Robust Freeway Travel time Prediction with State-Space Neural Networks
A Recurrent Neural Network Approach

TRAIL Research School, Delft, November 2002

Authors
Ir. Hans J.W.C. van Lint, Dr. ir. S.P. Hoogendoorn, Prof. dr. H.J. van Zuylen
Faculty of Civil Engineering and Geosciences, Delft University of Technology, Transportation Planning and Traffic Engineering

Abstract

Robustness to missing or faulty input, due to failures in the data collection system, is a key characteristic for any travel time prediction model that is to be applied in a real time environment. Previous research by van Lint et al (2002) has shown that so-called State-Space Neural Networks (SSNN) are capable of accurately predicting experienced travel times. Our paper shows that incorporating corrupt data into the training procedure does increase the robustness of these SSNN models, but at the cost of predictive performance: there is a clear trade off between robustness and model accuracy. More over, inclusion of (small) amounts of corrupted data in the training procedure makes the internal states of the SSNN model, which are closely related to the expected traffic conditions, more difficult to interpret.
Because of the SSNN's intrinsic robustness to small deviations in the input, application of pre-processing (input-repairing) strategies before feeding the input into the model does lead to a robust but still accurate model. The combined framework of pre-processing and SSNN model is robust to various kinds of input failure, even though the proposed pre-processing strategies are naïve non-parameterized procedures such as exponential smoothing and spatial interpolation. Further research should emphasize on enhancing the proposed pre-processing procedures and applying the travel time prediction framework in real time.

Keywords

robustness, travel time prediction, state-space neural networks, missing values

There is an increasing need for systems that can provide road-users with accurate real time traffic information. Many research efforts emphasize the significance of traffic information and the potential effect of advanced traveler information systems (ATIS) on travel choice. In (van Berkum, van der Mede, 1993) the *credibility* and the *net value* of the information is argued to influence the traveler's response to the information presented. *Net value* is defined by the difference between the information and individual expectations, implying that different drivers value the same information differently. The *credibility of information* strongly relates to the *quality* (e.g. accuracy) of the information. Drivers tend to make less use of the information, once they have had bad experiences with it. Thus, for traffic information to have an effect on driver behavior, it should consist of clear and unambiguous messages based on accurate and reliable predictions. With respect to the accuracy and reliability of traffic information, the sensitivity of the quality of that information to faulty and missing input data is of particular interest.

So-called State-Space Neural Networks (SSNN) are capable of accurately predicting experienced travel times, producing approximately zero mean normally distributed residuals, generally not outside a range of 10% of the real expected travel times (van Lint et. al., 2002). These results, however, are obtained by feeding the models with 100% accurate and reliable data. The input data to travel time prediction models, collected by a *real time* traffic monitoring system, will often consist of corrupted or missing values. In illustration: Let us note that on average 15% of the inductive loops of the Dutch Freeway monitoring system (MONICA) may be out of operation or producing unreliable measurements[1].

The goal of this paper is therefore to present a travel time prediction framework, based on the SSNN model, which is robust to various kinds of input failure, as well as to noisy and possibly biased input. Robustness in this context is defined as the capability of the SSNN based framework to produce accurate predictions under all kinds and degrees of input failure. To this end we first give a brief overview of methods for dealing with data-corruption from literature, next we present a classification of different kinds of input failure (incidental, structural and intrinsic). Subsequently, we propose various pre-processing and training strategies, and finally, we critically discuss results and present conclusions and recommendations for further research.

[1] Statistic from a week of 1 minute aggregate measurements of detectors on the A13 highway between Den Haag and Rotterdam, January 2002; 9746 of 65536 measurements classified unreliable (missing or faulty) =14.9%

2 Brief overview of methods for dealing with missing data

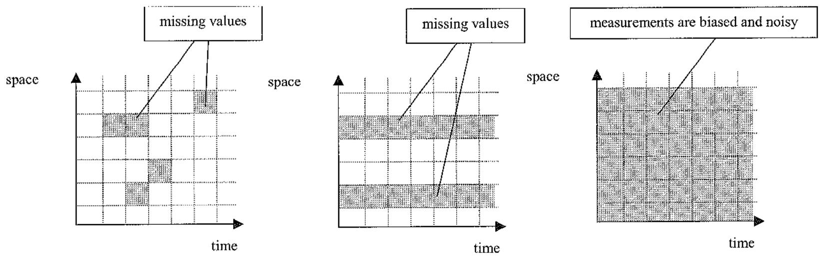
In practice the most commonly used approach to remedy input-failure is *imputation* (substituting missing values with sensible replacements, such as regression forecasts or sample mean). This pre-processing procedure allows one to treat the input after imputation as if it were 100% complete. Schafer (1997) shows that simple *imputation* schemes tend to change the covariance structure of the input-data and may induce bias. Therefore, Schafer proposes EM-based and Markov Chain Monte Carlo based approaches that account for the missing values, and the uncertainty they inherently introduce. Examples of these and other approaches to remedy the missing data problem can be found in many fields, including neuro-computing (Armitage, 1994) and (Meert, 1996), pattern recognition (Gabrys, 2002), climatology (Jeffrey et al, 2001), and medical statistics (Faris et al, 2001), to name a few.

Despite the clear theoretical shortcomings of *simple* imputation schemes (Schafer (1997) and Armitage (1994)), the results in (Chen et al, 2001) indicate that such simple imputation schemes combined with a *neural network* based travel time predictor, do yield accurate travel time predictions even when up to 30% of the input data to the model is missing. One (tentative) hypothesis may be that the neural network travel time predictor is robust to the "damage" caused by the imputation scheme applied. Slight changes in the statistical properties of the input data do not cause the neural network to produce inaccurate results. Another (again tentative) hypothesis could be that the spatiotemporal patterns formed by traffic measurements have statistical properties (covariance structure, correlations through time and space) that are more invariant to simple data-repair techniques as compared to for instance the multivariate datasets used throughout Schafer (1997), which stem predominantly from medical statistics, and social sciences. From traffic flow theory this does makes sense: we do expect that traffic measurements are highly correlated through space and time. Travel time predictions may be induced from different subsets of measurements along the route of interest

Based on these findings, we will explore two different approaches in the remainder of the paper. The first one involves inclusion of corrupted data[2] in the training procedure of our neural network based travel time predictor (the SSNN). The idea behind it is that the SSNN will find learn to predict travel time in a redundant way. The second one involves a simple imputation scheme that corrects corrupted input values before it is fed to the SSNN, by means of simple non-parameterized pre-processing procedures. This approach relies on the hypothesis that the SSNN is robust to the "damage" done by these procedures. Before we elaborate on these two approaches, we will first classify the various kinds of input-failure we may encounter in practice.

3 Classification of input failure

We propose a classification of input failure as presented in Figure 1.



(a) Incidental failures (b) Structural failures (c) Intrinsic failures

Figure 1: Incidental, Structural and Intrinsic failures in detection system

The first type of detection failure (Figure 1a – Incidental failures) occurs due to, for example, temporary power or communication failures in the freeway monitoring system. The second type (Figure 1b – Structural failures) occurs mainly due to physical damage or maintenance backlogs to the inductive loops or roadside equipment. The distinction between structural and incidental failure may in fact not be as crisp as presented here. It is more likely that detection devices will exhibit failure for a number of consecutive time periods, rather than produce random failure patterns over time. Furthermore, it is also more likely that failure will occur on a specific set of detectors due to maintenance backlog or the specific geographical and environmental circumstances on their locations, rather than randomly on all detectors on a specific route. Nonetheless, the proposed distinction expresses two extreme configurations of input failure that we can expect to encounter, and is hence useful in the investigation of the robustness of travel time prediction models to input failure. A third type of failure (Figure 1c – Intrinsic failure), measurement noise and bias, is inherent to detection devices and averaging measurements over time in general. For example: in MONICA the arithmetic time mean speed per measurement period is calculated. The result is a biased estimate of the space mean speed (the mean speed on a particular link or stretch), which is in fact the quantity of interest. This bias can exceed 13% in low speed regimes with high speed-variability, which is the case in congested or near-congested conditions: the conditions where accurate measurements are most desirable. Another known source of intrinsic data corruption is miscounts, double counts or false counts of vehicles, device calibration errors, round off errors, etc.

5 Experimental Setup

5.1 Data

We will use synthetic data obtained from a micro-simulation to test the robustness of the framework above. To this end we have set up a network in the microscopic simulation model FOSIM (developed at the Delft University of Technology, see e.g. (Vermijs et. al., 1994)), for the southbound stretch of the A13 highway between two of the major Dutch cities in the western part of the Netherlands, The Hague and Rotterdam. This three-lane stretch with a total length of 7,3 km is one of the most intensively used motorways in the Netherlands, with recurring congestion in most afternoon peaks on weekdays. It contains four on ramps and four off ramps, and two weaving sections. Usage of data from a traffic micro-simulation model allows us to collect and process all the data we need for the experiments, including the experienced travel times. The input to FOSIM (Dynamic OD matrices) was scaled to fit collected data from inductive loop-detectors on a number of typical afternoon peaks of January 2000. The scaling was done heuristically. All data used for training and testing the neural networks (both inputs and outputs) are linearly scaled to the interval [+0.1, +0.9], based on the rule-of-thumb that this leads to faster and more stable learning.

5.2 Representation of input failure

All Input failures result in input values of −1, allowing the pre-processing layer and/or travel time prediction model to detect these values as "outliers". If a measurement from detector d at time period p is dubbed corrupt then all values (i.e. both speed and flow) measured at $\{d, p\}$ are replaced with −1. *Incidental* input failure is generated with a random generator ϑ, producing numbers from a uniform distribution on [0, 1], such that each measurement $\{d, p\}$ has an equal probability to be labelled corrupt. If, for example, the required level of corruption is set to 20% then a measurement is labelled corrupt if $\vartheta(d,p) \leq 0.2$. The maximum amount of incidental input failure considered is set at 40%. In case of *structural* detection failure ALL measurements $\{d_k, p\}$ from a specific detector d_k are labelled corrupt. Therefore, structural detector failure is generated deterministically. First the number of corrupt detectors N^c is selected from {0, 1, .., D}, where D is the total number of detectors. Next all possible combinations of N^c out of D detectors are generated. Since this leads to a very large amount of test data, we will only consider cases where 1 or 2 detectors (out of 13) are structurally down, yielding a total of

$$\frac{13!}{1!\cdot 12!} + \frac{13!}{2!\cdot 11!} = 13 + 78 = 91 \text{ test-sets}$$

We will not include structural failure of detectors located on- or off-ramps here. Obviously, spatial interpolation does not apply to structural detector failure at onramps, since no adjacent detectors are available. We will address this important issue in following studies.

In this section we will propose a number of strategies in handling faults or incompleteness in the input data based on the framework in Figure 2. Each strategy (listed in Table 1) consists of a naïve and non-parameterized pre-processing method and a neural network training scheme[3], and is consequently tested against a dataset consisting of (a) 100% clean data, (b) a dataset with increasing amount of incidental input failure (detector failure is randomly drawn from a uniform probability distribution), and – in three of the five strategies - (c) a dataset with structural detector failure (one or two detectors on the main carriage way structurally breaks down). In the remainder of the paper the abbreviation MA is used to indicate the pre-processing method discussed in section 4.1.1, Interpolation refers to the pre- method depicted in section 4.1.2, and MA+Interpolation refers to the one discussed in section 4.1.3

Table 1: Combined strategies of pre-processing and training

Strategy	pre-processing	training	testing		
			100% clean	incidental	structural
S0	None	100% clean	x	x	x
S1	None	10-40% incidental	x	x	x
S2	MA	100% clean		x	
S3	Interpolation	100% clean		x	x
S4	MA+Interpolation	100% clean		x	

Note that we do not need to test the performance of S2, S3 and S4 on clean data, since pre-processing has no effect if no corruption is present.

5.4 Performance indicators

Let T_p and T_p^{est} denote the experienced and the predicted mean travel time for a vehicle departing in time period p on the route described above, and let P the total number of periods. The prediction error for departure time p then equals $\Delta_p^{est} = T_p^{est} - T_p$. We will make use of the following performance measures to assess the various strategies:

RMSE (Root of Mean Squared Error) $\sqrt{\frac{1}{P}\sum_p \left(\Delta_p^{est}\right)^2}$

ME (Mean Error) $\frac{1}{P}\sum_p \Delta_p^{est}$

SE (Standard Deviation of Error) $\sqrt{\frac{\sum_p (\Delta_p^{est} - ME)^2}{P-1}}$

[3] The SSNN model is trained by Levenberg-Marquardt and Bayesian Regularization (Foresee, Hagan, 1997). More details and references are given in (van Lint et al, 2002).

6 Results

6.1 Testing on Incidental Failure

The three tables below show the ME, SE and RMSE on one of the two test datasets for all proposed strategies.

Table 2: Mean prediction Error of strategies S0 (top row), S1 (2nd to 5th row), S2 (6th row), S3 (7th row) and S4 (8th row) to incidental input failure at increasing amounts of corrupted data in the test datasets (total nr records in test data: 5 x 396)

		% of incidental failure in test-set					ME on ALL
	ME (seconds)	0%	10%	20%	30%	40%	test-sets
	S0	1	43	91	120	152	81
training with corrupted data	S1 (10%)	0	0	-4	-5	-10	39
	S1 (20%)	-6	-3	-4	1	1	25
	S1 (30%)	-3	-2	-4	-1	-3	18
	S1 (40%)	17	13	6	3	-4	16
pre processing	S2 (ma)	1	1	1	2	2	1
	S3 (interp)	1	-8	-19	-25	-26	-15
	S4 (int/ma)	1	3	6	12	18	8

Table 3: Standard deviation of prediction Error of strategies S0 (top row), S1 (2nd to 5th row), S2 (6th row), S3 (7th row) and S4 (8th row) to incidental input failure at increasing amounts of corrupted data in the test datasets (total nr records in test data: 5 x 396)

		% of incidental failure in test-set					SE on ALL
	SE (seconds)	0%	10%	20%	30%	40%	test-sets
	S0	17	43	62	76	89	83
training with corrupted data	S1 (10%)	22	22	26	31	42	75
	S1 (20%)	27	25	26	27	32	66
	S1 (30%)	32	30	28	25	26	60
	S1 (40%)	50	44	37	32	28	57
pre processing	S2 (ma)	17	18	19	21	24	20
	S3 (interp)	17	30	39	50	62	44
	S4 (int/ma)	17	19	23	32	36	27

increasing amounts of corrupted data in the test datasets (total nr records in test data: 5 x 396)

			% of incidental failure in test-set				RMSE on
	RMSE (seconds)	0%	10%	20%	30%	40%	ALL test-sets
	S0	17	60	110	141	176	116
training with corrupted data	S1 (10%)	22	22	26	32	43	85
training with corrupted data	S1 (20%)	28	25	26	27	32	71
training with corrupted data	S1 (30%)	33	30	28	25	27	63
training with corrupted data	S1 (40%)	53	46	38	32	28	59
pre processing	S2 (ma)	17	18	19	22	24	20
pre processing	S3 (interp)	17	31	43	56	67	46
pre processing	S4 (int/ma)	17	19	24	34	40	28

The best results are obtained with strategy S2, which performs well at all degrees of incidental input failure corruptness. Due to the random nature of the input failure, the MA procedure is capable of reconstructing the corrupted input signals such, that the SSNN model is still capable of producing accurate predictions, even at 40% incidental corruption. We hypothesize that the good results of this strategy are partly due to the good generalization capabilities of the SSNN model (section 4.2), implicating that slight deviations (caused by the MA procedure) in on or more input signals do not translate into similar deviations in the output. The fact that it infers expected travel times not only from current input signals but also from its own internal states (its short term memory), makes it "extra" robust to temporary flaws in one or more input signals.

We can also conclude that incorporating corrupt values into the training datasets (strategy S1), does increase the robustness of the model to incidental input failure. There is, however, a price for this increase in robustness in terms of model accuracy. Clearly, the SSNN trained with 40% corrupt input data, produces a larger bias and larger variance when tested against clean test-data, than a SSNN model trained with 100% clean data. Nonetheless, the inclusion of small amounts (10 - 20%) of corruptness in the training datasets offers a good balance between accuracy and robustness.

6.2 Testing on structural detector failure

As noted before, structural detector failure may require a neural network to learn the complex dynamics of traffic flow in a different way, by "looking" at different detectors to detect congestion and thus delays. In other words: structural detector failure might require an entirely different SSNN model, than a situation with clean or randomly corrupted data. Consequently, we a-priori expect that a pre-processing strategy might prove more useful than a strategy where SSNN's are trained with various degrees of incidental corruption.

Table 5 shows the performance for a SSNN model trained with clean data, four different SSNN models, trained with 10 – 40% (incidental) corruption, and a SSNN model trained with clean data fed with pre-processed data using spatial interpolation on test sets in which one out of the 13 detectors of the main carriage way is structurally down.

Again we see that the simple pre-processing strategy (S3 – spatial interpolation) outperforms the models resulting from strategy S1 on all performance indicators. Obviously, the deviations in the input patterns caused by the interpolation procedure still allow the SSNN model to infer expected travel times quite accurately, although structural input failure at some of the significant detectors still causes an increase in bias. This increase is never larger than 30 seconds and always less than in any of the "S1 models".

Nonetheless, all "S1 models" still perform substantially better than the model trained with clean data, but no clear picture emerges as to how much corruption in the training datasets leads to the best overall result. If we look at the RMSE cost for failure of detectors connected to so-called "significant" links[4], {2,3,5,6,7,8,10,11}, then the SSNN trained with 20% corruption can be considered the best model for strategy S1.

A couple of remarks need to be made. First, it is important to note that structural failure of the significant detectors leads to a larger bias (ME), rather than larger variance (SE), for all strategies. This is a result from the fact that these detectors are associated with those links where congestion sets in first (bottlenecks), and thus provide crucial information on the progression of congestion and hence the expected travel time. Secondly, a remarkable result is that structural failure of detectors connected to these significant links result in the largest errors in the SSNN trained with *clean* data, but do not necessarily yield the largest errors in SSNN models trained with more than 10% corruption. We have no clear explanation for this phenomenon.

[4] For example: detector 10 and 11 are connected to link 10, which is considered the principle bottleneck

Table 5: Performance of strategies S0, S1, and S3 to structural input failure at 1 out of 13 detectors on the main carriage way (total nr records in test data: 13 x 396)

		detector structurally down													Performance on ALL test-data
		1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	
S0	*ME*	-50	-107	64	134	127	209	96	38	74	139	198	28	19	75
	SE	26	63	31	41	38	70	40	23	34	56	77	18	18	98
	RMSE	56	124	71	140	132	220	104	44	81	150	213	33	26	123
S1 (10%)	*ME*	-1	2	2	-41	-6	-27	-44	15	85	71	-26	39	-1	5
	SE	22	25	22	26	20	23	33	38	35	29	36	27	22	47
	RMSE	22	25	22	49	21	35	55	41	92	76	44	47	22	47
S1 (20%)	*ME*	-4	6	-70	17	-3	-12	-36	-78	33	80	-18	29	-2	-4
	SE	27	27	42	36	24	23	33	36	24	36	23	34	28	51
	RMSE	27	27	82	40	24	25	49	86	41	87	29	44	28	51
S1 (30%)	*ME*	-4	25	-36	7	27	13	-11	31	43	45	21	32	-7	14
	SE	30	60	33	38	47	37	28	36	31	39	28	41	32	44
	RMSE	30	65	49	38	54	39	30	47	53	59	35	52	33	46
S1 (40%)	*ME*	17	52	-12	29	2	18	-32	-3	55	53	-77	111	8	17
	SE	50	73	40	46	35	51	35	48	47	43	54	72	47	67
	RMSE	53	90	41	55	35	53	48	48	72	68	94	132	48	69
S3 (interp)	*ME*	0	23	-1	3	-6	16	-6	2	6	-12	25	-2	4	4
	SE	17	24	17	17	18	28	18	18	17	21	24	17	17	23
	RMSE	17	33	17	17	19	33	19	18	18	25	34	17	17	23

Let us now consider structural failure of two detectors on the main carriageway, for which Table 6 presents the total performance on all 78 test-sets.

Table 6: Total Performance on structural detector failure at 2 out of 13 detectors on the main carriageway, indicators calculated on 78 test-sets of 396 records each

	ME (seconds)	SE (seconds)	RMSE (seconds)
S0	136	126	185
S1 (10%)	5	47	47
S1 (20%)	8	65	65
S1 (30%)	35	55	66
S1 (40%)	19	81	83
S3	6	28	29

Again we see that the inclusion of small amounts of corruption in the training datasets leads to an improvement of the performance and thus robustness of the model to corrupted input. As could be expected, the preprocessing strategy (S3 – spatial interpolation) outperforms the S1 strategies.

6.3 Overall comparison and discussion

In the previous sections we tested two general approaches to increase the robustness of the SSNN travel time prediction model to both incidental and structural input failure. The first involved incorporating corrupted data in the SSNN training procedure, the second pre-processing strategies with naïve non-parameterized procedures such as an exponentially moving average and spatial interpolation.

It appears that both approaches do improve the robustness to both incidental and structural input failure considerably, albeit that the pre-processing strategies performed best. This is remarkable since the proposed pre-processing algorithms are very simple naïve non-parameterized procedures. This supports both hypotheses posed in section 2 on the robustness of a neural network to the "damage" caused by simple imputation schemes, and the invariant nature of traffic data to imputation schemes (to a degree). There is yet another, more qualitative, argument that favours the pre-processing approach to the training approach. The internal states of SSNN travel time prediction model (Figure 4) are strongly correlated to the actual traffic processes and can be interpreted as metrics representative for the expected traffic conditions on the route of interest (van Lint et al, 2001). Incorporating corrupt data in the training procedure makes these internal states more difficult to interpret, and makes the model less useful for analytical purposes.

As stated in section 3, the distinction between incidental and structural detector failure may not be as crisp as presented here. The results of the previous section show that a time series approach works best for incidental failure, but is useless for structural failure, where spatial interpolation is the preferable alternative. Thus, in real life a more sophisticated pre-processing strategy is required. In this strategy, we need to keep track of detector failure, classify it on-line as incidental or structural, and consequently apply the appropriate pre-processing algorithm, which could be a time series model, a spatial interpolation model or even more sophisticated (spatio-temporal) models such as Kalman filters, macroscopic traffic models or even neural networks.

Robustness to missing or faulty input, due to failures in the data collection system, is a key characteristic for any travel time prediction model that is to be applied in a real time environment. We have tested two strands of approaches to deal with different kinds of input failure (incidental, structural) in conjunction with a neural network based travel time prediction model (the SSNN model). The main findings are:

1. Both naïve non-parameterized imputation schemes, and incorporation of corrupt data in the training data of the SSNN, improve the robustness of the travel time prediction model to incidental and structural input-failure considerably.
2. The pre-processing strategies give the best overall results. Although there are clear theoretical shortcomings to simple imputation schemes, their use may be justified in this particular application: the results indicate that the SSNN is robust to the "damage" done by naïve imputation schemes.
3. Nonetheless, in real life more sophisticated pre-processing procedures need to be developed since the distinction between incidental and structural detector failure will not be as crisp as presented here.
4. Inclusion of corruption in the training procedure of the SSNN model also improves robustness, but at the cost of predictive accuracy.

Further research topics include: enhancing the proposed pre-processing strategies such that the appropriate procedure is used for each kind of input failure encountered; research into more sophisticated spatio-temporal pre-processors based on for instance Kalman filtering and macroscopic traffic flow modeling; a mechanism to assign confidence intervals to predictions based on the level of corruptness in the input. Further research should also focus on real time application of the robust travel time prediction framework proposed here.

References

Berkum, P. van, P. van der Mede (1993) *The impact of Traffic Information: Dynamics in route and departure time choice*, Delft University Press.

Lint, J.W.C. van, S.P. Hoogendoorn, H.J. van Zuylen (2002) *Freeway travel time prediction with State-Space Neural Networks,* Proceedings of the Transportation Research Board Annual Meeting (CD Rom), 14-17 January 2002, Washington D.C., USA.

Schafer, J.L. (1997) *Analysis of Incomplete Multivariate Data,* Chapman & Hall, London, UK.

Gabrys, B. (2002) Neuro-fuzzy approach to processing inputs with missing values in pattern recognition problems, in: *International Journal of Approximate Reasoning,* Volume 30, Issue 3, September 2002, pp. 149-179

Meert, K. (1996) A Real-Time Recurrent Learning Network Structure for Dealing with Missing Sensor Data, in: *Proceedings of the 1996 IEEE Conference on Neural Networks,* part 3, Washington D.C., USA, pp. 1600-1605.

Jeffrey, S. J., J.O. Carter, K.B. Moodie, A.R. Beswick (2001) Using spatial interpolation to construct a comprehensive archive of Australian climate data, in: *Environmental Modeling & Software*, Volume 16, Issue 4, June 2001, pp. 309-330.

Faris, P.D., W.A. Ghali, R. Brant, C.M. Norris, P.D. Galbraith, M.L. Knudtson, (2001) Multiple imputation versus data enhancement for dealing with missing data in observational health care outcome analyses, in: *Journal of Clinical Epidemiology*, Volume 55, Issue 2, February 2002, pp. 184-191.

Armitage, W.D. (1994) Enhancing the robustness of a Feedforward Neural Network in the Presence of Missing Data, in: *Proceedings of the 1994 IEEE Conference on Neural Network,* part 2, Orlando, Florida, USA, pp836-839.

Bishop, C.M. (1995) *Neural Networks for pattern recognition*, Oxford University Press, United Kingdom.

Elman, J. (1990) Finding structure in Time, in: *Cognitive Science 14*, pp. 179-211.

Foresee, F.D., M.T. Hagan (1997) Gauss-Newton approximation to Bayesian learning, in: *International Conference on Neural Networks*, Vol 3, pp. 1930 –1935.

Vermijs, R.G.M.M., H. Schuurman (1994) Evaluating capacity of freeway weaving sections and on-ramps using the microscopic simulation model FOSIM, in: *Proceedings of the second international symposium on highway capacity,* Vol 2., pp 651-670.