Selecting the Shortest Itinerary in a Cloud-Based Distributed Mobility Network

Jelena Fiosina and Maksims Fiosins *

Abstract New Internet technologies can considerably enhance contemporary traffic control and management systems (TCMS). Such systems need to process increasing volumes of data available in clouds, and so new algorithms and techniques for statistical data analysis are required. A very important problem for cloud-based TCMS is the selection of the shortest itinerary, which requires route comparison on the basis of historical data and dynamic observations. In the paper we compare two non-overlapping routes in a stochastic graph. The weights of the edges are considered to be independent random variables with unknown distributions. Only historical samples of the weights are available, and some edges may have common samples. Our purpose is to estimate the probability that the weight of the first route is greater than that of the second one. We consider the resampling estimator of the probability in the case of small samples and compare it with the parametric plug-in estimator. The analytical expressions for the expectations and variances of the proposed estimators are derived, which allow theoretical evaluation of the estimators' quality. The experimental results demonstrate that the resampling estimator is a suitable alternative to the parametric plug-in estimator. This problem is very important for a vehicle decision-making procedure to choose route from the available alternatives.

Key words: traffic control and management, future Internet, stochastic graph, shortest route, resampling, small samples, estimation, simulation

---

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

* The research leading to these results has received funding from the European Union Seventh Framework Programme (FP7/2007-2013) under grant agreement No. PIEF-GA-2010-274881.

1 Introduction

Future Internet opportunities open new perspectives on the development of intelligent transport systems (ITS). Technologies such as cloud and grid computing, the Internet of Things concept and ambient intelligence methods allow the development of new applications, to hide the complexity of data and algorithms in the network. This allows traffic participants to run simple applications on their mobile devices, which provide clear recommendations on how they should act in the current situation. These simple applications are based on the aggregation and processing of large amounts of data, which are collected from different traffic participants and objects. These data are physically distributed and available in virtual clouds. This creates a need for innovative data analysis, processing, and mining techniques, which run in clouds and prepare necessary information for end-user applications.

In this study, we deal with route recommendation systems, which are essential applications in cloud-based ITS. This system includes optimization of the booked itinerary with respect to user preferences, time, fuel consumption, cost, and air pollution to provide better (i.e., quicker, more comfortable, cheaper, and greener) mobility. The recommendations can be made on the basis of static information about the network (traffic lights, public transport schedules, etc.) combined with dynamic information about the current situation and historically stored data about traveling under equivalent conditions. If necessarily, the recommendations of other travelers. can be included. Booking the shortest itinerary is a key aspect in many traffic scenarios with different participants: a dynamic multi-modal journey, a simple private drive through a transport network, or smart city logistical operations. We consider an example of driving through a transport network segment considering the time consumption as the optimization criterion in itinerary comparisons and shortest route selection. In this case, the route recommendation is based on the estimates of the travel time along the route.

For this purpose, an artificial transport network is created, the travel times for alternative routes are estimated, and the best route is selected. Different methods of travel-time forecasting can be used, such as regression models, and neural networks. Most of these are sensitive to outliers or incorrect model selection(e.g. wrong distribution). In these situations, the methods of computational statistics can be effective.

Computational statistics includes a set of methods for non-parametric statistical estimation. The main idea is to use data in different combinations to replace complex statistical inferences by computations. The resampling approach supposes that the available data are used in different combinations to obtain model-free estimators that are robust to outliers. The quality of the estimators obtained is also important.

In the present study, we demonstrate data flows in cloud-based ITS for route recommendations and propose a resampling-based approach for the route comparison in such systems. We derive the properties of the proposed resampling estimators and compare these with traditional plug-in estimators.

The remainder of this study is organized as follows: Section 2 formulates the problem, Sections 3-4 describe the resampling procedure and its properties, Section 5 contains a numerical example, and Section 6 presents the conclusion.

2 Problem Formulation

We consider a cloud-based ITS architecture [8]. In terms of the Internet of Things, the real-world users are represented in the cloud system as virtual agents, which act in the cloud and virtual traffic network. The street network is presented by the virtual transport network, which consists of a digital map as well as the associated ad-hoc network models that allow estimation and forecasting of the important network characteristics for each problem [6]. The virtual agents store the real-time information, which is collected and constantly processed in the cloud. Moreover, the strategies for execution of the cloud application are constantly pre-calculated and checked in the virtual network (e.g., the shortest routes are pre-calculated). When a user runs the cloud application, the pre-calculated strategy is updated with the real-time data and is executed, with respect to the corresponding changes. Data flows and corresponding optimization methods in the cloud-based ITS architecture are presented in Fig. 1.

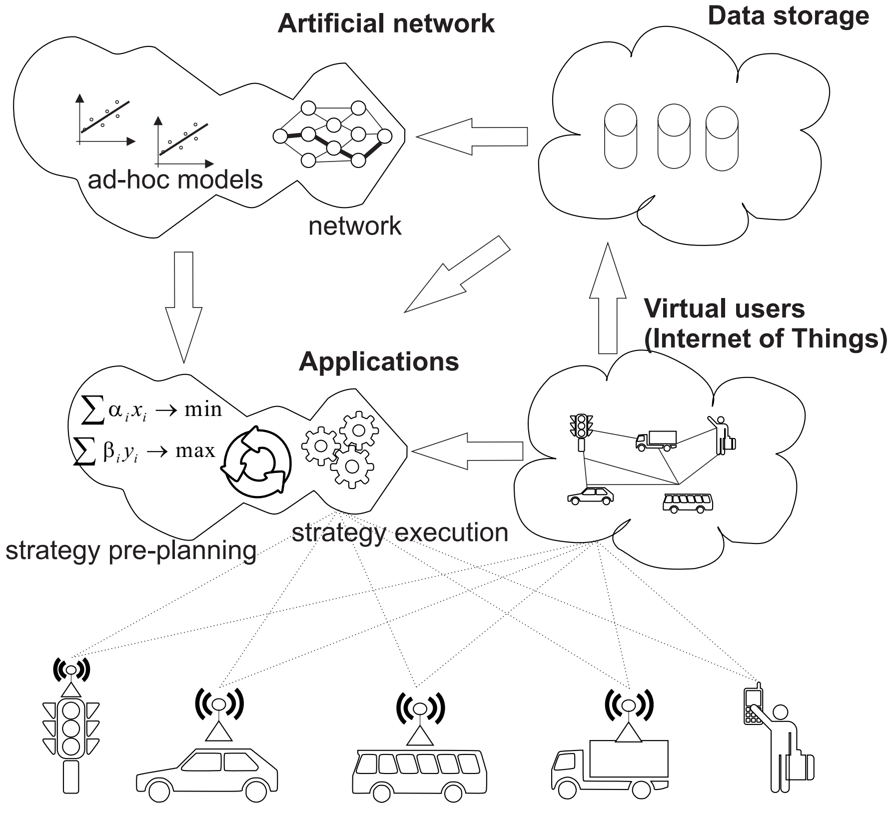

Fig. 1 Data flows and corresponding optimization methods in cloud-based ITS

We consider an application, that provides route recommendations to vehicle drivers. The essential process of this application is the comparison of pre-defined routes. It is based on historical samples of the route segments, which are collected from the virtual users. The candidate routes are compared in the virtual transport network in order to recommend the best route to a user. As the travel times are random, a stochastic comparison should be used. We consider a directed graph $G = (V,E)$ with n edges, $|E| = n$, where each edge $e_i \in E$ has an associated weight X_i (e.g. travel time). We assume that the weights $\{X_1, X_2, \ldots, X_n\}$ are independent random variables (r.v.). A route in the graph is a sequence of edges such that the next edge in the sequence starts from the node, where the previous edge ends. Let us denote a route b as a sequence k^b of edge indices in the initial graph: $k^b = (k^b_1, k^b_2, \ldots, k^b_{n_b})$, which consists of n_b edges. Hence, a route b is the sequence of edges $r^b = \{e_{k^b_1}, e_{k^b_2}, \ldots, e_{k^b_{n_b}}\}$. The route weight S^b is the sum of the corresponding edge weights, so $S^b = \sum_{i \in k^b} X_i$.

We compare two non-overlapping routes by calculating the probability that the weight of route 1 is greater than that of route 2: $\Theta = P\{S^1 > S^2\}$.

The distributions of the edge weights are unknown, only the samples are available: $H_i = \{H_{i,1}, H_{i,2}, \ldots, H_{i,m_i}\}$, where $i = 1,2,\ldots,c$, $c \leq n$. Each sample may correspond to one or several edges. An (unknown) cumulative distribution function (cdf) of the sample H_i elements is denoted by $F_i(x)$, $i = 1,2,\ldots,c$.

The traditional plug-in approach supposes a choice of distribution type and estimation of its parameters. In the case of small samples, it is difficult to choose the distribution law correctly; hence the estimators obtained are usually inaccurate.

Hence, it is preferable to use the non-parametric resampling procedure ([7]), which is a variant of the bootstrap method ([3], [4]). The implementation of this approach to various problems was considered in the studies reported in ([1], [2], [5]). We employ the usual simulation technique without parameter estimation and use this in the simulation process to extract elements randomly from the samples of random variables. We produce a series of independent experiments and accept the average over all realizations as the resampling estimator of the parameter of interest.

Two cases are considered: (1) each edge has different samples, so only one element is extracted from the sample H_i; and (2) edges may correspond to common samples, including the common samples for two routes.

3 Resampling Procedure

We propose an N-step resampling procedure. At each step, we randomly without replacement choose $\eta^1_i + \eta^2_i$ elements from each sample H_i: η^1_i elements for route 1, and η^2_i elements for route 2: $\eta_i = (\eta^1_i, \eta^2_i)$.

Let $J^b_i(l)$, $|J^b_i(l)| = \eta^b_i$ be a set of element indices extracted from the sample H_i, for a route b, $b = 1,2$, during resampling step l, $i = 1,\ldots,c$. Let $\mathbf{X}^{*l} = \bigcup_{i=1}^{c} \{H_{i,j} : j \in J^1_i(l)\} \cup \bigcup_{i=1}^{c} \{-H_{i,j} : j \in J^2_i(l)\}$ be the l-th resample of the edge weights for both routes, with the weights of route 2 assumed to be negative.

Let $\boldsymbol{\Psi}(\mathbf{x})$ be an indicator function, where $\mathbf{x} = (x_1, x_2, \ldots)$ is a vector of real numbers: $\boldsymbol{\Psi}(\mathbf{x})$ is unity if $\sum_i x_i > 0$; otherwise, it is zero. The average of $\boldsymbol{\Psi}(\mathbf{X}^{*l})$ over all N steps is accepted as the resampling estimator of the probability of interest: $\Theta^* = \frac{1}{N}\sum_{l=1}^{N} \boldsymbol{\Psi}(\mathbf{X}^{*l})$. The resampling procedure is presented as Algorithm 1.

Algorithm 1 Function RESAMPLE

function RESAMPLE$(H_i, \eta_i, i = 1, \ldots, c, N)$
    for all $l \in 1, \ldots, N$ **do**
        for all $i \in 1 \ldots c$ **do**
            $X_i^{*l} \leftarrow \ extract(H_i, \eta_i^1 + \eta_i^2)$
            $X1_i^{*l} \leftarrow \ subsample(X_i^{*l}, 1, \eta_i^1)$; $X2_i^{*l} \leftarrow \ subsample(X_i^{*l}, \eta_i^1 + 1, \eta_i^2)$
        end for
        $\mathbf{X}^{*\mathbf{l}} = \bigcup X1_i^{*l} \bigcup - X2_i^{*l}$; $\Theta_l \leftarrow \boldsymbol{\Psi}(\mathbf{X}^{*\mathbf{l}})$
    end for
    $\Theta^* \leftarrow \frac{1}{N}\sum_{l=1}^{N} \Theta_l$
    return Θ^*
end function

The function $extract(X, n)$ randomly chooses n elements without replacement from the set X. The function $subsample(X, a, n)$ returns n elements from X, starting from position a. These two cases differ with the parameters of the $extract$ procedure.

4 Properties of the Resampling Estimator

The estimator Θ^* is obviously unbiased: $E(\Theta^*) = \Theta$, so we are interested in its variance. Consider the elements extracted at two different steps $l \neq l'$. Moreover, we denote: $\mu = E\ \boldsymbol{\Psi}(\mathbf{X}^{*l})$, $\mu_2 = E\ \boldsymbol{\Psi}(\mathbf{X}^{*l})^2$, $\mu_{11} = E\ \boldsymbol{\Psi}(\mathbf{X}^{*l}) \cdot \boldsymbol{\Psi}(\mathbf{X}^{*l'}), l \neq l'$. Then, the variance is $V(\Theta^*) = E(\Theta^{*2}) - \mu^2 = \left\{\frac{1}{N}\mu_2 + \frac{N-1}{N}\mu_{11}\right\} - \mu^2$, for the estimation of which we need the mixed moment μ_{11} depending on the resampling procedure.

Different Samples for Each Edge

In this case, $J_i^b(l)$ consists of one element, denoted as $j_i^b(l)$. This is the index of an element extracted from the sample H_i at step l for route b.

Let $M_i = \{1, 2, \ldots, m_i\}$, $U^b : \{i : \eta_i^b \neq \emptyset\}$, $M^b = \prod_{i \in U^b} M_i$ and $\mathbf{j}^b(l) = \{j_i^b(l) : i \in U^b\}$, $\mathbf{j}(l) = (\mathbf{j}^1(l), \mathbf{j}^2(l))$, where $\mathbf{j}^b(l) \in M^b$ and $b = 1, 2$.

We use a modification of the ω-pair notation [5]. Let $\omega^b \subset U^b$, $\omega = (\omega^1, \omega^2)$. We assume that two vectors $\mathbf{j}(l)$ and $\mathbf{j}(l')$ produce an ω-pair, if $j_i^b(l) = j_i^b(l')$ for $i \in \omega^b$ and $j_i^b(l) \neq j_i^b(l')$ for $i \notin \omega^b$. In other words, the components of the vectors $\mathbf{j}(l)$ and $\mathbf{j}(l')$ produce the ω-pair if they have the same elements from the samples, whose indices are contained by ω.

Let $A(\omega)$ be an event 'resamples $\mathbf{j}(l)$ and $\mathbf{j}(l')$ for the different steps $l \neq l'$ produce the ω-pair', let $P\{\omega\}$ be the probability of this event, and let $\mu_{11}(\omega)$ be the corresponding mixed moment. The probability of producing the ω-pair is

$$P\{\omega\} = \frac{1}{|M^1||M^2|} \prod_{i \in \bigcup_b \{U^b \setminus \omega^b\}} (m_i - 1).$$

The mixed moment μ_{11} can be calculated with the formula $\mu_{11} = \sum_{\omega \subset U^1 \times U^2} P(\omega)\mu_{11}(\omega)$. Next, we intend to calculate $\mu_{11}(\omega)$, $\omega \subset U^1 \times U^2$. Let

$$S_l^{dif}(\omega) = \sum_{i \in U^1 \setminus \omega^1} H_{i,j_i^1(l)} - \sum_{i \in U^2 \setminus \omega^2} H_{i,j_i^2(l)},$$
$$S_{ll'}^{com}(\omega) = \sum_{i \in \omega^1} H_{i,j_i^1(l)} - \sum_{i \in \omega^2} H_{i,j_i^2(l)}.$$

Then, $\mu_{11}(\omega)$ can be calculated as

$$\mu_{11}(\omega) = E(\Psi(\mathbf{X}^{*l}) \cdot \Psi(\mathbf{X}^{*l'})|\omega) = \int_{-\infty}^{+\infty} \left(1 - F_\omega^d(-x)\right)^2 dF_{\omega(x)}^c,$$

where $F_\omega^d(x)$ is cdf of $S_l^{dif}(\omega)$, $F_\omega^c(x)$ is cdf of $S_{ll'}^{com}(\omega)$ given ω-pair.

Common Samples for Edges

Here, we use the notation of α-pairs ([1], [2], [5]) instead of ω-pairs. Let $J_i^b(l) = \{j_{i,1}^b(l), j_{i,2}^b(l), \ldots, j_{i,\eta_i^b}^b(l)\}$, $J^b(l) = \{J_i^b(l) : i \in U^b\}$, $\mathbf{J}(l) = \{J^1(l), J^2(l)\}$, where $J_i^b(l) \subset M^b$, $b = 1,2$, $l = 1,\ldots,N$, $i = 1,2,\ldots,c$.

Let $A_i^b(ll')$ be a set of indices of the common elements, extracted from the sample H_i for route b at steps l and l'. Let $A_i^{bp}(ll')$ be a set of indices of the common elements, extracted from the sample H_i for route b at step l and for route p and at step l'. Let $\overline{A}_i^{bp}(l)$ be a set of indices of the elements from route b at step l, which were in neither route b nor route p at step l', $b,p \in \{1,2\}$ and $b \neq p$:

$A_i^b(ll') = J_i^b(l) \cap J_i^b(l')$, $\overline{A}_i^{bp}(l) = J_i^b(l) \setminus (A_i^b(ll') \cup A_i^{bp}(ll'))$, $A_i^{bp}(ll') = J_i^b(l) \cap J_i^p(l')$, $\overline{A}_i^{pb}(l) = J_i^p(l) \setminus (A_i^p(ll') \cup A_i^{pb}(ll'))$. Let $0 \le \alpha_i^b \le \eta_i^b$, $0 \le \alpha_i^{bp} \le \min(\eta_i^b, \eta_i^p)$, $b,p \in \{1,2\}$ and $b \neq p$. Let $\alpha_i = \{\alpha_i^1, \alpha_i^2, \alpha_i^{12}, \alpha_i^{21}\}$, $\alpha = \{\alpha_i\}, i = 1,2,\ldots,c$. Next, we say that $\mathbf{J}(l)$ and $\mathbf{J}(l')$ produce an α-pair, if and only if: $\alpha_i^1 = |A_i^1(ll')|$, $\alpha_i^2 = |A_i^2(ll')|$, $\alpha_i^{12} = |A_i^{12}(ll')|$, $\alpha_i^{21} = |A_i^{21}(ll')|$. Let $A_{ll'}(\alpha)$ denote the event 'sub-samples $\mathbf{J}(l)$ and $\mathbf{J}(l')$ produce an α-pair', and let $P_{ll'}\{\alpha\}$ be the probability of this event: $P_{ll'}\{\alpha\} = P_{ll'}\{A_{ll'}(\alpha)\}$.

To calculate $\mu_{11}(\alpha)$ we replace ω-pairs with α-pairs. Therefore we need to calculate $P\{\alpha\}$ and $\mu_{11}(\alpha)$. The probability $P\{\alpha\}$ is

$$P\{\alpha\} = \prod_{i \in 1,2,\ldots,c} \frac{\binom{\eta_i^1}{\alpha_i^1}\binom{\eta_i^2}{\alpha_i^{21}}\binom{m_i - \eta_i^1 - \eta_i^2}{\eta_i^1 - \alpha_i^1 - \alpha_i^{21}}}{\binom{m_i}{\eta_i^1}} \times$$
$$\times \binom{\eta_i^1 - \alpha_i^1}{\alpha_i^{12}}\binom{\eta_i^2 - \alpha_i^{21}}{\alpha_i^2} \frac{\binom{m_i - 2\eta_i^1 - \eta_i^2 + \alpha_i^1 + \alpha_i^{21}}{\eta_i^2 - \alpha_i^{12} - \alpha_i^2}}{\binom{m_i - \eta_i^1}{\eta_i^2}},$$

where $\binom{n}{m}$ is a binomial coefficient.

To calculate $\mu_{11}(\alpha)$ we divide each sum into three subsums: $S_l^{dif}(\alpha)$ contains different elements for steps l and l'; $S_{ll'}^{com}(\alpha)$ - the common elements for the same route; $S_{ll'}^{com12}(\alpha)$ - the common elements for different routes. Let $S_l^{dif}(\alpha) = \sum_{i=1}^{c}\left\{\sum_{j\in\bar{A}_i^{12}(l)} H_{i,j} - \sum_{j\in\bar{A}_i^{21}(l)} H_{i,j}\right\}$, $S_{ll'}^{com}(\alpha) = \sum_{i=1}^{c}\left\{\sum_{j\in A_i^{1}(ll')} H_{i,j} - \sum_{j\in A_i^{2}(ll')} H_{i,j}\right\}$, $S_{ll'}^{com12}(\alpha) = \sum_{i=1}^{c}\left\{\sum_{j\in A_i^{12}(ll')} H_{i,j} - \sum_{j\in A_i^{21}(ll')} H_{i,j}\right\}$.

As $S_{ll'}^{com}(\alpha) = S_{l'l}^{com}(\alpha)$ and $S_{ll'}^{com12}(\alpha) = -S_{l'l}^{com12}(\alpha)$, $\mu_{11}(\alpha)$ is:

$$\mu_{11}(\alpha) = E\{\Psi(\mathbf{X}^{*l})\cdot\Psi(\mathbf{X}^{*l'})|\alpha\} = P\left\{\Psi(\mathbf{X}^{*l}) = 1, \Psi(\mathbf{X}^{*l'}) = 1|\alpha\right\} =$$
$$= \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty}(1-F_\alpha^d(-x-y))\times(1-F_\alpha^d(-x+y))dF_\alpha^c(x)dF_\alpha^{c12}(y),$$

where $F_\alpha^d(x)$ is cdf of $S_l^{dif}(\alpha)$, $F_\alpha^c(x)$ is cdf of $S_{ll'}^{com}(\alpha)$, $F_\alpha^{c12}(x)$ is cdf of $S_{ll'}^{com12}(\alpha)$.

5 Numerical Example

We model a route recommendation in the southern part of the city of Hanover, (Germany), which is shown in Fig. 2 (left), and represented by the graph in Fig. 2 (right). We compare two routes: nodes 9,8,7,5,3,1 (solid) and nodes 9,6,4,2,1 (dashed) for vehicles travelling from 9 to 1.

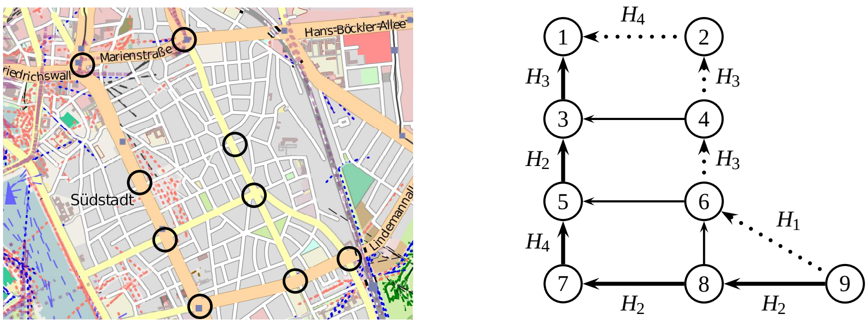

Fig. 2 Street network of the south part of Hanover city and corresponding graph

The cloud-based ITS collects information about travel times for different road segments. We assume that due to technical or organizational limitations, the travel times on different roads are indistinguishable. The travel times are collected into four samples H_1, H_2, H_3 and H_4, as demonstrated in the graph in Fig. 2 (right).

The traditional methods for route comparison give a biased estimator. As an alternative, we apply the resampling approach. For comparison, we use the mean squared errors of the plug-in $MSE(\tilde{\Theta})$ estimator and the resampling estimator $MSE(\Theta^*) = V(\Theta^*)$ because of $E(\Theta^*) = \Theta$. The experimental results are shown in Fig. 3. We can see that the resampling estimator is effective in most situations.

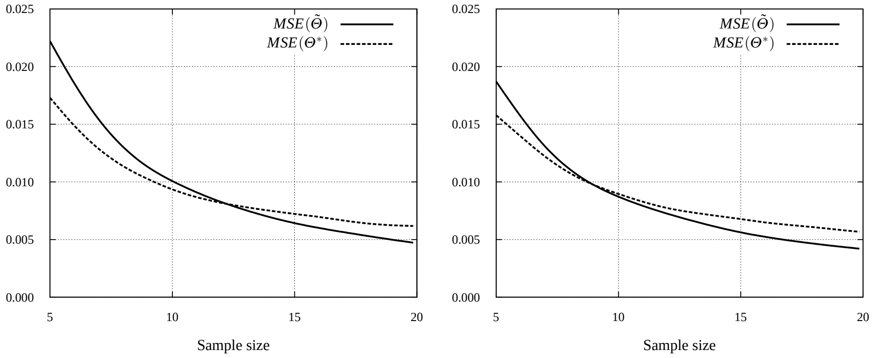

Fig. 3 MSE (vertical axis) of plug-in and res. estimators for $\Theta = 0.5$ (left) and $\Theta = 0.34$ (right)

Conclusion

Cloud applications open new perspectives on intelligent transportation services. Data mining is one of the most important problems for such systems. We demonstrated the application of the resampling approach to the problem of route comparison in route recommendation systems. The formulas obtained allow calculation and comparison of the properties of the estimators considered. Future work will be devoted to the integration of the proposed algorithms to cloud-based TCMS and their validation on large-scale transport networks.

References

1. Afanasyeva, H.: Resampling-approach to a task of comparison of two renewal processes. In: Proc. of the 12th Int. Conf, on Analytical and Stochastic Modelling Techniques and Applications, pp. 94–100. Riga (2005)
2. Andronov, A., Fioshina, H., Fioshin, M.: Statistical estimation for a failure model with damage accumulation in a case of small samples. Journal of Statistical Planning and Inference **139**(5), 1685 – 1692 (2009). DOI DOI: 10.1016/j.jspi.2008.05.026
3. Davison, A., Hinkley, D.: Bootstrap Methods and their Application. Cambridge university Press (1997)
4. Efron, B., Tibshirani, R.: Introduction to the Bootstrap. Chapman & Hall (1993)
5. Fioshin, M.: Efficiency of resampling estimators of sequential-parallel systems reliability. In: Proc. of 2nd Int. Conf. on Simulation, Gaming, Training and Business Process Reengineering in Operations, pp. 112–117. Riga (2000)
6. Fiosins, M., Fiosina, J., Müller, J., Görmer, J.: Reconciling strategic and tactical decision making in agent-oriented simulation of vehicles in urban traffic. In: Proceedings of the 4th International ICST Conference on Simulation Tools and Techniques, pp. 144–151 (2011)
7. Gentle, J.E.: Elements of Computational Statistics. Springer (2002)
8. Li, Z., Chen, C., Wang, K.: Cloud computing for agent-based urban transportation systems. IEEE Intelligent Systems, IEEE Computer Society **26**(1), 73–79 (2011)