Automated Setting of Bus Schedule Coverage Using Unsupervised Machine Learning

Jihed Khiari[1], Luis Moreira-Matias[1](✉), Vitor Cerqueira[1], and Oded Cats[2]

[1] NEC Laboratories Europe, 69115 Heidelberg, Germany
{jihed.khiari,luis.matias,vitor.cerqueira}@neclab.eu
[2] Department of Transport and Planning, TU Delft, 2600 Delft, Netherlands
o.cats@tudelft.nl

Abstract. The efficiency of Public Transportation (PT) Networks is a major goal of any urban area authority. Advances on both location and communication devices drastically increased the availability of the data generated by their operations. Adequate Machine Learning methods can thus be applied to identify patterns useful to improve the Schedule Plan. In this paper, the authors propose a fully automated learning framework to determine the best Schedule Coverage to be assigned to a given PT network based on Automatic Vehicle location (AVL) and Automatic Passenger Counting (APC) data. We formulate this problem as a clustering one, where the best number of clusters is selected through an *ad-hoc* metric. This metric takes into account multiple domain constraints, computed using Sequence Mining and Probabilistic Reasoning. A case study from a large operator in Sweden was selected to validate our methodology. Experimental results suggest necessary changes on the Schedule coverage. Moreover, an impact study was conducted through a large-scale simulation over the affected time period. Its results uncovered potential improvements of the schedule reliability on a large scale.

Keywords: Unsupervised learning · Public transportation · Big data · Schedule plan · Schedule coverage · Sequence mining · Probabilistic reasoning

1 Introduction

Public Transport (PT) reliability is a major issue in modern cities. A good operational planning is necessary to deliver such service quality requirements while maintaining a balanced relationship between resource usage and obtained revenues. Nowadays, major PT operators have their fleets equipped with Global Positioning System (GPS) antennas, communicational devices (e.g. 3G) and Radio-frequency Identification readers able communicate the vehicle's positioning (i.e., Automatic Vehicle Location (AVL)) and its ridership (i.e., Automatic Passenger Counting (APC)) to a central server [1].

To mine this novel source of data is a massive challenge. It contains information about the patterns of human behavior while traveling (as drivers or passengers) on an urban environment. Such patterns can provide useful insights to

J. Bailey et al. (Eds.): PAKDD 2016, Part I, LNAI 9651, pp. 552–564, 2016.
DOI: 10.1007/978-3-319-31753-3_44

improve the operational planning of mass transit agencies - namely, its **Schedule Plan (SP)**. Such improvement may bring multiple benefits by providing ways of reducing costs (e.g. fleet (re)sizing or fuel saving due to a decrease of the necessary number of trips) and/or improving the passenger experience.

A Schedule Planning (SP) process for a given route relies on two main steps [2]: (1) the first step is to define the number k of schedules and their individual coverage, S_i. Consequently, this first step defines different schedules for days that are characterized by different traffic and demand patterns due to seasonal variations, for instance. Secondly, (2) the timetables are assigned for each route schedule containing the time the buses pass at each schedule time point (per trip). This process is done for all routes. While the timetables are defined *routewise* (e.g. high/low frequency routes), the number of schedules (i.e. k) and their coverage ($S_i, \forall i \in \{1, .., k\}$) must be defined *networkwise*. Such definition is key to ease PT operations (e.g. maintenance tasks) and, most of all, to facilitate the SP memorization by the passengers.

Automated data driven frameworks that aim to improve the SP are commonly focused on timetabling tasks, thus *skipping* the coverage definition. Some of the most well-known approaches include finding the optimal slack time and round-trip time to put into the schedule using Genetic/Ant Colony Algorithms [3,4], mining distribution rules able to discover feature subspaces (i.e. scenarios) for an increased travel time uncertainty [5], or clustering trips based on APC data regarding their frequency setting, i.e. high/low [6]. However, the coverage definition can easily constrain the timetable construction (e.g. two days with distinct demand peak periods should have different timetables). At the best of our knowledge, only Mendes-Moreira *et al.* [2] covers the improvement of Schedule Coverage: a Consensual Clustering framework groups days with similar behavior (using AVL data standalone) given a predefined number of schedules k.

This paper is a comprehensive extension of the work in [2]. It aims to generalize this framework's usage for every scenario that fully exploits the information available on the data repository while still minimizing the required human input to reach a decision. The contributions are threefold:

1. a novel *ad-hoc* domain-oriented metric to select the most adequate number of schedules to put in place based on Sequential Itemset Mining [7] and Probabilistic Reasoning. It settles on a trade-off between the entropy within the clusters and the operational adequacy of the resulting coverage.
2. a hybrid computation of the daily profiles using APC/AVL data simultaneously by decomposing the round trip times into a sum of link travel times (the run times between two consecutive stops) and dwell times[1]. Their computation may highlight demand peaks which would be smoothed otherwise.
3. the application of a Gaussian Mixture Model (GMM) [8] to perform the necessary clustering for the individual routes, thus replacing the originally proposed k-Means (see Sect. 5 in [2]). By doing so, we obtain a soft assignment of the samples, reducing the overfitting chances.

[1] Reports stoppage time at stops. Includes a fixed delay due to door opening and closing time, and a variable delay caused by passengers boarding/alighting activities.

The proposed framework was evaluated using data acquired from a large bus operator in Sweden throughout a period of six months. Numerical experiments suggested a change to the agency's original coverage. The impact of such change was measured by assigning a theoretical timetable to the affected period. A *before-and-after* schedule reliability study was conducted. The results are promising.

The remainder of the paper is structured as follows: methodology is described in Sect. 2, by doing an analysis of the previous work and a formal explanation of our contributions. The case study is presented in Sect. 3, along with some summary statistics of the used datasets. The results are presented in the Sect. 4, followed by a brief discussion. Finally, conclusions are drawn.

2 Methodology

A stepwise methodology is hereby proposed to automatically set both the number of schedules and their daily coverage. This description follows closely the one proposed in Sect. 4 of [2]. It elaborates on the principle that days where the route trips have a similar behavior (e.g. round-trip times) throughout the day should be assigned to the same schedule. Let $\mathbb{L} = \{r_1, ..., r_n\}$ denote a set of routes of interest. Firstly, for each $r \in \mathbb{L}$, the running times and the boardings/alightings at each stop (if existing) are extracted from its original AVL/APC dataset. Secondly, the daily profiles are generated. If there is no APC data available for a specific route, the procedure originally suggested in [9] is used. Otherwise, a biased dwell time model is generated based on APC data to account demand peaks/valleys. Its output is added to the link travel times computed through the AVL data - as described in Sect. 2.1.

The next two steps generate a distance matrix between the days (using their daily profiles) and cluster them. The first task is conducted using a Euclidean-flavoured Dynamic Time Warping, while the latter is addressed using a GMM. Conversely to previous works, the clustering is made for a user-defined set of admissible number of schedules $\mathbb{K} \subset \mathbb{N}$, i.e. $\forall k \in \mathbb{K}$ instead of a single predefined k value. The above mentioned steps are repeated for all routes.

Step 5 selects the best possible $k \in \mathbb{K}$ to define the best number of schedules to put in place. This is made using a two-stage process, where an *ad-hoc* metric is devised to evaluate the clustering result for each pair $(r, k), \forall r \in \mathbb{L}, k \in \mathbb{K}$. Then, a consensual k, i.e. K is found through a domain-oriented weighted mean of the previously computed metrics - as described in Sect. 2.3. Finally, a Consensual Clustering procedure is devised using the clustering pieces obtained for $k = K$ to compute the suggested Schedule Coverage, following the original procedure proposed in [2]. An illustration of our methodology is presented in Fig. 1. The remainder of this Section describes our contributions.

2.1 Modeling the Daily Profiles

Let $L = \{L_1, L_2, ..., L_n\}$ be a set of the available AVL datasets for n considered routes, and $C = \{C_1, C_2, .., C_n\}$ a set of the corresponding APC datasets.

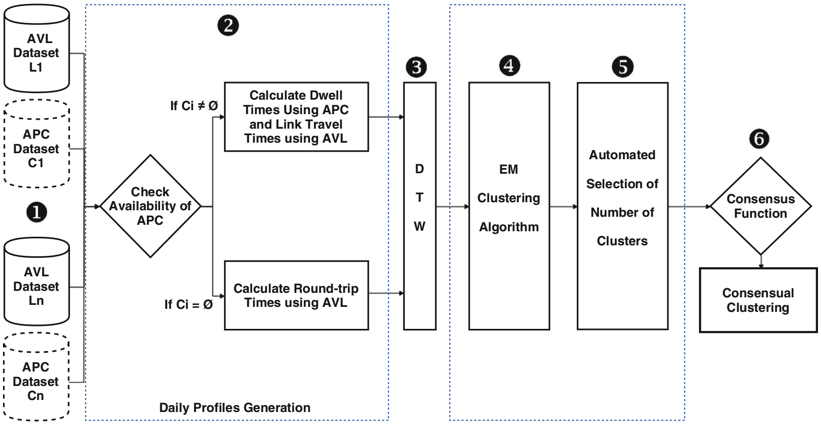

AVL Dataset L1
APC Dataset C1
1
AVL Dataset Ln
APC Dataset Cn
2
Check Availability of APC
If Ci ≠ Ø
If Ci = Ø
Calculate Dwell Times Using APC and Link Travel Times using AVL
Calculate Round-trip Times using AVL
Daily Profiles Generation
3
D T W
4
EM Clustering Algorithm
5
Automated Selection of Number of Clusters
6
Consensus Function
Consensual Clustering

Fig. 1. A generic representation of the proposed methodology. The contributions of this paper are highlighted by the dashed blue rectangles.

If $C_i \neq \varnothing$, the round-trip time for every trip is obtained by adding the dwell times at stops and the link travel times as they are described in the AVL data.

By using trip-level APC data, we expect to *express* the demand peaks/valleys as slight increases/decreases of the computed round-trip time. Let r be a route of interest with the associated datasets (L_i, C_i) where t is the number of trips and s is its number of stops. This procedure starts by modeling the dwell time at stop through a decomposition in multiple factors. It can be computed as follows:

$$\delta_{o,j} = \max(\alpha \times a_{o,j}, \beta \times b_{o,j}) + doc \tag{1}$$

where α and β are constants that denote the alighting and boarding time per passenger, respectively, and doc denotes the time allocated for operations that take place on every stop, e.g. the opening and closing of doors. On the other hand, $a_{o,j}$ and $b_{o,j}$ are the number of passengers that alight/board on a stop j during a trip o, respectively, where $o \in \{1, 2, ..., t\}$ and $j \in \{1, 2, ..., s\}$.

Using the available values for dwell times (AVL) $\delta_{o,j}$ and the values of $a_{o,j}$ and $b_{o,j}$ (APC), we perform a linear regression procedure to estimate the values of α, β and doc. It consists of three steps: firstly, we isolate the samples (i.e. boardings/alightings and dwell times for every pair of [trips/stops] available) where $a_{o,j} = 0$ and $b_{o,j} = 0$ into two different partitions. This allows to transform Eq. 1 into a linear one. Secondly, we estimate values for α, β and two possible values for doc, i.e. doc_a, doc_b. Finally, the doc value is computed as $doc = \frac{doc_a + doc_b}{2}$. Then, we use the resulting constants to compose a novel function for the dwell time (i.e. $\hat{\delta}_{o,j}$). This function is used with the original APC data to compute novel dwell time estimations, which are summed up to the link travel times observed in the original AVL data.

The induction model used to do the abovementioned linear regression procedure is a modified version of the well-known least squares, where we replace its typical loss-function (a sum of the squared residuals) for the mean absolute deviation (MAD) (i.e. which results in a simple sum of the residuals). This change increases the framework's tolerance to large errors (i.e. demand peak/valleys), which will result in an under/overestimation of the dwell times under such conditions. This effect aims to model the demand peaks/valleys inside the daily profiles of round trip times typically used by [2]. By producing a daily profile based on heterogeneous sources of data, we aim to adequately express the differences between the route behavior - both in terms of cruising time and in its demand - on the schedule coverage definition.

2.2 Expectation-Maximization (EM) for Clustering Analysis

[2] proposed k-Means algorithm to perform the routewise clustering in the context of this application. This approach assumes a deterministic clustering step where the model is only given by the Euclidean Distance to the incrementally computed centroids (i.e. spherical clusters, parametric). Such characteristics may easily lead to an undesired overfitting, where the samples are erroneously initially assigned to a non-homogeneous cluster, potentially increasing the variance within. To overcome this limitation, we propose a GMM (a general version of k-Means), which (briefly) operates as follows: firstly, it (a) softly assigns a sample to a cluster, i.e., computing the probability of any point belonging to every centroid; then, it (b) estimates the parameters of the probability distribution, taking the sample-based covariances into account.

2.3 Automated Selection of Number of Schedules

The selection of the best number of clusters is a complex problem in data analysis. One of the most well-known metrics to do it so is the Bayesian Information Criterion (BIC) [10], which computes an entropy-based probabilistic score that, when maximized over a set of values, i.e. $\mathbb{K}$, aims to return the optimal k by minimizing the entropy between samples of the same cluster and maximizing the one between samples of different ones. However, such optimization problem may not lead to a good solution for a real-world context, given the constraints that each application domain encloses. Consequently, *ad-hoc* metrics are often devised to address such issues (e.g. market segmentation in [11]).

In this context, we depart from BIC to set up an *ad-hoc* metric, i.e. m for this problem as a linear combination of multiple factors. These factors were considered in light of two main constraints: (1) the cost of increasing the number of defined schedules (which reduces the schedule's interpretability as well as its easy memorization, the operators' ability to easily put it in place, and consequently, the route's riderships) must be necessarily balanced by a *gain* on the punctuality of the offered service, by reducing significantly the entropy on the produced clusters; (2) the cluster's output must model a *frequent pattern* (e.g. the Saturdays should be grouped with the Sundays throughout five months of an year). Such factors can be expressed as follows:

$$m(k,r) = \big(nbic(k,r) - f(k,r)^2\big) + \big(q(k,r) - \hat{\sigma}(k,r)\big), k \in \mathbb{K}, r \in \mathbb{L} \qquad (2)$$

where $nbic(k,r)$ is the normalized[2] value of BIC. (1) The first term of Eq. 2 addresses the number of clusters. High values of *nbic* will bring a gain on the punctuality of a suitable timetable defined for such partitioning. On the other hand, the increase of the number of schedules to maximize such punctuality must be done if and only if such *gain* is **significant**. Consequently, we need to model a *trade-off* between an eventual gain given by increasing the number of schedules and the associated cost of decreasing its interpretability. We do it so by introducing a penalty term $f(k,r)^2$ that favors lower values of k, where $f(k,r) = k/max(\mathbb{K}), \forall r \in \mathbb{L}$.

The second term of Eq. 2 addresses the cohesion and consistency of the partitioning for a number of schedules k. Empirically, we know that a SP in PT should cover a static set of *daytypes* (e.g. Mondays) throughout a relatively long set of weeks. Consequently, a suitable cluster would be one that provides such *frequent pattern*. The suitability of each cluster is given by an *ad-hoc quality* metric, i.e. $q(k,r)$. It is computed in two stages: (2a) frequent itemset mining and (2b) compatible pattern merging. This procedure is detailed as follows.

Cluster Quality Computation. A *frequent pattern* in this problem can be modeled through a sequence mining problem to find *frequent itemsets* of daytypes among the weeks (i.e. *transactions*) covered by the input data (e.g. Mondays to Fridays). Let $\gamma, \phi \in [0,1]$ denote two user-defined parameters for the minimum *support* to consider a given itemset as frequent (i.e. the minimum amount of weeks to define a schedule) and for the minimum cluster's mass ratio to be covered by it, respectively. The PrefixSpan algorithm [7] is hereby adopted to find such frequent itemsets, i.e. FI_i among the daytype's transactions obtained from each partition S_i. Let N denote the number of weeks in the input data. The *frequent pattern* of each cluster, i.e. FP is then selected as follows:

$$FP_i = \underset{FI_i \subseteq S_i}{\arg\max} \left(\frac{\varGamma(FI_i) \cdot |FI_i|}{N} \right) \text{ subject to: } \varGamma(FI_i) \geq \gamma, FP_i \geq \phi \qquad (3)$$

where $\varGamma(FI_i)$ is the support of the frequent itemset FI_i on the partition S_i.

After such procedure, each cluster possesses a FP_i (which may be $\emptyset$). The quality of each cluster is then computed as $q(k,r) = \sum_{i=1}^{k} \frac{\varGamma(FP_i)}{k}$. However, in this domain, it is very common to find **complementary** schedules (e.g. workdays for all year and workdays during summer vacations, with a support of 0.9 and 0.1, respectively). **Together**, these complementary clusters would present a very meaningful *frequent pattern* which is penalized by the $q(k,r)$ computation formula introduced above. Consequently, we introduced a merging step which aims to find such clusters and to merge them in order to obtain the overall quality of the coverage proposed by a given value of k. This merging step aims to find clusters which have frequent itemsets complementary to a given FP_i by relaxing, at most, one of the two constraints imposed in Eq. 3. The algorithm to do it so is introduced by Fig. 2. Note that two clusters are considered as complementary if they overlap, at most, 10 % of the weeks of the input data.

[2] All the normalizations done throughout this section used the Euclidean distance.

```
function Merging-Comp($k, \gamma, \phi, S$)
    $k' \leftarrow k$;
    for $\big(i$ in $\{1, ..., k'\}\big)$ do
        if $\big(FP_i \neq \emptyset \wedge \Gamma(FP_i) < 1\big)$ then
            for $\big(j$ in $\{1, ..., k'\}\big)$ do
                if $\Big(j \neq i \wedge \big(\exists\ cFI = FI_j \subseteq FP_i : FI_j(\gamma = 0) \vee FI_j(\phi = 0)\big)\Big)$ then
                    if AreCoveringComplementaryPeriods?$(S_i, S_j)$ then
                        $S_i \leftarrow S_i \cup S_j; S_j \leftarrow \emptyset; k' \leftarrow k' - 1$;
                        return Merging-Comp$(k', \gamma, \phi, S)$;
                    end if
            end for
    end for
    return $\{k', S\}$;
end function
```

Fig. 2. Merging Procedure for Complementary Clusters/Coverages.

Given the resulting clusters after the merging procedure (with a number of k' clusters), we can compute the final cluster's quality as

$$q(k, r) = \begin{cases} \sum_{i=1}^{k'} \frac{\Gamma(FP_i)}{k'} \text{ if} & k' = k \\ \left(\sum_{i=1}^{k'} \frac{\Gamma(FP_i)}{k'}\right)^{\left(1-\frac{\chi}{2}\right)}, & \chi = \max(FPM_i) \text{ otherwise.} \end{cases} \tag{4}$$

where FPM_i denotes the support of the frequent itemset of a *merged* cluster. Obviously, the resulting clusters may also contain other samples regarding daytypes not included in the frequent itemset (e.g. a cluster modeling the weekends which have two Mondays within). These samples are referred to as *noise* in this context. Such *noise* naturally decreases the adequacy of the *frequent pattern* modeled by each cluster. This effect is introduced by term $\hat{\sigma}(k, r)$ in Eq. 2. $\hat{\sigma}(k, r)$ is calculated based on the standard deviation between the relative frequencies of every day within a particular cluster. It can be computed as:

$$\hat{\sigma}(k, r) = \frac{1}{2} \times \sqrt{\sum_{i=1}^{k} \frac{\sigma(fr_{k,S_i,r})}{k}} \tag{5}$$

where $fr_{k,S_i,r}$ is the vector of relative frequencies of the days within the cluster i, where a relative frequency of a daytype d within a cluster S_i is given by the number of days of daytype d divided by the cluster's mass.

Given such metric computation for all pairs (r, k), we can now compute a consensual number of clusters K. Let $\eta(r)$ denote the normalized (see Footnote 2) number of trips for the route r. The consensual number of clusters K is defined by a weighted average of $k \in \mathbb{K}$. We can express $K \in \mathbb{N}$ as follows:

$$\left\lceil \sum_{r \in \mathbb{L}} \sum_{k \in \mathbb{K}} \frac{m(k, r)^2 \times k \times \eta(r)}{\Psi} \Big/ \sum_{r \in \mathbb{L}} \eta(r), \Psi = \sum_{k \in \mathbb{K}} m(k, r)^2 \right\rceil \tag{6}$$

3 Case Study

Our case study was a large urban bus operator in Sweden. We used data from four high-frequency (maximum planned headway of 10 min between 7:00–19:00) routes A1/A2/B1/B2, i.e. two bus lines A/B. Line A links residential areas to a PT hub as well as major shopping areas. B connects the southern parts of the city to the city center, traversing by a PT hub, major hospitals as well as a logistic center. This study covers six months between August 2011 and January 2012. The coverages in place are relative to two time periods: Summer, from 19 June till 14 December and Winter: from 15 December till 18 June. Two schedules are defined for each period: workdays and weekends/holidays.

As preprocessing, a trip pruning was performed by removing trips with more than 80 % of missing link travel times. Reversely, we performed data imputation on the remaining samples by following the interpolation procedure suggested in [2]. The dwell times were also pruned by using the 99 % percentile to remove erroneous measurements. APC data was used as is.

Table 1 presents an overview of the resulting dataset, detailed per route. It contains the (i) total number of trips (NT), (ii) its number of stops, (iii) the Daily Trips (DT), (iv) the Round Trip Times (RTT) and (v) the loads (i.e. total number of boarding passengers). Both have a similar NT, while line A has a larger RTT than B.

4 Experiments

The experiments were conducted using the R language [12]. The model-based clustering was performed using the GMM implementation of `mclust` package [13]. To compute the frequent itemsets used in the cluster's quality computation, a C++ implementation of *PrefixSpan* [14] was employed. This framework has three parameters: $\mathbb{K}, \gamma$ and ϕ. Their values were set to $2 \leq k \leq 7, \forall k \in \mathbb{K}$, 0.25 and 0.4, respectively. The first used the range suggested by the original experimental setup in [2]. The value of γ was empirically set such that a schedule can only be set for a period of, at least, four weeks; on the other hand, ϕ was selected out of three possible values $0.4, 0.5, 0.6$ through an iterative parameter tuning setting conducted on a small subset of the training data.

The application of the proposed methodology to the available dataset suggested a novel SP - as detailed further in this Section. Its impact on the agency's operations in terms of schedule reliability was assessed through a simulation procedure, described in the next section.

4.1 Impact Evaluation Through a Data-Driven Simulation

Any change of the schedule coverage will result in one of two scenarios: (i) a group of days B changes from one coverage to another among the ones that were already in place or (ii) it will take a completely novel timetable. The procedure that we describe hereby is focused on the type-i Scenarios. Let A and Z be two

Table 1. Statistics per Route. The values are as mean ± s.d.. Times in seconds.

	Nr. Trips	Stops	DT	RTT	Loads
A1	17953	33	134 ± 27	3017 ± 425	101 ± 50
A2	16353	33	133 ± 30	2755 ± 480	98 ± 51
B1	16280	25	127 ± 23	2607 ± 465	70 ± 37
B2	16353	25	124 ± 22	2746 ± 448	60 ± 29

groups of days with different coverages and, consequently, distinct timetables assigned where $B \subseteq A$. Our goal is to test whether the time period B would benefit from having the same timetable of Z instead of its original one (i.e. from B). This procedure is done in three steps: firstly, we need to assign a timetable to B - which will change from the one in place in A to the one used in Z[3]. Then, we need to simulate which would be the (a) link travel times and (b) the dwell times generated by such timetable given the available AVL/APC data.

The (a) link travel times are generated through a k-Nearest Neighbors regression [16] ($k = 1$), where the departure time of each stop is used as an independent variable. The demand on each stop is generated by using the headways computed through (a). These headways correspond to the idle time on a given bus stop bs_i, τ_i. The passenger arrivals at stops are modeled by iteratively sampling passenger arrival times pav^i from an exponential distribution, i.e. $pav^i \sim \text{Exp}(\lambda_i)$. Then, the number of boardings on each stop is computed as follows:

$$bo_i = \arg\max_x \sum\nolimits_{j=1}^{x} pav_j^i \text{ , subject to: } \sum\nolimits_{j=1}^{x} pav_j^i \leq \tau_i \wedge pav_j^i \sim \text{Exp}\ (\lambda_i) \quad (7)$$

where λ_i is computed as time-dependent Poisson process for every specific pair (r, bs) by considering averages of boardings on one hour periods of the days with similar daytypes (e.g. the number of passengers boarded on a given route between 8am and 9am of every Monday) - which are linearly normalized according to the amount of idle time available to compute each $bo_i \simeq x$. The alightings are then computed based on an assumption that the passengers traverse up to 25 % of the route. The resulting dwell times are computed using the Eq. 1 and the constant values obtained through the procedure described in Sect. 2.1.

The impact evaluation study is conducted on a *before-and-after* fashion, where schedule reliability metrics are firstly computed for the current case study (using the original AVL/APC data, as well as the SP in place). Then, the same metrics are also computed for the simulated data obtained through the abovementioned procedure. Four schedule reliability metrics were employed: On-Time Performance, Run-Time Variation, Headway Variation and Excess Waiting Time. Details about these metrics can be found in Sect. 4 of the Survey in [1].

[3] Note that this *naive* timetabling procedure is done only for this specific purpose. Once the coverage is changed, the entire timetable of the affected periods need to be recomputed. The reader can consult the work in [15] to know more about this topic.

4.2 Results

This framework typically runs in linear time, where a single-core CPU processed the 16 k trips of our case study in ~ 600 s. Figure 3 illustrates the computed values for the *ad-hoc* metric hereby devised to assess the quality of the partitioning provided by each value of k. These values resulted in a consensual $K = 3$. Figure 4 shows an example of the clustering results obtained for a particular route using its best value of k, i.e. $k = 5$. The consensual clustering results are exhibited in Fig. 5. Finally, Fig. 6 presents the schedule reliability evaluation metrics of the *before-and-after* study performed through the simulation described in the above Section.

4.3 Discussion

Figure 3 clearly exhibits the penalty effects of the term $f(k, r)^2$ as there is a clear trend of reducing the computed score with the increase of k. Yet, the weighted voting schema proposed in Eq. 6 ends up by finding a *consensus* around $K = 3$ - and not 2 as the charts may empirically suggest. As it is detailed by Fig. 4, this happens mainly due to a particular merge between the S_2 and S_4. Figure 5 illustrates the obtained coverage. It differs largely from the one in place by suggesting that the winter schedule should be in place four weeks earlier than it is (i.e. a change from mid-December to mid-November). The affected period was used as case study to conduct the simulation-based impact study described along Sect. 4.1. The obtained results (exhibited by Fig. 6) clearly outline high

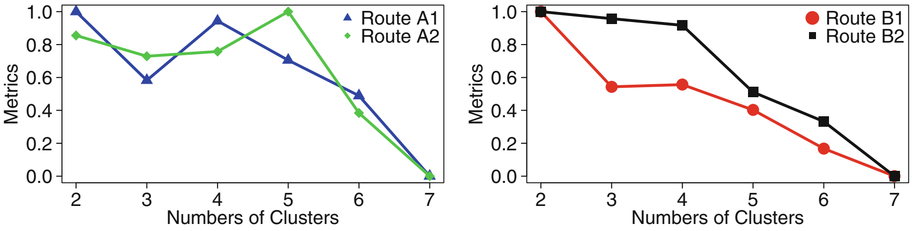


Fig. 3. Cluster quality metrics computed for every route and $k \in \mathbb{K} = \{2, .., 7\}$.

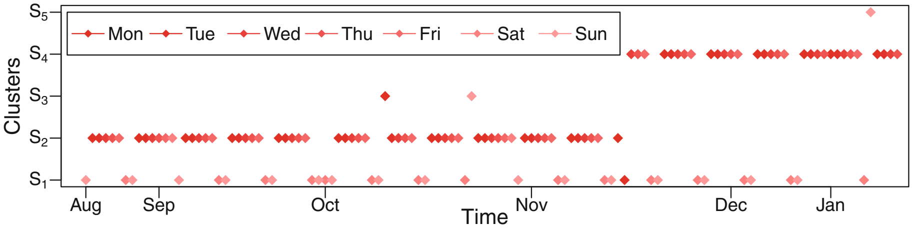


Fig. 4. Clustering results for route A2, $k = 5$.

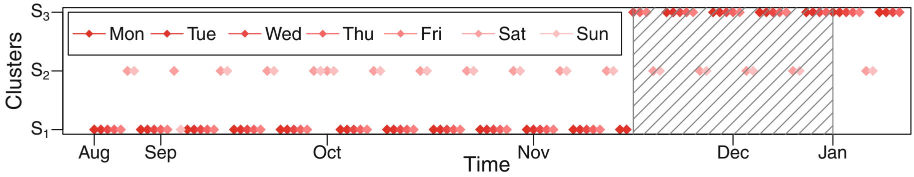
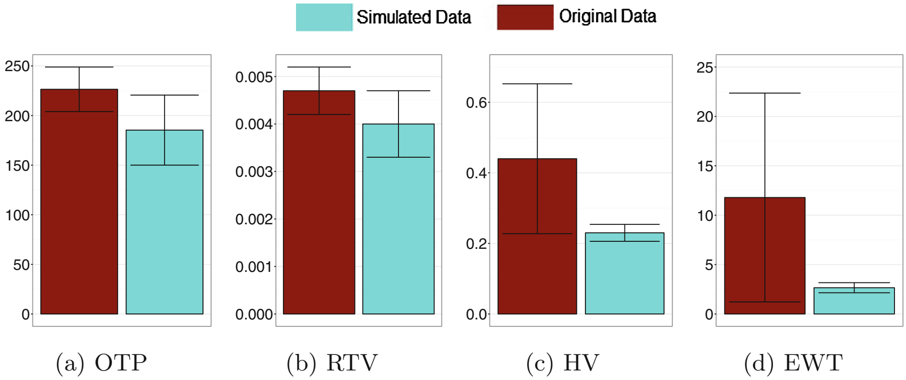


Fig. 5. Consensual clustering results, $K = 3$. Note the coverage's change on the workdays from Summer to Winter period suggested by the highlighted area.



(a) OTP (b) RTV (c) HV (d) EWT

Fig. 6. *Before-and-after* impact evaluation of the novel schedule coverage reliability assessed using a data-driven simulation procedure.

potential gains of performing such change. However, such gains are mainly theoretical boundaries. They may be biased by the multiple constraints of daily PT operations, as well as by the oversimplification of the dwell time's computation (i.e. used the constants computed as described in Sect. 2.1). Consequently, an on-field deployment of this new coverage would be necessary to determine the exact impact of the suggested changes.

5 Final Remarks

This paper introduces a novel procedure to improve schedule coverage on PT networks. It is based solely on AVL/APC data. The final goal is to improve PT reliability and, consequently, their ridership and cost efficiency. Our main contribution is an *ad-hoc* metric to select the best number of schedules to put in place giving four decision factors - punctuality, adequacy, interpretability and reliability - modeled throughout sequence mining and probabilistic reasoning. To the best of our knowledge, this is first data driven framework to automatically select the number of schedules to be put in place using real-world data from a PT operator. Experimental results uncovered the potential gains introduced by this framework. As future work, the authors intend to evaluate it on a real-world testbed. Moreover, we also expect to create adequate exceptions on the concept of frequent itemset to relevant *outliers* on this domain (e.g. a schedule for the Christmas week) and identify when changes in round-trip times require introducing a novel schedule. This is still an open research question.

Acknowledgements. This work was also supported by the European Commission under TEAM, a large scale integrated project part of the Seventh Framework Programme for research, technological development and demonstration [Grant Agreement No. 318621]. The authors would like to thank all partners within TEAM for their cooperation and valuable contribution.

References

1. Moreira-Matias, L., Mendes-Moreira, J., Freire de Sousa, J., Gama, J.: Improving mass transit operations by using avl-based systems: a survey. IEEE Trans. Intell. Transp. Syst. **16**(4), 1636–1653 (2015)
2. Mendes-Moreira, J., Moreira-Matias, L., Gama, J., Freire de Sousa, J.: Validating the coverage of bus schedules: a machine learning approach. Inf. Sci. **293**, 299–313 (2015)
3. Mazloumi, E., Mesbah, M., Ceder, A., Moridpour, S., Currie, G.: Efficient transit schedule design of timing points: A comparison of ant colony and genetic algorithms. Transp. Res. Part B: Methodol. **46**(1), 217–234 (2012)
4. Cats, O., Mach Rufi, F., Koutsopoulos, H.: Optimizing the number and location of time point stops. Public Transp. **6**(3), 215–235 (2014)
5. Jorge, A.M., Mendes-Moreira, J., de Sousa, J.F., Soares, C., Azevedo, P.J.: Finding interesting contexts for explaining deviations in bus trip duration using distribution rules. In: Hollmén, J., Klawonn, F., Tucker, A. (eds.) IDA 2012. LNCS, vol. 7619, pp. 139–149. Springer, Heidelberg (2012)
6. Patnaik, J., Chien, S., Bladikas, A.: Using data mining techniques on apc data to develop effective bus scheduling. J. Syst. Cybern. Inf. **4**(1), 86–90 (2006)
7. Pei, J., Han, J., Mortazavi-Asl, N., Pinto, H., Chen, Q., Dayal, U., Hsu, M.: Prefixspan: mining sequential patterns efficiently by prefix-projected pattern growth. In: ICCCN, p. 0215. IEEE (2001)
8. Fraley, C., Raftery, A.: Model-based clustering, discriminant analysis, and density estimation. J. Am. Stat. Assoc. **97**(458), 611–631 (2002)
9. Matias, L., Gama, J., Mendes-Moreira, J., Freire de Sousa, J.: Validation of both number and coverage of bus schedules using avl data. In: 13th IEEE Conference on Intelligent Transportation Systems (ITSC), pp. 131–136 (2010)
10. Schwarz, G., et al.: Estimating the dimension of a model. Ann. Stat. **6**(2), 461–464 (1978)
11. Wagner, R., Scholz, S., Decker, R.: The number of clusters in market segmentation. In: Baier, D., Decker, R., Schmidt-Thieme, L. (eds.) Data Analysis and Decision Support. Studies in Classification, Data Analysis, and Knowledge Organization, pp. 157–176. Springer, Heidelberg (2005)
12. R Core Team: R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria (2012). ISBN 3-900051-07-0
13. Fraley, C., Raftery, A., Scrucca, L.: Normal mixture modeling for model-based clustering, classification, and density estimation. Department of Statistics, University of Washington **23**, 2012 (2012)

14. Tabei, Y.: An imprementation of prefixspan (prefix-projected sequential pattern mining), August 2015. https://code.google.com/p/prefixspan/people/list. last access at August 2015
15. Ceder, A.: Urban transit scheduling: framework, review and examples. J. Urban Plann. Dev. **128**(4), 225–244 (2002)
16. Cover, T., Hart, P.: Nearest neighbor pattern classification. IEEE Trans. Inf. Theory **13**(1), 21–27 (1967)