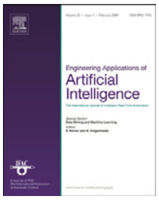
Engineering Applications of Artificial Intelligence 24 (2011) 534–542

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

journal homepage: www.elsevier.com/locate/engappai



Prediction intervals to account for uncertainties in neural network predictions: Methodology and application in bus travel time prediction

Ehsan Mazloumi [a],*, Geoff Rose [a], Graham Currie [a], Sara Moridpour [b]

[a] Institute of Transport Studies, Monash University, Melbourne, Australia
[b] School of Civil, Environmental and Chemical Engineering, RMIT University, Australia

ARTICLE INFO

Article history:
Received 23 April 2010
Received in revised form
29 October 2010
Accepted 12 November 2010
Available online 4 December 2010

Keywords:
Bus travel time
Prediction uncertainty
Neural network
Prediction intervals

ABSTRACT

Neural networks have been employed in a multitude of transportation engineering applications because of their powerful capabilities to replicate patterns in field data. Predictions are always subject to uncertainty arising from two sources: model structure and training data. For each prediction point, the former can be quantified by a confidence interval, whereas total prediction uncertainty can be represented by constructing a prediction interval. While confidence intervals are well known in the transportation engineering context, very little attention has been paid to construction of prediction intervals for neural networks. The proposed methodology in this paper provides a foundation for constructing prediction intervals for neural networks and quantifying the extent that each source of uncertainty contributes to total prediction uncertainty. The application of the proposed methodology to predict bus travel time over four bus route sections in Melbourne, Australia, leads to quantitative decomposition of total prediction uncertainty into the component sources. Overall, the results demonstrate the capability of the proposed method to provide robust prediction intervals.

Crown Copyright © 2010 Published by Elsevier Ltd. All rights reserved.

1. Introduction

1.1. *Artificial neural networks—prediction tools in Transportation engineering*

Artificial neural network models (or neural networks hereafter) are receiving more and more attention in the various aspects of transportation engineering due to their modelling flexibility, predictive ability and generalization potential. Their application ranges from traffic operations (Van Lint et al., 2005; Smith and Demetsky, 1995; Chien et al., 1994; Dharia and Adeli, 2003), incident detection and prediction (Xie et al., 2007; Dia and Rose, 1998) and transportation planning (Dia and Panwai, 2007; Tillema et al., 2006) to infrastructure management (Yang et al., 2006; Mukkamala and Sung, 2003) and environmental studies (Cai et al., 2009; Shiva Nagendra and Khare, 2004). Neural networks have also been adopted in the public transport context to model bus travel times (Kalaputapu and Demetsky, 1995; Jeong and Rilett, 2004; Chen et al., 2007).

Traditionally, neural networks used for prediction purposes give rise to a point prediction when they are presented with a set of input values. However, there is always a degree of uncertainty associated with any point prediction. That uncertainty, as will be discussed shortly in Section 2, is attributable to either structure of the model or the inherent uncertainty in the dataset used for model development. Due to these reasons, point prediction performance deteriorates and predictions become unreliable.

1.2. *Knowledge of prediction uncertainty—applications in Transportation engineering*

A common problem associated with point predictions is that they deliver no information about different kinds of uncertainty affecting the prediction performance. However, the reliability of point predictions can be enhanced through providing a measure of prediction uncertainty (Khosravi et al., 2010a), or at least by quantifying the extent that each different source contributes into prediction unreliability. This issue has motivated some studies in the transportation literature to provide a prediction range, rather than a point prediction, for the relevant dependent variable. Inherently, the width of these ranges is directly related to the degree of confidence in the point predictions. For instance, studies focussed on predicting travel time variability provide a measure of uncertainty in travel time prediction by quantifying the variance of travel times (e.g. Fu and Rilett, 1998; Pattanamekar et al., 2003; Liu et al., 2005; Li, 2006). These variance values, which indicate the extent of variability/reliability of travel times, would then benefit passengers by helping them to better plan their trips, hence would have a range of applications in intelligent transportation systems.

* Corresponding author.
E-mail address: ehsan.mazloumi@eng.monash.edu.au (E. Mazloumi).

0952-1976/$ - see front matter Crown Copyright © 2010 Published by Elsevier Ltd. All rights reserved.
doi:10.1016/j.engappai.2010.11.004

In public transport operations, predicting a range for travel times can assist in defining slack times needed in the scheduling process to maximize on-time performance (Mazloumi et al., 2010). The quality of transit signal priority schemes can also be enhanced by providing an arrival time interval for individual busses at a certain downstream signalized intersection (Kim and Rilett, 2005).

1.3. *Quantifying the uncertainty in predictions made by neural networks—research objectives*

To cope with the weakness of neural networks in providing prediction confidence, one approach is to specify intervals (rather than points) where predictions may lie with a predefined likelihood. Depending on what source of uncertainty is considered by these intervals, different terms are used to specify these measures of confidence, i.e. confidence intervals or prediction intervals. Many previous researchers have quantified confidence intervals. For instance, Van Hinsbergen et al. (2009) and Park and Lee (2004) used Bayesian technique to construct confidence intervals for travel time predictions made by neural networks. From a Bayesian inference perspective, each parameter in a neural network is conceived as a distribution rather than a single value. Consequently, neural network outcomes will also form a distribution, which can be further used to construct intervals around each prediction point. However, the computationally intensive nature of Bayesian technique has limited the application of this approach in confidence estimation for neural network predictions (Dybowski and Roberts, 2001). However, to the best of our knowledge, no work has been completed to construct prediction intervals for neural networks employed in transportation applications.

This paper contributes to understand this domain by demonstrating a relatively straightforward approach, founded in maximum likelihood techniques, for constructing prediction intervals. The maximum likelihood approach, as opposed to the Bayesian algorithm, will give rise to a single value (rather than a distribution) for each model parameter and hence for output values. The paper first discusses the possible sources of uncertainty in neural network predictions. Then, following a general description of neural networks, the concepts of confidence intervals and prediction intervals are presented and techniques to quantify each of them are discussed. The proposed methodology is then applied to predict bus travel times along a bus route in Melbourne, Australia, and its performance is evaluated. The final section of the paper presents the conclusions and identifies directions for future research.

2. Sources of uncertainty

In the neural network community, it is common to consider two sources for uncertainty associated with neural network outcomes: uncertainty in training dataset and uncertainty in model structure (Heskes, 1997; Papadopoulos et al., 2001; Dybowski and Roberts, 2001). Those sources are discussed separately in the subsections which follow.

2.1. *Uncertainty in training dataset*

A portion of the total uncertainty in prediction values is attributable to the inherent uncertainty in the input data. This uncertainty (measured by the variance denoted by σ_e^2) may arise from variability in randomly selecting a training dataset from the associated population. The implication of this uncertainty for the problem of bus travel time prediction might be related to a range of stochastic factors, e.g. signal delay or dwell time experienced by different buses, which results in travel times fluctuating around a mean value. Note that these variables have stochastic characteristics and are difficult, or even impossible, to model with deterministic variables.

Consequently, not all the possible realizations of the dependent variable are available in the training dataset, and the training dataset used to train a neural network is only one of a large number of possibilities. Since each possible training dataset will give rise to a different neural network, there could be a distribution for output values when certain input values are given. Note that in many practical problems, both input and target values might be associated with an uncertainty or a noise because of imperfections in data collection tools or techniques. However, in the analysis presented in this paper, it is assumed that there is no noise associated with data acquisition.

2.2. *Uncertainty in model structure*

Model uncertainty variance (measured by the variance denoted by σ_m^2) contributes to total prediction uncertainty in two ways:

- There is always an uncertainty in the values of the neural network parameters since an error function can have many local minima resulting in a number of possible values for network weights. In addition, suboptimal training, e.g. premature termination of a training algorithm, may introduce bias in model weights which is another source of uncertainty.
- As for regression techniques, neural networks are also prone to structural misspecification. The lack of input variables that can adequately model the dependent variable is an example of a model misspecification, which introduces uncertainty in prediction outcomes.

It is common to assume that σ_e^2 and σ_m^2 are independent and total prediction variance σ_t^2 can be obtained from $\sigma_t^2=\sigma_e^2+\sigma_m^2$ (Heskes, 1997; Papadopoulos et al., 2001; Dybowski and Roberts, 2001). Given these uncertainty sources, it is informative to explore the prediction confidence of a neural network, and to quantify how these sources might affect the prediction confidence. To this end, there could be two statistical measures of confidence, *confidence intervals* and *prediction intervals,* to explore predictive behaviour of a neural network. The next section differentiates between those two measures.

3. Methodology

In this section, the general mathematical structure of a neural network is presented as a foundation for the discussion which follows. Subsequent sections focus on quantification of σ_e^2 and σ_m^2.

3.1. *General description of feed forward neural networks*

Let us assume that we are interested in developing a neural network to model target values $t^n=(t_1^n,\ldots,t_d^n)$drawn from N data points in target dataset of $T=(t^1,\ldots,t^N)$ on the basis of input values $x^n=(x_1^n,\ldots,x_e^n)$ from a dataset of $X=(x^1,\ldots,x^N)$. For modelling purposes, we consider a neural network that consists of input, hidden and output layers as shown in Fig. 1. The input layer may contain e input nodes, the hidden layer h and the output layer d nodes. The input layer receives the input values which are processed through the network to give rise to output values $y(x^n)=(y_1(x^n),\ldots,y_d(x^n))$. The hidden layer is actually the main estimation core of the model. There could be several hidden layers, but one hidden layer is sufficient to closely map most relationships (Jain and Nag, 1997).

The mathematical expression of how the neural network calculates an output value $y_k,k=(1,\ldots,d)$ can be presented by the

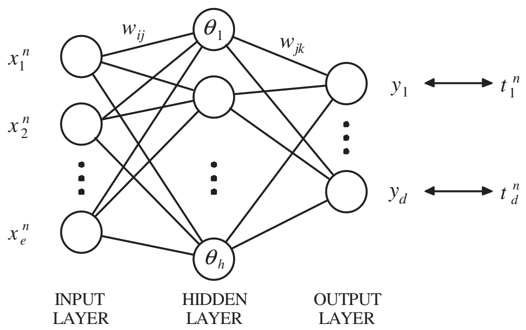


Fig. 1. A typical network with e input nodes, single hidden layer with h nodes and d output layer nodes.

following equations:

$$y_k(x) = f_o\left(\sum_{j=1}^{h} w_{jk} z_j\right) \quad z_j = f_h\left(\sum_{i=1}^{e} w_{ij} x_i\right) \tag{1}$$

where w_{jk} and w_{ij} are the network weights whose values are calibrated from the training data. The functionsf_o and f_h (corresponding to output and hidden nodes, respectively) are activation functions transforming the weighted sum of the outputs in the left of each node. While a range of transfer functions are available (including logistic, hyperbolic and sigmoid functions), the sigmoid function $f_h()$ commonly associates hidden layers in most travel time prediction models (Van Lint, 2004; Liu, 2008). A linear activation function $f_o()$ is also commonly used for the output nodes

$$f_h(a) = \frac{1}{1+e^{\theta_j - a}} \quad f_o(a) = a \tag{2}$$

where θ_j is called bias associated with hidden node j, which also needs to be adjusted based on the data. Biases together with weight values will form the vector of model parameters W.

3.2. Training a neural network

Calibration of neural network parameters is achieved through a training process. This training mechanism is normally based on maximum likelihood approach that seeks to find the optimum parameter values by maximizing likelihood derived from the training data. As shown by Bishop (1995), this approach is equivalent to the minimization of an error function such as the sum of squared errors of the predictions as formulated in the following:

$$E_D = \frac{1}{2}\sum_{n=1}^{N} [t^n - y^n]^2 \tag{3}$$

However, other error functions exist in literature such as root mean squared error (RMSE), mean absolute percentage error (MAPE), etc. Qi and Zhang (2001) have compared these criteria for selecting the optimal neural network parameters and concluded that there is no best method.

Training of a neural network ultimately aims to develop a model that predicts well on new, unseen test examples, i.e. it generalizes well. A superior approach to ensure a good generalization ability is Bayesian regularized back-propagation approach (Bishop, 1995) that uses the Levenberg–Marquardt (Hagan and Menhaj, 1999) algorithm to minimize a linear combination of prediction error E_D and weights

$$E = E_D + \alpha E_W \tag{4}$$

where $E_W = \frac{1}{2}\sum_{w \in W} w^2$ and the parameter α controls the influence of the weight regularizer on the solution, and is determined such that the model generalizes well. It has been found that a regularizer of this form can lead to considerable improvements in network generalization ability (Hinton, 1987). The use of Eq. (4) in training a network will lead to small weights that in turn have the potential to help in having a good generalization (Bishop, 1995).

3.3. Measures of prediction confidence: confidence intervals versus prediction intervals

First, assume that $g(x^n; w_o)$ is the 'true' unknown neural network that generated target vector t^n, where w_o is the 'true' vector of model parameters. For simplicity, assume that both vectors x^n and t^n are one dimensional, so

$$t_i = g(x_i; w_o) + \varepsilon_i \quad i = 1,\ldots,N \tag{5}$$

where ε_i is an error term with the average zero and variance σ_e^2. Within the training process, the unknown function $g(x_i; w_o)$is estimated by finding the best estimates $\hat{w}_o$ for w_o. Minimizing the error function in Eq. (3) will yield a set of outputs y_i being the average of target values given the input vector x_i, $E[t|x_i]$

$$y_i = g(x_i; \hat{w}_o) \equiv E[t|x_i] \tag{6}$$

with this notation, a *confidence interval* (*CI*) shows the accuracy of the estimation of the true but unknown function $g(x_i; w_o)$. It is concerned with the distribution of the quantity

$$g(x_i; w_o) - g(x_i; \hat{w}_o) = g(x_i; w_o) - y_i \tag{7}$$

On the other hand, a *prediction interval* (*PI*) is concerned with the accuracy of prediction outputs by focussing on the distribution of the quantity

$$t_i - g(x_i; w_o) = t_i - y_i \tag{8}$$

Thus, a *CI* is concerned with that part of the prediction uncertainty which is caused by the model inability to capture the 'true' unknown $g(x_i; w_o)$, whereas a *PI* deals with the difference between the target t_i and the output y_i by considering ε_i as well. It can be concluded that a *CI* is always enclosed in its corresponding *PI*. To better understand different prediction components discussed above, Fig. 2 illustrates the relationship between the network prediction$y_i = g_\lambda(x_i; \hat{w}_o)$, target value t_i, the unknown underlying function $g(x_i; w_o)$ and ε_i.

3.4. Quantifying model structure uncertainty variance

This section discusses how model structure uncertainty σ_m^2can be quantified through constructing confidence intervals.

Generally, three main approaches have been used for estimating prediction confidence: Bayesian framework based on Bayesian statistics (Van Hinsbergen et al., 2009), the Delta method which is based on a Taylor expansion of the regression function (Khosravi et al., 2010b) and the Bootstrap method which is essentially a resampling method (Van Lint, 2004, Li, 2006). Papadopoulos et al.

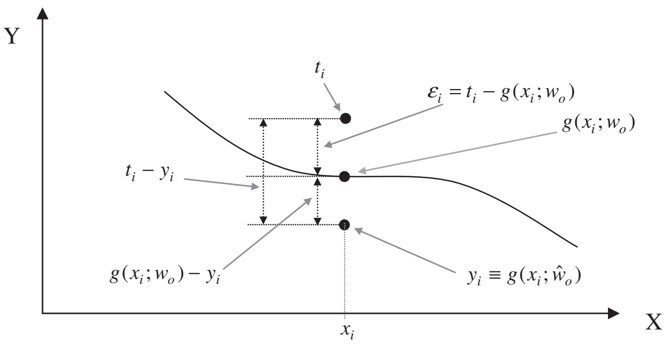

Y
t_i
$\varepsilon_i = t_i - g(x_i; w_o)$
$g(x_i; w_o)$
$t_i - y_i$
$g(x_i; w_o) - y_i$
$y_i \equiv g(x_i; \hat{w}_o)$
x_i
X

Fig. 2. The relationship between prediction y_i, target t_i , the unknown underlying function $g(x_i; w_o)$ and ε_i.

(2001) and Dybowski and Roberts (2001) elaborate on the pros and cons of these techniques. However, there is no distinct preference reported in the literature for any of these techniques.

The latter approach, the Bootstrap method, is chosen in this study to construct the confidence intervals for two reasons. First, unlike the Bayesian framework that involves the computation of a Hessian matrix, which limits its use in many applications (Dybowski and Roberts, 2001), the bootstrap is efficient and relatively easy method to implement in both research and practice (Li, 2006). In addition, unlike the Delta method, the Bootstrap method gives rise to input-dependent values for variance (Dybowski and Roberts, 2001).

The Bootstrap technique estimates the standard error of a model parameter (Efron, 1982), so it can be used to estimate σ_m^2 which actually reflects the standard error of $E[t|x_i]$. This technique is based on resampling with replacement of the available dataset and training an individual network on each resampled subset of the original dataset. A resample dataset D has the form of (x_i,t_i), where i can take any value in $[1,N]$ with the probability of $1/N$. As the size of the resampled sets is equal to the size of the original dataset N, a resampled set may contain some input–output pairs more than once while other pairs may not be present at all. If there are B resampled sets, a collection of B networks of the same architecture is formed, and the variance of the outputs of individual networks is the estimate of σ_m^2. The general steps to implement this technique for a neural network are as follows:

- Randomly draw B independent bootstrap samples from the training dataset of size N. Each sample consists of N pairs of input–output variables.
- Train a neural network with each sample. Corresponding to a certain input value x_i, the estimate from neural network trained on the bth, b=1,…,B bootstrap sample is $y_b(x_i)$.
- Estimate the model uncertainty variance σ_m^2 associated with input value x_i using the outputs of the B alternative networks

$$\sigma_m^2(x_i) = \sum_{b=1}^{B} \frac{[y_b(x_i)-\bar{y}(x_i)]^2}{B-1} \tag{9}$$

$$\bar{y}(x_i) = \sum_{b=1}^{B} \frac{y_b(x_i)}{B} \tag{10}$$

Assuming that target values follow a Normal distribution, the 95% confidence intervals are then constructed using $(\bar{y}_i \pm Z_\alpha \times \sigma_m)$ and $Z_\alpha = 1.96$. Corresponding to each input vector x_i, a 95% confidence interval implicates a range where the mean of a population of dependent values will occur with 95% probability. It is worth noting that bus travel times have been shown to follow a normal distribution under normal conditions and narrow departure time windows (Mazloumi et al., 2010). For the case when travel times follow a skewed distribution (e.g. lognormal or Gamma distributions), the intervals constructed as above will underestimate the real range.

3.5. Quantifying input data noise variance

To quantify the uncertainty σ_e^2caused by error term ε_i in the input data, maximum likelihood based approaches can be adopted which unlike the computationally intensive Bayesian mechanism provides an explicit alternative to estimate σ_e^2 (Bishop, 1995). One of the early studies in this realm is (Nix and Weigend, 1994), which extended the traditional network structure and used a new set of hidden units to compute σ_e^2. Their approach changes the network error function leading to the reformulation of weight updating equations in training algorithms which in turn required massive computational effort. Unlike that method, which is not easily implementable in many existing commercial tools, the method adopted in this study is more straightforward.

The variance of t conditioned on (or given) x_i,$Var(t|x_i)$, can be calculated from

$$Var(t|x_i) = E[(E[t|x_i]-t)^2|x_i] \tag{11}$$

On the other hand, from Eqs. (3) and (6), one can conclude that if Eq. (3) is used to train a neural network, for each input x_i, the network will give rise to an output $y_i = E[t|x_i]$ which is the average of target values given the input x_i. Therefore, in Eq. (3), replacing t with $(E(t|x_i)-t)^2$will result in the neural network estimating $E[(E[t|x_i]-t)^2|x_i]$ instead of $E(t|x_i)$. Thus, training a separate neural network $f(x_i;v_o)$with the same input values and the objective function in Eq. (12) will give rise to estimates of $Var(t|x_i)$

$$E_D = \frac{1}{2}\sum_{i=1}^{N} [s_i^2-(y_i-t_i)^2]^2 \tag{12}$$

where s_i^2 is the estimate of $Var(t|x_i)$. To put this in other words, $f(x_i;v_o)$ uses the same input values and the same objective function (as shown in Eq. (3)), but the target values are the square of the prediction errors from the first model $g(x_i;w_o)$. Note, however, $f(x_i;v_o)$ is also associated with uncertainty in its structure σ_{m2}^2, which can be also captured through a bootstrap analysis. Therefore

$$\sigma_e^2 = Var(t|x_i) + \sigma_{m2}^2 \tag{13}$$

After σ_m^2 and σ_e^2 have been quantified, total prediction uncertainty can be obtained from $\sigma_t^2 = \sigma_m^2 + \sigma_e^2$ and 95% prediction intervals are then constructed by adopting $(y_i \pm 1.96 \times \sigma_t)$ for each input vector x_i. Corresponding to each input vector x_i, such an interval is a range where the dependent variable would occur with 95% probability.

4. Case study

The methodology detailed in the previous section is used to examine the impact of different sources of uncertainty in travel time predictions in the context of an 8-km-long portion of a bus route in inner Melbourne, Australia. This portion of the route comprises four sections (which are similar in length) and are demarcated by five timing point stops (see Fig. 3). Those timing

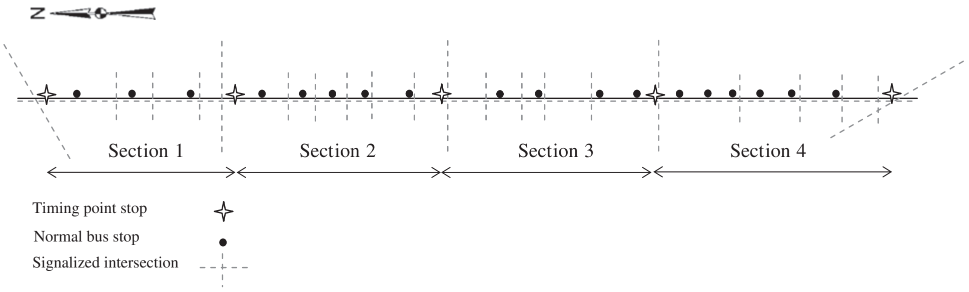


Fig. 3. A schematic presentation of the study test bed.

points are the major bus stops and at each of them, bus arrival/departure times are monitored to maintain consistency and track schedule adherence. Bus headways vary from 10 min in peak hours to about half an hour in the off-peak. Buses on this portion of the route operate in mixed traffic and there is no separate lane allocated to buses.

4.1. Data

Some of the buses (not all of them) over this route were equipped with GPS devices, and their arrival/departure times corresponding to timing point stops were recorded. About 1800 weekday travel time observations were obtained for each route section over a six month period (starting from February 2007). Fig. 4 shows each section's travel times over the day. This graphical presentation reveals the considerable variability of travel times at the same time of day for each section. For instance, travel times in Section 2 at 2:00 pm, range from about 200 s to just over 500 s over different days. This may be due to a range of factors including variations in passenger demand and traffic flow over different days, various signal delays experienced by different buses, and variation in driving style of bus drivers on different days. Fig. 4 also suggests that travel times for each section can be classified into four time periods: AM peak (7 am–10 am), inter peak (10 am–4 pm), PM peak (4 pm–7 pm) and off-peak (before 7 am and after 7 pm). This taxonomy will be used later to assess the performance of the proposed methodology by time period.

The study aims to construct confidence and prediction intervals for each route section. To this end, a set of explanatory variables was needed. The selection of explanatory variables was guided by evidence found in literature (Mazloumi et al., 2010) of key variables in bus travel time prediction. Traffic flow data collected by SCATS loop detectors was available from each intermediate signalized intersection. SCATS data included traffic counts and traffic degree of saturation values averaged over a predefined aggregation period before departure of each bus from an upstream timing point stop. Four different aggregation period lengths are considered: 2, 15, 30 and 60 min. It is worth noting that while traffic counts reflect the fluctuations in demand, degree of saturation values, defined as the ratio of the effectively used green time to the total available green time for each movement (Lee et al., 2002), capture the changes in both demand and capacity (signal cycle and green time lengths).

Also available were data on weather conditions in terms of the hourly amount of rain that fell over the corresponding one hour period, and a measure of schedule adherence quantified by subtracting the observed arrival time from scheduled arrival time for each bus at the upstream timing point stop (i.e. before entering the section). The next section briefly explains how the study

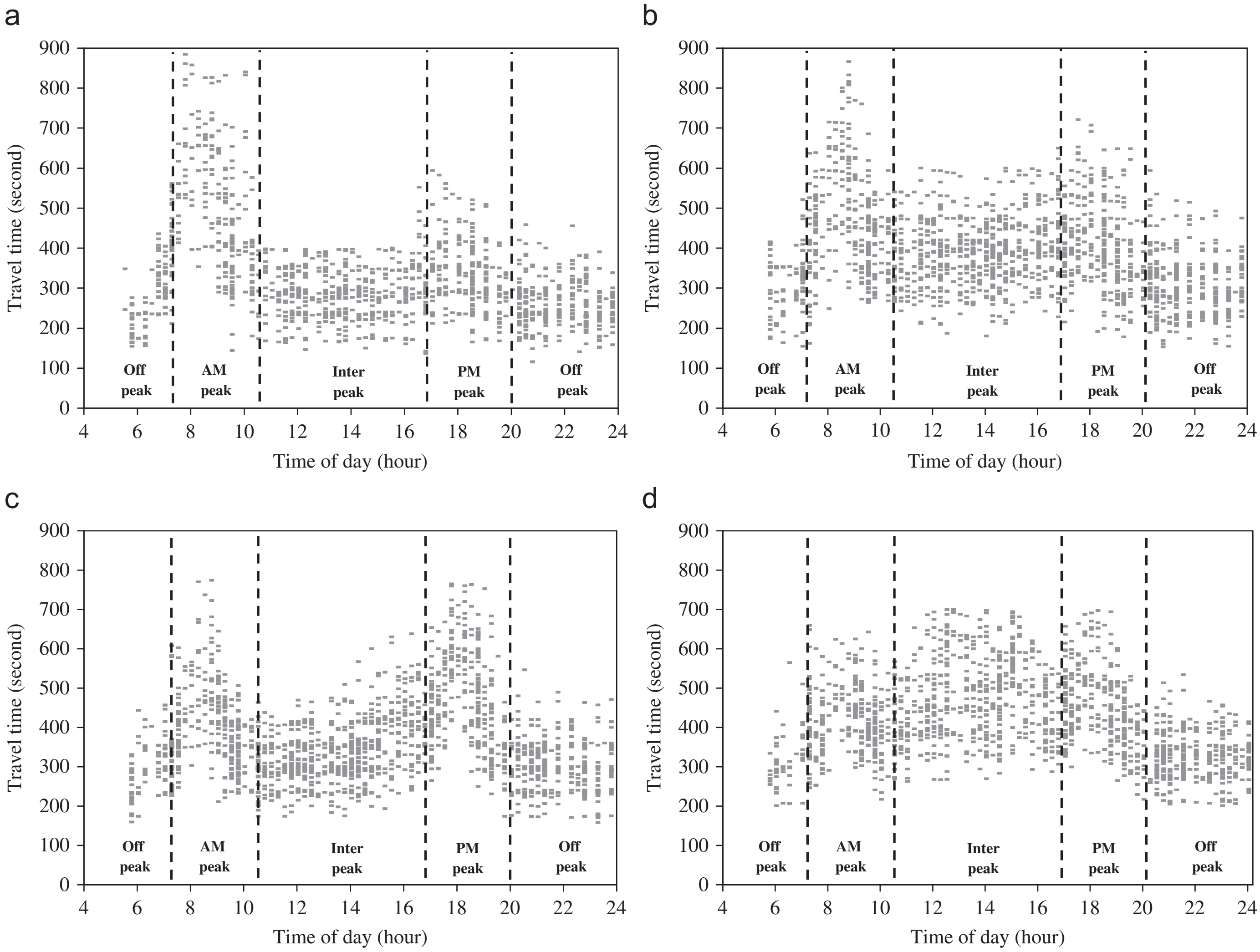


Fig. 4. Travel time observations of the four sections across the day. (a) Section 1, (b) Section 2, (c) Section 3, and (d) Section 4.

selected the input variables and the aggregation period which lead to the most accurate predictions.

4.2. Input variable (feature) selection

The performance of a neural network often deteriorates when the number of input variables increases, i.e. the dimension of input space increases. This has been referred as the *curse of dimensionality* in the literature (Bellman, 1961; Bishop, 1995). This phenomenon may lead to the choice of irrelevant input variables for modelling, which may unnecessarily increase model complexity and hence lead to a poor generalization. Increasing the number of input variables also leads to neural networks needing more training examples to effectively understand the input–output relationship. However, dataset size is limited in many applications.

Feature selection techniques have been applied in many problems to reduce dimensionality (Srivastava et al., 2000; Khosravi et al., 2007) to either the original input variables or to a set of features constructed through combining original input variables. These techniques discard those inputs which carry little useful information to solve the problem (Peng et al., 2005). One approach in this realm is principal component analysis (PCA) which combines inputs together to produce a smaller set of features, called principal components (Jolliffe, 2002). Principal components are linear combinations of the original variables, and are orthogonal to each other, so there is no redundant information. However, the results of this technique will never give the importance of original input variables, and their contribution level to the variations of the dependent variable.

An alternative approach to identify irrelevant input variables and to explore the impact of individual input variables on a dependent variable is to use statistical techniques such as a regression analysis (Li, 2006). These techniques compensate for the inability of the neural network approach to provide insight into the statistical significance of particular explanatory variables. Mazloumi et al. (2009) used the same dataset (as used in this paper) to evaluate the value of traffic flow data in predicting bus travel time and employed a regression analysis to identify the input variables for their neural network prediction model. It was suggested that to predict bus travel time for each section, the input variables should include the average of saturation degree values in the last 15 min interval prior to the departure of each bus (i.e. aggregation period equals to 15 min) from the upstream timing point stop along with the schedule adherence at the upstream timing point. Noteworthy is that all these input values are available before each bus enters each route section. The next section adopts these variables for modelling purposes.

Table 1
The effect of different numbers of hidden neurons on $g(x_i;w_o)$performance (RMSE in seconds).

Section	Number of hidden neurons									
	1	2	3	4	5	6	7	8	9	10
1	82	81	**80**	117	82	123	82	146	84	85
2	126	98	**92**	92	106	95	102	116	104	96
3	88	**76**	79	84	92	94	100	105	90	97
4	101	102	**91**	102	101	103	129	113	97	101

Note: $g(x_i;w_o)$predicts average travel time and the best model results are underlined for each section.

4.3. Development of neural network models

The proposed method works with two neural networks. The first network $g(x_i;w_o)$ predicts the average travel time given a certain set of input valuesx_i, whereas the second neural network $f(x_i;v_o)$ predicts the variance imposed by training data noise. Following a conventional approach to train neural networks, we randomly split up the travel time dataset of each section into a training dataset and a testing dataset. Out of the 1800 travel time observations available for each section, 80% were randomly selected for training the networks, and the remaining 20% were set aside for testing the networks. To make sure that the prediction results are not biased to certain travel times, this task is undertaken 10 times (i.e. we randomly select the training and testing sets 10 times). Therefore, the results reported hereafter are the average on 10 different training sets (when developing the models in this section) and on 10 testing sets (when testing the models in Section 5).

To develop a neural network, two features of the network have to be determined including the number of hidden layers and the number of neurons in each layer. This is a trade-off between model estimation capability and its generalizing ability. As the number of hidden layer neurons increases, the model becomes more prone to over-fitting of the data hence to poor generalization. On the other hand, a small number of neurons in a model may not be sufficient to efficiently describe the complexity of the underlying problem.

To assist in determining the optimum number of hidden neurons, constructive/destructive algorithms (also known as growing and pruning algorithms) and evolution techniques can be employed. The former algorithms start with an extreme network (either small or large) and neurons are added or removed step by step until a predetermined criterion is met. The problem with constructive algorithms is that they are usually trapped in local minima and often lead to big networks (Angeline et al., 1994). The drawback of destructive algorithms is also related to the assignment of credit to the structural components of the network in order to decide whether a connection or node must be removed (Garcia-Pedrajas et al., 2003). The problematic issue with evolutionary based techniques is their massive computation requirements that make them impractical in many real world applications (Khosravi et al., 2010a).

The networks developed in this paper consist of one hidden layer neuron that has been shown to be sufficient to closely map any relationship (Bishop, 1995). To determine the number of hidden layer neurons, a traditional approach is the widely accepted

Table 2
The effect of different numbers of hidden neurons on $f(x_i;v_o)$performance (RMSE in seconds^2).

Section	Number of hidden neurons									
	1	2	3	4	5	6	7	8	9	10
1	14,340	**13,727**	14,312	14,035	14,613	14,102	14,449	14,809	14,562	14,741
2	14,066	14,075	**13,058**	13,260	14,783	13,432	14,012	14,374	14,279	13,076
3	8656	8767	**7746**	8574	8008	8463	8446	8476	7807	8216
4	19,466	**18,783**	19,102	19,450	19,134	19,808	19,581	19,295	18,812	19,279

Note: $f(x_i;v_o)$ predicts the input data variance and the best model results are underlined for each section.

trial and error process. In this study, we use a k-fold cross-validation method (k=5) applied on the training dataset, to test several network structures with differing numbers of hidden layer neuron. For each model, the root mean squared error (RMSE) is used to quantify the error, and those models with the least error are selected

$$RMSE=\sqrt{\frac{\sum_{i=1}^{N}(t_i-y_i)^2}{N}} \tag{14}$$

Table 1 reports the change in the performance of $g(x_i;w_o)$ (which predicts the average travel time) with respect to the number of hidden layer neurons adopted. Similarly, the results reported in Table 2 show the sensitivity of$f(x_i;v_o)$ (which predicts the variance of travel times) to the number of hidden layer neurons. In each table, the best model results are underlined for each section.

5. Results

To predict the average travel time for each section, the best $g(x_i;w_o)$model is selected on the basis of the results reported in Table 1. To test the predictive ability of each model on unseen data, each model is now applied on the testing dataset (i.e. 20% of the travel times that have been put aside). The results reported in Table 3 illustrate model performance (in terms of RMSE) by time period. Except for Section 4, the poorest model performance is in peak periods. In Section 4 there is also a peak in inter peak travel times (clearly visible in Fig. 4). The higher travel times during that period (when the RMSE reaches 113 s), may be explained by factors which are not captured in the model.

We now turn to quantify separately the effect of model structure $\sqrt{\sigma_m^2}$and input data noise $\sqrt{\sigma_e^2}$ on prediction uncertainty. In addition, total prediction uncertainty is also computed as $\sqrt{\sigma_e^2+\sigma_m^2}$ to better understand the contribution of each source of uncertainty to total uncertainty. Table 4 shows the results of this investigation, where model structure is shown to have a minor effect on total uncertainty compared to that produced by training data noise. For example, consider the results of Section 1, access the whole day (that is the 'All day' column). While training data noise gives rise to an uncertainty of 94 s, the total uncertainty (considering uncertainty from both sources) is only 95 s highlighting that the model structure is only a minor component of total uncertainty.

To quantitatively explore the performance of PIs, one approach is to compare the prediction interval coverage probability (PICP) to the expected coverage probability. Mathematically, PICP is defined as

$$PICP=\left(\frac{1}{n}\sum_{i=1}^{n}c_i\right)\times 100 \tag{15}$$

where $c_i=1$ if $y_i\in[L(x_i),U(x_i)]$, otherwise $c_i=0$. Corresponding to an input value x_i, y_i is the model prediction, and $L(x_i)$ and $U(x_i)$ are the lower and upper bounds of the prediction interval. It is expected that the 95% prediction intervals encompass the observed travel times on 95% of the occasions. As seen in Table 5, application of the method outlined in this paper leads to very robust prediction intervals capable of encompassing observed travel times in accordance with expectations. However, there are persistent signs of overestimation in off-peak hours where the PICP values are higher than 95%. We theorize that the proposed method results could be improved if data on other potentially relevant explanatory variables, such as passenger demand, were available.

6. Summary and conclusion

Despite existing reports of the successful exploitation of neural networks, the predictions made by neural networks are always prone to uncertainty. In this study, different sources of uncertainty associated with neural network outcomes were discussed,

Table 3
The performance of $g(x_i;w_o)$in terms of RMSE (seconds).

Section	AM peak 7–10 am	Inter peak 10 am to 4 pm	PM peak 4–7 pm	Off-peak 7 pm to 7 am	All day
1	140	65	108	59	90
2	133	81	101	78	95
3	94	74	94	69	81
4	105	113	94	88	97

Note: $g(x_i;w_o)$predicts average travel time given the input vector x_i.

Table 5
The obtained coverage probability of the 95% prediction interval (percent).

Section	AM peak 7–10 am	Inter peak 10 am to 4 pm	PM peak 4–7 pm	Off-peak 7 pm to 7 am	All day
1	97	96	92	98	97
2	93	97	95	98	96
3	93	95	98	98	97
4	94	94	98	99	95

Note: the expected coverage probability is 95%.

Table 4
The contribution of different sources of prediction uncertainty (seconds).

Route section	Source of uncertainty[a]	AM peak 7–10 am	Inter peak 10 am to 4 pm	PM peak 4–7 pm	Off-peak 7 pm to 7 am	All day
1	Model structure	46	10	14	13	18
	Input data noise	141	80	96	74	94
	Total	148	80	97	76	95
2	Model structure	13	6	12	5	9
	Input data noise	106	95	103	83	96
	Total	106	96	103	84	96
3	Model structure	32	12	58	13	24
	Input data noise	83	80	86	77	82
	Total	89	81	104	78	85
4	Model structure	24	20	19	23	21
	Input data noise	108	117	121	105	113
	Total	111	119	122	107	115

[a] Model structure impact is measured by$\sqrt{\sigma_m^2}$—used for confidence interval construction. Input data noise impact is quantified by$\sqrt{\sigma_e^2}$. Total uncertainty is $\sqrt{\sigma_e^2+\sigma_m^2}$—used for prediction interval construction.

including uncertainty arising from inherent noise in input data, and that due to model structure. Two alternative measures were also introduced to quantify how different uncertainty sources contribute to total prediction uncertainty. Confidence intervals measure the uncertainty in model structure, while prediction intervals are concerned with total variance. To construct confidence intervals corresponding to each input value set, a Bootstraping analysis was employed, while a maximum likelihood based approach was used to quantify the variance in input data.

The proposed methodology was applied in the context of predicting bus travel time over different sections of a bus route in Melbourne, Australia. The travel time for each section of the route was modelled as a function of traffic saturation degree values and a measure of schedule adherence. The results of this application revealed that a major portion of uncertainty in predictions is related to noise in input data. This could be related to the stochastic nature of some key factors like signal delay or dwell time. Overall, the proposed approach has proved to provide robust prediction intervals around prediction values.

The approach proposed here to construct prediction intervals is effective (in terms of accuracy) and efficient (from an ease of implementation perspective). As a result, it can be regarded as a promising means for both researchers and practitioners to statistically explore neural network outputs. The prediction intervals it provides can be disseminated to travellers through traveller information systems to enable them to efficiently plan their trips. In the context of public transport, the proposed framework can be a promising means to help operators in developing timetables and defining slack times to maximize on-time performance. Comparative research exploring other techniques for confidence estimation is a promising direction for future research. Meanwhile, inclusion of other variables affecting bus travel time, such as passenger demand, may lead to more reliable model outcomes and more accurate prediction intervals.

Acknowledgment

The authors would like to acknowledge Ventura Bus Company and VicRoads for supplying the GPS and SCATS data, respectively, for this research.

References

Angeline, P.J., Saunders, G.M., Pollack, J.B., 1994. An evolutionary algorithm that constructs recurrent neural networks. IEEE Transactions on Neural Networks 5, 54–65.

Bellman, R., 1961. Adaptive Control Processes: A Guided Tour. Princeton University Press.

Bishop, C.M., 1995. Neural networks for pattern recognition. Clarendon Press, Oxford.

Cai, M., Yin, Y., Xie, M., 2009. Prediction of hourly air pollutant concentrations near urban arterials using artificial neural network approach. Transportation Research Part D 14 (1), 32–41.

Chen, M., Yaw, J., Chien, S.I., Liu, X., 2007. Using automatic passenger counter data in bus arrival time prediction. Journal of Advanced Transportation 41 (3), 267–283.

Chien, S., Hwang, H., Pei, T., 1994. Using neural networks to synthesize origin-destination flows in a traffic circle. Transportation Research Record 1457, 134–142.

Dharia, A., Adeli, H., 2003. Neural network model for rapid forecasting of freeway link travel time. Engineering Applications of Artificial Intelligence 16 (7–8), 607–613.

Dia, H., Panwai, S., 2007. 'Modelling drivers' compliance and route choice behavior in response to travel information. Nonlinear Dynamics 49, 493–509.

Dia, H., Rose, G., 1998. Development and evaluation of neural network freeway incident detection models using field data. Transportation Research Part C 5, 313–331.

Dybowski, R., Roberts, S.J., 2001. Confidence intervals and prediction intervals for feed-forward neural networks. In: Dybowski, R., Gant, V. (Eds.), Clinical Applications of Artificial Neural Networks'. Cambridge University Press, pp. 298–327.

Efron, B., 1982. The jacknife, the bootstrap, and other resampling plans, Society for Industrial and Applied Mathematics. SIAM, Philadelphia, PA, USA.

Fu, L., Rilett, L.R., 1998. Expected shortest paths in dynamic and stochastic traffic networks. Transportation Research Part B 32 (7), 499–516.

Garcia-Pedrajas, N., Hervas-Martinez, C., Munoz-Perez, J., 2003. COVNET: a cooperative coevolutionary model for evolving artificial neural networks. IEEE Transactions on Neural Networks 14 (3), 575–596.

Hagan, M.T., Menhaj, M., 1999. Training feed-forward networks with the Marquardt algorithm. IEEE Transactions on Neural Networks 5 (6), 989–993.

Heskes, T., 1997. Practical confidence and prediction interval. Advances in Neural Information Processing Systems 9, 176–182.

Hinton, G., 1987. Learning translation invariant recognition in a massively parallel networks, in: 'PARLE Parallel Architectures and Languages Europe', Edited by, pp. 1–13.

Jain, B.A., Nag, B.N., 1997. A performance evaluation of neural network decision models. Journal of Management Information Systems 14, 201–216.

Jeong, R.,Rilett, R.L., 2004., Bus arrival time prediction using artificial neural network model. In: Proceedings of the IEEE Intelligent Transportation System Conference, Washington, D.C., USA.

Jolliffe, I.T., 2002. Principal Component AnalysisSpringer, New York.

Kalaputapu, R., Demetsky, M.J., 1995. Modelling schedule deviations of buses using automatic vehicle location data and artificial neural networks. Transportation Research Record 1497, 44–52.

Khosravi, A., Martinez, T., Melendez, J., Colomer, J., Sanchez, J., 2007. Integrating a feature selection algorithm for classification of voltage sags originated in transmission and distribution networks. In: Proceeding of the 2007 conference on Artificial Intelligence Research and Development, IOS Press.

Khosravi, A., Nahavandi, S., Creighton, D., 2010a. A prediction interval-based approach to determine optimal structures of neural network metamodels. Expert Systems with Applications 37 (3), 2377–2387.

Khosravi, A., Nahavandi, S., Creighton, D., 2010b. 'Construction of optimal prediction intervals for load forecasting problems. IEEE Transactions on Power Systems 25 (3), 1496–1503.

Kim, W., Rilett, L., 2005. Improved transit signal priority system for networks with nearside bus stops. Transportation Research Record: Journal of the Transportation Research Board 1925, 205–214.

Lee, S.S., Lee, S.H., Oh, Y.T., Choi, K.C., 2002. Development of degree of saturation estimation models for adaptive signal systems. KSCE Journal of Civil Engineering 6 (3), 337–345.

Li, R., 2006. Enhancing motorway travel time prediction models through explicit incorporation of travel time variability, Ph.D. thesis, Monash University, Melbourne, Australia.

Liu, H., 2008. Travel time prediction for urban networks, Ph.D. thesis, Delft University of Technology, the Netherlands.

Liu, H., Van Zuylen, H.J., Van Lint, H., Chen, Y., Zhang, K., 2005. Prediction of urban travel times with intersection delays. In: Proceedings of the Eighth International IEEE Conference on Intelligent Transportation Systems, Vienna, Austria.

Mazloumi, E., Currie, G., Rose, G., 2010. Using GPS data to gain insight into public transport travel time variability. Journal of Transportation Engineering 136, 623–6317, 623–631.

Mazloumi, E., Currie, G., Rose, G., Sarvi, M., 2009. Using SCATS data to predict bus travel time. In: Proceedings of the 32nd Australian Transport Research Forum (ATRF), Auckland, New Zealand.

Mukkamala, S., Sung, A.H., 2003. Feature selection for intrusion detection using neural networks and support vector machines. Transportation Research Record 1823, 33–39.

Nix, A.D., Weigend, A.S., 1994. Estimating the mean and variance of the target probability distribution. In: Proceedings of the IEEE International Conference on Neural Networks.

Papadopoulos, G., Edwards, P.J., Murray, A.F., 2001. Confidence estimation methods for neural networks: a practical comparison. IEEE Transactions on Neural Networks 12 (6), 1278–1287.

Park, T., Lee, S., 2004. A Bayesian approach for estimating link travel time on urban arterial road network. In: Computational Science and Its Applications – ICCSA 2004, Edited by, pp. 1017–1025.

Pattanamekar, P., Park, D., Rilett, L.R., Lee, J., Lee, C., 2003. Dynamic and stochastic shortest path in transportation networks with two components of travel time uncertainty. Transportation Research Part C: Emerging Technologies 11 (5), 331–354.

Peng, H., Long, F., Ding, C., 2005. Feature selection based on mutual information: criteria of max-dependency, max-relevance, and min-redundancy. IEEE Transactions on Pattern Analysis and Machine Intelligence 27 (8), 1226–1238.

Qi, M., Zhang, G.P., 2001. An investigation of model selection criteria for neural network time series forecasting. European Journal of Operational Research 132 (3), 666–680.

Shiva Nagendra, S.M., Khare, M., 2004. Artificial neural network based line source models for vehicular exhaust emission predictions of an urban roadway. Transportation Research Part D 9 (3), 199–208.

Smith, B.L., Demetsky, M.J., 1995. Short term traffic flow prediction: neural network approach. Transportation Research Record 1453, 98–104.

Srivastava, L., Singh, S.N., Sharma, J., 2000. Comparison of feature selection techniques for ANN-based voltage estimation. Electric Power Systems Research 53 (3), 187–195.

Tillema, F., Van Zuilekom, K.M., Van Maarseveen, M.F.A.M., 2006. Comparison of neural networks and gravity models in trip distribution. Computer-Aided Civil and Infrastructure Engineering 21, 104–119.

Van Hinsbergen, C.P.I., Van Lint, J.W.C., Van Zuylen, H.J., 2009. Bayesian committee of neural networks to predict travel times with confidence intervals. Transportation Research Part C 17 (5), 498–509.

Van Lint, J.W.C., 2004. Reliable travel time prediction for freeways, Ph.D. thesis, Delft University of Technology, the Netherlands.

Van Lint, J.W.C., Hoogendoorn, S.P., Van Zuylen, H.J., 2005. Accurate freeway travel time prediction with state-space neural networks under missing data. Transportation Research Part C 13 (5–6), 347–369.

Xie, Y., Lord, D., Zhang, Y., 2007. Predicting motor vehicle collisions using Bayesian neural networks: an empirical analysis. Accident Analysis & Prevention 39, 922–9335, 922–933.

Yang, J., Lu, J.J., Gunaratne, M., Dietrich, B., 2006. Modeling crack deterioration of flexible pavements: comparison of recurrent Markov chains and artificial neural networks. Transportation Research Record 1974, 18–25.