reconstruction efficiency of the four low-dimensional models presented in the previous section. Fig. 2.7 shows the performance of these

Figure 2.6: Compression efficiency of HO-SVD in terms of element ratio and relative error.

low-dimensional models for different element ratios (ER). DCT and wavelets have comparable performance in terms of both relative error and MAPE. For DCT and wavelets, we only need to store the transformed coefficients for reconstruction. For SVD, we need to store matrices $\mathbf{U}$, $\mathbf{V}$ as well as singular values. Hence, SVD has a lower element ratio for a particular threshold of relative error.

To visualize the reconstruction patterns for these low-dimensional models, we show the data from a typical link in Fig. 2.8. The figure shows around two days of actual speed data from the link along with the reconstructed speed profile. The ER for this particular representation $\hat{\mathbf{A}}$ is around 9.5 for the four methods.

NMF provides an interesting case. The low-dimensional model obtained from NMF was able to capture the dominant trend in the speed profile. However, the model failed to incorporate localized variations in the speed profile (see Fig. 2.8d). Moreover, the curve representing the reconstruction error of NMF remains flat for different element ratios (see Fig. 2.7). We observe that although NMF can provide reasonable reconstruction performance with a small number of factors, it cannot model the local variations efficiently.

Lossy reconstruction does not provide any guarantee that the maximum loss of information $|z(s_i, t_j) - \hat{z}(s_i, t_j)|$ will remain below a certain tolerance limit δ . For

(a) Relative error.

(b) MAPE.

Figure 2.7: Comparison of the reconstruction efficiency of different low-dimensional models.

Figure 2.8: Reconstructed speed profile of a typical road segment using different low-dimensional models. The units for speed values are km/h. The speed data in this figure is taken from 1st and 2nd August, 2011.

Figure 2.9: Schematic block diagram of the near-lossless compression of traffic data obtained from large road networks. $Q(\cdot)$ refers to the quantization scheme used for residuals (see (2.16)).

instance, although it may seem from the example that the maximum error for DCT, wavelets and SVD should be small. However, all four methods reported MAE in excess of 60 km/h for the reconstructed network profile $\hat{\mathbf{A}}$. In the next section, we propose a two-step compression algorithm to mitigate this issue by performing near-lossless compression for traffic data sets.

2.4 Near-lossless data compression

In this section, we discuss a two-step algorithm for near-lossless compression of traffic data obtained from large networks. Fig. 2.9 shows the steps involved during the encoding and decoding phases. We now briefly discuss the design of encoder and decoder for near-lossless compression of traffic data sets.

2.4.1 Encoder

The encoder consists of two stages: (1) lossy compression and (2) residual coding to keep the maximum reconstruction error below the tolerance limit δ . We start

by creating the network profile $\mathbf{A}$ for traffic speed data obtained from the set of links $\{s_i | s_i \in E\}$. In the first step, we perform lossy compression by means of different subspace methods such as DCT, SVD, wavelets, and NMF. The bitstream χ_{en} represents the compressed representation obtained from these methods. For SVD the factors $\{(\mathbf{u}_i, \sigma_i, \mathbf{v}_i)\}_{i=1}^r$ can be encoded in straightforward manner. The same goes for factors $(\mathbf{W}, \mathbf{H})$ obtained from NMF. For DCT and wavelets, techniques such as run length encoding or sparse representation can be used to store the appropriate set of coefficients and their positions Ω . Many ITS applications such as feature selection, estimation and prediction only require transformed set of variables instead of the complete data set [97, 14, 30, 4, 114, 115, 116]. These applications can directly use the compressed state χ_{en} instead of actual network profile $\mathbf{A}$ or the decompressed representation $\hat{\mathbf{A}}$. However, by only storing χ_{en} , we cannot control the maximum loss of information $|z(s_j, t_i) - \hat{z}(s_j, t_i)|$. In the second step, we encode the residual error so that the maximum reconstruction error remains below the tolerance limit δ . The elements e_{ij} of the residual matrix $\mathbf{E}$ are calculated as:

$$e_{ij} = z(s_j, t_i) - \hat{z}(s_j, t_i), \quad (2.15)$$

where $\hat{z}(s_j, t_i) = \hat{a}_{ij}$ represents the speed value obtained from the low-dimensional representation $\hat{\mathbf{A}}$. We then obtain the quantized residuals $\hat{e}_{ij} = Q(e_{ij}, \delta)$ as follows:

$$\hat{e}_{ij} = \begin{cases} 0 & |e_{ij}| \leq \delta \\ \lfloor e_{ij} \rfloor & \text{otherwise.} \end{cases} \quad (2.16)$$

where $\lfloor \cdot \rfloor$ rounds the argument to the nearest integer value. In this quantization scheme, we discard the residuals e_{ij} whose absolute value is equal to or less than the tolerance limit. For the rest of the residuals $\{e_{ij}\}_{|e_{ij}| > \delta}$, we round them to the nearest integer value. We apply Huffman coding to encode the quantized residuals $\hat{e}_{ij}$ in a lossless manner [117]. We represent the resultant bitstream by ε_{qen} . Let us now briefly discuss the decoder design.

Table 2.2: Compression ratios of different methods for $\delta = 0$. HC refers to the case, in which Huffman coding is directly applied to perform lossless compression.

Low-Dim model	DCT	NMF	SVD	Wavelets	HC
CR	1.66	1.57	1.65	1.62	1.2

2.4.2 Decoder

At the decoder end, we decompress the streams χ_{en} and ε_{gen} to obtain the matrices $\hat{\mathbf{A}}$ and $\hat{\mathbf{E}}$ respectively. Decompressing the bitstream χ_{en} will yield the low-dimensional representation $\hat{\mathbf{A}}$. Decompressing ε_{gen} will provide the quantized residuals stored in matrix $\hat{\mathbf{E}}$. Consequently, the reconstructed network profile will be $\mathbf{D} = \hat{\mathbf{A}} + \hat{\mathbf{E}}$. The decompressed speed value d_{ij} for a link s_j at time instance t_i will be:

$$d_{ij} = \hat{z}(s_j, t_i) + \hat{e}_{ij}. \quad (2.17)$$

The maximum absolute error (MAE) between the network profile $\mathbf{A}$ and the reconstructed profile $\mathbf{D}$ can be calculated as:

$$\begin{aligned}
 \text{MAE}(\mathbf{A}, \mathbf{D}) &= \max_{i,j} |a_{ij} - d_{ij}|, \\
 &= \max_{i,j} |(a_{ij} - \hat{a}_{ij}) - (d_{ij} - \hat{a}_{ij})|, \\
 &= \max_{i,j} |e_{ij} - \hat{e}_{ij}|, \\
 &\leq \delta.
 \end{aligned} \quad (2.18)$$

Selecting different tolerance limits will result in different compression ratios. We calculate the compression ratio (CR) as:

$$\text{CR} = \frac{L_{\text{org}}}{L_{\text{comp}}}, \quad (2.19)$$

where L_{org} and L_{comp} represent the bitstream lengths of the original and compressed sources respectively. For calculating CR, we set the resolution of speed data to be

Table 2.3: Near-lossless compression performance of low-dimensional models for different tolerance levels $\delta = \{1 \text{ km/h}, \dots, 15 \text{ km/h}\}$.

Method	$\delta = 1$		$\delta = 2$		$\delta = 3$		$\delta = 4$		$\delta = 5$	
	CR	PSNR	CR	PSNR	CR	PSNR	CR	PSNR	CR	PSNR
DCT	1.82	54.23	2.05	48.93	2.40	45.20	2.80	42.55	3.22	40.59
SVD	1.83	53.95	2.10	48.69	2.46	45.07	2.86	42.53	3.26	40.66
NMF	1.69	54.56	1.87	49.29	2.14	45.46	2.46	42.69	2.81	40.61
Wavelets	1.75	54.64	1.93	49.37	2.21	45.52	2.56	42.72	2.94	40.62

Method	$\delta = 6$		$\delta = 7$		$\delta = 8$		$\delta = 9$		$\delta = 10$	
	CR	PSNR	CR	PSNR	CR	PSNR	CR	PSNR	CR	PSNR
DCT	3.64	39.10	4.05	37.95	4.43	37.05	4.77	36.34	5.07	35.76
SVD	3.64	39.26	4.00	38.18	4.32	37.34	4.60	36.67	4.84	36.14
NMF	3.16	39.01	3.51	37.77	3.84	36.80	4.14	36.02	4.42	35.39
Wavelets	3.34	39.01	3.74	37.74	4.14	36.74	4.51	35.94	4.85	35.30

Method	$\delta = 11$		$\delta = 12$		$\delta = 13$		$\delta = 14$		$\delta = 15$	
	CR	PSNR	CR	PSNR	CR	PSNR	CR	PSNR	CR	PSNR
DCT	5.33	35.30	5.55	34.91	5.74	34.59	5.90	34.32	6.04	34.09
SVD	5.04	35.71	5.21	35.35	5.35	35.05	5.47	34.79	5.57	34.57
NMF	4.65	34.88	4.86	34.45	5.04	34.10	5.19	33.80	5.32	33.55
Wavelets	5.15	34.78	5.42	34.35	5.65	33.98	5.85	33.68	6.03	33.41

$B = 8 \text{ bits}$.

In the next section, we analyze the performance of the proposed near-lossless compression method for different tolerance limits.

2.5 Discussion

In this section, we analyze the performance of the proposed algorithms for the test network. We also compare the compression efficiency of these algorithms for different road categories, weekdays, and weekends. All the compression ratios are calculated by keeping the resolution of field data to $B = 8 \text{ bits}$. Considering higher resolution for field data will automatically result in higher albeit inflated CR. Table 2.2 shows the compression ratios for different subspace methods by setting $\delta = 0$. If we ignore the rounding off error due to (2.16), then this scenario can be considered as the lossless case. If we directly apply Huffman coding (HC) on the network profile $\mathbf{A}$, we obtain a CR of 1.2. Hence, the two step compression method straight away provides

Table 2.4: PSNR for different road categories.

Method	CR	Tol. δ	CATA	CATB	CATC	Slip Roads	Other
DCT	3.22	5	41.09	40.41	40.42	40.81	40.77
SVD	3.26		41.59	40.45	40.41	40.84	40.75
NMF	2.81		40.80	40.45	40.47	40.88	40.84
Wavelets	2.94		40.60	40.49	40.50	40.89	40.90
Method	CR	Tol. δ	CATA	CATB	CATC	Slip Roads	Other
DCT	5.07	10	37.20	35.70	35.47	35.50	35.47
SVD	4.84		37.95	36.12	35.77	35.78	35.72
NMF	4.42		35.54	35.41	35.32	35.22	35.36
Wavelets	4.85		35.62	35.31	35.22	35.11	35.24

Table 2.5: PSNR for different methods during weekdays and weekends for the month of August, 2011.

Method	CR	Tol. δ	Weekdays	Weekends
DCT	3.22	5	40.60	40.57
SVD	3.26		40.66	40.66
NMF	2.81		40.63	40.54
Wavelets	2.94		40.65	40.54
Method	CR	Tol. δ	Weekdays	Weekends
DCT	5.07	10	35.73	35.86
SVD	4.84		36.12	36.21
NMF	4.42		35.38	35.43
Wavelets	4.85		35.28	35.38

(a) Compression ratio and quantizer step size δ (km/h).

(b) PSNR for different compression ratios.

Figure 2.10: Performance of near-lossless compression methods based on different low-dimensional models (SVD/DCT/wavelets/NMF).

around 30 % to 38 % (CR: 1.57 – 1.66) improvement in CR. Table 2.3 shows the CR and PSNR achieved by the four low-dimensional models for different tolerance levels. For tolerance level of 5 km/h, the algorithms yield a CR of around 3. In this case, the SVD based method provides the highest compression ratio and PSNR. The other three methods also report similar PSNR. DCT has comparable CR to that of SVD. The other two methods (wavelets and NMF) report slightly smaller CR for this particular tolerance level. For tolerance level of 15 km/h, DCT and wavelets achieve CR of more than 6 (see Table 2.3). While SVD has the best PSNR for this tolerance level, it reports slightly smaller CR (5.57) as compared to DCT (6.04) and wavelets (6.03).

To observe these trends in more detail, we plot the compression ratios achieved by these algorithms against different tolerance levels in Fig. 2.10a. All the algorithms report similar CR for tolerance levels close to zero. For higher tolerance levels, DCT and wavelets achieve higher CR as compared to matrix decomposition methods (SVD, NMF). For low tolerance levels, residual coding can easily compensate for the inefficiency of a particular low-dimensional model. For higher CR, such inefficiencies become prominent. The tolerance level (MAE) alone does not provide all the information about the overall reconstruction error. Two methods with the same MAE can still have different mean square error. To this end, we plot PSNR against the compression ratios for these methods. PSNR values for different compression ratios are shown in Fig. 2.10b. The plot shows that SVD provides similar PSNR to DCT for a given CR. However, SVD yields higher maximum error as compared to DCT (see Fig. 2.10a).

Table 2.4 shows the category wise analysis of different near-lossless compression methods for tolerance levels of 5 km/h and 10 km/h. Expressways (CATA roads) achieve higher PSNR as compared to roads from CATB and CATC for all four algorithms. This difference in PSNR is more visible for $\delta = 10$ as opposed to $\delta = 5$. For $\delta = 10$, expressways have the highest PSNR in comparison with CATB, CATC, slip roads and other categories for each algorithm. Nonetheless, the PSNR gain for CATA as compared to other categories varies from one algorithm to another. These

values have some intuitive sense as well. Normally, traffic on expressways behaves much more smoothly in comparison with other roads. Therefore, subspace methods can model traffic conditions on expressways with high accuracy.

Table 2.5 shows the PSNR values for weekdays and weekends for different algorithms. We observe that reconstructed data for both weekdays and weekends has similar PSNR. This trend is observed for all the compression algorithms. Weekdays and weekends tend to have distinct traffic patterns. However, the similar compression performance indicates that traffic patterns across different time periods (from one week to another) remain quite similar. Consequently, traffic data sets can be easily compressed (see Tables 2.3 and 2.5).

Various studies [118, 119, 120] have identified management and storage of traffic data as one of the major big data problems currently faced by ITS. The proposed compression methods may prove to be useful for efficient storage and management of traffic data for systems dealing with large-scale networks. Furthermore, many ITS applications such as feature selection, estimation and prediction only require information related to transformed variables instead of the actual data. These applications can directly utilize the compressed data streams obtained from the first step.

Chapter 3

Missing data estimation for road networks

3.1 Introduction

With advancements in sensor technologies, intelligent transportation systems (ITS) can now collect traffic data from a wide range of stationary and mobile sensors [121, 96, 122, 22, 123, 124, 15, 125]. Stationary sensors such as loop detectors and road side cameras tend to have limited spatial coverage, whereas mobile sensors such as GPS probes collect data with highly erratic spatial and temporal resolution. These issues make the problem of missing data unavoidable in traffic data sets. Furthermore, failures such as detector malfunction and lossy communication systems may also result in incomplete traffic information [22, 19]. This can result in situations, where a high percentage of data is missing. Consequently, missing data is a commonly reported problem in traffic data sets [17, 18, 126, 16, 20, 19]. Different studies in this regard have reported that missing data percentages can be as high as 90% [16]. For traffic management systems, this is a critical issue [127, 128].

We propose various matrix and tensor based methods that can extract global traffic patterns from incomplete data. To this end, we consider different techniques such as

Table 3.1: Size of different test networks. Each test network is composed of roads from a specific category. These roads belong to the city-state network of Singapore. Primary and local access roads are referred as “Other Roads” in the table.

CATA	CATB	CATC	Slip Roads	Other Roads
2,175	2,500	1,428	1,572	2,221

fixed point continuation with approximate singular value decomposition (FPCA), canonical polyadic (CP) decomposition, least squares (LS) and variational Bayesian principal component analysis (VBPCA) to extract these global traffic states from incomplete data sets. We analyze the performance of these methods for different road categories and for days of the week. We also analyze the variance and bias induced by these methods in the imputed speed data. Furthermore, we also discuss the impact of rank selection on the performance of different methods.

3.2 Traffic data set and performance measures

In this section, we briefly explain the traffic data considered in this chapter. We also discuss different measures to evaluate the performance of the imputation methods.

3.2.1 Data set

We represent the test road network of size p by a set E of road segments s_i , such that $E = \{s_i\}_{i=1}^p$. In this study, we consider average speed data. The average speed on a link s_i during the interval $(t_j - \Delta t, t_j)$ is represented by $z(s_i, t_j)$. The sampling interval Δt is 5 minutes. For each link s_i , we create a speed profile $\mathbf{a}_i \in \mathbb{R}^n$ such that $\mathbf{a}_i = [z(s_i, t_1) \dots z(s_i, t_n)]^T$. The speed profiles contain one day of speed data for each link. We stack these speed profiles to obtain the network profile matrix $\mathbf{A} \in \mathbb{R}^{n \times p}$ such that $\mathbf{A} = [\mathbf{a}_1 \dots \mathbf{a}_p]$. Let $\mathbf{D} \in \mathbb{R}^{n \times p}$ be the corresponding incomplete observed data matrix. The set Ω contains the location of the entries in $\mathbf{D}$ for which speed data is available and the set $\Theta = \Omega^c$ represents the location of the

missing speed values in $\mathbf{D}$. We then consider various matrix completion methods to obtain an estimated network profile matrix $\hat{\mathbf{A}}$ from the incomplete matrix $\mathbf{D}$.

Fig. 3.1 and 3.2 illustrate the concept of recovering speed information from incomplete network profile by utilizing low-dimensional models.

For the tensor completion method, we create the network profile tensor $\underline{\mathbf{A}} \in \mathbb{R}^{n \times p \times q}$ by stacking together network profile matrices $\{\mathbf{A}_1, \mathbf{A}_2, \dots, \mathbf{A}_q\}$ from different days to form a 3-way tensor. To this end, we use $q = 7$ days of data. In this case, the incomplete tensor is represented by $\underline{\mathbf{D}} \in \mathbb{R}^{n \times p \times q}$.

For the analysis, we consider 5 test networks. The roads in each network belong to the city-state road network of Singapore for which sufficient data was available. The first test network consists of expressways (CATA). The second and third networks are composed of major and minor arterial roads respectively. We refer to major arterial roads as CATB and minor arterial roads as CATC. The fourth network contains slip roads, while the fifth network contains primary access and local access roads. Table 3.1 shows the size of each test network. The network consisting of primary/local access roads is referred as “Other Roads” in the table. The speed data was provided courtesy of Singapore’s Land Transportation Authority (LTA). In this study, we consider speed data from August 1, 2011 to August 7, 2011.

3.2.2 Performance measures

In this section, we briefly describe different performance measures to assess the proposed methods. For matrices, we define the weighted relative error (WRE) between actual $\mathbf{A}$ and estimated network profile $\hat{\mathbf{A}}$ as:

$$\text{WRE} = \frac{\|\mathbf{W} \circ (\mathbf{A} - \hat{\mathbf{A}})\|_F}{\|\mathbf{W} \circ \mathbf{A}\|_F}, \quad (3.1)$$

Figure 3.1: Incomplete and reconstructed speed profiles (from incomplete data) of a subset of road segments from $\mathbf{D}^T$ and $\hat{\mathbf{A}}^T$, respectively. The colorbars represent the colors corresponding to different speed values. The maroon (brownish-red) slots in the top image represent the missing values. A row in the image represents the speed values (incomplete $\mathbf{d}_i^T$ and reconstructed $\hat{\mathbf{a}}_i^T$) of an individual link for 3rd and 4th August, 2011.

Figure 3.2: Actual and reconstructed speed profiles (from incomplete data) of a subset of road segments from $\mathbf{A}^T$ and $\hat{\mathbf{A}}^T$, respectively. The colorbars represent the colors corresponding to different speed values. A row in the image represents the speed values (actual $\mathbf{a}_i^T$ and reconstructed $\hat{\mathbf{a}}_i^T$) of an individual link for 3rd and 4th August, 2011.

where the symbol $\circ$ represents the element wise multiplication between the two matrices. The matrix $\mathbf{W} \in \mathbb{R}^{n \times p}$ is the weight matrix with values:

$$w_{ij} = \begin{cases} 0 & (i, j) \in \Omega \\ 1 & (i, j) \in \Theta. \end{cases} \quad (3.2)$$

The Frobenius Norm $\|\mathbf{A}\|_F$ of a matrix $\mathbf{A} \in \mathbb{R}^{n \times p}$ is defined as:

$$\|\mathbf{A}\|_F = \sqrt{\sum_{i=1}^n \sum_{j=1}^p a_{ij}^2} \quad (3.3)$$

Similarly, we define WRE for tensors as follows:

$$\text{WRE} = \frac{\|\underline{\mathbf{W}} \circ (\underline{\mathbf{A}} - \hat{\underline{\mathbf{A}}})\|_F}{\|\underline{\mathbf{W}} \circ \underline{\mathbf{A}}\|_F}, \quad (3.4)$$

where the symbol $\circ$ represents the element wise multiplication between the two tensors. The tensor $\underline{\mathbf{W}} \in \mathbb{R}^{n \times p \times q}$ is the weight tensor with values:

$$w_{ijk} = \begin{cases} 0 & (i, j, k) \in \Omega \\ 1 & (i, j, k) \in \Theta. \end{cases} \quad (3.5)$$

The Frobenius Norm $\|\underline{\mathbf{A}}\|_F$ of a tensor $\underline{\mathbf{A}} \in \mathbb{R}^{n \times p \times q}$ is defined as:

$$\|\underline{\mathbf{A}}\|_F = \sqrt{\sum_{i=1}^n \sum_{j=1}^p \sum_{k=1}^q a_{ijk}^2} \quad (3.6)$$

Weighted relative error is commonly used to evaluate the performance of matrix and tensor completion algorithms [79, 129]. We calculate the bias induced in the imputed speed data as follows:

$$\text{Bias}_{\text{mat}} = \frac{1}{|\Theta|} \sum_{(i,j) \in \Theta} a_{ij} - \hat{a}_{ij}, \quad (3.7)$$

$$\text{Bias}_{\text{ten}} = \frac{1}{|\Theta|} \sum_{(i,j,k) \in \Theta} a_{ijk} - \hat{a}_{ijk}, \quad (3.8)$$

where $|\Theta|$ represents the size of the set Θ . We also calculate the variance of the imputed values and compare it with the variance of the actual speed data.

3.3 Missing data estimation

In this section, we briefly discuss various matrix and tensor completion algorithms for estimation of missing data in matrices and tensors. We apply least squares (LS), fixed point continuation with approximate singular value decomposition (FPCA) and variational Bayesian principal component analysis (VBPCA) to recover missing speed information in the incomplete matrices. For tensor completion, we apply canonical polyadic weighted optimization (CP-WOPT) to recover the missing traffic information.

3.3.1 Least squares method (LS)

Traffic parameters such as speed tend to behave similarly across an interconnected network [28, 11]. We aim to utilize these latent patterns to recover the missing speed information in the incomplete matrix $\mathbf{D}$. To this end, let us first consider the complete network profile matrix $\mathbf{A}$. By applying principal component analysis (PCA), we can obtain a low rank (with rank- r) approximation $\hat{\mathbf{A}}$ of the network profile matrix $\mathbf{A}$ as follows:

$$\hat{\mathbf{A}} = \mathbf{W}\mathbf{X} + \mathbf{M}, \quad (3.9)$$

where $\mathbf{W} \in \mathbb{R}^{n \times r}$ and $\mathbf{X} \in \mathbb{R}^{r \times p}$ are two low-rank matrices and $\mathbf{M} \in \mathbb{R}^{n \times p}$ contains the row-wise mean values of $\mathbf{A}$. The decomposition in (3.9), can be obtained by

solving the following least squares optimization problem:

$$\min_{\hat{\mathbf{A}}} \sum_{i=1}^n \sum_{j=1}^p (a_{ij} - \hat{a}_{ij})^2, \quad (3.10)$$

$$\hat{a}_{ij} = \mathbf{w}_i^T \mathbf{x}_j + m_{ij},$$

with the constraint that the vectors $\{\mathbf{w}_i\}_{i=1}^r$ remain orthonormal [130]. In the case of incomplete matrix $\mathbf{D}$, we can reformulate the problem by only minimizing the reconstruction error for observed speed data $\{d_{ij}\}_{(i,j) \in \Omega}$. Hence, the optimization problem will become [131]:

$$\min_{\hat{\mathbf{A}}} \sum_{(i,j) \in \Omega} (d_{ij} - \hat{a}_{ij})^2, \quad (3.11)$$

$$\hat{a}_{ij} = \mathbf{w}_i^T \mathbf{x}_j + m_{ij}.$$

In this study, we solve the optimization problem in (3.11) by the means of commonly applied gradient descent algorithm.

3.3.2 Variational bayesian principal component analysis (VBPCA)

In the previous section, we discussed the least squares method to obtain the low-rank approximation $\hat{\mathbf{A}}$ of matrix $\mathbf{A}$ from the incomplete network profile matrix $\mathbf{D}$. However, the least squares approach is prone to overfitting [80]. The problem of overfitting can be avoided by using probabilistic methods to perform PCA on incomplete matrices. These methods have been previously used to estimate missing traffic information from incomplete data sets [24, 22]. However, these studies only considered small networks and did not evaluate the performance of probabilistic methods for different road categories and during different days of the week. Furthermore, analysis in terms of induced bias and variance in the recovered speed data also needs to be considered.

We apply VBPCA to estimate missing speed data in the incomplete network profile

matrix $\mathbf{D}$. VBPCA is more resilient to overfitting in comparison with other probabilistic methods such as probabilistic principal component analysis (PPCA) and maximum a posteriori PCA (MAPPCA) [80]. In this study, we apply a variant of VBPCA proposed by Ilin and Raiko [80], which they termed as VBPCAd. This approach has faster convergence rates as opposed to traditional VBPCA implementation [80].

VBPCA avoids the problem of overfitting by penalizing complex representation of data. Thus it has a built-in mechanism for rank regularization. However, this rank selection approach can sometimes lead to suboptimal solutions (local minima) [80]. Secondly, the network profile matrix $\mathbf{A}$ (or the incomplete profile matrix $\mathbf{D}$) is not a low-rank matrix in the strict sense. In section 3.3.6, we will discuss the effect of the number of latent factors on the imputation performance of the algorithm.

3.3.3 Fixed point continuation with approximate singular value decomposition (FPCA)

In this section, we discuss an alternative way to estimate the missing traffic information. We aim to recover these missing speed values in the incomplete data matrix $\mathbf{D}$ by utilizing the common traffic behavior across different roads $\{s_i\}_{i=1}^p$. To this end, we need to obtain a suitable low-rank approximation $\hat{\mathbf{A}}$ from the incomplete speed data $\{d_{ij}\}_{(i,j) \in \Omega}$. Furthermore, the estimated network profile $\hat{\mathbf{A}}$ should also conserve the speed information already available in the incomplete data matrix $\mathbf{D}$ within a certain tolerance limit ϵ , such that $\{|\hat{a}_{ij} - d_{ij}| < \epsilon\}_{(i,j) \in \Omega}$. Hence, we can setup the optimization problem as follows:

$$\begin{aligned} & \min \text{rank}(\hat{\mathbf{A}}), \\ \text{s.t. } & |\hat{a}_{ij} - d_{ij}| < \epsilon, \quad \forall (i, j) \in \Omega. \end{aligned} \tag{3.12}$$

The above mentioned optimization problem tries to recover the missing speed data with the smallest number of latent components while preserving the speed information provided by the observed data $\{d_{ij}\}_{(i,j) \in \Omega}$. However, this is a non-convex

Figure 3.3: Network profile tensor and the corresponding unfolded matrix form $\mathbf{U}$. The matrices $\{\mathbf{A}\}_{i=1}^q$ each contain one day of speed data from the test network.

and NP-hard problem [132, 133]. To make the problem tractable, we can replace $\text{rank}(\hat{\mathbf{A}})$ by its convex envelope, which turns out to be the nuclear norm $\|\hat{\mathbf{A}}\|_*$ of the estimated matrix $\hat{\mathbf{A}}$ [132]. This way, the problem in (3.12) can be reformulated as:

$$\begin{aligned} \min \quad & \|\hat{\mathbf{A}}\|_*, \\ \text{s.t.} \quad & |\hat{a}_{ij} - d_{ij}| < \epsilon, \quad \forall (i, j) \in \Omega, \end{aligned} \tag{3.13}$$

where the nuclear norm of the matrix $\hat{\mathbf{A}}$ of rank r is defined as:

$$\|\hat{\mathbf{A}}\|_* = \sum_{i=1}^r \sigma_i, \tag{3.14}$$

and σ_i is the i^{th} singular value of the matrix $\hat{\mathbf{A}}$. We consider fixed point continuation with approximate singular value decomposition (FPCA) to solve the optimization problem defined in (3.13) [79].

3.3.4 Tensor decomposition

So far, we have discussed different matrix completion methods to extract the underlying traffic patterns in road networks. However, these methods cannot efficiently utilize multi-way dependencies in traffic data sets. For instance, consider the behavior of road traffic during different days of the week. Naturally, traffic parameters such as speed tend to follow similar daily patterns [134]. These temporal relationships can be extracted in a more efficient manner by creating a multi-way structure for traffic data. To this end, we represent the speed data in the form of a 3-way tensor $\underline{\mathbf{A}} \in \mathbb{R}^{n \times p \times q}$. This tensor profile is obtained by stacking together the network profile matrices $\{\mathbf{A}_1, \mathbf{A}_2, \dots, \mathbf{A}_q\}$ from different days (see Fig. 3.3). Canonical polyadic (CP) decomposition is commonly used to obtain low-rank approximations for tensors [113]. For the incomplete tensor profile $\underline{\mathbf{D}}$, we can obtain a suitable low-rank approximation $\hat{\underline{\mathbf{A}}}$ by minimizing the reconstruction error for the observed speed data in the following manner:

$$\begin{aligned} \min_{\hat{\underline{\mathbf{A}}}} \quad & \frac{1}{2} \|\underline{\mathbf{W}} \circ (\underline{\mathbf{D}} - \hat{\underline{\mathbf{A}}})\|_F^2, \\ \hat{\underline{\mathbf{A}}} = \quad & \sum_{i=1}^r \mathbf{b}_i^{(1)} \otimes \mathbf{b}_i^{(2)} \otimes \mathbf{b}_i^{(3)}, \end{aligned} \quad (3.15)$$

where $\mathbf{b}_i^{(m)}$ is the i^{th} column vector of mode- m factor matrix $\mathbf{B}^{(m)}$ [113, 129]. In (3.15), the symbol $\otimes$ denotes the vector outer product, whereas the symbol $\circ$ represents element wise multiplication between two tensors [113]. The factor matrices $\mathbf{B}^{(1)}$, $\mathbf{B}^{(2)}$ and $\mathbf{B}^{(3)}$ contain the common traffic patterns across different modes of the tensor. These patterns include common traffic behavior across different days and between different roads. We apply CP weighted optimization (CP-WOPT) to obtain a suitable estimation $\hat{\underline{\mathbf{A}}}$ from the incomplete network profile tensor $\underline{\mathbf{D}}$. We refer to this technique as CP (3D).

We also apply CP-WOPT on the unfolded tensor to study the impact of multi-way representation on the imputation performance. To this end, we create another

Figure 3.4: Impact of restricting the maximum number of factors on the imputation accuracy of VBPCA. The WRE is calculated for the case when 90% of speed data is missing.

network profile matrix $\mathbf{U} \in \mathbb{R}^{n \times pq}$ | $\mathbf{U} = [\mathbf{A}_1 \ \dots \ \mathbf{A}_q]$ by combining speed data from multiple days. This network profile matrix $\mathbf{U}$ is essentially an unfolded representation of the network profile tensor $\underline{\mathbf{A}}$ (see Fig. 3.3). In this case, the corresponding incomplete data matrix is represented by $\mathbf{D}_u$. Similar to CP (3D), the low-rank approximation $\hat{\mathbf{U}}$ of the matrix $\mathbf{U}$ from the incomplete speed data $\mathbf{D}_u$ is obtained by minimizing the reconstruction error for the observed speed data:

$$\min_{\hat{\mathbf{U}}} \frac{1}{2} \|\mathbf{W} \circ (\mathbf{D}_u - \hat{\mathbf{U}})\|_F^2, \quad (3.16)$$

$$\hat{\mathbf{U}} = \sum_{i=1}^r \mathbf{b}_i^{(1)} \otimes \mathbf{b}_i^{(2)}.$$

We apply CP-WOPT to obtain the estimated network profile matrix $\hat{\mathbf{U}}$. This formulation will be referred to as CP (unfold).

3.3.5 Mean substitution

In mean substitution (MS), we replace the missing speed information for the link s_j by the average speed of that road, which is obtained from the available speed

Figure 3.5: Weighted relative error by considering different number of latent factors (rank) for different percentages of missing data. The test network is composed of expressways (CATA).

Figure 3.6: Weighted relative error by considering different number of latent factors (rank) for different percentages of missing data. The test network is composed of major arterial roads (CATB).

Figure 3.7: Weighted relative error by considering different number of latent factors (rank) for different percentages of missing data. The test network is composed of minor arterial roads (CATC).

Figure 3.8: Weighted relative error by considering different number of latent factors (rank) for different percentages of missing data. The test network is composed of slip roads.

Figure 3.9: Weighted relative error by considering different number of latent factors (rank) for different percentages of missing data. The test network is composed of access roads.

data. For instance, the missing speed value $\hat{a}_{ij} = \hat{z}(s_j, t_i)$ of a link s_j at time t_i is estimated as follows:

$$\hat{a}_{ij} = \frac{1}{n - n_{s_j}} \sum_{i=1}^n (1 - w_{ij}) d_{ij} \quad , \quad (3.17)$$

where n_{s_j} is the number of missing speed values in the incomplete speed profile $\mathbf{d}_j$ of the link s_j .

3.3.6 Latent factors

In this section, we discuss the impact of the choice of rank (number of latent factors) on the imputation performance of the proposed algorithms. Fig. 3.5 shows the variations in reconstruction performance of different algorithms caused by the choice of rank for speed data obtained from expressways. Fig. 3.6 shows these variations for speed data obtained from major arterial roads. Let us first discuss the performance of LS, CP (3D) and CP (Unfold). These three methods try to extract common patterns in data by minimizing the mean squared error for the observed speed information. For large percentages of missing data, the reconstruction error of these algorithms can vary significantly, depending upon the choice of rank. Furthermore, the imputation performance of these algorithms fluctuates more in the case of arterial roads as compared to expressways (see Fig. 3.5 and 3.6). On the other hand, the reconstruction error for FPCA and VBPCA does not vary significantly for different rank values. The results for rank sensitivity analysis of minor arterial roads, slip roads and access roads are shown in Fig. 3.7, 3.8 and 3.9. The sensitivity patterns for minor roads, slip roads and access roads are quite similar to those of expressways and major arterial roads.

The rank values for VBPCA in Fig. 3.5e, 3.6e, 3.7e, 3.8e and 3.9e represent the limit on the maximum number of factors that can be used to reconstruct the estimated network profile matrix $\hat{\mathbf{A}}$. VBPCA can automatically choose the optimal number of factors while estimating missing values in the incomplete speed data matrix $\mathbf{D}$.

Hence, it might be tempting to assume that the information about the maximum number of factors (rank) is redundant. However, this assumption does not hold in all cases. Fig. 3.4 shows the impact of setting a limit on the maximum number of latent factors for VBPCA. The reconstruction error is shown for the scenario when 90% of speed data is missing. We can conclude that VBPCA is also prone to overfitting if a suitable cut-off value for the rank is not available.

3.4 Discussion

In this section, we compare the imputation performance of various matrix and tensor completion methods for different road networks. We compare their performances for different percentages of missing data and during different days of the week. We also analyze the variance as well as the bias imparted by these methods in the imputed speed data for different types of roads. To this end, we consider five test networks each comprising of around 1,500 roads.

Let us start by analyzing the estimation accuracy of the proposed algorithms in terms of weighted relative error (WRE). Fig. 3.10 shows the imputation accuracy of these methods for different road types. For expressways, VBPCA achieves the lowest WRE followed by FPCA. The imputation error for expressways is lower for all the algorithms as compared to other road categories. For major and minor arterial roads, CP (3D) and FPCA provide slightly better performance as compared to other methods (see Fig. 3.10b and Fig. 3.10c). FPCA also achieves better performance for slip roads (see Fig. 3.10d). In the case of access roads, all the algorithms suffered from large imputation error.

CP (3D), CP (Unfold) and LS all try to impute missing values by finding those traffic patterns that can minimize the reconstructed squared error for the observed speed data $\{(d_{ij} - \hat{a}_{ij})^2\}_{(i,j) \in \Omega}$. Out of these three least squared based methods, multi-way representation (tensor method) tends to achieve the best performance. Furthermore, for arterial roads and access roads, multi-way representation (tensor

Figure 3.10: Weighted relative error of the proposed algorithms for different percentages of missing data and various road networks. Mean sub. refers to the substitution performed by applying (3.17).

Figure 3.11: Weighted relative error of the proposed algorithms during different days of the week for various road networks. The reconstruction error is for the case when 50% of data is missing.

Figure 3.12: Weighted relative error of the proposed algorithms during different days of the week for various road networks. The reconstruction error is for the case when 70% of data is missing.

Figure 3.13: Weighted relative error of the proposed algorithms during different days of the week for various road networks. The reconstruction error is for the case when 90% of data is missing.

Table 3.2: Variance of the imputed speed data for different road types. The units for variance are km^2/hr^2 . The values in the brackets represent the variance of imputed speed data w.r.t the actual speed data.

Type	Missing	Variance (%age of actual variance)				
		FPCA	LS	CP(3D)	CP(Unfold)	VBPCA
CATA Var 154	50%	121.32 (79)	121.27 (79)	94.38 (61)	120.97 (79)	128.12 (83)
	60%	119.00 (77)	110.30 (72)	95.15 (62)	115.48 (75)	126.95 (83)
	70%	116.99 (76)	111.29 (72)	95.10 (62)	117.09 (76)	124.84 (81)
	80%	112.10 (73)	114.81 (75)	95.52 (62)	96.37 (63)	121.89 (79)
	90%	93.62 (61)	79.24 (52)	97.17 (63)	105.09 (68)	115.64 (75)
Type	Missing	FPCA	LS	CP(3D)	CP(Unfold)	VBPCA
CATB Var 163	50%	130.66 (80)	118.41 (73)	120.61 (74)	120.43 (74)	121.29 (74)
	60%	129.42 (79)	117.92 (72)	119.86 (73)	120.41 (74)	120.93 (74)
	70%	128.76 (79)	106.49 (65)	119.95 (73)	120.61 (74)	118.31 (72)
	80%	127.01 (78)	107.63 (66)	118.13 (72)	109.69 (67)	116.15 (71)
	90%	123.37 (76)	112.21 (69)	111.05 (68)	110.27 (68)	115.60 (71)
Type	Missing	FPCA	LS	CP(3D)	CP(Unfold)	VBPCA
CATC Var 112	50%	79.46 (71)	70.14 (63)	68.47 (61)	69.12 (62)	72.05 (64)
	60%	77.93 (69)	57.74 (51)	66.52 (59)	62.05 (55)	69.70 (62)
	70%	77.50 (69)	60.80 (54)	67.80 (60)	61.73 (55)	66.79 (60)
	80%	75.11 (67)	62.10 (55)	66.38 (59)	61.81 (55)	63.95 (57)
	90%	71.02 (63)	341.92 (300)	61.68 (55)	61.73 (55)	61.00 (54)
Type	Missing	FPCA	LS	CP(3D)	CP(Unfold)	VBPCA
Slip Var 412	50%	360.21 (86)	333.86 (80)	338.77 (81)	340.07 (81)	342.62 (82)
	60%	356.92 (85)	335.59 (80)	339.15 (81)	340.42 (81)	339.31 (81)
	70%	355.76 (85)	315.75 (75)	342.67 (82)	312.50 (75)	336.64 (80)
	80%	351.26 (84)	318.51 (76)	343.60 (82)	313.48 (75)	331.79 (79)
	90%	339.72 (81)	321.65 (77)	358.10 (85)	314.51 (75)	323.02 (77)
Type	Missing	FPCA	LS	CP(3D)	CP(Unfold)	VBPCA
Oth Var 206	50%	130.96 (64)	110.45 (54)	110.23 (54)	110.70 (54)	117.56 (57)
	60%	127.36 (62)	111.55 (54)	110.55 (54)	111.20 (54)	113.10 (55)
	70%	124.18 (60)	95.14 (46)	108.29 (53)	111.71 (54)	108.81 (53)
	80%	119.51 (58)	96.87 (47)	107.93 (53)	99.67 (48)	105.20 (51)
	90%	116.18 (57)	103.59 (50)	111.82 (54)	101.17 (49)	100.10 (49)

Table 3.3: Bias of the imputed speed data for different road types. The units for bias are km/hr.

Road Type	Missing Data	Bias				
		FPCA	LS	CP(3D)	CP(Unfold)	VBPCA
CATA Avg. Speed (86 km/hr)	50%	0.007	0.067	0.004	0.004	0.069
	60%	0.011	0.069	0.006	0.012	0.073
	70%	0.006	0.074	0.009	0.003	0.066
	80%	0.004	0.062	-0.003	-0.010	0.064
	90%	0.008	0.091	-0.003	0.000	0.062
Road Type	Missing Data	FPCA	LS	CP(3D)	CP(Unfold)	VBPCA
CATB Avg. Speed (43 km/hr)	50%	0.003	0.375	0.004	0.005	0.408
	60%	-0.015	0.384	-0.019	-0.008	0.365
	70%	-0.004	0.377	0.003	0.007	0.360
	80%	-0.006	0.377	0.004	0.018	0.344
	90%	-0.031	0.309	0.013	0.018	0.278
Road Type	Missing Data	FPCA	LS	CP(3D)	CP(Unfold)	VBPCA
CATC Avg. Speed (38 km/hr)	50%	0.001	0.476	0.009	0.007	0.398
	60%	-0.010	0.349	0.012	0.014	0.341
	70%	-0.003	0.418	0.007	0.016	0.360
	80%	-0.029	0.430	-0.006	0.001	0.334
	90%	-0.038	0.137	0.009	0.009	0.284
Road Type	Missing Data	FPCA	LS	CP(3D)	CP(Unfold)	VBPCA
Slip roads Avg. Speed (55 km/hr)	50%	-0.001	0.402	-0.002	-0.001	0.431
	60%	-0.007	0.416	0.002	-0.003	0.412
	70%	-0.003	0.418	0.010	0.050	0.439
	80%	-0.002	0.419	0.013	0.047	0.476
	90%	-0.002	0.428	0.044	0.074	0.438
Road Type	Missing Data	FPCA	LS	CP(3D)	CP(Unfold)	VBPCA
Access roads Avg. Speed (41 km/hr)	50%	-0.032	0.433	-0.001	0.000	0.408
	60%	-0.004	0.461	0.019	0.021	0.433
	70%	-0.022	0.361	0.005	0.003	0.367
	80%	-0.031	0.359	-0.001	0.012	0.361
	90%	-0.061	0.431	-0.958	0.020	0.354

method) also achieves better imputation accuracy than other methods such as FPCA and VBPCA. However, in case of expressways, the advantage of considering multi-way representation is not that apparent (see Fig. 3.10a). It seems that tensor representation is more useful for smaller roads where traffic behaves more erratically. In such cases, multi-way representation of speed data is an efficient way to extract underlying traffic patterns.

Let us now analyze the performance of different imputation methods across the week. Fig. 3.11 shows the imputation error of FPCA, LS and VBPCA during different days of the week. The results are shown for different road categories with speed data obtained from August 1, 2011 to August 7, 2011. In this scenario, the missing data percentage was 50%. The results for 70% and 90% missing data are shown in Fig. 3.12 and 3.13. Let us take the example of the scenario where 70% data is missing. For expressways, VBPCA has lower WRE as compared to other methods during most of the days. This is expected as VBPCA has the lowest overall imputation error for speed data obtained from expressways (see Fig. 3.10a). For arterial roads, all three methods have similar performance during most of the days (see Fig. 3.13b and 3.13c). However, VBPCA and LS tend to suffer from large estimation error on certain days. On the other hand, the estimation performance of FPCA does not vary significantly from one day to another. We also observe similar trend in the performances of LS, FPCA and VBPCA for slip roads (see Fig. 3.13d). For primary and local access roads, all three methods reported large imputation error during all seven days (see Fig. 3.13e). We see similar trends for the cases with 50% and 90% missing data. We can conclude that imputation performance of FPCA is more robust to daily variations in traffic conditions in comparison with other methods such as LS and VBPCA.

Let us now analyze the variance of the imputed speed data and the bias imparted by various algorithms. Table 3.2 shows the variance of the estimated data for different road types. It also shows the variance of the actual speed data. As expected, the imputation algorithms underestimate the variance of the imputed data. For instance, the actual variance of the speed data obtained from expressways was around

153 km²/hr². However, the variance of the speed data obtained from different imputation methods was around 100 – 130 km²/hr². Moreover, the difference between the variance of actual and imputed speed data becomes larger as the percentage of missing data increases. For expressways, VBPCA provided the best estimate of the variance in the speed data. For other road types such as arterial roads (CATA, CATB), access roads and slip roads, the variance of the imputed data obtained from FPCA was the closest to the variance of the actual speed data. Nonetheless, all the five methods had comparable performance in terms of conserving the variance of the speed data.

Table 3.3 shows the bias induced in the recovered speed data by various proposed methods. The results show that the proposed algorithms do not add significant bias in the imputed data as the bias-value remains less than 1 km/hr for all test cases. The imputed speed data obtained from VBPCA and LS had slightly higher bias (≈ 0.5 km/hr) in comparison with other methods.

Missing data is a common problem faced by many transportation management systems. In this chapter, we compared various methods to estimate missing traffic information in data sets obtained from large road networks. To this end, we extracted common global patterns from incomplete speed data by applying various matrix and tensor completion algorithms such as FPCA, VBPCA, LS and CP-WOPT. Matrix and tensor completion methods have been previously applied to solve the problem of missing data in transportation systems [16, 22, 23, 24, 25]. However, these studies typically consider small test networks comprising of a few roads. Furthermore, the performance of these methods for different road categories as well as during different days of the week is usually not analyzed.

In this study, we considered five large test networks each comprising of around 1,500 road segments. We analyzed the reconstruction accuracy of various matrix and tensor completion methods for different types of roads as well as during different days of the week. We also analyzed the impact of the choice of latent factors on the estimation accuracy of recovered speed data. Moreover, we compared the variance

and bias induced in the imputed speed data by the proposed methods.

The results show that multi-way representations can prove helpful in achieving better reconstruction accuracy (in terms of WRE) for small roads. However, for expressways, tensor representation does not offer any clear advantage. We also observed that the reconstruction accuracy of FPCA and VBPCA does not change significantly due to the choice of the rank. For methods such as LS, CP (3D) and CP (Unfold) the choice of latent factors can have significant impact on the imputation accuracy. The results also show that the imputed speed data obtained from different proposed algorithms (matrix and tensor) had similar bias and variance.

FPCA is particularly useful for imputation of traffic data sets as its performance is least sensitive to daily variations in traffic data. Furthermore, it also provides better or comparable performance (in terms of WRE, variance and bias) to other algorithms for different road categories.

Chapter 4

Spatial and temporal patterns in large-scale prediction

4.1 Introduction

Intelligent Transport Systems (ITS) can provide enhanced performance by incorporating data related to future state of the road networks [1]. Traffic prediction is useful for many applications such as congestion avoidance and travel time prediction [1, 48, 38]. Data driven methods such as Support Vector Regression (SVR) tend to provide better prediction results than competing methods [47, 38, 44, 48, 49, 41, 42, 46]. However, existing studies usually consider scenarios such as expressways, or a few intersections. In this study, we analyze the performance of data driven methods such as SVR for large-scale prediction. The network comprises of roads with different speed limits, capacities and covering different areas (urban, rural, downtown).

Traffic prediction studies usually consider point error measures like Mean Absolute Percentage Error (MAPE) to analyze prediction performance [28, 135, 38, 39, 40, 41, 42, 44, 46, 47, 48, 49]. These measures are helpful for comparing the overall performance. However, they fail to provide any insight into underlying spatial and

temporal prediction patterns. Forecasting methods may not provide a uniform prediction performance across the road network. Moreover, the prediction accuracy may also depend upon the time and the day. These spatial and temporal performance trends contain useful information about the predictability of the network. ITS applications can provide more robust and accurate solutions by utilizing such trends.

We apply temporal window SVR to perform large-scale traffic prediction. For analysis, we consider a large road network from Outram Park to Changi in Singapore (see Fig. 4.1 and 4.2). The road network consists of more than 5,000 road segments. We compare the performance of SVR with other commonly used time series prediction algorithms such as Artificial Neural Networks (ANN) and exponential smoothing. We also provide a performance comparison for different road categories in this study. To extract spatial and temporal prediction patterns for large networks, we propose unsupervised learning methods such as k-means clustering and Principal Component Analysis (PCA). For link level temporal prediction patterns, we apply Self Organizing Maps (SOM). We apply these data mining algorithms to extract performance trends in SVR prediction data.

4.2 Large-scale prediction

In this section, we briefly discuss the problem of large-scale prediction. We also discuss the selection of training and test data for supervised learning algorithms.

4.2.1 Traffic prediction

We represent the road network by a directed graph $G = (N, E)$. The set E contains p road segments (links) $\{s_i\}_{i=1}^p$. Speed value $z(s_i, t_j)$, represents the average speed on the road segment s_i , during the interval $(t_j - \delta_t, t_j)$. The sampling interval δ_t is 5 minutes. The future traffic trends strongly depend on current and past

behavior of that road and its neighbors [48, 38, 28]. Suppose $\{\theta_u \in \Theta_{s_i}\}_{u=1}^l$ is the set of road segments containing s_i and its neighbors, such that $\Theta_{s_i} \subseteq E$. Our aim will be to find the relationship function f between current/past traffic data $\{z(\theta_u, t_j - q\delta_t) | u = 1, \dots, l, q = 0, \dots, m_{\theta_u}\}$ and the future traffic variations $\hat{z}(s_i, t_j + k\delta_t)$ such that:

$$\hat{z}(s_i, t_j + k\delta_t) = f(z(\theta_1, t_j), \dots, z(\theta_l, t_j - m_{\theta_l}\delta_t)). \quad (4.1)$$

The feature set $\{m_{\theta_u}\}_{u=1}^l$ determines the horizon of the past speed values of link θ_u which are used for predicting k -step ahead speed values of s_i . We will refer to k -step ahead prediction as k^{th} prediction horizon.

We need to determine relevant links Θ_{s_i} (spatial features) and time lags m_{θ_u} (temporal features) to predict $\hat{z}(s_i, t_j + k\delta_t)$. Extracting spatial features is a computationally expensive task[136]. This additional computational cost severely limits the scalability of prediction algorithm for large and generic road networks. Therefore, we will not consider spatial features in this study. For large scale prediction, we consider the following variant of (4.1), termed as the temporal window method [49]:

$$\hat{z}(s_i, t_j + k\delta_t) = f(z(s_i, t_j), \dots, z(s_i, t_j - m_{s_i}\delta_t)). \quad (4.2)$$

In (4.2), we only consider past historical trends of s_i to predict $\hat{z}(s_i, t_j + k\delta_t)$. The temporal window method for feature selection has been demonstrated to work effectively for data driven traffic prediction algorithms [38, 41, 49, 37, 46, 47]. Different methods have been proposed to take further advantage of inherent temporal traffic patterns for enhanced prediction accuracy [65, 137, 39, 66, 138, 115, 42, 139]. These methods employ different feature selection techniques to pre-partition the data according to temporal patterns (time of the day, weekdays/weekends etc.). The feature selection algorithms include Self Organizing Maps (SOM) [137, 139, 66], genetic algorithms [138, 42], wavelets [115] and committees of neural networks [39]. Another proposed method combines Kalman filter with Seasonal Autoregressive Integrated Moving Average (SARIMA) [65, 66]. These techniques, however, are computational

expensive, which limits their scalability for large networks. Furthermore, M. Lippi et al. showed that traditional SVR can provide similar performance to these ensemble methods without suffering from extra computational overheads [66]. Consequently, we will consider SVR with temporal window for large scale prediction. We train separate predictors for each link s_i and for each prediction horizon k . Temporal window method for feature selection allows predictors from different links and prediction horizons to run in parallel. Furthermore, these algorithms are independent of each other. Therefore, they can efficiently run on distributed platforms with minimum communication overhead.

4.2.2 Training and test data for supervised learning

Supervised learning methods such as SVR and Artificial Neural Networks (ANN) assume that the labeled training data and the test data come from the same distribution [140, 141, 142]. Hence, it is unnecessary to retrain the algorithm every time new data becomes available. Traffic prediction methods also follow the same assumption [47, 143, 28, 38, 44, 48, 49, 41, 37, 42, 46]. Similar to other studies, we train the algorithm with 50 days of data and perform prediction for 10 days [28, 38, 44]. It is important to point out that this assumption may not hold true in the long term. Factors such as changes in transportation infrastructure, residential location, fuel prices and car ownership can significantly affect long term traffic patterns [143, 144]. Supervised learning methods may not work well in such cases. Techniques based on transfer learning might prove useful in such scenarios [142].

4.3 Traffic prediction algorithms

We apply SVR to perform large scale prediction. We briefly explain the algorithm in this section. We compare the performance of SVR with ANN and exponential smoothing. We also briefly discuss these algorithms in this section.

4.3.1 Support vector regression

SVR is a data driven prediction algorithm. It is commonly employed for time series prediction [145]. With temporal feature selection, the input feature vector $\mathbf{x}_j \in \mathbb{R}^n$ at time t_j for link s_i will be $\mathbf{x}_j = [z(s_i, t_j) \dots z(s_i, t_j - m_{s_i} \delta_t)]^T$. The feature vector $\mathbf{x}_j$ contains current average speed of the road $z(s_i, t_j)$ and m_{s_i} past speed values. Let $y_{jk} = z(s_i, t_j + k\delta_t)$ be the future speed value at time $t_j + k\delta_t$. We aim to find the relationship between y_{jk} and $\mathbf{x}_j$. To this end, we use historical speed data of s_i to train SVR. The training data contains r 2-tuples $\{(\mathbf{x}_j, y_{jk})\}_{j=1}^r$. We use SVR to infer non-linear relationships between $\mathbf{x}_j$ and y_{jk} , to find f_k in (4.2) for k^{th} prediction horizon.

We briefly explain the SVR algorithm here. More rigorous treatment of the topic can be found in [140, 141]. Let us first consider the formulation of SVR called ϵ -SVR, which is formulated as [140]:

$$\begin{aligned} & \text{minimize} \quad \frac{1}{2} \mathbf{w} \cdot \mathbf{w} + C \sum_{j=1}^r (\xi_j + \xi_j^*), \\ & \text{subject to} \quad \begin{cases} y_{jk} - \mathbf{w} \cdot \mathbf{x}_j - b \leq \epsilon + \xi_j \\ \mathbf{w} \cdot \mathbf{x}_j + b - y_{jk} \leq \epsilon + \xi_j^* \\ \xi_j, \xi_j^* \geq 0, \end{cases} \end{aligned} \quad (4.3)$$

where, $\mathbf{w}$ is the required hyperplane and ξ_j, ξ_j^* are the slack variables. It minimizes the insensitive loss function which imposes cost C on training points having deviation of more than $|\epsilon|$. It is often hard to predefine the exact value of error bound ϵ [146]. This problem can be avoided by adopting a variant of SVR called ν -SVR [146]. The two main properties of the parameter ν are as follows [146]:

- ν is an upper bound on the fraction of errors.
- ν is a lower bound on the fraction of Support Vectors.

Consequently, with ν -SVR, we have direct control over model generalization (support vectors) and data fitting (error). In ϵ -SVR, this would require tuning the

“tube radius” [146]. Due to these properties, we will employ ν -SVR to perform speed prediction.

In general, SVR non-linearly maps (not explicitly) the input speed data into some higher dimensional feature space Φ [146, 140]. It then finds the optimal hyperplane in that high dimensional feature space Φ . The kernel trick helps SVR to avoid this explicit mapping in Φ . Let us chose κ as the desired kernel function. Then we can replace dot products in the feature space by the relation $\kappa(\mathbf{x}_i, \mathbf{x}_j) = \Phi(\mathbf{x}_i) \cdot \Phi(\mathbf{x}_j)$ [140]. The function f_k will be [146, 140]:

$$f_k(\mathbf{x}) = \sum_{j=1}^r (\alpha_j - \alpha_j^*) \kappa(\mathbf{x}, \mathbf{x}_j) + b, \quad (4.4)$$

where α_j , α_j^* are the Lagrange multipliers. We employ (4.4) to perform speed prediction. We utilize the MATLAB package LIBSVM for SVR implementation [147].

For this study, we consider Radial Basis Function (RBF) kernel. It is highly effective in mapping non-linear relationships [148]. Consequently, it is commonly employed for performing traffic prediction [47, 38].

4.3.2 Artificial neural networks

Artificial Neural Networks (ANN) can perform time series prediction with high accuracy [149]. Consequently, they have been extensively applied for traffic parameter forecasting in different configurations [37, 150, 40, 47, 42, 49, 39, 48]. Multi-layer feed forward neural networks is the most commonly employed configuration for traffic prediction [37, 150, 47].

We applied feed forward neural network for large scale speed prediction of the network G across multiple prediction horizons. The neural network was composed of one hidden layer. We found that the hidden layer with 100 neurons provided the best results. We used sigmoid function as the activation function. Furthermore, the

Figure 4.1: The map of region for large scale prediction.

predictors were trained using Levenberg-Marquardt (LM) backpropagation method. Root mean square error (rmse) was used as the measure of goodness-of-fit (GoF) during the training phase.

We consider temporal window for feature selection for ANN. We train separate neural networks for different links and prediction horizons. The training was performed in the form of Batch processing, where 50 days of speed data from each road was used for training and 10 days of data was used for prediction.

4.3.3 Exponential smoothing

Exponential smoothing is a commonly employed method to perform time series prediction. It is also applied for traffic parameter prediction [151]. The prediction is computed as a weighted average of past data points. Specifically, weights of past values decay exponentially with decay factor χ_k for k^{th} prediction horizon.

Figure 4.2: Different road categories in the network.

4.4 Data set and performance measures

In this section, we explain the data set considered for this study. We consider a large sub-network in Singapore, which covers region from Outram park to Changi (see Fig. 4.1 and 4.2). The road network G consists of a diverse set of roads having different lane count, speed limits and capacities. It includes three expressways, which are East Coast Parkway, Pan Island Expressway and Kallang-Paya Lebar Expressway. The network also includes areas carrying significant traffic volumes, such as Changi Airport and the central business district.

Overall, the network G consists of $p = 5,024$ road segments. These road segments are grouped into different categories by the Land Transport Authority (LTA) of Singapore. Table 4.1 shows the number of road segments for each category in G . Fig. 4.2 shows the location of roads belonging to various categories. In this study, we consider speed data provided by LTA. The data set has an averaging interval of 5 minutes. We choose speed data from the months of March and April, 2011.

Table 4.1: Categories of road segments.

Category	CATA	CATB	CATC	Slip Roads	Other
No. of links	703	2,818	841	592	70

We now explain the performance measures for assessing the prediction accuracy of the proposed algorithms. We calculate Absolute Percentage Error (APE) for s_i at time t_j for k^{th} prediction horizon $e(s_i, k, t_j)$ as follows:

$$e(s_i, k, t_j) = \frac{|\hat{z}_k(s_i, t_j) - z(s_i, t_j)|}{z(s_i, t_j)}. \quad (4.5)$$

The Mean Absolute Percentage Error (MAPE) for link s_i and k^{th} prediction horizon $e_s(s_i, k)$ is defined as:

$$e_s(s_i, k) = \frac{1}{d} \sum_{j=1}^d \frac{|\hat{z}_k(s_i, t_j) - z(s_i, t_j)|}{z(s_i, t_j)}, \quad (4.6)$$

where d is the number of test samples and $\hat{z}_k(s_i, t_j)$ is the predicted speed value at time t_j for the k^{th} prediction horizon. MAPE is a commonly used metric to assess the accuracy of traffic prediction algorithms [38, 41, 47, 135]. For the whole network G containing p links, we calculate the MAPE $e_G(k)$ for the k^{th} prediction horizon as:

$$e_G(k) = \frac{1}{p} \sum_{i=1}^p e_s(s_i, k). \quad (4.7)$$

We calculate the Standard Deviation (SD) of error σ_k for the k^{th} prediction horizon as:

$$\sigma_k = \sqrt{\frac{1}{p} \sum_{i=1}^p (e_s(s_i, k) - e_G(k))^2}. \quad (4.8)$$

We use these measures to assess the performance of SVR, ANN and exponential smoothing models for large-scale prediction.

4.5 Comparison of different prediction algorithms

In this section, we compare the prediction performance of SVR to ANN and exponential smoothing. Fig. 4.3 shows a comparison of the performance of the algorithms on the network level. Fig. 4.4 shows the distribution of the prediction error across different horizons. Fig. 4.5 and 4.6 show the performance of the proposed methods for different road categories.

SVR has the smallest MAPE across different prediction horizons for the whole network G (see Fig 4.3a). It also has the smallest SD of error between different links (see Fig. 4.3b). ANN exhibits a slightly larger error as compared to SVR. This can be attributed to the problem of local minima associated with ANN training algorithms [152]. Overall, the prediction performance for all three algorithms degrades as the prediction horizon increases. Performance degradation tends to flatten out especially for data driven methods (SVR, ANN) for large prediction horizons (see Fig. 4.3a, 4.5).

Error distribution plots (see Fig. 4.4) show variations in prediction error across the road network. We observe that the distribution of the prediction error (MAPE) varies from one prediction horizon to another. We also observe more than one peak for each prediction horizon. This implies that there might exist different groups of links with similar prediction performance. We now analyze the prediction behavior of different road categories.

Fig. 4.5 and 4.6 show how the prediction error and standard deviation vary from one road category to another. SVR still provides the lowest MAPE (see Fig. 4.5) and error standard deviation (Fig. 4.6 except for CATA see Fig. 4.6a). As expected expressways are relatively easy to predict as compared to other road categories. This can be verified by comparing the performance of the predictors for CATA roads (expressways; see Fig. 4.5a and 4.6a) with other categories. However, we still observe high SD of error within expressways, especially for large prediction horizons. We find similar patterns for other road categories. Overall SD of error

(a) Mean error for different algorithms.

(b) Error standard deviation within the network.

Figure 4.3: Performance comparison of prediction algorithms for different prediction horizons.

Figure 4.4: Error distribution of algorithms for different prediction horizons.

Figure 4.5: Road category wise performance comparison.

Figure 4.6: Standard deviation of error within each road category.

within different road categories is not that different from network wide SD (see Fig. 4.6 and 4.3b). We observe that roads belonging to the same category can still have dissimilar prediction performance.

Point error measures work well to evaluate and compare prediction performance of different methods. However, they provide little insight into the spatial distribution of the performance. For instance, we are unable to identify which set of links provide worse prediction performance and vice versa (see Fig. 4.7). Moreover, we cannot extract temporal performance patterns across the network and for individual roads.

In the next section, we propose unsupervised learning methods to analyze these trends.

4.6 Spatial and temporal patterns in prediction performance

In this section, we consider the analysis of the prediction performance as a data-mining problem. For this purpose, we apply unsupervised learning methods to find spatial and temporal performance patterns in large-scale prediction. AS SVR exhibited the best average prediction performance, hence, we analyze the efficiency of proposed algorithms by applying them to the prediction data of SVR.

Traffic prediction studies commonly employ measures such as MAPE to evaluate the performance of algorithms [28, 135, 37, 38, 39, 40, 41, 42, 44, 46, 47, 48, 49]. These measures are inadequate to provide any information about the underlying spatial and temporal behavior of prediction algorithms. If $e_s(s_i, k)$ represents the mean prediction error observed for s_i , then the MAPE $e_G(k)$ across the test network G can be written as:

$$e_G(k) = \frac{1}{pd} \sum_{i=1}^p \sum_{j=1}^d \frac{|\hat{z}_k(s_i, t_j) - z(s_i, t_j)|}{z(s_i, t_j)}. \quad (4.9)$$

Figure 4.7: An example of variations in prediction error across the network.

From (4.9), we can see that the errors are averaged across different links and during different time periods. Consider a large network G containing thousands of links and prediction performed for multiple prediction horizons. The prediction error of a particular link s_i at time t_j might be different from that at $t_{j'}$. It might vary from day to day or change during different hours. Similarly, the prediction performance between any two links s_i and s_j may also vary significantly. However, the MAPE (4.9) averages out all these trends. We observed earlier that prediction performance may not remain uniform across large networks (see Fig. 4.3, 4.4, 4.5 and 4.6). We also observed that point error measures provide little detail about these spatial variations. Moreover, these measures do not give any information about temporal performance variations.

The spatial and temporal patterns provide insight about short-term and long-term predictability. Such insights can be highly useful for ITS applications like route guidance and traffic management.

For extracting performance patterns, we consider three components in (4.9), which are space (s_i), time (t_j) and prediction horizon (k). We perform cluster analysis to obtain spatial prediction patterns. This will help to find roads with overall similar predictability across different prediction horizons. To find temporal performance patterns, we combine PCA with k -means clustering. Temporal patterns help us to identify roads with inconsistent prediction performance during different time periods. We also analyze daily and hourly performance patterns for individual links by applying Self Organizing Maps (SOM).

4.6.1 Analysis of spatial prediction patterns

In this section, we apply k -means clustering to find spatial prediction patterns. The method creates different groups (clusters) of road segments. We represent these clusters by labels $\{\omega_i\}_{i=1}^w$. Each group (cluster) contains roads that provide similar prediction performance across different prediction horizons. To compare the links, we represent each link s_i , by a vector $\mathbf{e}_{s_i} = [e_s(s_i, 1) \dots e_s(s_i, k)]^T$, where $e_s(s_i, k)$ is

the MAPE for k^{th} prediction horizon for s_i . The distance measure $\Delta_s(s_i, s_j)$ between the links s_i and s_j is defined as:

$$\Delta_s(s_i, s_j) = \sqrt{(\mathbf{e}_{s_i} - \mathbf{e}_{s_j})^T (\mathbf{e}_{s_i} - \mathbf{e}_{s_j})}. \quad (4.10)$$

We apply unsupervised clustering method as no prior knowledge about the groups of links $\{\omega_i\}_{i=1}^w$ is available. Unsupervised learning approach creates clusters of links depending on their predictability (mean prediction error).

We use k -means clustering to find the roads with similar performance [153]. For k -means clustering we need to specify the number of clusters w beforehand. However, this information is not available for the network G . We require the clusters to be composed of roads with similar performance. Moreover, performance of different clusters should be different from one another. This problem is usually referred as cluster validation [154, 155].

We consider commonly applied cluster validation techniques such as the Silhouette index [156], Hartigan index [157], and homogeneity and separation index [155, 154] in this study.

The Silhouette index $\Psi_{\text{sil}}(s_i)$ for link s_i belonging to cluster ω_j is defined as:

$$\Psi_{\text{sil}}(s_i) = \frac{\beta_2(s_i, \omega'_j) - \beta_1(s_i)}{\max(\beta_1(s_i), \beta_2(s_i, \omega'_j))}, \quad (4.11)$$

where $\beta_1(s_i)$ is the mean distance (in sense of (4.10)) of s_i with other links in the cluster ω_j . In (4.11), $\beta_2(s_i, \omega'_j)$ is the mean distance of s_i with links in the nearest cluster ω'_j . We chose the clustering structure with the highest mean value $\zeta_{\text{sil}}(w)$ such that:

$$\zeta_{\text{sil}}(w) = \frac{1}{p} \sum_{s_i \in G} \Psi_{\text{sil}}(s_i). \quad (4.12)$$

The Hartigan index $\zeta_{\text{har}}(w)$ for data size N is calculated as [158]:

$$\zeta_{\text{har}}(w) = (N - w - 1) \frac{\Omega(w) - \Omega(w + 1)}{\Omega(w + 1)}. \quad (4.13)$$

It considers changes in mean intra-cluster dispersion $\Omega(w)$ due to changes in the number of clusters w [158, 157]. Consider a clustering structure with w clusters and $\{g_i\}_{i=1}^w$ links in each cluster. Intra-cluster dispersion for the structure is given by:

$$\Omega(w) = \sum_{j=1}^w \sum_{i=1}^{g_j} \Delta_s(s_i, c_j)^2, \quad (4.14)$$

where $\{c_i\}_{i=1}^w$ are the cluster centroids.

We use these indices to select the optimal number of clusters. To this end, we require that the indices agree upon on a certain w^* . We will treat the corresponding clustering structure $\{\omega_i\}_{i=1}^{w^*}$ as the best model for the network G .

4.6.2 Analysis of temporal prediction patterns

In this section, we propose methods to infer variations in the prediction performance for different time intervals.

The prediction error for a certain set of links τ_i may not change significantly during different times of the day and across different days. For the other group τ_j prediction performance might vary significantly from one period to another. We refer to these as consistent and inconsistent clusters respectively. We combine PCA and k -means clustering to identify consistent and inconsistent clusters.

We also analyze the temporal performance trends for individual roads. For a given link s_i some days (hours) will have similar performance patterns. We apply SOM to extract these trends.

We now explain our proposed methods to infer network level and link level temporal prediction patterns.

4.6.2.1 Network level temporal prediction patterns

We consider variations in prediction error of the links during different days and hours to group them together. We use Principal Component Analysis (PCA) to

deduce these daily and hourly performance patterns.

We define daily $\{\mathbf{d}_j(s_i) \in \mathbb{R}^{n_d}\}_{j=1}^{m_d}$ and hourly $\{\mathbf{h}_l(s_i) \in \mathbb{R}^{n_h}\}_{l=1}^{m_h}$ performance patterns as follows. The vector $\{\mathbf{d}_j(s_i)\}_{j=1}^{m_d}$ comprises the APE for all the time periods and prediction horizons for that day $\{j\}_{j=1}^{m_d}$ for link s_i . The vector $\{\mathbf{h}_l(s_i)\}_{l=1}^{m_h}$ contains APE across all the days and prediction horizons for the link s_i during the hour $\{l\}_{l=1}^{m_h}$.

The daily variation matrix $\mathbf{D}(s_i) = [\mathbf{d}_1(s_i) \dots \mathbf{d}_{m_d}(s_i)]$ and the hourly variation matrix $\mathbf{H}(s_i) = [\mathbf{h}_1(s_i) \dots \mathbf{h}_{m_h}(s_i)]$ contain such patterns for the link s_i . To quantify performance variations within different days $\{\mathbf{d}_j(s_i)\}_{j=1}^{m_d}$ and hours $\{\mathbf{h}_l(s_i)\}_{l=1}^{m_h}$, we construct corresponding covariance matrices $\Sigma_d(s_i)$ and $\Sigma_h(s_i)$. By centralizing $\{\mathbf{d}_j(s_i)\}_{j=1}^{m_d}$ and $\{\mathbf{h}_l(s_i)\}_{l=1}^{m_h}$ about their means we obtain $\mathbf{D}'(s_i)$ and $\mathbf{H}'(s_i)$ respectively. The covariance matrices $\Sigma_d(s_i)$ and $\Sigma_h(s_i)$ are calculated as follows:

$$\Sigma_d(s_i) = \frac{1}{n_d} \mathbf{D}'(s_i)^T \mathbf{D}'(s_i), \quad (4.15)$$

$$\Sigma_h(s_i) = \frac{1}{n_h} \mathbf{H}'(s_i)^T \mathbf{H}'(s_i). \quad (4.16)$$

Eigenvalue decomposition of covariance matrices will yield:

$$\Sigma_d(s_i) = \mathbf{U}_d(s_i) \Lambda_d(s_i) \mathbf{U}_d(s_i)^T, \quad (4.17)$$

$$\Sigma_h(s_i) = \mathbf{U}_h(s_i) \Lambda_h(s_i) \mathbf{U}_h(s_i)^T, \quad (4.18)$$

where matrices $\{\mathbf{U}_j(s_i) = [\varphi_{j1}(s_i) \dots \varphi_{jm_j}(s_i)]\}_{j \in \{d,h\}}$ and $\{\Lambda_j(s_i)\}_{j \in \{d,h\}}$ contain the normalized eigenvectors and the corresponding eigenvalues of $\{\Sigma_j(s_i)\}_{j \in \{d,h\}}$ respectively. We calculate Principal Components (PC) by rotating the data along the direction of eigenvectors (direction of maximum variance) of the covariance matrix [159]:

$$\mathbf{P}_d(s_i) = \mathbf{D}'(s_i) \mathbf{U}_d(s_i), \quad (4.19)$$

$$\mathbf{P}_h(s_i) = \mathbf{H}'(s_i) \mathbf{U}_h(s_i). \quad (4.20)$$

Each eigenvalue $\lambda_j(s_i)$ represents the amount of variance in the data explained by the corresponding PC $\mathbf{p}_j(s_i)$. For instance, let us consider daily performance patterns. Strongly correlated (pointing in the similar direction) error profiles $\{\mathbf{d}_j(s_i)\}_{j=1}^{m_d}$ for link s_i imply that prediction errors across different days $\{j\}_{j=1}^{m_d}$ follow similar patterns. In this case, few PC $f_d(s_i)$ can cover most of the variance in the daily error performance data $\mathbf{D}'(s_i)$ of s_i [159]. If most days show independent behavior, then we would require more PC $f'_d(s_i)$ to explain the same percentage of variance in data.

The same goes for hourly error patterns. For a link s_i with similar performance across different hours, we require a small number of hourly PC $f_h(s_i)$. In case of large hourly performance variations, we will need a large number of hourly PC to explain the same amount of variance.

The number of PC $\{f_j(s_i)\}_{j \in \{d,h\}}$ are chosen using a certain threshold of total variance η_σ (typically 80%) in the data [159]. We define the following distance measure to compare consistency in prediction performance of two links:

$$\Delta_t(s_i, s_j) = \sqrt{(f_d(s_i) - f_d(s_j))^2 + (f_h(s_i) - f_h(s_j))^2}. \quad (4.21)$$

We find the clusters of consistent τ_1 and inconsistent τ_2 links by applying (4.21) and k -means clustering. Consistent (inconsistent) links will have similar (variable) performance patterns across days and during different hours.

4.6.2.2 Link level temporal prediction patterns

In the previous subsection, we proposed a method to find consistent and inconsistent links. In this section, we propose an algorithm to cluster days/hours with similar performance for each road segment s_i . The algorithm also conserves the topological relation between the clusters. Topological relations are considered in the sense of mean prediction performance of different clusters [160]. To this end, we use Self Organizing Maps (SOM). Self Organizing Maps belong to category of neural networks that can perform unsupervised clustering. In SOM, each cluster is represented by

Table 4.2: Spatial cluster validation

Cluster validation index	Number of optimal clusters (w)
Silhouette index	4
Hartigan index	4
Homogeneity and separation index	4

a neuron. Neurons are organized in a grid pattern $\mathfrak{M}$. The weight $\{\mathbf{q}_\rho\}_{\rho \in \mathfrak{M}}$ of the neuron represents the center of the cluster ρ . We use the Kohonen rule [160] to find the optimal weights (cluster centers).

Consider a road segment s_i with prediction performance matrix $\mathbf{D}(s_i)$. The matrix $\mathbf{D}(s_i)$ is composed of daily prediction error profiles $\{\mathbf{d}_j(s_i)\}_{j=1}^{m_d}$ for m_d days. We represent each day by index $\{j\}_{j=1}^{m_d}$. We aim to identify subset (cluster) of days $\rho \subseteq \{j\}_{j=1}^{m_d}$ with similar performance patterns. Secondly, we aim to find a 2-D grid $\mathfrak{M}$ for clusters. In the grid, clusters $\rho \in \mathfrak{M}$ with similar behavior (daily prediction performance) will be placed adjacent to each other. However, each daily performance profile contains data for multiple prediction horizons and time instances. It is hard to visualize the data in such high dimensional representation. We apply SOM to visualize and map daily performance patterns on a 2-D clustering grid $\mathfrak{M}$ [161].

We apply the same procedure to find different groups of hourly patterns with similar prediction performance for each road segment s_i . To this end, SOM performs clustering by considering hourly profile matrix $\mathbf{H}(s_i)$ for each link s_i .

In this section, we have proposed unsupervised learning methods to find spatial and temporal performance patterns. In the next section, we apply these proposed performance analysis methods to prediction data of SVR and provide results.

4.7 Discussion

Let us start with the spatial performance patterns. We apply k -means clustering to find road segments with similar performance across the prediction horizons. We

Table 4.3: Cluster centers for temporal clusters.

Temporal Cluster (τ)	Cluster Center		Total links
	f_d	f_h	
Cluster 1 (τ_1)	5	6	1,500
Cluster 2 (τ_2)	13	12	3,524

Table 4.4: Distribution of links in temporal clusters.

Temporal Cluster	Spatial Cluster (ω)				Total links
	ω_1	ω_2	ω_3	ω_4	
Cluster 1 (τ_1)	735	610	125	30	1,500
Cluster 2 (τ_2)	66	2,054	1,176	228	3,524

Table 4.5: Performance for different spatial clusters. The MAPE values are in percentage

Cluster	Prediction Horizon							
	5 min	10 min	15 min	20 min	25 min	30 min	45 min	60 min
(ω_1)	2.69	3.24	3.66	3.86	4.05	4.20	4.61	4.89
(ω_2)	6.79	9.16	10.38	10.54	10.64	10.72	10.95	11.14
(ω_3)	11.06	14.60	15.90	16.01	16.05	16.08	16.18	16.28
(ω_4)	17.18	23.04	24.58	24.81	24.98	25.10	25.41	25.62

Figure 4.8: Error distribution for different clusters.

(a) Intra Cluster error standard deviation.

(b) Percentage Distribution of road segments.

(c) Road categories and spatial clusters.

(d) Sections belonging to different clusters at a particular intersection.

Figure 4.9: Properties of spatial clusters. In Fig. 4.9d, blue links (s_b, s_d, s_g) and red links (s_a, s_c, s_e, s_f, s_h) correspond to spatial clusters ω_2 and ω_3 respectively.

Figure 4.10: Road segments belonging to different spatial clusters.

consider three different validation indices to obtain the optimal number of clusters for the test network. All three validation methods yield 4 clusters as the optimal structure (see Table 4.2). The spatial distribution of different clusters is shown in Fig. 4.10. Error distributions for each cluster across different prediction horizons are shown in Fig. 4.8.

The first cluster (ω_1) consists of roads with high prediction accuracy (see Table 4.5). We refer to this group of links as best performing cluster (cluster 1, ω_1). Most of the roads in the network (around 75%, see Fig. 4.9b) belong to clusters 2 (ω_2) and 3 (ω_3). These two clusters represent the performance trends of the majority of roads in the network. It is interesting to note that the combined prediction performance of these two clusters (see Table 4.5) is worse than the mean prediction accuracy of the whole network (see Fig. 4.3a). In this case, the network wise MAPE (see Fig. 4.3a) provides a slightly inflated depiction of overall prediction accuracy. This is due to the high prediction accuracy of the best performing cluster. Finally, roads belonging to cluster 4 (ω_4) have the highest prediction error from each road category (see Table 4.5). The proposed prediction algorithm performs poorly for this set of road segments. We refer to this cluster as the worst performing cluster.

Let us briefly discuss the composition of these spatial clusters. Most of the expressways belong to the best performing cluster (around 80%, see Fig. 4.9c). However, a small group of roads from other categories also belong to this cluster. Interestingly, a small percentage of expressways also appear in other clusters (see Fig. 4.9c). Some expressway sections even appear in cluster 4 (ω_4). The expressway sections belonging to ω_4 are mostly situated at busy exits. Naturally, it is relatively hard to predict traffic on such sections. Moreover, a higher ratio of CATC roads belong to cluster ω_3 , as compared to roads in CATB (see Fig. 4.9c). Likewise, the majority of CATD and CATE roads (referred as others in Fig. 4.9c) also belong to spatial cluster 3.

Overall, roads within each spatial cluster show similar performance (see Fig. 4.8). Consequently, we find small SD of error within each cluster across the prediction

horizons (see Fig. 4.9a). The behavior of the worst performing cluster is an exception in this case.

Spatial clusters can also provide useful information about the relative predictability of different road segments. Consider the intersection shown in Fig. 4.9d. It shows that roads carrying inbound traffic may have different prediction performance as compared to roads carrying outbound traffic. In this case, links carrying traffic towards downtown area (s_a, s_c, s_f, s_h) tend to show degraded prediction performance (cluster ω_3).

We apply PCA and k -means clustering to find temporal performance patterns. We create two clusters for the roads segments in this regard. We refer to these clusters as consistent cluster (τ_1) and inconsistent cluster (τ_2). Table 4.3 and 4.4 summarize the properties of these two clusters. Prediction performance of road segments in consistent cluster remains uniform across days and during different hours. Links in the inconsistent cluster have variable prediction performance during different time periods.

We observe that roads with similar mean prediction performance (see Fig. 4.9a) can still have different temporal performance patterns (see Table 4.4). All these spatial clusters (1, 2, 3) have small intra-cluster SD (see Fig. 4.9a). Majority of roads in spatial cluster 1 (best performing cluster) are part of consistent cluster. However, a small proportion of roads from best performing cluster are also part of inconsistent cluster. We observe this trend in other spatial clusters as well. Although the majority of the links in spatial clusters 2 and 3 are part of inconsistent cluster. Still they both have a sizable proportion of consistent links (see Table 4.4). Even in the case of worst performing cluster, a small subset of roads are part of the consistent cluster. These road segments report high prediction error consistently during most of the time periods.

Temporal performance analysis shows that links with similar overall prediction behavior can still have variable temporal performance. To analyze such trends in details, we focus on two specific links s_1 and s_2 in the network which have the

following properties. They both belong to the same road category (CATA). Furthermore, both of them are from the best performing cluster. However, road segment s_1 is part of consistent cluster and s_2 is from inconsistent cluster. We apply SOM to analyze variations in daily performance patterns of these two links.

Fig. 4.11a and 4.11b show different properties of daily performance patterns for the consistent link s_1 . In Fig. 4.11a, we present the composition of each cluster for s_1 . The entry within each hexagon denotes the number of days belonging to that cluster. Fig. 4.11b shows the relative similarity of each cluster and its neighbors. For the consistent link we find that most of the days fall into four clusters (see Fig. 4.11a). SOM helps us to conserve the topological relations of these clusters. The four main clusters are positioned adjacent to each other (see Fig. 4.11a). This implies that these clusters represent days with similar daily performance patterns (see Fig. 4.11b). For this road segment, we observe similar performance patterns for most of the days (see Fig. 4.11b).

Now let us consider the behavior of inconsistent link s_2 . Fig. 4.12a and 4.12b show the prediction patterns for the road segment. In this case, we find three major clusters. The rest of the days are scattered into other small clusters (see Fig. 4.12a). Even these three clusters represent quite different daily performance patterns (see Fig. 4.12b).

Both of these links belong to same road category and spatial cluster. However, their daily performance patterns are quite different from each other. In case of consistent link we observe that prediction error patterns do not vary significantly on daily basis. For inconsistent link the performance patterns vary significantly, from one day to another.

Traffic prediction can potentially improve the performance of ITS applications [2]. However, these application remain vulnerable to variations in prediction error. Spatial and temporal performance patterns provide insight into prediction behavior of different road segments. Consider the example of a driver advisory system. This system can assign large penalties to links belonging to clusters with large prediction

errors (for instance, see clusters ω_3, ω_4). For instance, with spatial clustering, we can see that traveling through expressway sections belonging to cluster ω_4 (worst performing links) may not be a good idea. By utilizing this information, the planned journey will be less susceptible to the performance of prediction engine. The spatial clusters also serve another important purpose. They provide information about the relative predictability of different road segments in a particular network. Furthermore, temporal clusters can help the algorithm to avoid inconsistent links. These links might have low average prediction error. However, their prediction performance may vary widely from one time instance to another. Again, consider the example of a route guidance algorithm. It would be better to plan a route by incorporating roads with known performance patterns, even if they have slightly larger prediction error than inconsistent roads. The application can utilize these spatial and temporal markers to provide routes which are more robust to variations in prediction performance.

(a) Distribution of different clusters (road segment s_1). (b) Similarity between different clusters (road segment s_1).

Figure 4.11: Daily performance patterns for road segment s_1 (consistent link). Hexagons represent the clusters and the entry within each hexagon denotes the number of days in that cluster. Color bar represents cluster similarity in terms of Euclidean distance.

(a) Distribution of different clusters (road segment s_2). (b) Similarity between different clusters (road segment s_2).

Figure 4.12: Daily performance patterns for road segment s_2 (inconsistent link). Hexagons represent the clusters and the entry within each hexagon denotes the number of days in that cluster. Color bar represents cluster similarity in terms of Euclidean distance.

Chapter 5

Conclusion

5.1 Summary

In this thesis, we studied the problems of data compression and estimation (missing data imputation) for large-scale and heterogenous road networks. We also developed unsupervised learning algorithms to help predictive ITS in reducing the uncertainty caused by variations in prediction performance of data-driven algorithms. For each problem, we proposed algorithms that can model traffic patterns in heterogenous city-scale networks. By contrast, the previous works have mostly addressed the problems of compression, imputation and prediction for small test networks involving a few intersection or expressways. This is a major limiting factor in these studies. Practical traffic management systems are mostly deployed for city-scale networks which are composed of various types of roads. To deal with such scenarios, ITS would require generic algorithms that can extract useful information from sensor data with minimal help from customized models. Secondly, due to their generic nature, the performance of these data-driven models may vary for different road categories and during different days of the week. We have developed models that can solve various problems for ITS deployed for urban road networks. The applications include efficient data compression, missing data estimation and reducing uncertainty of data-driven predictors. our main contributions are as follows:

- In chapter 2, we proposed a two-step compression algorithm for traffic data-sets. In the first step, we obtain a suitable low-dimensional representation of large-scale road networks. To this end, we analyzed various subspace methods in terms of their compression efficiency. In the second step, we performed Huffman coding to keep the residual errors below a certain tolerance level. The algorithm guarantees that the maximum loss of information, for any link and at any time instance, in the decompressed data will remain below a pre-defined tolerance limit. We analyzed the performance of the proposed near-lossless algorithm for various road categories such as expressways, arterial roads, access roads and slip roads. Furthermore, we also assessed the efficiency of the compression algorithm for data collected during weekdays and weekends.
- In chapter 3, we proposed various matrix and tensor completion algorithms to recover missing information in incomplete traffic data-sets. We proposed different methods that can deal with diverse urban networks. We analyzed their performance for different road categories and during different days of the week. We also studied the impact of the choice of latent factors on the estimation efficiency of these algorithms. The suitable low-rank approximation of a road network is a non-trivial problem. Hence, a certain degree of robustness to the choice of rank would be a highly desirable feature for matrix/tensor completion algorithm. Lastly, we also analyzed the variance and bias induced by different algorithms in the recovered data.
- In chapter 4, we studied the problem of traffic prediction in the context of large-scale networks. In previous works, data-driven methods have been shown to achieve high prediction accuracy. However, these studies were mostly done on small custom test networks. Hence the spatial and temporal variations in prediction performance were not considered by these works. Naturally, the performance of predictors may not remain same across different roads. Sec-

only, on certain roads the performance may vary during different parts of the day/week as well. We proposed various unsupervised learning methods that can extract these performance patterns on the network level (high performance links v/s low performance links) as well as on individual link level (performance during weekdays v/s weekends). Such information can prove highly useful for predictive ITS systems in hedging the uncertainty caused by inaccurate predictions.

5.2 Conclusion and future directions

5.2.1 Low-dimensional traffic models

During our analysis, we found some interesting results that can prove helpful in future studies. For instance, we found that considering a large road network as a low-rank matrix/tensor can work as a viable formulation. Our main observation was that traffic conditions across city-state networks are mostly composed of two types of patterns. Granted, there are seasonal variations, daily variations, and week-days/weekends patterns amongst others and due to these patterns it is not advisable to model a large road network as rank-one matrix/tensor. Nonetheless, with sufficient data, generic subspace methods can efficiently model these components in traffic data. For instance, consider the example of peak-hour/off-peak variations. These variations tend to be highly correlated across the network as peak hour congestion would affect most of the roads. Traffic conditions during morning peak hours may or may not be similar to congestion levels during the evening rush hour and vice versa. Nonetheless, these patterns are highly repetitive, can be observed on global (network-wide) level and can explain most of the variation in the data. In fact, these patterns can also be used to validate data from field sensors. The second type of patterns are highly localized patterns that are mostly unique to individual roads (and their immediate neighbors). Developing generic subspace methods that

can efficiently extract these local patterns across large networks still remains an open problem.

We also found that while estimating missing traffic information in case of large-scale networks, robustness of the algorithm (against uncertainty in parameter selection) should be given high importance. We found that different matrix/tensor completion methods tend to provide similar results (some methods proving more accurate than others) if we can estimate the value of rank of the underlying low-dimensional model accurately. However, finding the correct rank value in presence of medium/large amount of missing data is a difficult task.

5.2.2 Combining data-driven prediction with microscopic simulation

In terms of modeling uncertainty in traffic prediction, we found that the location of road can in certain cases play an important role in impacting the predictability of traffic on that particular section. This observation can prove useful when using simulation based approach for modeling traffic. In ITS applications, there are two parallel methodologies to perform short term prediction: data-driven and simulations based models (microscopic and mesoscopic). Simulation based models tend to be highly accurate as they incorporate detailed information about driver behavior, traffic lights, route choice, pedestrian models amongst other factors. However, simulation based models are not scalable. This is where data-driven predictors can prove helpful. The first step would be to identify those parts of the test network where data-driven predictors can achieve high accuracy. The simulator can then ignore those sections and incorporate the information from data-driven predictors as boundary conditions and use that information as additional constraints when calculating the traffic parameters for those sections where data-driven predictors did not perform well.

5.2.3 Models for non-nominal conditions (incidents / extreme weather conditions)

Figure 5.1: Concept map showing the effect of external factors (weather, incidents) on network-wide traffic conditions. Incorporating information about these factors can potentially improve the effectiveness of low-dimensional models.

In this dissertation, we developed models by utilizing a single data source (for instance, traffic speed). External factors such as rain and accidents tend to effect traffic conditions by certain degree as well (see Fig. 5.1). Extending the low-dimensional models presented in this thesis to incorporate these external factors can prove to be highly useful for traffic management systems. To this end, let us start with the example of traffic incidents. Different incidents disrupt traffic in different ways. Major road works and accidents would cause significant traffic diversion to alternative routes thus causing congestion in that part of the network. On the other hand, minor incidents such as car break down or lane blockage may not effect the traffic conditions on the neighboring roads. Similarly, consider the example of a city such

as Singapore. Singapore receives rainfall across the year. However, light showers and moderate amount of rainfall may not have any significant effect on road traffic whereas heavy rainfall can significantly disrupt normal traffic flow. The aim could be to develop generic models for large-scale that can incorporate information about these external factors. One interesting direction in this regard would be to incorporate these external factors (weather / accidents / car breakdowns) in missing data estimation methods. The estimation methods proposed in this thesis rely on nominal patterns to recover missing information. However, traffic on a road may not behave normally during accidents/heavy rainfall-snow/breakdowns. The challenge would be to learn the traffic behavior from incomplete data during such events. An alternative way would be to learn the level of uncertainty due to such events by grouping together data from similar roads and incorporate this information while recovering missing data. In this scenario, tensor completion methods might prove useful as we can create these tensors by combining multiple space-time matrices (one containing incomplete data, the other containing estimated uncertainty due to incident).

5.2.4 Connected vehicles

Recent advances in wireless communication and vehicular sensor technology are leading way for applications that can potentially improve driving and safety conditions on the roads [162, 163]. With the help of these technologies, drivers and traffic management centers would get access to a wealth of data with high granularity. One potential application in this case would be dynamic toll adjustment for traffic management. Many cities across the world are moving towards electronic road pricing (ERP) infrastructure to control congestion in crowded cities. Currently, these systems have pre-defined tolls. With the availability of vehicle to infrastructure (V2I) technologies and high granularity data, predictive modeling applications would allow traffic management centers to adjust the tolls in real time by anticipating short-term future traffic demand.

Bibliography

- [1] J. Zhang, F.-Y. Wang, K. Wang, W.-H. Lin, X. Xu, and C. Chen, “Data-driven intelligent transportation systems: A survey,” *Intelligent Transportation Systems, IEEE Transactions on*, vol. 12, no. 4, pp. 1624–1639, 2011.
- [2] E. I. Vlahogianni, M. G. Karlaftis, and J. C. Golias, “Short-term traffic forecasting: Where we are and where we are going,” *Transportation Research Part C: Emerging Technologies*, 2014.
- [3] Q. Li, H. Jianming, and Z. Yi, “A flow volumes data compression approach for traffic network based on principal component analysis,” in *Intelligent Transportation Systems Conference, 2007. ITSC 2007. IEEE*. IEEE, 2007, pp. 125–130.
- [4] X. Jin, Y. Zhang, and D. Yao, “Simultaneously prediction of network traffic flow based on PCA-SVR,” in *Advances in Neural Networks-ISNN 2007*. Springer, 2007, pp. 1022–1031.
- [5] Y. Xiao, Y.-m. Xie, L. Lu, and S. Gao, “The traffic data compression and decompression for intelligent traffic systems based on two-dimensional wavelet transformation,” in *Signal Processing, 2004. Proceedings. ICSP’04. 2004 7th International Conference on*, vol. 3. IEEE, 2004, pp. 2560–2563.
- [6] T. Djukic, J. van Lint, and S. Hoogendoorn, “Exploring application perspectives of principal component analysis in predicting dynamic origin-destination matrices,” in *Transportation Research Board 91st Annual Meeting*, no. 12-2702, 2012.

- [7] T. Tsekeris and A. Stathopoulos, “Measuring variability in urban traffic flow by use of principal component analysis,” *Journal of Transportation and Statistics*, vol. 9, no. 1, p. 49, 2006.
- [8] X. Xing, X. Zhou, H. Hong, W. Huang, K. Bian, and K. Xie, “Traffic flow decomposition and prediction based on robust principal component analysis.”
- [9] Z.-q. ZHAO, Y. ZHANG, J.-m. HU, and L. Li, “Comparative study of pca and ica based traffic flow compression [j],” *Journal of Highway and Transportation Research and Development*, vol. 11, p. 022, 2008.
- [10] X.-f. Shi and L.-l. Liang, “A data compression method for traffic loop detectors’ signals based on lifting wavelet transformation and entropy coding,” in *Information Science and Engineering (ISISE), 2010 International Symposium on*. IEEE, 2010, pp. 129–132.
- [11] A. Hoffleitner, R. Herring, A. Bayen, Y. Han, F. Moutarde, and A. de La Fortelle, “Large-scale estimation of arterial traffic and structural analysis of traffic patterns from probe vehicles,” in *Transportation Research Board 91st Annual Meeting*, no. 12-0598, 2012.
- [12] L. Li, X. Su, Y. Wang, Y. Lin, Z. Li, and Y. Li, “Robust causal dependence mining in big data network and its application to traffic flow predictions,” *Transportation Research Part C: Emerging Technologies*, 2015.
- [13] J. Ding, Z. Zhang, and X. Ma, “A method for urban traffic data compression based on wavelet-pca,” in *Computational Sciences and Optimization (CSO), 2011 Fourth International Joint Conference on*. IEEE, 2011, pp. 1030–1034.
- [14] T. Djukic, G. Flotterod, H. van Lint, and S. Hoogendoorn, “Efficient real time od matrix estimation based on principal component analysis,” in *Intelligent Transportation Systems (ITSC), 2012 15th International IEEE Conference on*. IEEE, 2012, pp. 115–121.

- [15] Y. Wang, Y. Zhu, Z. He, Y. Yue, and Q. Li, “Challenges and opportunities in exploiting large-scale gps probe data,” *HP Laboratories, Technical Report HPL-2011-109*, vol. 21, 2011.
- [16] H. Tan, G. Feng, J. Feng, W. Wang, Y.-J. Zhang, and F. Li, “A tensor-based method for missing traffic data completion,” *Transportation Research Part C: Emerging Technologies*, vol. 28, pp. 15–27, 2013.
- [17] B. L. Smith, W. T. Scherer, and J. H. Conklin, “Exploring imputation techniques for missing data in transportation management systems,” *Transportation Research Record: Journal of the Transportation Research Board*, vol. 1836, no. 1, pp. 132–142, 2003.
- [18] D. Ni, J. D. Leonard, A. Guin, and C. Feng, “Multiple imputation scheme for overcoming the missing values and variability issues in its data,” *Journal of transportation engineering*, vol. 131, no. 12, pp. 931–938, 2005.
- [19] C. Chen, J. Kwon, J. Rice, A. Skabardonis, and P. Varaiya, “Detecting errors and imputing missing data for single-loop surveillance systems,” *Transportation Research Record: Journal of the Transportation Research Board*, vol. 1855, no. 1, pp. 160–167, 2003.
- [20] M. Zhong, P. Lingras, and S. Sharma, “Estimation of missing traffic counts using factor, genetic, neural, and regression techniques,” *Transportation Research Part C: Emerging Technologies*, vol. 12, no. 2, pp. 139–166, 2004.
- [21] Y. Zhang and Y. Liu, “Missing traffic flow data prediction using least squares support vector machines in urban arterial streets,” in *Computational Intelligence and Data Mining, 2009. CIDM '09. IEEE Symposium on*, March 2009, pp. 76–83.
- [22] L. Qu, L. Li, Y. Zhang, and J. Hu, “Ppca-based missing data imputation for traffic flow volume: A systematical approach,” *Intelligent Transportation Systems, IEEE Transactions on*, vol. 10, no. 3, pp. 512–522, 2009.

- [23] L. Qu, Y. Zhang, J. Hu, L. Jia, and L. Li, “A bpca based missing value imputing method for traffic flow volume data,” in *Intelligent Vehicles Symposium, 2008 IEEE*, June 2008, pp. 985–990.
- [24] L. Li, Y. Li, and Z. Li, “Efficient missing data imputing for traffic flow by considering temporal and spatial dependence,” *Transportation Research Part C: Emerging Technologies*, vol. 34, pp. 108–120, 2013.
- [25] G. Chang and T. Ge, “Comparison of missing data imputation methods for traffic flow,” in *Transportation, Mechanical, and Electrical Engineering (TMEE), 2011 International Conference on*, Dec 2011, pp. 639–642.
- [26] Z. Li, Y. Zhu, H. Zhu, and M. Li, “Compressive sensing approach to urban traffic sensing,” in *Distributed Computing Systems (ICDCS), 2011 31st International Conference on*, 2011, pp. 889–898.
- [27] M. T. Asif, N. Mitrovic, L. Garg, J. Dauwels, and P. Jaillet, “Low-dimensional models for missing data imputation in road networks,” in *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, 2013, pp. 3527–3531.
- [28] W. Min and L. Wynter, “Real-time road traffic prediction with spatio-temporal correlations,” *Transportation Research Part C: Emerging Technologies*, vol. 19, no. 4, pp. 606–616, 2011.
- [29] M. Asif, K. Srinivasan, N. Mitrovic, J. Dauwels, and P. Jaillet, “Near-lossless compression for large traffic networks,” *Intelligent Transportation Systems, IEEE Transactions on*, vol. 16, no. 4, pp. 1817–1826, Aug 2015.
- [30] Y. Han and F. Moutarde, “Analysis of network-level traffic states using locality preservative non-negative matrix factorization,” in *Intelligent Transportation Systems (ITSC), 2011 14th International IEEE Conference on*. IEEE, 2011, pp. 501–506.

- [31] C. Furtlehner, Y. Han, J.-M. Lasgouttes, V. Martin, F. Marchal, and F. Moutarde, "Spatial and temporal analysis of traffic states on large scale networks," in *Intelligent Transportation Systems (ITSC), 2010 13th International IEEE Conference on*, Sept 2010, pp. 1215–1220.
- [32] M. Zhong, S. Sharma, and P. Lingras, "Genetically designed models for accurate imputation of missing traffic counts," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 1879, no. 1, pp. 71–79, 2004.
- [33] Z. Li, Y. Zhu, H. Zhu, and M. Li, "Compressive sensing approach to urban traffic sensing," in *Distributed Computing Systems (ICDCS), 2011 31st International Conference on*. IEEE, 2011, pp. 889–898.
- [34] N. Zhang, F.-Y. Wang, F. Zhu, D. Zhao, and S. Tang, "Dynacas: Computational experiments and decision support for its," *Intelligent Systems, IEEE*, vol. 23, no. 6, pp. 19–23, Nov 2008.
- [35] B. Atasoy, T. Ikeda, X. Song, and M. E. Ben-Akiva, "The concept and impact analysis of a flexible mobility on demand system," *Transportation Research Part C: Emerging Technologies*, vol. 56, pp. 373–392, 2015.
- [36] E. I. Vlahogianni, J. C. Golias, and M. G. Karlaftis, "Short-term traffic forecasting: Overview of objectives and methods," *Transport reviews*, vol. 24, no. 5, pp. 533–557, 2004.
- [37] J. McFadden, W. Yang, and S. Durrans, "Application of artificial neural networks to predict speeds on two-lane rural highways," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 1751, no. -1, pp. 9–17, 2001.
- [38] C. Wu, J. Ho, and D. Lee, "Travel-time prediction with support vector regression," *Intelligent Transportation Systems, IEEE Transactions on*, vol. 5, no. 4, pp. 276–281, 2004.

- [39] C. van Hinsbergen, J. Van Lint, and H. Van Zuylen, “Bayesian committee of neural networks to predict travel times with confidence intervals,” *Transportation Research Part C: Emerging Technologies*, vol. 17, no. 5, pp. 498–509, 2009.
- [40] D. Park and L. Rilett, “Forecasting freeway link travel times with a multi-layer feedforward neural network,” *Computer-Aided Civil and Infrastructure Engineering*, vol. 14, no. 5, pp. 357–367, 1999.
- [41] L. Vanajakshi and L. Rilett, “Support vector machine technique for the short term prediction of travel time,” in *Intelligent Vehicles Symposium, 2007 IEEE*. IEEE, 2007, pp. 600–605.
- [42] E. Vlahogianni, M. Karlaftis, and J. Golias, “Optimized and meta-optimized neural networks for short-term traffic flow prediction: A genetic approach,” *Transportation Research Part C: Emerging Technologies*, vol. 13, no. 3, pp. 211–234, 2005.
- [43] B. Smith and M. Demetsky, “Short-term traffic flow prediction models-a comparison of neural network and nonparametric regression approaches,” in *Systems, Man, and Cybernetics, 1994. 'Humans, Information and Technology'. 1994 IEEE International Conference on*, vol. 2. IEEE, 1994, pp. 1706–1709.
- [44] Y. Zhang and Y. Liu, “Traffic forecasting using least squares support vector machines,” *Transportmetrica*, vol. 5, no. 3, pp. 193–213, 2009.
- [45] M. Dougherty and M. Cobbett, “Short-term inter-urban traffic forecasts using neural networks,” *International journal of forecasting*, vol. 13, no. 1, pp. 21–31, 1997.
- [46] A. Ding, X. Zhao, and L. Jiao, “Traffic flow time series prediction based on statistics learning theory,” in *Intelligent Transportation Systems, 2002. Proceedings. The IEEE 5th International Conference on*. IEEE, 2002, pp. 727–730.

- [47] L. Vanajakshi and L. Rilett, “A comparison of the performance of artificial neural networks and support vector machines for the prediction of traffic speed,” in *Intelligent Vehicles Symposium, 2004 IEEE*. Ieee, 2004, pp. 194–199.
- [48] J. Park, D. Li, Y. Murphey, J. Kristinsson, R. McGee, M. Kuang, and T. Phillips, “Real time vehicle speed prediction using a neural network traffic model,” in *Neural Networks (IJCNN), The 2011 International Joint Conference on*. IEEE, 2011, pp. 2991–2996.
- [49] C. Quek, M. Pasquier, and B. Lim, “Pop-traffic: A novel fuzzy neural approach to road traffic analysis and prediction,” *Intelligent Transportation Systems, IEEE Transactions on*, vol. 7, no. 2, pp. 133–146, 2006.
- [50] J. Wang and Q. Shi, “Short-term traffic speed forecasting hybrid model based on chaos-wavelet analysis-support vector machine theory,” *Transportation Research Part C: Emerging Technologies*, vol. 27, pp. 219–232, 2013.
- [51] G. Gopi, J. Dauwels, M. Asif, S. Ashwin, N. Mitrovic, U. Rasheed, and P. Jaillet, “Bayesian support vector regression for traffic speed prediction with error bars,” in *Intelligent Transportation Systems - (ITSC), 2013 16th International IEEE Conference on*, Oct 2013, pp. 136–141.
- [52] M. Castro-Neto, Y.-S. Jeong, M.-K. Jeong, and L. D. Han, “Online-svr for short-term traffic flow prediction under typical and atypical traffic conditions,” *Expert systems with applications*, vol. 36, no. 3, pp. 6164–6173, 2009.
- [53] H. Su, L. Zhang, and S. Yu, “Short-term traffic flow prediction based on incremental support vector regression,” in *Natural Computation, 2007. ICNC 2007. Third International Conference on*, vol. 1, Aug 2007, pp. 640–645.
- [54] W.-C. Hong, “Traffic flow forecasting by seasonal svr with chaotic simulated annealing algorithm,” *Neurocomputing*, vol. 74, no. 12, pp. 2096–2107, 2011.

- [55] ———, “Application of seasonal svr with chaotic immune algorithm in traffic flow forecasting,” *Neural Computing and Applications*, vol. 21, no. 3, pp. 583–593, 2012.
- [56] E. I. Vlahogianni, “Enhancing predictions in signalized arterials with information on short-term traffic flow dynamics,” *Journal of Intelligent Transportation Systems*, vol. 13, no. 2, pp. 73–84, 2009.
- [57] C.-S. Li and M.-C. Chen, “Identifying important variables for predicting travel time of freeway with non-recurrent congestion with neural networks,” *Neural Computing and Applications*, vol. 23, no. 6, pp. 1611–1629, 2013.
- [58] S. Dunne and B. Ghosh, “Regime-based short-term multivariate traffic condition forecasting algorithm,” *Journal of Transportation Engineering*, vol. 138, no. 4, pp. 455–466, 2011.
- [59] G. Yu, J. Hu, C. Zhang, L. Zhuang, and J. Song, “Short-term traffic flow forecasting based on markov chain model,” in *Intelligent Vehicles Symposium, 2003. Proceedings. IEEE*. IEEE, 2003, pp. 208–212.
- [60] Y.-A. Daraghmi, C.-W. Yi, and T.-C. Chiang, “Negative binomial additive models for short-term traffic flow forecasting in urban areas,” *Intelligent Transportation Systems, IEEE Transactions on*, vol. 15, no. 2, pp. 784–793, 2014.
- [61] R. Tahmasbi and S. M. Hashemi, “Modeling and forecasting the urban volume using stochastic differential equations,” *Intelligent Transportation Systems, IEEE Transactions on*, vol. 15, no. 1, pp. 250–259, 2014.
- [62] L. Li, X. Chen, and L. Zhang, “Multimodel ensemble for freeway traffic state estimations,” *Intelligent Transportation Systems, IEEE Transactions on*, vol. 15, no. 3, pp. 1323–1336, 2014.

- [63] Y. Xie, Y. Zhang, and Z. Ye, “Short-term traffic volume forecasting using kalman filter with discrete wavelet decomposition,” *Computer-Aided Civil and Infrastructure Engineering*, vol. 22, no. 5, pp. 326–334, 2007.
- [64] Y. Wang, J. H. Van Schuppen, and J. Vrancken, “Prediction of traffic flow at the boundary of a motorway network,” *Intelligent Transportation Systems, IEEE Transactions on*, vol. 15, no. 1, pp. 214–227, 2014.
- [65] B. M. Williams and L. A. Hoel, “Modeling and forecasting vehicular traffic flow as a seasonal arima process: Theoretical basis and empirical results,” *Journal of Transportation Engineering*, vol. 129, no. 6, pp. 664–672, 2003.
- [66] M. Lippi, M. Bertini, and P. Frasconi, “Short-term traffic flow forecasting: An experimental comparison of time-series analysis and supervised learning,” *Intelligent Transportation Systems, IEEE Transactions on*, vol. 14, no. 2, pp. 871–882, 2013.
- [67] —, “Collective traffic forecasting,” in *Machine Learning and Knowledge Discovery in Databases*. Springer, 2010, pp. 259–273.
- [68] W. Liu, Y. Zheng, S. Chawla, J. Yuan, and X. Xing, “Discovering spatio-temporal causal interactions in traffic data streams,” in *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2011, pp. 1010–1018.
- [69] J. Dauwels, A. Aslam, M. T. Asif, X. Zhao, N. M. Vie, A. Cichocki, and P. Jaillet, “Predicting traffic speed in urban transportation subnetworks for multiple horizons,” in *Control Automation Robotics Vision (ICARCV), 2014 13th International Conference on*, Dec 2014, pp. 547–552.
- [70] L. Safonov, E. Tomer, V. Strygin, Y. Ashkenazy, and S. Havlin, “Multifractal chaotic attractors in a system of delay-differential equations modeling road traffic,” *Chaos*, vol. 12, no. 4, pp. 1006–1014, 2002.

- [71] M. P. Miska, *Microscopic online simulation for real-time traffic management*. TU Delft, Delft University of Technology, 2007.
- [72] J. Barceló, J. Casas, J. Ferrer, and D. García, “Modelling advanced transport telematic applications with microscopic simulators: The case of aimsun2,” in *Traffic and Mobility*. Springer, 1999, pp. 205–221.
- [73] Q. Yang and H. N. Koutsopoulos, “A microscopic traffic simulator for evaluation of dynamic traffic management systems,” *Transportation Research Part C: Emerging Technologies*, vol. 4, no. 3, pp. 113–129, 1996.
- [74] G. D. Cameron and G. I. Duncan, “Paramicsparallel microscopic simulation of road traffic,” *The Journal of Supercomputing*, vol. 10, no. 1, pp. 25–53, 1996.
- [75] M. Schreckenberg, “Online traffic simulation based on cellular automata,” *Human-Machine-Interface Implications*, p. 196, 1997.
- [76] M. Behrisch, L. Bieker, J. Erdmann, and D. Krajzewicz, “Sumo—simulation of urban mobility,” in *The Third International Conference on Advances in System Simulation (SIMUL 2011)*, Barcelona, Spain, 2011.
- [77] M. Fellendorf and P. Vortisch, “Validation of the microscopic traffic flow model vissim in different real-world situations,” in *Transportation Research Board 80th Annual Meeting*, 2001.
- [78] M. Fellendorf, “Vissim: A microscopic simulation tool to evaluate actuated signal control including bus priority,” in *64th Institute of Transportation Engineers Annual Meeting*, 1994, pp. 1–9.
- [79] S. Ma, D. Goldfarb, and L. Chen, “Fixed point and bregman iterative methods for matrix rank minimization,” *Mathematical Programming*, vol. 128, no. 1-2, pp. 321–353, 2011.
- [80] A. Ilin and T. Raiko, “Practical approaches to principal component analysis in the presence of missing values,” *The Journal of Machine Learning Research*, vol. 11, pp. 1957–2000, 2010.

- [81] C. Y. Goh, J. Dauwels, N. Mitrovic, M. T. Asif, A. Oran, and P. Jaillet, “Online map-matching based on hidden markov model for real-time traffic sensing applications,” in *Intelligent Transportation Systems (ITSC), 2012 15th International IEEE Conference on*, 2012, pp. 776–781.
- [82] N. Mitrovic, M. Asif, J. Dauwels, and P. Jaillet, “Low-dimensional models for compressed sensing and prediction of large-scale traffic data,” *Intelligent Transportation Systems, IEEE Transactions on*, vol. 16, no. 5, pp. 2949–2954, Oct 2015.
- [83] W. L. Leong, K. Lee, K. Swapnil, X. Li, H. Y. Victor, N. Mitrovic, M. T. Asif, J. Dauwels, and P. Jaillet, “Improving traffic prediction by including rainfall data,” in *14th Intelligent Transportation Systems Asia-Pacific Forum*, 2015.
- [84] S. Jere, J. Dauwels, M. T. Asif, N. M. Vie, A. Cichocki, and P. Jaillet, “Extracting commuting patterns in railway networks through matrix decompositions,” in *Control Automation Robotics Vision (ICARCV), 2014 13th International Conference on*, Dec 2014, pp. 541–546.
- [85] R. Ansar, P. Sarampakhul, S. Ghosh, N. Mitrovic, M. Asif, J. Dauwels, and P. Jaillet, “Evaluation of smart-phone performance for real-time traffic prediction,” in *Intelligent Transportation Systems (ITSC), 2014 IEEE 17th International Conference on*, Oct 2014, pp. 3010–3015.
- [86] D. Mohan, M. Asif, N. Mitrovic, J. Dauwels, and P. Jaillet, “Wavelets on graphs with application to transportation networks,” in *Intelligent Transportation Systems (ITSC), 2014 IEEE 17th International Conference on*, Oct 2014, pp. 1707–1712.
- [87] N. Mitrovic, M. Tayyab Asif, J. Dauwels, and P. Jaillet, “Compressed prediction of large-scale urban traffic,” in *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*, May 2014, pp. 5984–5988.

- [88] N. Mitrovic, M. Asif, U. Rasheed, J. Dauwels, and P. Jaillet, “Cur decomposition for compression and compressed sensing of large-scale traffic data,” in *Intelligent Transportation Systems - (ITSC), 2013 16th International IEEE Conference on*, Oct 2013, pp. 1475–1480.
- [89] P. J. Bickel, C. Chen, J. Kwon, J. Rice, E. Van Zwet, and P. Varaiya, “Measuring traffic,” *Statistical Science*, vol. 22, no. 4, pp. 581–597, 2007.
- [90] S. Reddy, M. Mun, J. Burke, D. Estrin, M. Hansen, and M. Srivastava, “Using mobile phones to determine transportation modes,” *ACM Transactions on Sensor Networks (TOSN)*, vol. 6, no. 2, p. 13, 2010.
- [91] A. Hofleitner, R. Herring, P. Abbeel, and A. Bayen, “Learning the dynamics of arterial traffic from probe data using a dynamic bayesian network,” *Intelligent Transportation Systems, IEEE Transactions on*, vol. 13, no. 4, pp. 1679–1693, 2012.
- [92] M. T. Asif, J. Dauwels, C. Y. Goh, A. Oran, E. Fathi, M. Xu, M. Dhanya, N. Mitrovic, and P. Jaillet, “Spatiotemporal patterns in large-scale traffic speed prediction,” *Intelligent Transportation Systems, IEEE Transactions on*, vol. 15, no. 2, pp. 794–804, April 2014.
- [93] L. Li, D. Wen, and D. Yao, “A survey of traffic control with vehicular communications,” *Intelligent Transportation Systems, IEEE Transactions on*, vol. PP, no. 99, pp. 1–8, 2013.
- [94] R. Herring, A. Hofleitner, P. Abbeel, and A. Bayen, “Estimating arterial traffic conditions using sparse probe data,” in *Intelligent Transportation Systems (ITSC), 2010 13th International IEEE Conference on*. IEEE, 2010, pp. 929–936.
- [95] J. Ribeiro Junior, M. Mitre Campista, and L. Costa, “Cotrams: A collaborative and opportunistic traffic monitoring system,” *Intelligent Transportation Systems, IEEE Transactions on*, vol. 15, no. 3, pp. 949–958, June 2014.

- [96] J. Rodrigues, A. Aguiar, F. Vieira, J. Barros, and J. P. S. Cunha, “A mobile sensing architecture for massive urban scanning,” in *Intelligent Transportation Systems (ITSC), 2011 14th International IEEE Conference on*, Oct 2011, pp. 1132–1137.
- [97] C. Furtlehner, Y. Han, J.-M. Lasgouttes, V. Martin, F. Marchal, and F. Moutarde, “Spatial and temporal analysis of traffic states on large scale networks,” in *Intelligent Transportation Systems (ITSC), 2010 13th International IEEE Conference on*. IEEE, 2010, pp. 1215–1220.
- [98] G. Ke, J. Hu, L. He, and Z. Li, “Data reduction in urban traffic sensor network,” in *Intelligent Vehicles Symposium, 2009 IEEE*, June 2009, pp. 1021–1026.
- [99] M. T. Asif, S. Kannan, J. Dauwels, and P. Jaillet, “Data compression techniques for urban traffic data,” in *Computational Intelligence in Vehicles and Transportation Systems (CIVTS), 2013 IEEE Symposium on*, 2013, pp. 44–49.
- [100] F. Qiao, H. Liu, and L. Yu, “Incorporating wavelet decomposition technique to compress transguide intelligent transportation system data,” *Transportation Research Record: Journal of the Transportation Research Board*, vol. 1968, no. 1, pp. 63–74, 2006.
- [101] —, “Intelligent transportation systems data compression using wavelet decomposition technique,” Tech. Rep., 2009.
- [102] <http://www.mytransport.sg/content/mytransport/home/dataMall.html>, accessed: 2015-10-25.
- [103] <http://www.lta.gov.sg/ltaacademy/doc/IS02-p07%20TrafficScan.pdf>, accessed: 2015-10-25.
- [104] N. Thomos, N. V. Boulgouris, and M. G. Strintzis, “Optimized transmission of jpeg2000 streams over wireless channels,” *Image Processing, IEEE Transactions on*, vol. 15, no. 1, pp. 54–67, 2006.

- [105] M. Antonini, M. Barlaud, P. Mathieu, and I. Daubechies, “Image coding using wavelet transform,” *Image Processing, IEEE Transactions on*, vol. 1, no. 2, pp. 205–220, 1992.
- [106] R. W. Buccigrossi and E. P. Simoncelli, “Image compression via joint statistical characterization in the wavelet domain,” *Image Processing, IEEE Transactions on*, vol. 8, no. 12, pp. 1688–1701, 1999.
- [107] K. Srinivasan, J. Dauwels, and M. Reddy, “Multichannel EEG compression: Wavelet-based image and volumetric coding approach,” *Biomedical and Health Informatics, IEEE Journal of*, vol. 17, no. 1, pp. 113–120, 2013.
- [108] J. Dauwels, K. Srinivasan, M. Reddy, and A. Cichocki, “Near-lossless multichannel EEG compression based on matrix and tensor decompositions,” *Biomedical and Health Informatics, IEEE Journal of*, vol. 17, no. 3, pp. 708–714, 2013.
- [109] M. W. Berry, M. Browne, A. N. Langville, V. P. Pauca, and R. J. Plemmons, “Algorithms and applications for approximate nonnegative matrix factorization,” *Computational Statistics & Data Analysis*, vol. 52, no. 1, pp. 155–173, 2007.
- [110] D. Donoho and V. Stodden, “When does non-negative matrix factorization give a correct decomposition into parts?” in *Advances in Neural Information Processing Systems*, 2004, pp. 1141–1148.
- [111] L. De Lathauwer, B. De Moor, and J. Vandewalle, “On the best rank-1 and rank-($r_1, r_2, \dots, r_n$) approximation of higher-order tensors,” *SIAM Journal on Matrix Analysis and Applications*, vol. 21, no. 4, pp. 1324–1342, 2000.
- [112] L. De Lathauwer and J. Vandewalle, “Dimensionality reduction in higher-order signal processing and rank-($r_1, r_2, \dots, r_n$) reduction in multilinear algebra,” *Linear Algebra and its Applications*, vol. 391, pp. 31–55, 2004.

- [113] T. Kolda and B. Bader, “Tensor decompositions and applications,” *SIAM Review*, vol. 51, no. 3, pp. 455–500, 2009.
- [114] T. T. Tchrakian, B. Basu, and M. O’Mahony, “Real-time traffic flow forecasting using spectral analysis,” *Intelligent Transportation Systems, IEEE Transactions on*, vol. 13, no. 2, pp. 519–526, 2012.
- [115] X. Jiang and H. Adeli, “Dynamic wavelet neural network model for traffic flow forecasting,” *Journal of Transportation Engineering*, vol. 131, no. 10, pp. 771–779, 2005.
- [116] Y. Zhang, Y. Zhang, and A. Haghani, “A hybrid short-term traffic flow forecasting method based on spectral analysis and statistical volatility model,” *Transportation Research Part C: Emerging Technologies*, 2013.
- [117] K. Sayood, *Introduction to data compression*. Access Online via Elsevier, 2012.
- [118] T. Qu, S. T. Parker, Y. Cheng, B. Ran, and D. A. Noyce, “Large-scale intelligent transportation system traffic detector data archiving,” in *Transportation Research Board 93rd Annual Meeting*, no. 14-5448, 2014.
- [119] I. Vilajosana, J. Llosa, B. Martinez, M. Domingo-Prieto, A. Angles, and X. Vilajosana, “Bootstrapping smart cities through a self-sustainable model based on big data flows,” *Communications Magazine, IEEE*, vol. 51, no. 6, pp. 128–134, June 2013.
- [120] H. Jagadish, J. Gehrke, A. Labrinidis, Y. Papakonstantinou, J. M. Patel, R. Ramakrishnan, and C. Shahabi, “Big data and its technical challenges,” *Communications of the ACM*, vol. 57, no. 7, pp. 86–94, 2014.
- [121] P.-A. Laharotte, R. Billot, E. Come, L. Oukhellou, A. Nantes, and N.-E. El Faouzi, “Spatiotemporal analysis of bluetooth data: Application to a large urban network,” *Intelligent Transportation Systems, IEEE Transactions on*, vol. PP, no. 99, pp. 1–10, 2014.

- [122] B. Płaczek, “Selective data collection in vehicular networks for traffic control applications,” *Transportation Research Part C: Emerging Technologies*, vol. 23, pp. 14–28, 2012.
- [123] T. Hunter, T. Das, M. Zaharia, P. Abbeel, and A. Bayen, “Large-scale estimation in cyberphysical systems using streaming data: A case study with arterial traffic estimation,” *Automation Science and Engineering, IEEE Transactions on*, vol. 10, no. 4, pp. 884–898, Oct 2013.
- [124] T. Cheng, G. Tanaksaranond, C. Brunsdon, and J. Haworth, “Exploratory visualisation of congestion evolutions on urban transport networks,” *Transportation Research Part C: Emerging Technologies*, vol. 36, pp. 296–306, 2013.
- [125] T. Hunter, T. Moldovan, M. Zaharia, S. Merzgui, J. Ma, M. J. Franklin, P. Abbeel, and A. M. Bayen, “Scaling the mobile millennium system in the cloud,” in *Proceedings of the 2Nd ACM Symposium on Cloud Computing*, ser. SOCC ’11. New York, NY, USA: ACM, 2011, pp. 28:1–28:8.
- [126] J. Conklin and B. Smith, “The use of local lane distribution patterns for the estimation of missing data in transportation management systems,” *Transportation Research Record*, vol. 1811, pp. 50–56, 2002.
- [127] C. Chen, Y. Wang, L. Li, J. Hu, and Z. Zhang, “The retrieval of intra-day trend and its influence on traffic prediction,” *Transportation research part C: emerging technologies*, vol. 22, pp. 103–118, 2012.
- [128] S. Sharma, P. Lingras, and M. Zhong, “Effect of missing values estimations on traffic parameters,” *Transportation Planning and Technology*, vol. 27, no. 2, pp. 119–144, 2004.
- [129] E. Acar, D. M. Dunlavy, T. G. Kolda, and M. Mørup, “Scalable tensor factorizations for incomplete data,” *Chemometrics and Intelligent Laboratory Systems*, vol. 106, no. 1, pp. 41–56, 2011.

- [130] A. Cichocki and S.-i. Amari, *Adaptive blind signal and image processing: learning algorithms and applications*. John Wiley & Sons, 2002, vol. 1.
- [131] T. Wiberg, “Computation of principal components when data are missing,” in *Proc. of Second Symp. Computational Statistics*, 1976, pp. 229–236.
- [132] B. Recht, M. Fazel, and P. A. Parrilo, “Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization,” *SIAM review*, vol. 52, no. 3, pp. 471–501, 2010.
- [133] E. J. Candès and B. Recht, “Exact matrix completion via convex optimization,” *Foundations of Computational mathematics*, vol. 9, no. 6, pp. 717–772, 2009.
- [134] S. Yang, K. Kalpakis, and A. Biem, “Spatio-temporal coupled bayesian robust principal component analysis for road traffic event detection,” in *Intelligent Transportation Systems - (ITSC), 2013 16th International IEEE Conference on*, Oct 2013, pp. 392–398.
- [135] B. Smith, B. Williams, and R. Keith Oswald, “Comparison of parametric and nonparametric models for traffic flow forecasting,” *Transportation Research Part C: Emerging Technologies*, vol. 10, no. 4, pp. 303–321, 2002.
- [136] Z. Sun, Y. Wang, and J. Pan, “Short-term traffic flow forecasting based on clustering and feature selection,” in *Neural Networks, 2008. IJCNN 2008.(IEEE World Congress on Computational Intelligence). IEEE International Joint Conference on*. IEEE, 2008, pp. 577–583.
- [137] M. Van Der Voort, M. Dougherty, and S. Watson, “Combining kohonen maps with arima time series models to forecast traffic flow,” *Transportation Research Part C: Emerging Technologies*, vol. 4, no. 5, pp. 307–318, 1996.
- [138] P. Lingras and P. Mountford, “Time delay neural networks designed using genetic algorithms for short term inter-city traffic forecasting,” in *Engineering of Intelligent Systems*. Springer, 2001, pp. 290–299.

- [139] J. Zhu and T. Zhang, “A layered neural network competitive algorithm for short-term traffic forecasting,” in *Computational Intelligence and Software Engineering, 2009. CiSE 2009. International Conference on*. IEEE, 2009, pp. 1–4.
- [140] A. Smola and B. Schölkopf, “A tutorial on support vector regression,” *Statistics and computing*, vol. 14, no. 3, pp. 199–222, 2004.
- [141] C. Chang and C. Lin, “Training v-support vector regression: theory and algorithms,” *Neural Computation*, vol. 14, no. 8, pp. 1959–1977, 2002.
- [142] S. J. Pan and Q. Yang, “A survey on transfer learning,” *Knowledge and Data Engineering, IEEE Transactions on*, vol. 22, no. 10, pp. 1345–1359, 2010.
- [143] S. Sundaram, H. N. Koutsopoulos, M. Ben-Akiva, C. Antoniou, and R. Balakrishna, “Simulation-based dynamic traffic assignment for short-term planning applications,” *Simulation Modelling Practice and Theory*, vol. 19, no. 1, pp. 450–462, 2011.
- [144] P. Moriarty and D. Honnery, “Low-mobility: The future of transport,” *Futures*, vol. 40, no. 10, pp. 865–872, 2008.
- [145] K. Müller, A. Smola, G. Rätsch, B. Schölkopf, J. Kohlmorgen, and V. Vapnik, “Predicting time series with support vector machines,” *Artificial Neural Networks ICANN’97*, pp. 999–1004, 1997.
- [146] B. Schölkopf, P. Bartlett, A. Smola, and R. Williamson, “Shrinking the tube: a new support vector regression algorithm,” *Advances in neural information processing systems*, pp. 330–336, 1999.
- [147] C. Chang and C. Lin, “Libsvm: a library for support vector machines,” *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 2, no. 3, p. 27, 2011.

- [148] S. Keerthi and C. Lin, “Asymptotic behaviors of support vector machines with gaussian kernel,” *Neural computation*, vol. 15, no. 7, pp. 1667–1689, 2003.
- [149] R. Frank, N. Davey, and S. Hunt, “Time series prediction and neural networks,” *Journal of Intelligent & Robotic Systems*, vol. 31, no. 1, pp. 91–103, 2001.
- [150] H. Chen and S. Grant-Muller, “Use of sequential learning for short-term traffic flow forecasting,” *Transportation Research Part C: Emerging Technologies*, vol. 9, no. 5, pp. 319–336, 2001.
- [151] B. Williams, P. Durvasula, and D. Brown, “Urban freeway traffic flow prediction: application of seasonal autoregressive integrated moving average and exponential smoothing models,” *Transportation Research Record: Journal of the Transportation Research Board*, vol. 1644, no. -1, pp. 132–141, 1998.
- [152] M. Gori and A. Tesi, “On the problem of local minima in backpropagation,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, no. 1, pp. 76–86, 1992.
- [153] T. Kanungo, D. Mount, N. Netanyahu, C. Piatko, R. Silverman, and A. Wu, “An efficient k-means clustering algorithm: Analysis and implementation,” *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 24, no. 7, pp. 881–892, 2002.
- [154] K. Wang, B. Wang, and L. Peng, “Cvap: Validation for cluster analyses,” *Data Science Journal*, no. 0, p. 904220071, 2009.
- [155] R. Sharan, A. Maron-Katz, and R. Shamir, “Click and expander: a system for clustering and visualizing gene expression data,” *Bioinformatics*, vol. 19, no. 14, pp. 1787–1799, 2003.
- [156] P. Rousseeuw, “Silhouettes: a graphical aid to the interpretation and validation of cluster analysis,” *Journal of computational and applied mathematics*, vol. 20, pp. 53–65, 1987.

- [157] J. Hartigan, *Clustering algorithms*. John Wiley & Sons, Inc., 1975.
- [158] I. Pelczer and H. Cisneros-Iturbe, “Identification of rainfall patterns over the valley of mexico,” in *Proceedings of 11th International Conference of Urban Drainage, Edinburgh, Scotland, UK*, 2008.
- [159] I. Jolliffe, *Principal component analysis*. Wiley Online Library, 2005.
- [160] T. Kohonen, “The self-organizing map,” *Proceedings of the IEEE*, vol. 78, no. 9, pp. 1464–1480, 1990.
- [161] B. S. Penn, “Using self-organizing maps to visualize high-dimensional data,” *Computers & Geosciences*, vol. 31, no. 5, pp. 531–544, 2005.
- [162] Y. Ma, M. Chowdhury, A. Sadek, and M. Jelihani, “Integrated traffic and communication performance evaluation of an intelligent vehicle infrastructure integration (vii) system for online travel-time prediction,” *Intelligent Transportation Systems, IEEE Transactions on*, vol. 13, no. 3, pp. 1369–1382, 2012.
- [163] —, “Real-time highway traffic condition assessment framework using vehicle–infrastructure integration (vii) with artificial intelligence (ai),” *Intelligent Transportation Systems, IEEE Transactions on*, vol. 10, no. 4, pp. 615–627, 2009.