Decentralised Regression Model for Intelligent Forecasting in Multi-Agent Traffic Networks

Jelena Fiosina *

Abstract The distributed nature of complex stochastic systems, such as traffic networks, can be suitably represented by multi-agent architecture. Centralised data processing and mining methods experience difficulties when data sources are geographically distributed and transmission is expensive or not feasible. It is also known from practice that most drivers rely primarily on their own experience (historical data). We consider the problem of decentralised travel time estimation. The vehicles in the system are modelled as autonomous agents consisting of an intellectual module for data processing and mining. Each agent uses a local linear regression model for prediction. Agents can adjust their model parameters with others on demand, using the proposed resampling-based consensus algorithm. We illustrate our approach with case studies, considering decentralised travel time prediction in the southern part of the city of Hanover (Germany).

Key words: Regression, parameter estimation, distributed data processing and mining, multi-agent systems

1 Introduction

Currently, travel time information plays a significant role in transportation and logistics and is applied in various fields and for different purposes. From the travellers' viewpoint, travel time information helps to select the most optimal route, which minimizes delays. In logistics, accurate travel time estimation helps to reduce transport

Clausthal University of Technology,
Institute of Informatics,
Julius-Albert Str. 4, D-38678, Clausthal-Zellerfeld, Germany
e-mail: Jelena.Fiosina@gmail.com

* This work was supported by European Commission FP7 People Marie Curie IEF Career Development Grant. We thank also Prof. J.P. Müller for his valuable comments.

delivery costs by avoiding congested route sections, and helps also to increase the service quality of commercial delivery. For traffic managers, travel time information is an important metric of the operational efficiency of the traffic system [8].

The distributed nature of traffic networks, which are complex stochastic systems, can be suitably represented by multi-agent architecture. Centralised data processing and mining methods experience difficulties when data sources are geographically distributed and transmission is very expensive or not feasible. From practice, it is also known that most drivers rely primarily on their own experience (historical data). We use a decentralised multi-agent system to model autonomous data mining of vehicle-agents, on the based of not only the centrally processed information but also historical data.

To achieve their common goals effectively, agents should cooperate with so-called gossip scenarios. We suppose that each agent autonomously estimates its parameters and then (on demand) adjusts them, by communicating with randomly chosen neighbour agents from the transmission radius. This random selection of neighbour agents is performed several times in different combinations, thereby performing the so-called resampling procedure. The obtained weighted resampling estimators are more reliable and robust in the case of nuisance observations. This detail is important because some agents can provide unreliable estimates, and it is better to average not all possible estimates, but a random subset of local estimates [1].

In this study, we do not deal with decision-making problems as we did in the study reported in [6]. The estimated travel times serve as input data to the decision-making module, which can also be incorporated into the intelligent agents.

The contribution of this study is toward the following: 1) development of the structure of a regression model for travel time forecasting; 2) development of an decentralised resampling-based regression parameter adjustment algorithm for streaming data; 3) application of the suggested algorithm for real-time data with the objective of demonstrating its efficiency.

The remainder of this paper is organised as follows. The second section contains a description of related work. The third section formulates the problem. In the fourth section, we present the multivariate linear regression model and the iterative least square algorithm for parameter estimation. The fifth section presents the resampling-based consensus algorithm of parameter adjustment. The sixth section presents case studies. The last section presents the conclusion.

2 Related Work

The need for research in the transportation area was considered by many authors. In this area, travel time prediction is one of the important challenges. From the architectural viewpoint, centralised and distributed approaches for travel time forecasting were considered. The centralised approach was applied in various intelligent transport systems, such as in-vehicle route guidance, or advanced traffic management systems. A detailed overview is presented in [8]. The estimation of actual travel time by using vehicle-to-vehicle communication is described in [9]. In contrast to centralisation, it was demonstrated that the representation of complex systems, such as traffic networks, in the form of decentralised multi-agent systems is of fundamental importance [4]. Decision-making in multi-agent traffic systems was considered in [6]. An example of the architecture of a distributed traffic system was presented in [7].

From the algorithmic viewpoint, numerous data mining and processing techniques were suggested for travel time prediction. Statistical methods, such as regression and time series, and artificial intelligence methods, such as neural networks, are successfully being implemented for similar problems. However, travel time is affected by a range of different factors. Thus, accurate prediction of travel time is difficult and needs considerable historical data. Understanding the factors affecting travel time is essential for improving prediction accuracy [8]. Travel time prediction for bus routes using a linear regression model was employed in [10].

In this study, we propose to use a decentralised regression model for solving the travel time prediction problem for streaming data. A similar decentralised approach was suggested in [11] for the estimation of the parameters of a wireless network.

3 Problem Formulation

We consider a traffic network with several vehicles, represented as autonomous agents, which predict their travel time on the basis of their current observations and history. Each agent locally estimates the parameters of the same traffic network. In order to make a forecast, each agent constructs a regression model, which explains the manner in which different explanatory variables (factors) influence the travel time. A detailed overview of such factors is provided in [8]. The following information is important for predicting the travel time [10]: average speed before the current segment, number of stops, number of left turns, number of traffic lights, average travel time estimated by traffic management centres (TMC). We should also take into account the possibility of an accident, network overload (”rush hour”) and weather conditions.

Let us consider a vehicle, whose goal is to drive through the defined road segment under specific environment conditions (day, time, city district, weather, etc.). Let us suppose, that it has no or little experience of driving in such conditions. For accurate travel time estimation, it contacts other traffic participants, which send their estimated parameters to it. The forecasting procedure of one such vehicle is shown in Fig. 1.

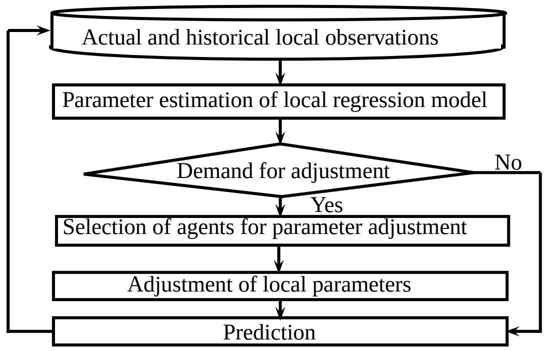

Fig. 1 Algorithm for local travel time prediction by an individual agent

4 Local Recursive Parameter Estimation

We describe the formal model, which is incorporated into each agent's local data processing and mining module. For this purpose, we first consider the classical multivariate linear regression model:

$$\mathbf{Y} = \mathbf{X}\beta + \varepsilon; \tag{1}$$

where $\mathbf{Y}$ is an $n \times 1$ vector of dependent variables (here, actual travel times); β is an $m \times 1$ vector of unknown parameters of the system to be estimated; ε is an $n \times 1$ vector of random errors; $\mathbf{X}$ is an $n \times m$ matrix of explanatory variables. The rows of the matrix $\mathbf{X}$ correspond to observations and the columns correspond to factors (here, length of the route, average speed in the system, average number of stops in the system, congestion level, etc.).

We suppose that $\{\varepsilon_i\}$ are mutually independent, have zero expectation, $E[\varepsilon] = 0$, and equal variances, $V[\varepsilon] = \sigma^2\mathbf{I}$, where $\mathbf{I}$ is an $n \times n$ identity matrix.

The well-known least square estimator (LSE) $\mathbf{b}$ of β is:

$$\mathbf{b} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Y}. \tag{2}$$

After the estimation of the parameters β, we forecast travel time for a certain k-th future time moment:

$$E(Y_k) = \mathbf{x}_k\mathbf{b}, \tag{3}$$

where, $\mathbf{x}_k$ is a vector of observed values of explanatory variables for the kth future time moment. The described estimation procedure, (2), requires information about all observations, i.e., the complete matrix $\mathbf{X}$. In practice, for real-time streaming data, the estimation is performed iteratively, being updated after each new observation. The recurrent iterative method for the LSE was suggested in [2], [3]. This method assumes the recalculation of system parameters for each new observation.

Let us briefly describe the key aspects of this algorithm. Let $\mathbf{b}_t$ be the estimate after t first observations. After receiving the $t+1$-th observation, we recalculate the estimates of β (Y_{t+1} - value of a dependent variable; and $\mathbf{x}_{t+1}$ - values of explanatory variables):

$$\mathbf{b}_{t+1} = \mathbf{b}_t + \mathbf{K}_{t+1}(Y_{t+1} - \mathbf{x}_{t+1}\mathbf{b}_t),\ t = 0,1,\ldots, \tag{4}$$

where $\mathbf{K}_{t+1}$ is an $m \times 1$ vector of proportionality, smoothness, or compensation.

From (4), one can observe, that $\mathbf{b}_{t+1}$ is represented as a sum of the previous estimate $\mathbf{b}_t$ and the correction term $\mathbf{K}_{t+1}(Y_{t+1} - \mathbf{x}_{t+1}\mathbf{b}_t)$. Formula (4)is based on exponential smoothness, an adaptive forecasting method [3].

To calculate $\mathbf{K}_{t+1}$, we need values of matrices $\mathbf{A}_t$ and $\mathbf{B}_t$, obtained after the last t-th iteration. $\mathbf{A}_t$ and $\mathbf{B}_t$ are square $m \times m$ matrices. $\mathbf{B}_t$ is equal to $(\mathbf{X}^T\mathbf{X})^{-1}$ if this matrix exists, else it is equal to a pseudo-inverse matrix. Matrix $\mathbf{A}_t$ is a projection matrix, therefore, if $\mathbf{x}_{t+1}$ is the linear combination of the rows of matrix $\mathbf{X_t}$, its projection is equal to zero: $\mathbf{x}_{t+1}\mathbf{A}_t = 0$. Starting the algorithm we set the following initial values $\mathbf{A}_0 = \mathbf{I}$, $\mathbf{B}_0 = \mathbf{0}$, $\mathbf{b}_0 = \mathbf{0}$.

If the condition $\mathbf{x}_{t+1}\mathbf{A}_t = 0$ is satisfied, then

$$\begin{aligned}
&\mathbf{B}_{t+1} = \mathbf{B}_t - (1 + \mathbf{x}_{t+1}\mathbf{B}_i\mathbf{x}_{t+1}^T)^{-1}\mathbf{B}_i\mathbf{x}_{t+1}^T(\mathbf{B}_t\mathbf{x}_{t+1}^T),\\
&\mathbf{A}_{t+1} = \mathbf{A}_t,\ \mathbf{K}_{t+1} = (1 + \mathbf{x}_{t+1}\mathbf{B}_t\mathbf{x}_{t+1}^T)^{-1}\mathbf{B}_t\mathbf{x}_{t+1}^T,
\end{aligned}$$

otherwise

$$\begin{aligned}
&\mathbf{B}_{t+1} = \mathbf{B}_t - (\mathbf{x}_{t+1}\mathbf{A}_t\mathbf{x}_{t+1}^T)^{-1}\left((\mathbf{B}_i\mathbf{x}_{t+1}^T)(\mathbf{A}_t\mathbf{x}_{t+1}^T)^T + (\mathbf{A}_t\mathbf{x}_{t+1}^T)(\mathbf{B}_t\mathbf{x}_{t+1}^T)^T\right) +\\
&+(\mathbf{x}_{t+1}\mathbf{A}_t\mathbf{x}_{t+1}^T)^{-2}(1 + \mathbf{x}_{t+1}\mathbf{B}_t\mathbf{x}_{t+1}^T)(\mathbf{A}_t\mathbf{x}_{t+1}^T)(\mathbf{A}_t\mathbf{x}_{t+1}^T)^T,\\
&\mathbf{A}_{t+1} = \mathbf{A}_t - (\mathbf{x}_{t+1}\mathbf{A}_t\mathbf{x}_{t+1}^T)^{-1}\mathbf{A}_i\mathbf{x}_{t+1}^T(\mathbf{A}_t\mathbf{x}_{t+1}^T)^T,\\
&\mathbf{K}_{t+1} = (\mathbf{x}_{t+1}\mathbf{A}_t\mathbf{x}_{t+1}^T)^{-1}\mathbf{A}_t\mathbf{x}_{t+1}^T.
\end{aligned}$$

5 Parameter Adjustment Algorithm

We are new going to introduce a notation for the local regression model of each of the s agents in the network. We use index (i,t) for the variables in formula (1), to refer to the i-th agent at time t:

$$\mathbf{Y}(i,t) = \mathbf{X}(i,t)\beta + \varepsilon(i,t),\ i = 1,\ldots,s. \tag{5}$$

Following (4), the i-th agent of s calculates the estimates $\mathbf{b}(i,t)$ of β and predicts the travel time $E[Y(i,t+1)]$ for the future time moment $t+1$, using (3).

Prior to forecasting, some agents may adjust their locally estimated parameters with other traffic participants. Let us describe this adjustment procedure more precisely.

First, the agent selects the other agents from a given transmission radius, contacts them, and requests them to send their estimated parameters. The agents can be in different situations and their observation may contain outliners. In order to make the adjustment procedure more reliable and robust to outliners, the agent perform the described selection several times in different combinations, forming so-called resamples from the available agents [1].

We implement N realisations of the following resampling procedure for the i-th agent. At the realisation q, the agent receives the parameter estimates of r randomly chosen neighbour agents. Let vector $\mathbf{L}_i^q$ contain the indices of the selected agents, $|\mathbf{L}_i^q| = r$.

The next step is the adjustment of the parameters. The agent that initialised the adjustment process considers the weighted estimates of other agents. The weights are time-varying and show the reliability level of each agent, depending on its forecasting experience as well as some other factors. Let $\mathbf{c}_i^{*q}(t)$ be a $1 \times r$ vector of the weights at the q-th realization at time t, $i = 1, \ldots, s$.

According to the logic of constructing discrete-time consensus, we assume that $\mathbf{c}_i^{*q}(t)$ is a stochastic vector for all t (the sum of its elements is equal to 1).

$$\mathbf{b}^{*q}(i,t+1) = \sum_{j=1}^{r} c_{i,j}^{*q}(t)\mathbf{b}(\mathbf{L}_{i,j}^q, t), \tag{6}$$

where $\mathbf{b}^{*q}(i,t)$ is the adjusted estimate of β calculated by the i-th agent at the q-th resampling realisation, $i = 1, \ldots, s$, $q = 1, \ldots, N$.

Finally, the resampling estimator is obtained as an average over all N realisations:

$$\mathbf{b}^R(i,t+1) = \frac{1}{N}\sum_{q=1}^{N} \mathbf{b}^{*q}(i,t+1). \tag{7}$$

The algorithm is a combination of the iterative LSE algorithm and resampling-based parameter adjustment. This adjustment procedure aims to increase the reliability of the estimates, especially for insufficient or missing historical data, and to contribute to the overall estimation accuracy [11].

6 Case Studies

We simulate a traffic network in the southern part of Hanover(Germany). The network contains three parallel and five perpendicular streets, creating fifteen intersections with a flow of approximately 5000 vehicles per hour. The vehicles solve a travel time prediction problem. They receive information about the centrally estimated system variables (such as average speed, number of stops, congestion level, etc.) for this city district from TMC, combine it with their historical information, and make adjustments adjust according to the information of other participants using the presented consensus algorithm. In this case, regression analysis is an essential part of the local travel time prediction process. We consider the local model (5) and implement the parameter adjustment algorithm (7).

The factors are listed in Table 1 (left). To improve the quality of the regression model, some non-linear transformations of the factors are performed. The resulting regression model remains linear by parameters, but becomes non-linear by factors.

All parameters satisfy the significance test, as a measure of influence to the target variable, i.e., travel time.

We simulate ten agents and train them on the observations taken from the available dataset of size 1790. We conduct three different experiments and compare the results using analysis of variance and adjusted coefficient of determination, R^2. This provides a well-known measure of the effectiveness of prediction of future outcomes by the model [5].

Case 1: The agents transmit their observations to the central unit and regularly obtain the updated parameters of the system. This requires the transmission of a large amount of data and is therefore very expensive. Nevertheless, this model provides the best prediction results (see Table 1)(right).

Case 2: The agents estimate the parameters of the same system locally without cooperation. This case requires no transmission costs; however, it assumes that each agent has a special data processing block for performing calculations. The main problem is that each agent builds its own model using a considerably small amount of data. The data should be homogeneous because each agent estimates the parameters of the same system by using different data. These estimates should converge to the parameter values estimated in Case 1. The experiments show that for historical data sets smaller than 25, the model parameters cannot be estimated well (see Table 1(right), worth agent). To estimate the quality of the *i*-th agent's model, the cross-validation technique was used for testing the model with the observations of other agents.

Case 3: The agents estimate the parameters of the same system locally and use cooperation mechanisms to adjust their parameter values with other agents. The amount of transmitted information is lower than in Case 1, because not all the data but only locally estimated parameters are transmitted. This case has the same assumptions concerning intelligent module and data, as in Case 2. Agents with more experience (historical dataset) help new agents with less experience to make better estimates. The prediction experience is used as the weights (reliability level) of agents. The results show that this cooperation helps to improve performance, especially for the less experienced agents. In this case four agents have less experience (data sets smaller than $n = 15$) and six agents are more experienced. The other parameters are r=3, N=10. This case gives better results than Case 2, but slightly worse results than Case 1, owing to the loss of some information in the process of averaging and the relatively small data sets of each agent (see Table 1)(right).

Table 1 Factors and corresponding parameters **b** (left). Efficiency criteria (right)

Var.	Description	Mod.	Koef.	Est. value
Y	travel time (min);	Y		
X_1	route length(km)	X_1	b_1	0.614
X_2	avg. speed in system (km/h)	X_2	b_2	-0.065
X_3	avg. number of stops(units/min)	X_3^2	b_3	0.09
X_4	congestion level(Veh/h)	$\sqrt{X_4}$	b_4	0.159
X_5	traffic lights in the route (num)	X_5^2	b_5	0.241
X_6	left turns in the route (num)	X_6^2	b_6	-0.058

Case	R^2 of the whole system	R^2 of the worth agent
1	0.66	-
2	0.55	0.28
3	0.64	0.58

7 Conclusions

A problem of decentralised travel time forecasting was considered. A multi-agent architecture with autonomous agents was used for this purpose.

A decentralised linear multivariate regression model was developed to forecast the travel time. The iterative LSE method was used for the regression parameter estimation, which is suitable for streaming data processing. The resampling-based consensus method was suggested for coordinated adjustment of estimates between neighbour agents. We illustrate the efficiency of the suggested approach using simulation with real data from the southern part of Hanover. The experiments show the efficiency of the proposed approach.

Our future work will be continued in three directions: (a) construction of the distributed model of multiple multivariate regression, which allows forecasting of several response variables simultaneously from the same set of explanatory variables (factors), as well as an implementation of such a model for real-time traffic data; (b) application of other regression model types (i.e., general regression models using kernel estimators); (c) modification of the parameter adjustment algorithm (new strategies for the calculation of the reliability level of agents, median resampling approach, etc.).

References

1. Afanasyeva, H., Andronov, A.: On robustness of resampling estimators for linear regression models. Communications in Dependability and Quality Management: An international Journal **9**(1), 5–11 (2006)
2. Albert, A.: Regression and the Moor-Penrose Pseudoinverse. Academic Press, New York and London (1972)
3. Andronov, A., Kiselenko, A., Mostivenko, E.: Forecasting of the development of Regional Transport System. KNZ UrO RAN, Sivtivkar (1991). (In Russian)
4. Bazzan, A., Wahle, J., Kluegl, F.: Agents in traffic modelling - from reactive to social behaviour. Adv. in Artificial Intelligence (LNAI) **1701**, 509–523 (1999)
5. Draper, N., Smith, H.: Applied Regression Analysis. John Wiley and Sons, New York (1986)
6. Fiosins, M., Fiosina, J., Müller, J., Görmer, J.: Agent-based integrated decision making for autonomous vehicles in urban traffic. Adv. in Intelligent and Soft Computing **88**, 173–178 (2011)
7. Lee, W., Tseng, S., Shieh, W.: Collaborative real-time traffic information generation and sharing framework for the intelligent transportation system. Inf. Scienses **180**, 62–70 (2010)
8. Lin, H.E., Zito, R., Taylor, M.A.: A review of travel-time prediction in transport and logistics. In: Proc. of the Eastern Asia Society for Transportation Studies, vol. 5, pp. 1433 – 1448. Hamburg (2005)
9. Malnati, G., Barberis, C., Cuva, C.M.: Gossip: Estimating actual travelling time using vehicle to vehicle communication. In: 4-th Int. Workshop on Intel. Transportation. Hamburg (2007)
10. McKnight, C.E., Levinson, H.S., Kamga, C., Paaswell, R.E.: Impact of traffic congestion on bus travel time in northern new jersey. Transportation Res. Record Journal **1884**, 27–35 (2004)
11. Stankovic, S.S., Stankovic, M.S., Stipanovic, D.M.: Decentralized parameter estimation by consensus based stochastic approximation. IEEE Trans. Automatic Controll **56** (2009)