Determination of Peak Times in Public Transportation

Kamer Özgün
Industrial Engineering Department
Antalya Bilim University
Antalya, Turkey
kamer.ozgun@antalya.edu.tr

Melih Günay
Computer Engineering Department
Akdeniz University
Antalya, Turkey
mgunay@akdeniz.edu.tr

Barış Doruk Başaran
Computer Engineering Department
Akdeniz University
Antalya, Turkey
doruk07@gmail.com

2021 Innovations in Intelligent Systems and Applications Conference (ASYU) | 978-1-6654-3405-8/21/$31.00 ©2021 IEEE | DOI: 10.1109/ASYU52992.2021.9599009

***Abstract*—Big data and advances in data mining promise to overcome major challenges in public transportation that result in improvements in network efficiency and rider satisfaction. In this study, we develop methods i) to determine peak and off-peak hours and ii) to cluster boarding profiles of demand throughout a day for all bus lines of the public transportation network. With the outcome of this study, it will be possible to propose bus scheduling algorithms that take into account the dynamic nature of ridership patterns by both time and location. As a case study, the passenger boarding patterns of the Antalya transportation network were analyzed. Among the bus lines, 2 are selected to demonstrate that the peak and off-peak times are not necessarily the same, and a dominant demand peak or several demand peaks are possible. Based on the peak demand characteristics, buses may be scheduled.**

***Index Terms*—Peak-Hour, Smart Card, Time Series Clustering, Public Transportation, Schedule**

I. Introduction

In recent years, the data and information has improved considerably with new technologies. Automated Fare Collection (AFC), Automatic Passenger Counter (APC), Automated Vehicle Location (AVL), and Geographical Positioning Systems (GPS) have accelerated research on transit systems [1, 2, 3]. It is now possible to examine big transportation data for developing novel approaches to improve transportation efficiency and comfort [4, 5, 6].

Performance measures that are of interest to transit planners could be generated at different scales, including transit system, neighborhoods, routes, and stop levels by the big data analytic and visualization of big data [7]. Performance of a transportation system as a whole requires, individual vehicles which are limited in numbers, run as efficient as possible while keeping passengers happy through accessibility to the service, quick arrival to the destination, connectivity and reach of the transit network and comfort during transportation. In general, improvements in system performance result with improvements in many of the perceived properties although there may be conflicting indicators such as speed vs safety. Performance indicators and quality attributes in transportation are discussed recently by [8, 9].

The authors would like to thank Ulaşım Planlama ve Raylı Sistem Dairesi Başkanlığı - Antalya for the data provided in this pilot study.

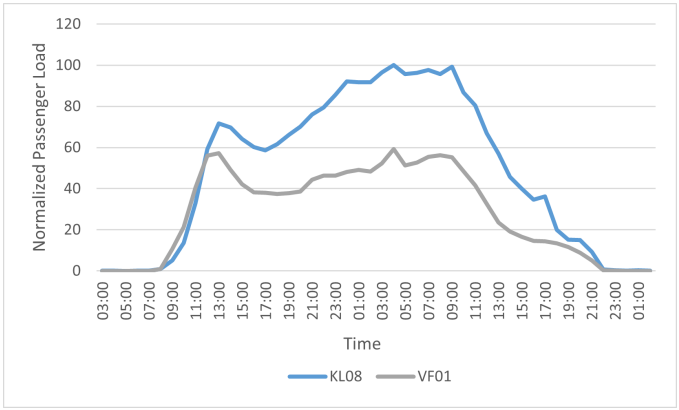

Fig. 1. Daily Demand Comparison Between Bus Lines KL08 and VF01

The most commonly studied quality attributes of public transportation are reliability (i.e. the matches between actual service and route timetable), frequency, speed, accessibility, waiting time at the bus stop, in-vehicle journey time, vehicle occupancy [10]. From the passenger's point of view, perhaps, comfort which is hard to measure and quantify is one of the most important perceived performance indicator, especially for potential users. We may assume that, the space available per passenger and the quality of trip determined via road and bus conditions determines the comfort. As a factor in direct relation to both service efficiency and quality, bus occupancy level is crucial to control and manage limited vehicles in public transport operations.

While most public transport authorities offer similar schedules for all weekdays, the demand for bus lines are constantly changing during the day and even between bus lines. As an example, Figure 1 shows the comparison of daily demands for two most frequently used bus lines, KL08 and VF01, in Antalya. Here, the boarding data on a bus line is aggregated in 30-minute time intervals, so that there are 48 time points on the x-axis in a 24-hour period starting at 3:00 and ending at 02:59. A normalized boarding count is the average of all boarding counts of a particular bus line in a particular time interval divided by the maximum demand over all bus lines and all time intervals. In Figure 1, the maximum demand is at 15:30

on KL08. VF01 reaches up to 60% of the maximum demand during the day. At 18:00, 99.27% and 55% of the maximum demand are realized on KL08 and VF01 respectively. In other words, if a schedule is created to meet the maximum demand for KL08 between 18:00 and 18:30, it is expected to realize occupancy levels around 55% for VF01. It can be concluded from Figure 1 that the number of trips needed to satisfy the passenger demand are different for different lines in each time interval. Once the time-slots have been determined, optimum frequencies may be set to meet the demand. However, note that the purpose of this study is not to offer optimum frequencies, but rather to identify the boundaries of rush hours for a particular bus line obtained by time clustering of smart card data.

Clustering on smart card data may provide a direct look at the passenger travel pattern [2, 1, 3, 11]. With a better knowledge of demands by time or location, services could be adjusted to optimize the public transportation system [12]. For an extensive literature review on the use of smart card data in strategic, tactical and operational aspects in recent years, readers may refer to [5].

References [13, 14, 15, 16, 17], use trip records of individuals to create passengers' trip chains, i.e. the trip sequences for individuals, then the clusters are discovered and labelled based on the temporal and spatial characteristics. For example, Mohamed et al. identify groups of passengers that have similar boarding times aggregated into weekly profiles and peak times have been identified using day-hour heat maps. Different from the travel chain approach, for example, Matias et al.'s effort focused on clustering bus stops, clustering time based on weekdays and/or clustering transportation demand reasons of bus schedules. As far as we know, time series clustering has not applied to characterize the daily demand profiles of routes for a bus network. Furthermore, establishing the time slots of boarding density for all bus lines of a public transportation network using the smart card data is the first in literature.

II. Methodology

The data set is provided by the Department of Transportation for the city of Antalya. Most of the analysis are done using Knime software [19] and custom developed Python scripts. The boarding data consist of passenger id, passenger's boarding stop id (origin), boarding time, bus id, route id (the direction on a particular line). We load the complete 60-day boarding data of May and October. The data set consists of 305 lines and 608 routes. A route is a sequential list of bus stops in either forward or backward (return) directions. In the data set, a total of 455,069 bus trips were made and with these trips a total of 21,629,402 passengers were carried. We have filtered bus lines that have a significant number of passengers and have nearly equal boarding counts in each direction.

In order to improve time-demand partition accuracy and visualization, boarding counts on a particular bus line are grouped into series of 30-minute time intervals as data fields. It is also assumed that the 24-hour period starts at 3:00 and ends at 2:59 the next day when passenger activities are minimum. [20],

The purpose of this study is to propose methods for clustering a typical commute day into a limited number of time slots, each of those includes a subset of sequential 30-minute time intervals. Please note that, a 30-minute time interval is a data field that includes boarding data of passengers, and a time slot is a cluster of peak or off-peak hours. While clustering, it is aimed to minimize variations in boarding counts within time slots relative to a chosen threshold. In other words, the goal is choosing a low number of clusters yet keeping sum of squared errors SSE, over the number of passengers low. The number of clusters results in a threshold value and vice versa. In following sections, two clustering algorithms that differ from each other in the way they use threshold values, are presented in detail.

A. Stepped Time Slot Clustering, STSC

Algorithm 1 is initialized with a particular bus line $LineID$, and a threshold value $Threshold$. It returns time slots as clusters of 30-minute time intervals with different numbers of passengers in each interval. Here, $Threshold$ value is defined in terms of number of passengers. The center of each time slot is the average number of passengers in that time slot. The absolute difference of centers between consecutive iterations is compared with $Threshold$ to decide the creation of a new cluster.

Algorithm 1 STSC

```
procedure STSC(LineID,Threshold)
    Load boarding dataset
    Filter dataset by LineID
    Group boarding data on LineID into the 30-minute TimeInterval
    Sort TimeInterval by ascending times
    Create new Cluster
    for all TimeInterval do
        Set CurrentMean equal to the mean of the boarding counts in the Cluster
        Set NewMean equal to mean boarding counts for Cluster and TimeInterval
        if | NewMean − CurrentMean |> Threshold then
            Create new Cluster
        Add new time slot Cluster to the Cluster
    return Clusters
```

B. Adaptive Time Slot Clustering, ATSC

Similar to STSC, given a bus line $LineID$, and a threshold value $ToleranceAngle$, Algorithm 2 returns time slots as clusters of 30-minute time intervals with different numbers of passengers in each interval. Here, $ToleranceAngle$ is defined in terms of angle, since the approach in ATSC is based on determination of the change in demand. The center of each time slot is the slope obtained by linear regression over the number of passengers in that time slot. The absolute difference

of centers between consecutive iterations is compared with $ToleranceAngle$ to decide the creation of a new cluster.

Algorithm 2 ATSC

procedure ATSC($LineID$,$ToleranceAngle$)
  Load boarding dataset
  Filter dataset by $LineID$
  Group boarding data on $LineID$ into the 30-minute $TimeInterval$
  Sort $TimeInterval$ by ascending times
  Create new $Cluster$
  $NodeList = []$
  for all $TimeInterval$ **do**
    if $NodeList$ has 1 element **then**
      Calculate the slope of a line which contains the node in $NodeList$ and $TimeInterval$ as $InitialAngle$
    if $NodeList$ has more than or equal to 2 elements **then**
      Calculate the angle of a linear regression of $TimeInterval$ and current elements of $NodeList$ as $LRAngle$
      if $|\ LRAngle - InitialAngle\ | > ToleranceAngle$ **then**
        Add all elements of the $NodeList$ to the $Cluster$
        Create new $Cluster$
        $NodeList = []$
    Add $TimeInterval$ to $NodeList$
  return $Clusters$

III. Results and Discussions

A. Results of STSC

Algorithm 1 denoted by STSC is executed with a given bus line and a given threshold value in passenger counts. Figure 2 presents the results of the elbow method over 60-day boarding data for the two most frequently used bus lines KL08 and VF01. In Figure 2, on y-axis, variations over passenger counts i.e. SSE, are presented as percent of the largest SSE in order to make a comparison between bus lines.

The goal is choosing a low number of clusters yet keeping the percent of largest error low. From Figure 2, when the number of clusters are 8 for both bus lines, the thresholds for passenger counts can be set to 3750 and 3000 for KL08 and VF01 respectively.

Threshold values may vary for different bus lines and may result in alternative number of clusters. To illustrate, seven of the most frequently used bus lines in Antalya, are arbitrarily picked and tested with a similar approach. The threshold values in passenger counts and corresponding number of clusters are given according to the bus lines in Table I.

Results of Algorithm 1 on KL08 are given in Table II when $Threshold$ is set to be 3750 passengers. In Table II, the maximum number of passengers is in cluster 4 that is between 16:00 and 20:00 (including nine 30-minute time intervals.) However, the average number of passengers is highest in cluster 3 between 13:00 and 15:30 (with six 30-minute time intervals). Thus, a significant change of boarding over time is expected in Cluster 3, which averaged larger than Cluster 4 with a narrower time frame. Furthermore, it makes sense to consider the cluster with the highest mean as the peak hour which is Cluster 3 on KL08.

Results of Algorithm 1 on VF01 are given in Table III when $Threshold$ is set to be 3000 passengers. In Table III, the maximum average number of passengers is realized in cluster 4 which is the time slot starts at 15:30 and ends at 18:30. Please note that, peak hour boundaries for bus lines KL08 and do not overlap VF01.

Similar to Figure 1 presented previously, Figure 3 shows the

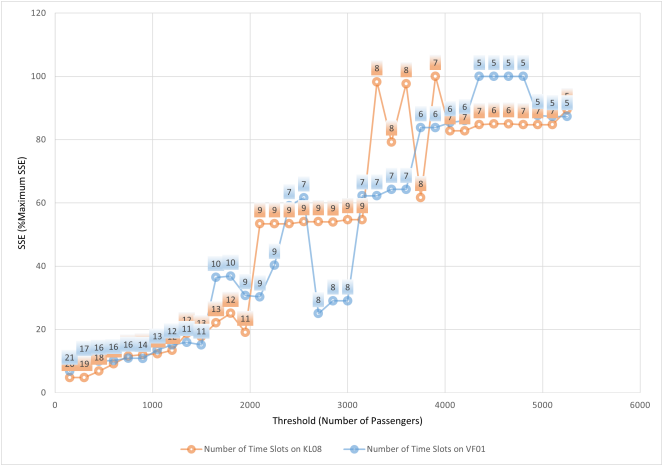

Fig. 2. The Results of Elbow Method on KL08 and VF01

TABLE I
THRESHOLDS IN PASSENGER COUNTS

Line Code	Clustering Input Selection	
	Threshold	*# of Clusters*
KL08	3750	8
VF01	3000	8
KC06	3060	6
LC07	3180	6
LF10	4500	6
UC11	2780	7
VL13	4020	5

TABLE II
RESULTS OF STSC ON KL08 WHEN $Threshold = 3750$ PASSENGERS

Time Slot	*Start Time*	*End Time*	*# of Passengers*	*Average # of Passengers*
0	03:00	07:30	1002	100
1	08:00	10:30	3443	574
2	11:00	12:30	2603	651
3	13:00	15:30	4973	829
4	16:00	20:00	6919	769
5	20:30	22:30	1574	315
6	23:00	01:00	363	73
7	01:30	02:30	7	2

TABLE III
RESULTS OF STSC ON VF01 WHEN $Threshold = 3000$ PASSENGERS

Time Slot	*Start Time*	*End Time*	*# of Passengers*	*Average # of Passengers*
0	03:30	06:30	296	42
1	07:00	07:30	862	431
2	08:00	09:00	1326	442
3	09:30	15:00	4683	390
4	15:30	18:30	3378	483
5	19:00	20:00	873	291
6	20:30	23:30	881	126
7	00:00	02:30	48	8

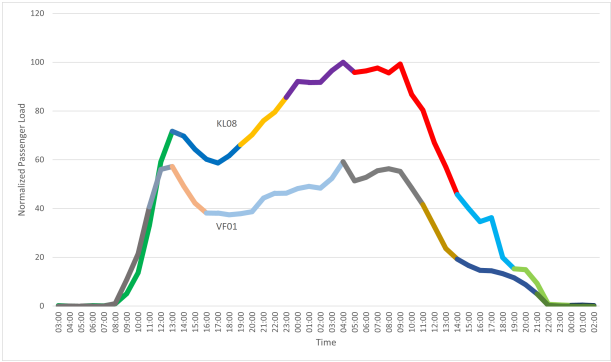

Fig. 3. STSC Result for KL08 and VF01

comparison of daily demands for KL08 and VF01, this time with time slots given by STSC algorithm. Same as before, in Figure 3, 30-minute time intervals are represented on the x-axis, and normalized boarding counts are given on the y-axis. The upper pattern in the figure belongs to KL08, and the pattern below is that of VF01. The inferences from Figure 3 are as follows:

- The morning peak is at 8:30 on KL08. It is at 8:00 on VF01 and the change in passenger counts breaks morning peak into clusters 1 and 2.
- Afternoon peak times are in clusters 3 and 4 at 15:30 and 18:00 on KL08, whereas it is at 17:30 on VF01.
- Evening peak times are in clusters 5, 6 and 7 on KL08. However, evening peak is not obvious for VF01.

B. Results of ATSC

The tolerance angle is the threshold and the main input for ATSC algorithm. It may be calculated again using the elbow method stated earlier. In case of ATSC, SSE is the difference between the slope of actual boarding counts at time (t) and the slope of regression line on boarding counts corresponding to time (t).

Table IV indicates tolerance angle in degrees and the corresponding cluster counts for seven lines that are selected previously in Table I. Note that, as the tolerance angle decreases the number of cluster increases and vice versa. After the threshold is set as tolerance angle for a route, Algorithm 2 denoted by ATSC is executed to establish the time slots. Table V shows the time slots on KL08 when $ToleranceAngle$ is 13 degrees. According to this table, mean boarding counts keep increasing until 18:00. The maximum mean is realized between 17:30 and 18:00 on KL08. In addition, the first and the last clusters have lowest boarding counts.

TABLE IV
THRESHOLDS IN DEGREES

Line Code	Clustering Input Selection	
	Tolerance Angle	*# of Clusters*
KL08	13	6
VF01	27	8
KC06	13	6
LC07	28	4
LF10	9	11
UC11	34	5
VL13	29	7

TABLE V
RESULTS OF ATSC ON KL08 WHEN $Threshold = 13$ DEGREES

Time Slot	*Start Time*	*End Time*	*# of Passengers*	*Average # of Passengers*
0	03:00	06:00	58	8
1	06:30	09:00	2778	463
2	09:30	10:00	1060	530
3	10:30	17:00	10710	765
4	17:30	18:00	1738	869
5	18:30	02:30	4541	267

TABLE VI
RESULTS OF ATSC ON VF01 WHEN $Threshold = 27$ DEGREE

Time Slot	*Start Time*	*End Time*	*# of Passengers*	*Average # of Passengers*
0	03:30	06:00	105	17
1	06:30	08:30	2002	400
2	09:00	10:00	1057	352
3	10:30	12:00	1411	353
4	12:30	15:00	2592	432
5	15:30	16:30	1457	486
6	17:00	18:00	1522	507
7	18:30	02:30	2234	131

In Table V, it is observed on KL08 that the average number of passengers increases over time and decreases after 18:00. However, the trend is changing in Table VI on VF01.

Figure 4, shows the comparison of daily demands for KL08 and VF01, with time slots given by ATSC algorithm. Same as before, the upper pattern belongs to KL08, and the pattern below is that of VF01. It is observed on KL08 that the trend is increasing until 18:00 excluding 1-hour-microtrends in clusters 2 and 4 and then starts decreasing. The changing trend on VF01 can be clearly observed form this figure.

By comparing the figures 3 and 4, it can be concluded that both algorithms cannot avoid outlier boarding counts at 15:30 and identify it as the peak. A general result is that STSC identifies passenger counts while ATSC identifies boarding trends in a daily transportation activity.

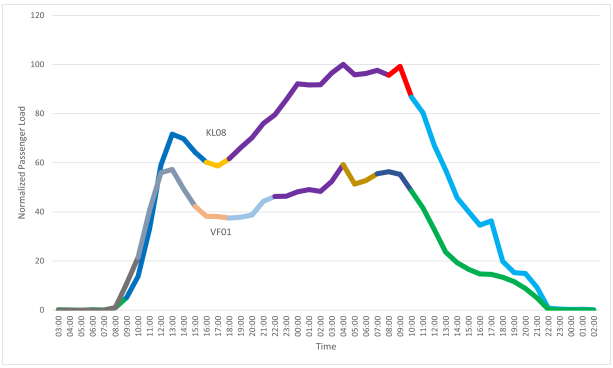

Fig. 4. ATSC Results for KL08 and VF01

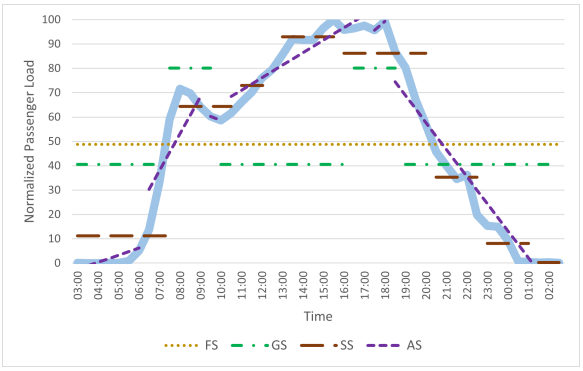

Fig. 5. Comparison of The Methods on KL08

C. Comparison of Methods on KL08

Once the time-slots have been determined, it may be possible to set bus service frequencies meeting the demand. Under the assumption that the demand can be represented by the mean passenger count in a time slot [21], performances of four possible methods to determine boundaries of time slots are illustrated on bus line KL08 in this section:

- ***Fixed Schedule (FS):***
  The first method FS assumes a single time slot for the entire day. A fixed schedule is adjusted to the average of boarding counts over the entire day [21].
- ***Given Schedule (GS):***
  The second method GS assumes given time slots for peak and off-peak hours. For examples, [11] states 6:00-7:00 and 16:30-17:30 as rush hours, and similarly, [16] determines first peak occurring at 12:00 and the evening peak occurring between 16:00–18:00. In this study, for GS, rush hours are stated as 7:00-9:00 and 16:00-18:00, and schedules are adjusted to the average of boarding counts for these given time slots
- ***Stepped Schedule (SS):***
  Stepped schedules are adjusted to the average of boarding counts in each time slot calculated by the Algorithm 1, STSC
- ***Adaptive Schedule (AS):***
  Adaptive schedules are adjusted to the linear regression fit of boarding counts in each time slot calculated by the Algorithm 2, ATSC

The performances of these methods can be simulated on a given bus line.

Figure 5 shows a breakdown of time slots for the bus line KL08 as a typical daily boarding request. The time slot boundaries vary significantly depending on the method chosen. For performance evaluation, the proximity of the lines of the methods to the demand line in Figure 5 can be compared. The first method FS, is expected to result in the lowest performance because most of the time the demand is significantly above average. The second method GS provides a slight improvement over FS. The third method SS, makes it possible to identify groups of time intervals that have similar boarding counts aggregated into time slots. It can be stated that the performance of the last method AS is the best as expected. Note that, depending on the purpose, it is always possible to trim or merge the time slots from both ends.

Conclusion

Peak and off-peak times and demand profiles are not same for each bus line. For a typical transportation network, some bus lines have single and other have more peak demand throughout a day. Moreover, peak time boundaries of bus lines do not necessarily overlap. Such knowledge extracted from the big public transportation data may be useful in public transportation planning. For instance, when scheduling transportation services, frequencies may be adjusted according to the changes in the boarding counts during a day. Such dynamic frequency scheduling increases not only transportation efficiency but also commuter comfort. Thus higher quality of service while reducing costs is achievable. As a future study, the demand profiling approach proposed in this paper may be used to establish the frequency for a bus line dynamically.

References

[1] G. Harrison, S. M. Grant-Muller, and F. C. Hodgson, "New and emerging data forms in transportation planning and policy: Opportunities and challenges for "track and trace" data," *Transportation Research Part C: Emerging Technologies*, vol. 117, p. 102672, 2020.

[2] K. Lu, J. Liu, X. Zhou, and B. Han, "A review of big data applications in urban transit systems," *IEEE Transactions on Intelligent Transportation Systems*, 2020.

[3] T. F. Welch and A. Widita, "Big data in public transportation: a review of sources and methods," *Transport Reviews*, vol. 39, no. 6, pp. 795–818, 2019. [Online]. Available: https://doi.org/10.1080/01441647.2019.1616849

[4] L. Zhu, F. R. Yu, Y. Wang, B. Ning, and T. Tang, "Big data analytics in intelligent transportation systems: A survey," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 1, pp. 383–398, 2019.

[5] M.-P. Pelletier, M. Trépanier, and C. Morency, "Smart card data use in public transit: A literature review,"

Transportation Research Part C: Emerging Technologies, vol. 19, no. 4, pp. 557–568, 2011.

[6] N. Van Oort and O. Cats, "Improving public transport decision making, planning and operations by using big data: Cases from sweden and the netherlands," in *2015 IEEE 18th international conference on intelligent transportation systems*. IEEE, 2015, pp. 19–24.

[7] C. Stewart, R. Bertini, A. El-Geneidy, and E. Diab, "Perspectives on transit: Potential benefits of visualizing transit data," *Transportation Research Record Journal of the Transportation Research Board*, vol. 2544, 01 2016. [Online]. Available: https://doi.org/10.3141/2544-11

[8] C. Daraio, M. Diana, F. Di Costa, C. Leporelli, G. Matteucci, and A. Nastasi, "Efficiency and effectiveness in the urban public transport sector: A critical review with directions for future research," *European Journal of Operational Research*, vol. 248, no. 1, pp. 1–20, 2016.

[9] L. Dell'Olio, A. Ibeas, and P. Cecin, "The quality of service desired by public transport users," *Transport Policy*, vol. 18, no. 1, pp. 217–227, 2011.

[10] L. Redman, M. Friman, T. Gärling, and T. Hartig, "Quality attributes of public transport that attract car users: A research review," *Transport Policy*, vol. 25, no. C, pp. 119–127, 2013. [Online]. Available: https://doi.org/10.1016/j.tranpol.2012.11.005

[11] T. Li, D. Sun, P. Jing, and K. Yang, "Smart card data mining of public transport destination: A literature review," *Information*, vol. 9, no. 1, p. 18, 2018.

[12] K. Özgün, M. Günay, B. Bulut, E. Yürüten, M. F. Baysan, and M. Kalemsiz, "Analysis of public transportation for efficiency," in *Trends in Data Engineering Methods for Intelligent Systems - ICAIAME 2020*, J. Hemanth, T. Yiğit, B. Patrut, and A. Angelopoulou, Eds., vol. 76. Springer International Publishing, 2021.

[13] H. Faroqi and M. Mesbah, "Inferring trip purpose by clustering sequences of smart card records," *Transportation Research Part C: Emerging Technologies*, vol. 127, p. 103131, 2021.

[14] L. He, B. Agard, and M. Trépanier, "A classification of public transit users with smart card data based on time series distance metrics and a hierarchical clustering method," *Transportmetrica A: Transport Science*, vol. 16, no. 1, pp. 56–75, 2020.

[15] A.-S. Briand, E. Côme, K. Mohamed, and L. Oukhellou, "A mixture model clustering approach for temporal passenger pattern characterization in public transport," *International Journal of Data Science and Analytics*, vol. 1, no. 1, pp. 37–50, 2016.

[16] K. Mohamed, E. Côme, L. Oukhellou, and M. Verleysen, "Clustering smart card data for urban mobility analysis," *IEEE Transactions on intelligent transportation systems*, vol. 18, no. 3, pp. 712–728, 2016.

[17] X. Ma, Y.-J. Wu, Y. Wang, F. Chen, and J. Liu, "Mining smart card data for transit riders' travel patterns," *Transportation Research Part C: Emerging Technologies*, vol. 36, pp. 1–12, 2013.

[18] L. Matias, J. Gama, J. Mendes-Moreira, and J. F. De Sousa, "Validation of both number and coverage of bus schedules using avl data," in *13th International IEEE Conference on Intelligent Transportation Systems*. IEEE, 2010, pp. 131–136.

[19] D. F. G. T. K. T. M. T. O. P. S. C. T. K. W. B. Berthold M.R., Cebron C., "Knime: The konstanz information miner," in *Studies in Classification, Data Analysis, and Knowledge Organization*, 2007.

[20] J. Barry, R. Freimer, and H. Slavin, "Use of entry-only automatic fare collection data to estimate linked transit trips in new york city," *Transportation Research Record*, vol. 2112, no. 1, pp. 53–61, 2009.

[21] A. Ceder, *Public transit planning and operation: Modeling, practice and behavior*. CRC press, 2016.