Two-Dimensional Inhomogeneous Markov Chains to Infer Timeliness of Buses in Real-Time

Rémi Dion [*1] **Alexander Peplowski** [*1] **Léa Ricard** [*1]

1. Introduction

Improvements in public transport service can have diverse positive impacts on communities. When a larger number of travelers choose public transport or active transport over private car, it can significantly reduce congestion as well as air pollution. Poor timeliness may discourage travelers from using public transport. One way to cope with this issue is to provide real-time information about delays to users. Thus, travelers can plan their trips ahead of time based on the updated arrival time and thereby reduce their waiting time.

In a work on traffic flow forecasting, Sun et al. (1) suggested that the flow on each road segment depends on the flow of adjacent upstream road segments. The joint probability of road segment flows is modeled by a Gaussian mixture model. Another work modeled passenger flow in a urban rail station using a dynamic Bayesian network with a graph structure directly derived from the topology of the transport network (2). In a recent work, Wiecek et al. (3) fitted an homogeneous Markov chain to predict on board bus comfort level, which is related to bus occupancy. They found that interchanging stops had a poor prediction accuracy and suggested using inhomogeneous Markov chains for future work. They stated that implementing Markov chains with transition matrices that vary over time and type of bus stop could improve results.

In this work, we will verify the spatiotemporal dependence between the timeliness at different bus stops using Markov chain processes. Three main axes along which dependence is anticipated will be explored: per-stop, per-trip, and per-day. The Markovian hypothesis is that the delay at a given spaciotemporal point depends only on the adjacent and upstream states. The remainder of this paper is organized as follows. Section 2 defines the bus delay prediction problem. Models, namely one-dimensional and two-dimensional finite Markov chains (4), are introduced in Section 3. Sections 4 and 5 present the dataset used in our experiments and its preparation. The experimental setup and our results are presented in Sections 6 and 7 before the main findings of this work are discussed in Section 8.

[*]Equal contribution [1]Department of Computer Science and Operations Research, University of Montréal, Montréal, Canada. Correspondence to: Rémi Dion <remi.dion.1@umontreal.ca>, Alexander Peplowski <alexander.peplowski@gmail.com>, Léa Ricard <lea.ricard@umontreal.ca>.

IFT 6269 project report - Université de Montréal

2. Problem Definition: Bus Delay Prediction

Given the knowledge about past stops, past trips and past days, the bus delay prediction problem attempts to infer the timeliness of either the next stop, the next trip or the next day. We define a bus route as an ordered sequence of road segments and bus stops, where the first and the last stops are called terminals. A bus line usually has two associated routes, each one going in opposite directions (e.g., outbound and inbound). A trip is defined as a unique voyage which follows a specific route according to a predefined schedule. We use the definition of delay, or timeliness, specified by some US transit authorities (5):

- Early ($>$ 1 min early)
- Late ($>$ 5 min late)
- On Time (otherwise)

We make three hypotheses that lead to three bus delay prediction subproblems, which can be combined at a later point:

Next stop prediction (x-axis): Following the hypothesis that a bus late (or early) at a given stop is likelier to be late (or early) at the next stop, the next stop prediction problem is to infer the state (i.e. timeliness) of the next stop as a function of the previous stop state.

Next trip prediction (y-axis): For a given stop of a bus route, we hypothesize that if a bus is late (or early) at that stop the subsequent trip is likelier to be late (or early) at that same stop. For example, if a bus is late at that stop because of unusual road congestion, it is likely that subsequent trips will suffer from the same road congestion. The next trip prediction problem is to infer, for a given bus stop, the state of the next trip given the previous trip state.

Next day prediction (z-axis): We hypothesize that a late bus (or early bus) at a given stop and time of day is likelier to be late (or early) the next day at that same stop and time. Thus, the next day prediction problem is to infer, based on the delay at a specific stop and a specific period of the previous day, the delay at the same stop and the same time period for the next day.

3. Models

The timeliness of a bus is computed by comparing the scheduled arrival time and the actual arrival time of a bus. The actual arrival time at a stop is for its part computed using the actual travel time on the previous segment. This measure, the actual travel time, is stochastic. Using the assumptions of (2), namely that the graph structure can be directly derived from the transport network topology, it is intuitive to model the timeliness of buses using Markov chains processes.

Consider a finite Markov chain along the x-axis, which is used to model the next stop timeliness. Let $X = (X_1, X_2, \dots, X_S)$ be a sequence of random variables, where X_s is the random variable associated with the timeliness at bus stop s. Let $\mathcal{T} = \{t_1, t_2, t_3\}$ be a discrete state space, with t_1 being early, t_2 being on time and t_3 being late. The Markovian assumption is that the probability of an event depends only on the previous event:

$$p(X_s \mid X_1, X_2, \dots, X_{s-1}) = p(X_s \mid X_{s-1}) \qquad (1)$$

In the homogeneous case, the transition matrix P is unique and $p(X_s = t_i \mid X_{s-1} = t_j) = P_{ij}$. In our model, we let P depend either on the time of the day, the stop identifier or both. Let $P^{(p,\cdot)}$ be the transition matrix during time period p, $P^{(\cdot,s)}$ be the transition matrix of stop s and $P^{(p,s)}$ be the transition matrix during time period p of stop p, for each case respectively. For example, we can consider time periods as bins of 1 hour.

Given a set of M observed past events sequences, $\{(\boldsymbol{x}_{11}, \boldsymbol{x}_{12}, \dots, \boldsymbol{x}_{1S}), (\boldsymbol{x}_{21}, \boldsymbol{x}_{22}, \dots, \boldsymbol{x}_{2S}), \dots, (\boldsymbol{x}_{M1}, \boldsymbol{x}_{M2}, \dots, \boldsymbol{x}_{MS})\}$ and their corresponding departure time $\{y_1, y_2, \dots, y_M\}$, the conditional event probability is either

$$p(X_s = t_i \mid X_{s-1} = t_j) = \sum_p P_{ij}^{(p,\cdot)} \delta(p, y_s), \qquad (2)$$

$$p(X_s = t_i \mid X_{s-1} = t_j) = P_{ij}^{(\cdot,s)}, \qquad (3)$$

$$p(X_s = t_i \mid X_{s-1} = t_j) = \sum_p P_{ij}^{(p,s)} \delta(p, y_s), \qquad (4)$$

with $\delta(p, y_s)$ a Kronecker delta function taking value 1 if y_s is in p. Equation 2 is used in the first scenario, when P is time-varying, Equation 3 is used when P is stop-varying and Equation 4 is used when P is both time and stop inhomogeneous. The initial distribution (i.e. distribution of the first stop) is given by π_x when P is not time varying and $\pi_x^{(p)}$ otherwise.

Equivalently, we could define finite Markov chains along the y-axis and the z-axis Along the y-axis, $X = (X_1, X_2, \dots, X_N)$ is a sequence of random variables where X_n is the random variable associated with the timeliness of the trip n at a given stop. Along the z-axis, $X = (X_1, X_2, \dots, X_D)$ is a sequence of random variables where X_d is the random variable associated with the time timeliness of day d at a given stop and a given time period.

We postulate that the combination of two dimensions, either x-axis and y-axis or x-axis and z-axis, could produce better models. From this, a lattice-like graphical model emerges. This lattice model is referred to as a two-dimensional finite Markov chain (4). If we take as example the lattice formed of the x- and y-axis, we have that this model has $S \times N$ random variables. Precisely, let $X = ((X_{11}, X_{12}, \dots, X_{1S}), (X_{21}, X_{22}, \dots X_{2S}) \dots (X_{N1}, X_{N2}, \dots, X_{NS}))$ be a sequence of N trips. Fornasini (4) showed that this two-dimensional finite Markov chain can simply and efficiently be implemented with:

$$\begin{aligned} T_i &= p(X_{n,s} = t_i \mid X_{n,s-1} = t_j, X_{n-1,s} = t_k) \\ &= \alpha P_{ij} + (1 - \alpha) Q_{ik} \end{aligned} \qquad (5)$$

where T is the resulting vector of probability for each state at stop s and trip n given the adjacent states (t_j, t_k), Q is the transition matrix on the y-axis and α is the weight given to the x-axis. Matrices P and Q can be replaced by their time-varying or stop-varying counterparts, using the same logic as in Equations 2, 3 and 4.

This produces a normalized linear combination of both 1-d models. The representation shown in figure 1 depicts the model where each node has two influences and influences two adjacent nodes. The first stop of any trip is influenced by the initial x-axis distribution (π_x or $\pi_x^{(p)}$ if time-varying with respect to the x-axis) and the adjacent node with respect to the y-axis. A stop from the first trip of the day is influenced by the combination of the inital y-axis distribution (π_y or $\pi_y^{(s)}$ is stop-varying with respect to the y-axis) and the adjacent node with respect to the x-axis.

Equivalently, the combination of x-axis and z-axis produces a lattice-like graphical model of $S \times D$ random variables. In this model, we define W as the transition matrix and π_z as the initial distribution on the z-axis.

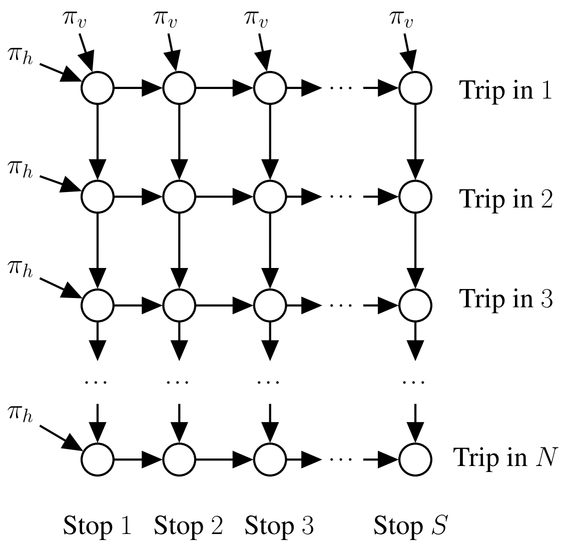


Figure 1. Graphical representation of the two-dimensional finite Markov chain along the x- and y-axis

4. Dataset

The data source we select for our experiments is the Massachusetts Bay Transportation Authority (MBTA) bus arrival and departure times dataset (6). The dataset contains bus arrival and departure times for the 2018 and 2019 seasons. Each sample in the dataset measures actual and scheduled arrival times of one bus arriving at one bus stop. Additionally, each sample has a unique trip identifier, which enables grouping of several (per-stop) samples into a single trip.

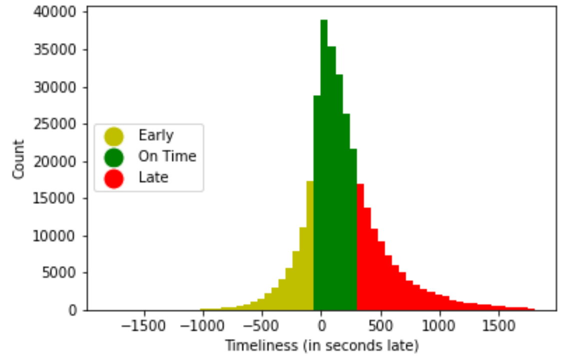
The MBTA dataset provides data on all bus lines in its network, however, we will focus on only a single route when performing our experiments. To maximize data availability, we select the route which has the most per-stop samples. In this case, we select bus routes 01 outbound and inbound.

To gain perspective on the data distribution, Figures 2(a) and 2(b) report the timeliness of the bus arrival at any stop on line 01. For both routes, we observe that the distribution is positive skewed (i.e. more samples are "Late" than "Early"). A majority of the samples of route 01 inbound are "On Time", while route 01 outbound has a majority of samples "Late". The timeliness ranges, for both routes, between 17 minutes early and 30 minutes late.

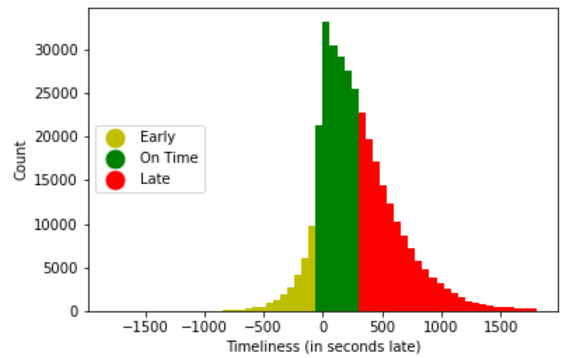
5. Data Preparation

In the course of analyzing the data, some issues were identified:

- The stop sequence (constituting a trip) is not consistent, some stops are missing and not all trips are recorded.



(a) Route 01 inbound



(b) Route 01 outbound

Figure 2. Timeliness Distributions of samples

- The scheduled departure times change as a function of the day (e.g. holidays, weekend)
- Scheduled departure times are occasionally modified by the transit authority.

Our approach using finite Markov chain models (see Section 3) requires a fixed set of stops per route and a fixed number of transitions per day. To make our dataset compatible with these assumptions, some preprocessing steps were applied to the MBTA dataset:

Consistency of stops per route The most common number of stops for a particular route was considered the "correct" number of stops. Additionally, trips with missing stops (incomplete trips) are disregarded.

Consistency of periodic information Since schedules differ on weekends when compared to weekday schedules, weekend information is separated.

Fixing dimensionality of route start time Because scheduled departure times change, it is difficult to have a one-to-one correspondence between trips across several days. To account for this problem, time binning is used to group together departure times into time periods. The first trip starting during a time period is chosen and the other trips in that time period are separated. This enables us to take into account the first departure of the day and the last departure of the day as special cases for which there is no transition into and out of, respectively.

We observe that with a bin size of 20 minutes, the samples per bin ratio is maintained above 1 for most of the day (6am to midnight) on route 01 outbound.

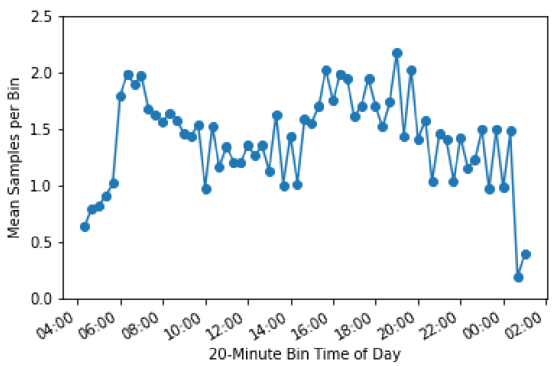


Figure 3. Number of samples per 20 minute bin for route 01 outbound

Data consistency after time binning Depending on the choice of a time bin size, not all time bins will be filled. In the case when a time bin on a particular day is empty, the entire day is dropped. Of course, the bin size should be chosen so that this is a rare event.

Applying the aforementioned processing steps reduces data availability considerably.

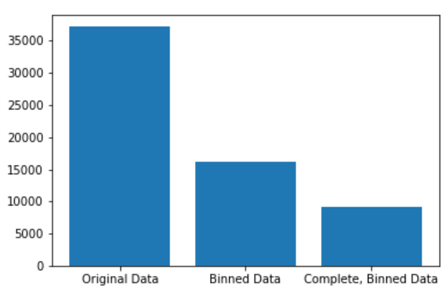


Figure 4. Data loss from preprocessing (measured in samples) for route 01 inbound

Once applying the preparation steps above, the dataset can be seen as a three-dimensional array, with stop 1 to S along the x-axis, trip 1 to N along the y-axis, and day 1 to D along the z-axis. To reiterate, the first axis (x-axis) represents the stops that a bus will visit, the second axis (y-axis) represents the times at which a bus trip begins its route, and the third axis (z-axis) represents sequential weekdays.

6. Experiment Setup

The data set was divided in a training set (40%), a validation set (40%) and a test set (20%).

During the parameters search, we pre-train our models on the training set. In this pre-training, P^{pre}, Q^{pre}, W^{pre}, π_x^{pre}, π_y^{pre} and π_z^{pre} and their respective time-varying and stop varying counterparts, are learned. Then, for each point in the validation set, we use the η recent past trips to learn P^{slid}, Q^{slid}, W^{slid}, π_x^{slid}, π_y^{slid} and π_z^{slid}. This second methodology is commonly called sliding window. Then, both sets of parameters are weighted using γ, such as

$$P = \gamma P^{pre} + (1-\gamma)P^{slid}, \tag{6}$$

and equivalently for Q, π_x, π_y, π_z.

During testing, the pre-training is done on both training and validation data while the sliding window methodology is applied on the test set. To maintain a sliding window containing the same number of trips (over η days), the window overlaps over the pre-train data for the first $(\eta-1)$ iterations.

We tested all combinations of parameters values or model choices listed in Table 1.

Table 1. Tested parameters values or model choices

Param. / model choice	Sym.	Values
Binning size (min)	-	$\{20, 30\}$
Weight of P over Q (or W)	α	$\{0, 0.1, 0.3, 0.5, 0.7, 0.9, 1\}$
x-axis homogeneity	-	$\{P, P^{(p,\cdot)}, P^{(\cdot,s)}, P^{(p,s)}, \pi_{p,s}\}$
y-axis homogeneity	-	$\{Q, Q^{(p,\cdot)}, Q^{(\cdot,s)}, Q^{(p,s)}\}$
z-axis homogeneity	-	$\{W, W^{(p,\cdot)}, W^{(\cdot,s)}, W^{(p,s)}\}$
Sliding window (days)	η	$\{5, 10, 30, 60\}$
Importance of recent trips vs train data	γ	$\{0.5, 0.7, 0.9, 1\}$

7. Empirical Results

Using the most frequent state from the training set as the baseline prediction to compare our results with, we find that this rudimentary approach would lead to an accuracy of roughly 50% for both bus routes.

Table 2 presents the accuracy of all models when predicting either the next trip, the next day or the next stop timeliness state, with the values in bold indicating for each problem and for each dataset the one with the best accuracy. For both route 01 inbound and outbound, the best model to predict the next trip state is the one-dimensional Markov chain on the y-axis, the best model to predict the next day state is the one-dimensional Markov chain on the z-axis and the best model to predict the next stop state is the one-dimensional Markov chain on the x-axis. This is true both on the training and on the test sets. The results show that using single-axis model is both an efficient and accurate way to predict the next state for all evaluated tasks.

Table 2. 1-dimensional and 2-dimensional Markov chains accuracy

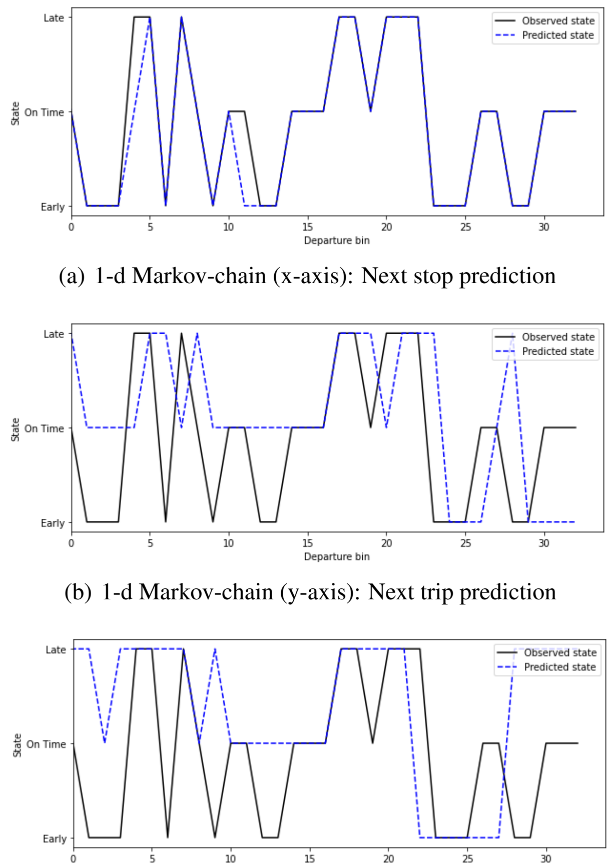
Dataset	**Model**	**Accuracy of next (%)**			
		Trip/ day*	Stop	Trip/ day*	Stop
		Training		*Test*	
Route 01 inbound	Baseline			50.2	50.2
	1-d (x-axis)	53.6	**82.0**	55.3	**82.4**
	1-d (y-axis)	**57.0**	57.0	**57.3**	57.3
	1-d (z-axis)	**57.8**	57.8	**57.9**	57.9
	2-d (x and y)	54.5	81.8	55.5	82.2
	2-d (x and z)	54.8	81.7	55.4	82.2
Route 01 outbound	Baseline			47.5	47.5
	1-d (x-axis)	44.3	**82.3**	52.0	**82.7**
	1-d (y-axis)	**57.9**	57.9	**62.1**	62.1
	1-d (z-axis)	**58.2**	58.2	**62.7**	62.7
	2-d (x and y)	52.7	81.4	56.8	82.2
	2-d (x and z)	53.4	81.1	59.5	81.6

* Next trip accuracy: y-axis; next day accuracy: z-axis

We observe in table 2 that the one-dimensional models on the y- and z-axis have the same accuracy for both problems, namely predicting the next trip (or day) and the next stop state. This is due to the fact that those models do not have a dependence with previous stops. Thus, to predict the next stop state, one-dimensional models on the y-axis and on the z-axis use the state of the last trip at this given stop or the state of the last day at this given stop and time of the day, respectively. Thus, the prediction of the next trip state (or day) is done exactly as the prediction of the next stop state.

During parameters and model selection, a η of 30 days was almost always selected. The most frequent γ value was 0.9 (i.e. recent trips have more importance). The best binning size was of 20 minutes for route 01 inbound and of 30 minutes for route 01 outbound. We suspect this is due to data availability; the former has more samples. Parameter α was always greater or equal to 0.5 in two-dimensional models, which means that more importance is given to the x-axis than to the y- or z-axis. Finally, most models selected transition matrices and initial distributions both time-varying and stop-varying. However, stop inhomogeneity has more importance and affects to a greater extend the accuracy.

Figure 5 compares the observed states and the predicted states of the one-dimensional Markov chains. If in Figure 5(a) the observed and the predicted states are usually the same except for two departure bins where the prediction is off by one level, in Figures 5(b) and 5(c), it is obvious that the prediction is somewhat random and that y-axis and z-axis models are bad at predicting the next trip and the next day timeliness, respectively. Indeed, the prediction is often off by two levels (i.e. the predicted state is late and the actual state is early, or reverse).



(a) 1-d Markov-chain (x-axis): Next stop prediction

(b) 1-d Markov-chain (y-axis): Next trip prediction

(c) 1-d Markov-chain (z-axis): Next day prediction

Figure 5. Observed and predicted timeliness for a given bus stop on a given day, route 01 inbound

8. Conclusions

The empirical results demonstrated that the dependence on the previous trip (y-axis) or on the previous day's trip (z-axis) are not as strong as anticipated. Indeed, table 2 shows that the single Markov chain predictions (1-d) on the y and z axis are about 25% less accurate than the prediction based on the previous stop (x-axis).

Although this is intuitively congruent with the conception that the previous stop's state has a high impact on the following stop's, it also hints towards the fact that the work of scheduling was carefully done by the MBTA planners. Undoubtedly, if we assume that planning is based on historic data, then a certain period of the day which is busier every day would have been taken into account at scheduling time, leaving only a small noise component for which our selected model on the z-axis may not be appropriate to model. This is something we hadn't factored in when considering the possible outcomes at the model design stage. If our interpretation of the results is right, we expect a weak dependence via the z-axis. Therefore, our model may prove useful to identify weaknesses in the planning when a strong z-axis dependence is observed.

To explain the y-axis results, we may interpret that the previous trip's dependence may simply have been hindered due to the time-binning that was applied in order to have day-to-day correspondence. If we think of times of the day with high ridership as having a higher frequency of bus trips, then the constant-time binning that we applied may have hidden useful data or added noise. Considering those results and hypothesis, future work would benefit from using a variable-size time binning method (for trip-dense periods of the day) or even no time binning at all, both of which would produce a number of trips per day which varies depending on the number of available recorded trips. However, this would make the y- and z-axis contribution less readily comparable as z-axis would still require time binning. To compensate for this, the z-axis may be replaced with a stop- & time-dependent input which would consider the influence from the seasonality at every node (e.g., via the estimation of a prior distribution over the past 30[1] days, similar to what was done with $\pi_x^{(p)}$). This methodology would also increase data availability which was significantly reduced with our current approach (See Figure 4).

Future work may also benefit from using more states from which to infer. For example, the actual state may be represented with a 5-state value ($>$2min early, 0-2min early, 0-4min late, 4-6min late, $>$6min late) in order to predict a 3-state value (early, on-time, late) which would, in effect, create a non-squared transition matrix and help take into account trickier cases where the bus is close to the early-to-on-time threshold (0-2min early) or to the on-time-to-Late threshold (4-6min late).

As one-dimensional finite Markov chains with transition matrices and initial probabilities varying with time and stop identifier outperformed two-dimensional models, we recommend training three models separately (one on each axis) and use x-axis, y-axis and z-axis models to predict the timeliness of the next stop, trip and day, respectively. These models are slightly faster to train than their two-dimensional counterparts and thus ideally suited for real-time prediction.

[1] Or some other relevant time period which may capture seasonality

References

[1] Shiliang Sun, Changshui Zhang, and Guoqiang Yu. A bayesian network approach to traffic flow forecasting. *IEEE Transactions on Intelligent Transportation Systems*, 7(1):124–132, 2006.

[2] Jérémy Roos, Gérald Gavin, and Stéphane Bonnevay. A dynamic bayesian network approach to forecast short-term urban rail passenger flows with incomplete data. *Transportation Research Procedia*, 26:53 – 61, 2017. Emerging technologies and models for transport and mobility.

[3] Paweł Wiecek, Daniel Kubek, Jan Aleksandrowicz, and Aleksandra Stróżek. Framework for onboard bus comfort level predictions using the markov chain concept. *Symmetry*, 11:755, 06 2019.

[4] Ettore Fornasini. 2d markov chains. *Mathematical Problems in Engineering*, 140:101 – 127, 1990.

[5] Transit service reliability: Analyzing automatic vehicle location (avl) data for on-time performance and to identify conditions leading to service degradation. Technical report, National Center for Transit Research (NCTR), 2016.

[6] Massachusetts Bay Transportation Authority. Mbta bus arrival departure times 2020.