algorithms

MDPI

Article

Representing and Inferring Massive Network Traffic Condition: A Case Study in Nashville, Tennessee

Hairuilong Zhang [1], Yangsong Gu [2] and Lee D. Han [2,*]

[1] The Bredesen Center for Interdisciplinary Research and Graduate Education, University of Tennessee, Knoxville, TN 37996, USA; hzhan101@vols.utk.edu

[2] Department of Civil and Environmental Engineering, University of Tennessee, Knoxville, TN 37996, USA; ygu17@vols.utk.edu

* Correspondence: lhan@utk.edu

Abstract: Intelligent transportation systems (ITSs) usually require monitoring of massive road networks and gathering traffic data at a high spatial and temporal resolution. This leads to the accumulation of substantial data volumes, necessitating the development of more concise data representations. Approaches like principal component analysis (PCA), which operate within subspaces, can construct precise low-dimensional models. However, interpreting these models can be challenging, primarily because the principal components often encompass a multitude of links within the traffic network. To overcome this issue, this study presents a novel approach for representing and indexing network traffic conditions through weighted CUR matrix decomposition integrated with clustering analysis. The proposed approach selects a subset group of detectors from the original network to represent and index traffic condition through a matrix decomposition method, allowing for more efficient management and analysis. The proposed method is evaluated using traffic detector data from the city of Nashville, TN. The results demonstrate that the approach is effective in representing and indexing network traffic conditions, with high accuracy and efficiency. Overall, this study contributes to the field of network traffic monitoring by proposing a novel approach for representing massive traffic networks and exploring the effects of incorporating clustering into CUR decomposition. The proposed approach can help traffic analysts and practitioners to more efficiently manage and analyze traffic conditions, ultimately leading to more effective transportation systems.

Keywords: network traffic monitoring; intelligent transportation system; CUR matrix decomposition; clustering analysis; big data

check for updates

Citation: Zhang, H.; Gu, Y.; Han, L.D. Representing and Inferring Massive Network Traffic Condition: A Case Study in Nashville, Tennessee. *Algorithms* **2023**, *16*, 485. https://doi.org/10.3390/a16100485

Academic Editor: Ayan Biswas

Received: 1 September 2023
Revised: 7 October 2023
Accepted: 17 October 2023
Published: 19 October 2023

1. Introduction

With the rapid development of intelligent transportation systems (ITSs) in the big data era, a huge amount of up-to-date data are being collected, archived, and analyzed from a variety of sources such as smart phones, probe vehicles, video cameras, and infrastructure-based detectors. Traditional detector loops and more advanced technologies, such as radar and microwave detectors, have been widely deployed in large metropolitan areas to monitor the traffic flow in a real-time manner. These systems deal with thousands of detector stations with high temporal resolution at 20 to 30 s per update, which poses great challenges for the efficiency and computational cost of analyzing ITS data. One of the main challenges is to assess the road network condition quickly and accurately for a given moment and efficiently infer and forecast meaningful spatial and temporal trends for the future, which can be useful for many ITS applications such as monitoring networks, planning traffic, and mitigating congestion [1,2].

Previous research has aimed to model the road network by considering every road segment within it. However, this approach may not be practical for large traffic networks and real-time applications. Furthermore, missing data will significantly impair the effectiveness of this method. To address this issue, our focus is on creating low-dimensional network models that only require monitoring of a selected subset of road segments.

To develop accurate low-dimensional models for large and diverse road networks, prior research has primarily focused on techniques such as principal component analysis (PCA) [3–7]. However, PCA models are difficult to interpret in terms of individual links in the network. Djukic et al. used PCA to analyze OD pair data in a small network [4,5], while Asif et al. applied various subspace methods to compress traffic speed data [8]. These studies demonstrated that subspace methods like PCA and DCT (discrete cosine transform) can effectively reduce the size of traffic data. However, they do not provide detailed insights into traffic patterns for specific roads and time periods. In addition, implementing PCA online still requires information from all detectors in the network because the principal component is basically a linear combination of all columns of the original data, which greatly reduces the efficiency of data processing. This is the reason why PCA is mostly used offline for dimension reduction purposes.

In contrast, the CUR decomposition [9] considers selecting subsets from individual links and time instances, enabling us to directly extract underlying spatial and temporal patterns in large road networks. In 2013, Mitrovic et al. explored the use of the CUR matrix decomposition method for compressing and sensing traffic speed data. They compared this method to PCA and showed that the resulting low-dimensional models from CUR are much more easily interpretable. They also demonstrated how CUR can be used for compressed sensing of traffic data [2]. In 2015, Mitrovic et al. employed column-based (CX) low-dimensional models based on their previous research to improve the scalability of compressed sensing and prediction. The researchers broke down the compressed prediction error into different parts and examined how they related to each other, and their numerical findings demonstrate that this approach considerably lowers computational expenses with little effect on prediction accuracy [10]. The benefits of the CUR method are achieved with a trade-off of greater prediction errors as the compression ratio (the ratio of original number of columns to number of columns in a low-dimensional representation) increases. Nevertheless, there is significant room for improving the process of selecting columns based on statistical leverage [9] calculated from singular value decomposition (SVD) [11]. Another issue is that the author only tested a limited range of compression ratios from 2 to 10 [2,10], which is not sufficient to depict the performance of the proposed method. In addition to leverage-based column selection, Couras et al. proposed [12] an algorithm to perform the approximation of the tensors based on the CX decomposition for matrices. Han and Huang [13] proposed a road network compression method to improve the efficiency of data processing based on correlation analysis and CX decomposition, which is then integrated into a deep learning network to predict the short-term traffic flow.

Apart from matrix decomposition, transformation techniques such as discrete Fourier transform (DFT) and discrete wavelet transform (DWT), which are commonly used for compressing signals and images, can be adapted for spatiotemporal data. In the context of road traffic data compression, methods involving DCT and SVD are utilized, leveraging Kronecker product and tensor decomposition [14]. Additionally, by organizing the data in a multidimensional format, algorithms based on DWT are integrated to achieve dimensionality reduction [15]. To harness the benefits provided by signal processing methods from other data types, the field of graph signal processing (GSP) [16] has been developed to conceptualize spatiotemporal data as a two-dimensional graph signal and establishes operations like linear filtering and linear prediction. Furthermore, Chindanur et al. [1] adopted graph Fourier transform (GFT) and achieved less than one percent reconstruction error (RE) on California's Interstate 605 (I-605) freeway data. Although the GFT-based method outperformed other methods, the problems are twofold: (1) it is difficult to interpret in terms of individual detectors, and (2) the model is too complex, possessing a mathematical form which is unlikely to be deployed and understood by practitioners in real world situations.

To summarize, previous studies have focused on the PCA approach, matrix decomposition methods, and signal transformation techniques to compress large datasets and represent the overall network condition using low-dimensional approximations. Three major research gaps were identified, as follows. First, the CUR decomposition method is widely used due to its strong interpretability, whereas the approach to selecting the optimal subset of columns remains to be improved. To this end, the clustering method is explored in this study to enhance the representing performance. Second, most previous studies failed to compare their method of selecting columns with a random sampling approach with equal probabilities. Finally, most studies only evaluated their algorithms in a limited range of compression ratios, but more interesting insights could be discovered if more cases, especially those under a high compression ratio, are tested.

To address these issues, this paper aims to utilize a column-based CUR matrix decomposition technique integrated with a clustering method to represent the original traffic network as a smaller subset of original columns with acceptable errors. The experiment was conducted using radar detector outputs (speed) aggregated at 5 min resolution from Nashville, TN. A group of representative detectors are selected using the proposed weighted average method to establish a relationship with the original network by analyzing the historical data and calculating the relationship matrix offline. In addition, this study incorporates clustering analysis into CUR decomposition inspired by the methodology of computing stock indexes such as the S&P 500 and Nasdaq. Analogous to selecting stocks from different sectors to represent the whole market, a clustering analysis was first conducted, followed by the normal CUR decomposition within each cluster, and results from each cluster would be merged as the final outputs. In this regard, we also reviewed several papers applying clustering algorithms to tackle problems in the traffic domain. Nguyen et al. [17] applied clustering algorithms to obtain labels automatically from the data and presented the results of clustering analysis using both point-based and area-based features, highlighting the superiority of the area-based approach in producing meaningful clusters. Cheng et al. [18] proposed an improved fuzzy c-means clustering method to classify urban traffic states with real world traffic flow data. Chen et al. [19] utilized dynamic time warping (DTW) k-means clustering to classify lane-changing risk profiles into several categories. In the traffic domain, clustering methods are mostly used to categorize time series over different locations.

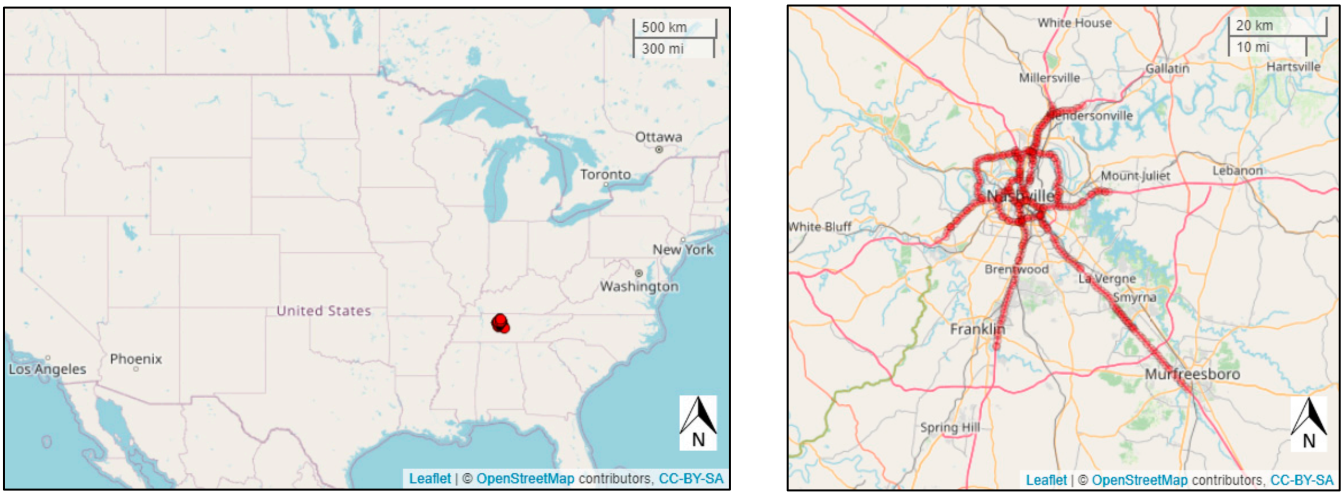
2. Materials and Methods

2.1. Data and Study Area

The radar detector system (RDS) in Nashville consists of 349 detectors covering 564 directional links. The geographic locations of the detectors used in this study and the shape of input data matrix are displayed in Figure 1. Figure 1a displays where Nashville, TN, is in the US, and Figure 1b shows the location of all detectors in Nashville, TN.

The objective of this study is to represent the traffic condition of a massive network, and traffic speed is one of the most widely used and most intuitive characteristics to reflect the traffic condition at a specific location. Thus, traffic speed data were sampled from 9 July 2023, to 29 July 2023. The first two weeks of data were used as training data to select the most representative detectors and learn the relationship matrix, while the remaining one week was used to evaluate the performance of the proposed framework on new data. Figure 2 shows the input data matrix.

The raw data, updated each 30 s, were first aggregated to 5 min to reduce randomness and variability, which will greatly reduce the computational cost as well. We will also implement an online application scenario by applying the relationship matrix learned from 5 min data to 30 s raw data in a near "real-time" manner.



(**a**) (**b**)

Figure 1. (**a**) Location of Nashville, TN, on US map; (**b**) RDS detector locations in Nashville, TN.

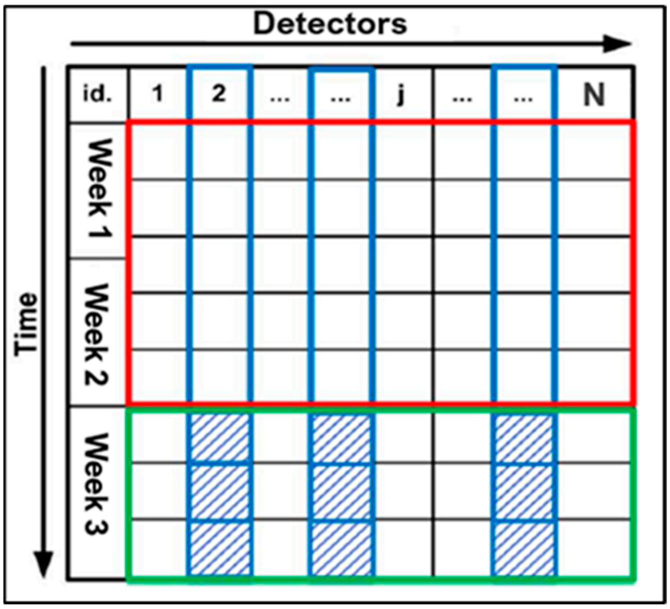


Figure 2. Input data matrix with training period in red rectangle and testing period in the green rectangle; columns in blue rectangles represent the selected group of detectors and blue shaded areas are data entries for testing period at selected detector locations.

2.2. CUR Decomposition

This section will briefly describe the details of different CUR decomposition methods and present the proposed column-based CUR decomposition algorithm integrated with clustering method.

As shown in Figure 3, let $\mathbf{A} \in \mathbb{R}^{t\times n}$ denote the original data matrix with t time instances in row and n detector links in column. Firstly, we define a compression rate as the ratio of number of selected columns to the total number of columns in matrix $\mathbf{A}$ (CR = c/n). The objective of column-based CUR is to find a submatrix $\mathbf{C} \in \mathbb{R}^{t\times c}$ consisting of c columns of $\mathbf{A}$ to create a low-rank approximation $\hat{\mathbf{A}}$ as shown in Equation (1):

$$\mathbf{A} \approx \hat{\mathbf{A}} = \mathbf{CX} = \mathbf{CC}^{+}\mathbf{A}, \tag{1}$$

where

- $\mathbf{C}^{+}$ is Moore–Penrose pseudo-inverse of matrix $\mathbf{C}$ [20].

- $\mathbf{X} = \mathbf{C}^{+}\mathbf{A}$ $\left(\mathbf{X} \in \mathbb{R}^{\mathbf{c}\times\mathbf{n}}\right)$ is the relationship matrix which projects the selected columns back onto all the columns in original data space [21]. For the given matrices **A** and **C**, the relationship matrix is computed as the matrix product of $\mathbf{C}^{+}$ and **A**.

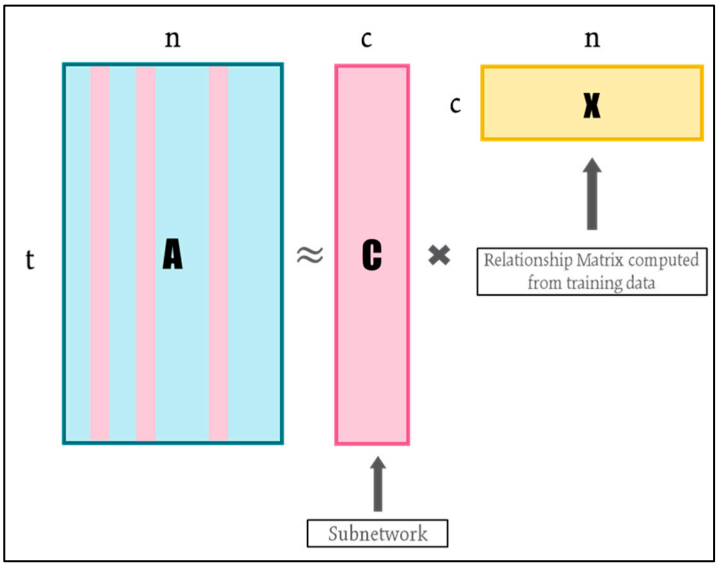


Figure 3. CUR decomposition.

The crucial problem with CUR method is how to select the most representative columns so that the relative error between approximation matrix and original matrix is as small as possible. The goal is to capture the essential information in the data while reducing dimensionality. To evaluate the approximation result, we calculated the percent-root-mean-square distortion (PRD), which is commonly used to assess reconstruction performance [22], provided by the following formula:

$$\mathrm{PRD}\ (\%) = \frac{\|\mathbf{A}-\mathbf{CX}\|_{\mathrm{F}}}{\|\mathbf{A}\|_{\mathrm{F}}}, \tag{2}$$

where

$$\|\mathbf{A}\|_{\mathrm{F}} = \left(\sum_{\mathrm{i}}\sum_{\mathrm{j}} \mathrm{a}_{\mathrm{i,j}}^{2}\right)^{\frac{1}{2}} \tag{3}$$

And it represents the Frobenius norm of matrix **A**. A lesser PRD represents a better compression result.

More specifically, we can break down the column-based CUR method into the following steps:

1. Determine the importance score of each column in the network matrix. Each column in the matrix **A** is assigned a probability score indicating its likelihood of being selected;
2. Create matrix **C** by selecting top c columns from matrix **A** in the descending order of probability scores calculated in the first step;
3. Calculate the relationship matrix $\mathbf{X} = \mathbf{C}^{+}\mathbf{A}$ $\left(\mathbf{X} \in \mathbb{R}^{\mathbf{c}\times\mathbf{n}}\right)$;
4. Assume traffic is stationary, infer the future traffic speed for the entire network using only speed measurements at the selected link, and calculate PRD for the testing data as the performance measurement.

Step 1 is clearly the most essential step in the CUR process. To identify the optimal set of columns for a specified number of columns, one would normally need to evaluate all possible $\binom{n}{c}$ combinations. Nevertheless, employing a brute-force approach entails a computational complexity of $O(n^{c})$ [23]. Given this computational burden, it is usually impractical to assess every potential selection of c columns. To tackle this challenge, two major randomized algorithms have been introduced [21,24] to compute the importance score of each column and then sample columns based on the score:

- L2 norm-based: this method calculates the square of L2 norm for each column and divides it by the sum of squares of all entries of the matrix, expressed as the following equation:

$$p_j^{L2} = \frac{\sum_{i=1}^{t} A(i,\ j)^2}{\sum_{i=1}^{t}\sum_{j=1}^{n} A(i,\ j)^2}, \tag{4}$$

where j = 1, 2, 3, . . ., n. The concept behind this method is to select detector links with high speed. L2 norm-based sampling offers benefits like rapid computation and a better understanding of the column magnitudes. The drawback lies in the neglect of detectors that record a significant portion of low-speed intervals, leading to substantial estimation errors during congestion periods.

- Leverage-based: this method utilizes statistical leverage [9], which measures the contribution of each column to the overall variance of the data, as the importance score for each column. The importance score can be conceptually understood as quantifying the "statistical leverage" or "impact" of a specific column on achieving the most accurate low-rank approximation of the data matrix. By prioritizing the selection of columns that has a disproportionately significant influence over the low-rank approximation, as opposed to L2 norm method that samples columns with higher Euclidean distances, we can ensure that CUR performs nearly as effectively as the best rank-k approximation $\mathbf{A_k}$ in capturing the predominant portion of the spectrum of A [9]. Columns with high leverage scores are often considered important. The underlying concept is to consider detector links with large variations in speed, thus covering various traffic conditions and providing a better capture of the detectors with different traffic states. However, the limitations associated with this approach include the substantial computational cost. The detailed steps are described as follows.

Firstly, singular value decomposition (SVD) is performed on the original data matrix **A** [11] to represent **A** as:

$$\mathbf{A} = \mathbf{U\Sigma V^T}, \tag{5}$$

where $\mathbf{U} \in \mathbb{R}^{t\times t}$ and $\mathbf{V} \in \mathbb{R}^{n\times n}$ are unitary matrices, and the columns of **U** and **V** are left and right singular vectors of **A**, respectively. Matrix $\mathbf{\Sigma} \in \mathbb{R}^{t\times n}$ is a rectangular diagonal matrix with non-negative diagonal entries, known as the singular values of matrix **A**. Then, the best rank-k approximation can be obtained by keeping top k columns of **U**, **Σ**, and **V** so that the explained variance is at least 80%:

$$\mathbf{A}_{t\times n}= \mathbf{U}_{t\times k}\mathbf{\Sigma}_{k\times k}\mathbf{V}^{\mathbf{T}}_{k\times n}. \tag{6}$$

Then, the leverage score of column j can be calculated as:

$$p_j^{SVD} = \frac{1}{k}\sum_{\xi=1}^{k}\left(\mathbf{v}_j^{\xi}\right)^2, \tag{7}$$

where $\mathbf{v}_j^{\xi}$ is the j^{th} coordinate of ξ^{th} right singular vector. It is obvious that the sum of p_j is equal to one because matrix **V** is unitary.

Based on the two previous options, we propose a weighted method considering both L2 norm and statistical leverage, expressed as Equation (8):

$$p_j^{W} = w\cdot p_j^{L2} + (1-w)\cdot p_j^{SVD}, \tag{8}$$

where w is a weight parameter, and w is set to be 0.5 in this study to assign equal weights to column magnitude and column variation.

In addition to the three options, a random sampling method to select c columns with equal probability for each column is also implemented. This process was repeated five times, and average performance was recorded. Prior research failed to consider the comparison with the simplest random sampling method, thus making the performance less persuasive.

2.3. CUR Integrated with Clustering

The primary objective of this study is to explore the effects of incorporating clustering method into CUR sampling process. The reason for considering clustering method is to investigate whether homogenous subnetworks can enhance the overall performance of the network representation. The idea of integrating clustering analysis into CUR method is inspired by the methodology of devising a composite index for stock market such as Nasdaq-100.

Stocks from different sectors will go through a rigorous screening process and then be selected to represent the whole market using a weighted average index. Similarly, to select most representative traffic detectors in a massive road network, the first step is to filter out columns with missing rate greater than 5% and to impute the remaining columns using data from adjacent time intervals (linear interpolation). Then, we can classify the detectors into different clusters based on their speed during the training period, followed by the normal CUR decomposition process within each individual cluster given a fixed compression rate (CR). Finally, the selected columns and corresponding relationship matrix will be merged as the final CUR outputs. The entire column selection process is displayed in Figure 4.

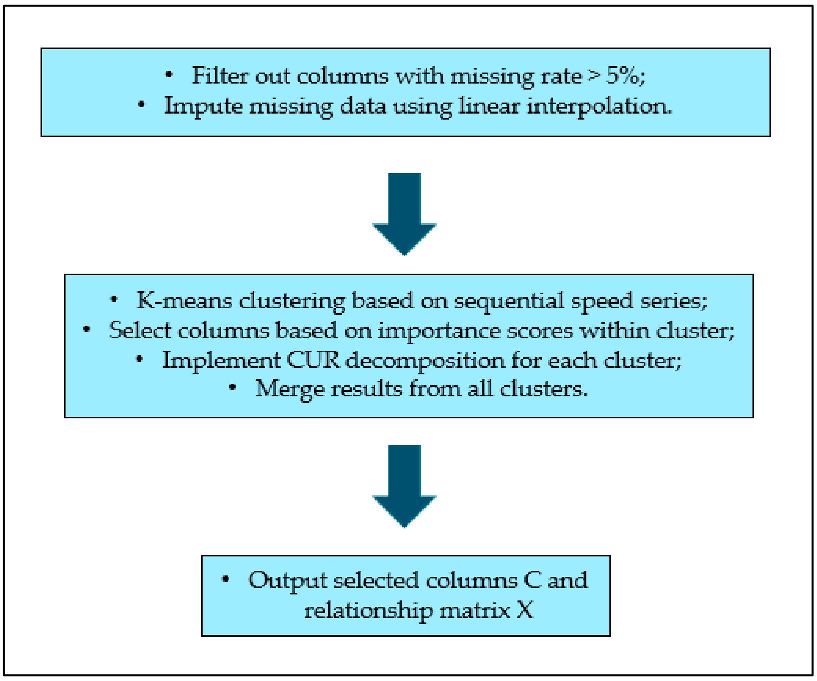

• Filter out columns with missing rate > 5%;
• Impute missing data using linear interpolation.
• K-means clustering based on sequential speed series;
• Select columns based on importance scores within cluster;
• Implement CUR decomposition for each cluster;
• Merge results from all clusters.
• Output selected columns C and relationship matrix X

Figure 4. Column selection and decomposition process.

Mitrovic et al. has taken into account the clustering method to compress large-scale traffic data [10]; however, they simply clustered the network according to road category instead of using a real clustering method based on the speed condition on the road. It is obvious that roads in the same category do not necessarily have similar traffic conditions.

In this study, k-means clustering was adopted due to its simplicity, computational efficiency, and interpretability. A detailed description of k-means clustering is provided as follows.

In k-means, clusters are defined so that the total intra-cluster variation (known as total within-cluster variation) is minimized. Within-cluster variation is defined as: $\sum||x - \boldsymbol{\mu}_{\boldsymbol{i}}||^2$, where we only consider x of a given cluster $\boldsymbol{S}_{\boldsymbol{i}}$, and $\boldsymbol{\mu}_{\boldsymbol{i}}$ is the mean (centroid) of points in cluster $\boldsymbol{S}_{\boldsymbol{i}}$. Then, total within-cluster variation is to perform the previous calculation for each cluster and take the sum of all clusters.

There are five overall steps in naïve k-means method:

- Step 1: specify k (number of clusters).
- Step 2: randomly select k instances from the data as the initial cluster centroids.
- Step 3: assign each instance to its closest centroid based on Euclidean distance.
- Step 4: for each of the k clusters, recompute the centroid by calculating the new mean of all the instances in the cluster.
- Step 5: repeat step 3 and 4 until centroids converge or the max number of iterations is reached.

3. Results

This section provides a detailed description of the results of all experiments. The first part of the study is representing the massive network using the CUR decomposition method based on the training data and inferring future conditions on the testing data. In addition, the second part is a new application scenario of indexing the network condition using the outputs from the CUR decomposition.

3.1. Performance of the Proposed Weighted Average Method

Figure 5 shows the performance of the proposed weighted method compared with the L2 norm, SVD-based leverage, and random sampling methods. The red line in the graph indicates that the weighted method outperforms all other three options when the compression rate (CR) is lesser than 32 ($2^5 = 32$). The proposed weighted method resulted in lower PRD than other methods when the CR does not exceed 32, whereas the drawback of the weighted sampling method is relatively longer computational time. When the CR surpasses 32, the proposed weighted method is only better than the SVD-based leverage method, partly because the presetting default value of the weight (w) is 0.5, which provides too much weight for SVD-based leverage. Random sampling is even better than the L2 norm option when the CR is less than 64, which might be due to randomness and fluctuation. It is worth noting that SVD-based leverage and proposed weighted sampling perform best with a CR less than 8, whereas the error for the SVD-based leverage option increases dramatically as the CR becomes greater than 32.

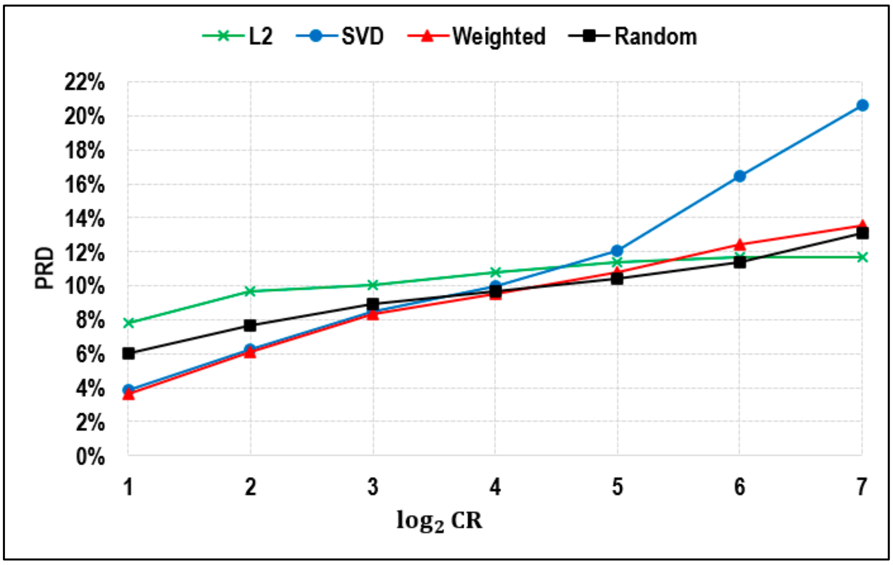


Figure 5. Performance of the proposed method without clustering.

Table 1 presents the summary statistics of the performance of the proposed method (weighted) compared to other three models (L2 norm, SVD-based leverage, and random sampling). It should be noted that random sampling is carried out to show the minimal time needed to generate the decomposition results, since we assume that the time consumed by column selection could be ignored under a random sampling scenario. However, random sampling is impractical because the performance fluctuates, and the selected columns are not fixed each time the sample is obtained. That is also why we repeated the process five times and used the average performance. Among the three methods, the L2 norm-based method is the most computation-friendly, and the proposed weighted method uses more computational resources than the other two methods. Overall, this is a trade-off between accuracy and computing cost.

Table 1. Comparison of the performance of different methods without clustering.

Compression Rate (CR)	**PRD (%)**				**Computing Time (Milliseconds)**			
	L2	**SVD**	**Weighted**	**Random**	**L2**	**SVD**	**Weighted**	**Random**
2	7.8%	3.8%	3.6%	5.8%	793.1	982.5	1075.5	645.9
4	9.7%	6.3%	6.1%	7.7%	692.0	870.9	964.9	562.6
8	10.1%	8.4%	8.3%	8.9%	705.0	899.3	939.8	534.3
16	10.8%	10.0%	9.5%	9.6%	648.2	860.0	973.0	512.7
32	11.4%	12.0%	10.8%	10.5%	691.7	915.1	939.8	512.0
64	11.7%	16.4%	12.5%	11.4%	631.2	835.1	940.4	567.0
128	11.7%	20.6%	13.5%	12.4%	624.3	882.4	905.8	522.4

Taking CR = 16 with the proposed weighted average method as an example, Figure 6 depicts the absolute error of the original matrix and inferred matrix for the testing period (23 July 2023, to 29 July 2023). The horizontal axis represents the dates and the vertical axis is the detectors. There are 31 links selected, and the PRD = 9.5%. The absolute error is less than 10 mph for 94% of time and detectors. Some recurring patterns can be found, probably due to the recurring peak-hour congestion at some detector links.

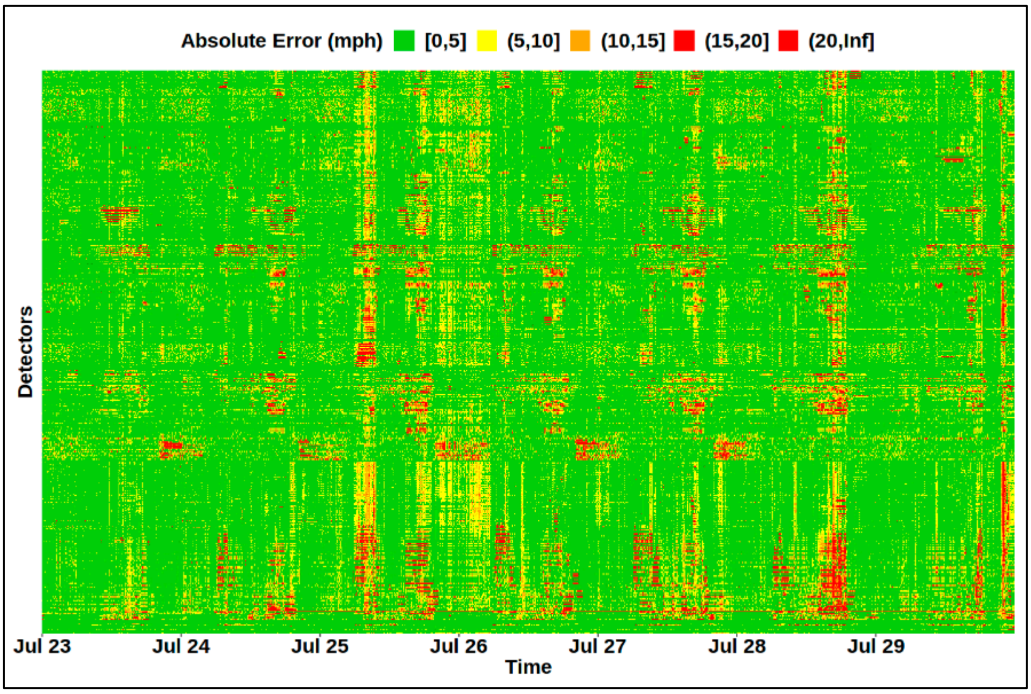
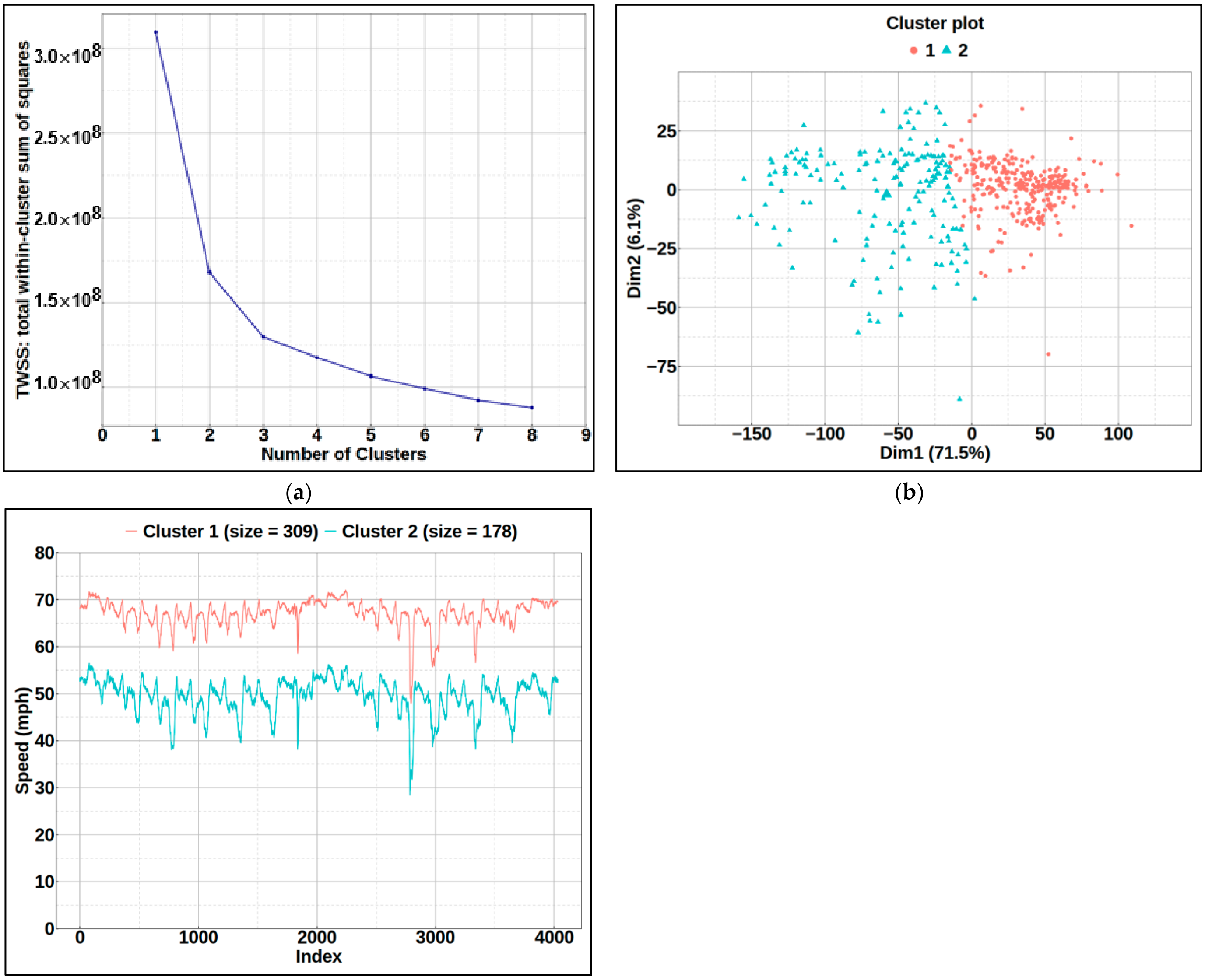


Figure 6. Performance of the proposed weighted method when CR = 16.

3.2. Effects of Clustering Method

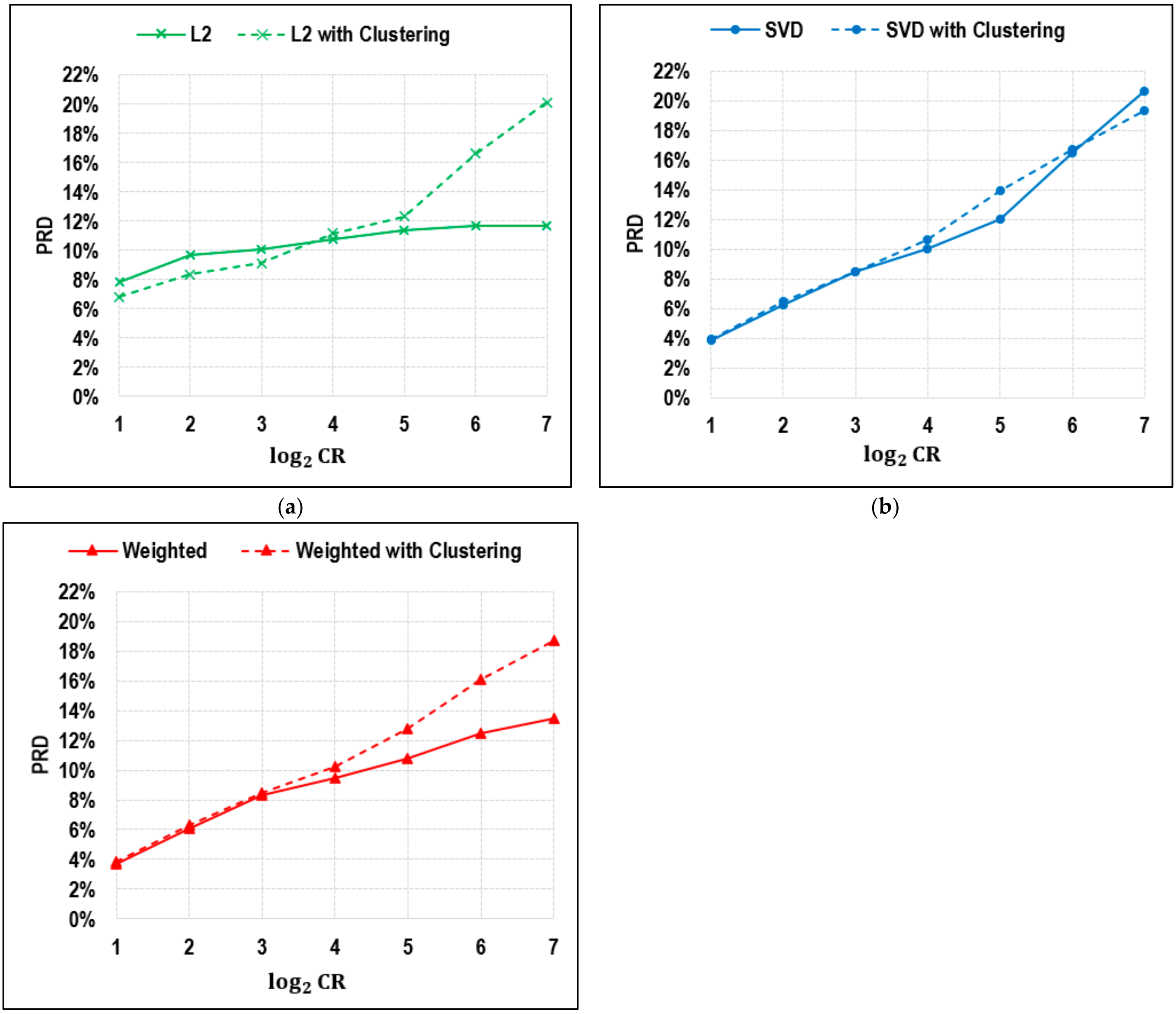
Firstly, K-means clustering was implemented on the original data to classify detector links into different groups based on their speed measurements during the training period. Figure 7a shows the results of the elbow method which implied that two is more likely to be the optimal number of clusters, and Figure 7b displays the cluster results for all observations projected onto the dimension of two principal components. As can be seen, there are overlaps between the two clusters. Figure 7c displays the centroids of two clusters, respectively. The red line is the center of the first cluster, with speed around 70 mph, and the speed of the second cluster's centroids, shown in blue, fluctuates around 50 mph.



(**a**) (**b**) (**c**)

Figure 7. Clustering results: (**a**) elbow method showing total with-cluster sum of squares versus number of clusters; (**b**) clustering results showing two principal components of observations; (**c**) cluster centroids versus time index.

After clustering, the normal CUR process, with three different options, was conducted to explore the effects of incorporating clustering into CUR decomposition. Figure 8 displays the comparison of the performance of the three options with and without clustering method. It was found that clustering almost failed to improve the performance for the SVD-based leverage and weighted average option, while the L2 norm-based option saw a minor enhancement when the CR was less than 16. This finding is somewhat intuitive and contradictory to Mitrovic's work [10], partly because Mitrovic only tested his method on CRs from 2 to 10. The reason why clustering failed to contribute will be discussed in the discussion section.



(**a**)

(**b**)

(**c**)

Figure 8. Comparison of three methods with and without clustering: (**a**) L2 norm-based importance; (**b**) SVD-based leverage importance; (**c**) weighted average of L2 norm and leverage.

4. Discussion

As shown in Figure 5 and Table 1, the proposed weighted approach demonstrated lower PRD compared to alternative methods when the CR remained below 32. However, the downside of employing the weighted sampling method is the comparatively longer computational time. When the CR threshold surpasses 32, the proposed weighted method only outperforms the SVD-based leverage method, partially because the default weight value (w) is set at 0.5, assigning too much weight to the SVD-based leverage approach. A possible future direction could be tuning the hyperparameter w to achieve better performance over the L2 norm-based and SVD-based leverage methods on training data, followed by an evaluation of the method on the testing period. The PRD of the L2 norm-based method remains relatively stable (between 8% and 12%) as the CR increases from 2 to 128, while the SVD-based leverage method yields good performance with a CR lesser than 16, but its error rises very quickly as the CR becomes greater than 16.

It is worth noting that the performance of the random sampling method is even better than that of the other three methods when the CR changes between 16 and 64. Previous researchers either failed to consider testing their method within this range or forgot to compare with the performance of the random sampling method.

Figure 6 displays the heatmap of the absolute error between the original and inferred test data. A compression rate of 16 yields a PRD of 9.5%, and the absolute error is within 10 mph for 94% of the time and detectors. Additionally, the largest error always occurred recurrently during the peak hour at some specific detectors, which implies that the CUR decomposition method does not perform well during congestion periods. There also exist some non-recurring large errors, which might be due to the occurrence of some atypical incidents such as severe weather, special events, and scheduled road work.

Another interesting finding is that clustering almost has no room for improvement for SVD-based leverage and the proposed weighted average method. As for the L2 norm-based method, clustering helps reduce the error to as low as 1% to 2%. Integrated with the clustering method, all three methods even demonstrated an increase in error when the CR was greater than 16, except for the SVD method with CR = 128.

There might be various reasons accounting for this phenomenon. First, if the data do not exhibit clear clusters or if the clusters are not well defined, clustering algorithms may not yield meaningful or useful groupings. CUR decomposition relies on selecting representative columns and rows, and if the data do not naturally cluster, selecting clusters may not provide an advantage. Second, determining the optimal number of clusters (k) is challenging. Choosing an incorrect value for k can lead to poor cluster quality and, consequently, suboptimal CUR selection. Another reason could be the choice of the clustering algorithm. If the clustering algorithm is not well-suited to the data or the specific goals of CUR decomposition, it may not lead to improvements. Since we are clustering time series in the context of traffic speed, a dynamic time warping distance could be considered as the similarity metric to perform the clustering analysis. In addition, k-means clustering is simple and basic, so it might be beneficial to try other advanced clustering algorithms, such as hierarchical clustering or DBSCAN, to yield better clustering results and compare the CUR performance with different clustering methods. This study opens the door to various intriguing possibilities for further investigation of how to improve clustering results.

The detectors used in this study are all from interstates without disruption by signal timing. Future work could be carried out to include different types of road networks to analyze their traffic speed patterns and investigate how clustering will impact the CUR decomposition performance on different types of road networks. As has been discussed in Mitrovic's work [10], clustering by different road categories improved the inferring performance for the entire network when the CR was between 2 and 10. Our future work could focus on analyzing the speed patterns on different road types and clustering based on speed profile and road type together.

Due to the data availability issues, this study only focused on a case study in Nashville, TN, and all detectors are located on urban interstates. To comprehensively evaluate the practical applicability, additional research could be carried out to investigate the generalizability of the proposed framework to other cities or regions and for other time periods. It is essential to extend this research to different locations and time frames to assess its spatial and temporal transferability.

In this study, external factors, such as adverse weather, special events and scheduled road work, were not included in the framework, which might impact the representing and inferring performance when there happened to be many non-recurring incidents during the training and testing period. One area for future work involves exploring the impact of external factors in greater depth to gain a more comprehensive understanding of its applications.

Another interesting avenue for future research is to explore the possibility of incorporating other traffic data, such as vehicle count and vehicular types, into the methodology for a more comprehensive representation of the traffic condition at specific detectors, thus resulting in a better representation of the entire network. This involves methods like column-based decomposition for multidimensional tensors, proposed by Couras et al. [12] in 2019.

5. Conclusions

In conclusion, this research addresses important research gaps in the field of massive road network compression and representation. The utilization of a column-based CUR matrix decomposition technique with a column selection method based on weighted average of L2 norm importance and SVD-based leverage provides a novel approach to efficiently represent the original traffic network with a reduced subset of columns while maintaining acceptable error levels. This approach offers enhanced interpretability and a more refined column selection process compared to previous methods. Moreover, by conducting a comprehensive comparison with random sampling, this study contributes to a better understanding of the advantages of the proposed approach.

Furthermore, the extension of the evaluation to a wider range of compression ratios adds valuable insights into the scalability and performance of the method. The application of these techniques to radar detector data from Nashville, TN demonstrates their practical utility in real-world scenarios. By drawing inspiration from stock market index computation methodology, the integration of clustering analysis into CUR decomposition is investigated with the anticipation of performance enhancements. Clustering was found to enhance performance, but, notably, this improvement was confined to a limited range of compression ratios. Interestingly, once the compression ratio exceeded a certain threshold, the application of clustering not only ceased to improve performance but also adversely affected it.

Overall, this research not only contributes to the advancement of data compression and network representation but also highlights the potential of combining CUR decomposition with clustering for various applications in the field of big data analytics and network science.

Author Contributions: Conceptualization, H.Z., Y.G. and L.D.H.; methodology, H.Z., Y.G. and L.D.H.; software, H.Z.; validation, H.Z., Y.G. and L.D.H.; formal analysis, H.Z.; investigation, H.Z.; resources, H.Z., Y.G. and L.D.H.; data curation, H.Z. and Y.G.; writing—original draft preparation, H.Z.; writing—review and editing, H.Z., Y.G. and L.D.H.; visualization, H.Z.; supervision, L.D.H.; project administration, L.D.H.; funding acquisition, L.D.H. All authors have read and agreed to the published version of the manuscript.

Funding: This research received funding from the Tennessee I-24 Smart Corridor evaluation project (RES2023-18) which is supported by the Tennessee Department of Transportation.

Data Availability Statement: Restrictions apply to the availability of these data. Data were obtained from the Tennessee Department of Transportation and are available from the authors with the permission of the Tennessee Department of Transportation.

Acknowledgments: We acknowledge the financial and data support from the Tennessee Department of Transportation.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

Abbreviation	Definition
CR	Compression rate
CX	Column-based
DBSCAN	Density-based spatial clustering of applications with noise
DCT	Discrete cosine transform
DFT	Discrete Fourier transform
DTW	Dynamic time warping
DWT	Discrete wavelet transform
GFT	Graph Fourier transform
GSP	Graph signal processing
ITS	Intelligent transportation system
OD	Origin–destination
PCA	Principal component analysis
PRD	Percent-root-mean-square distortion
RDS	Radar detector system
RE	Reconstruction error
SVD	Singular value decomposition
TN	Tennessee

Nomenclature

Symbol	Definition	Unit
$\mathrm{a}_{\mathrm{i,j}}$	The i^{th} row and j^{th} column of matrix **A**	-
$\mathbf{A}$	The original data matrix	-
$\hat{\mathbf{A}}$	The low-rank approximation of original matrix	-
$\mathbf{A}_{\mathbf{k}}$	The best rank-k approximation of matrix **A**	-
$\mathbf{C}$	The selected submatrix	-
$\mathbf{C}^{+}$	The Moore–Penrose pseudo-inverse of matrix **C**	-
$\mathrm{p}_{\mathrm{j}}^{\mathrm{L2}}$	The L2 norm-based importance score for column j	-
$\mathrm{p}_{\mathrm{j}}^{\mathrm{SVD}}$	The SVD leverage-based importance score for column j	-
$\mathrm{p}_{\mathrm{j}}^{\mathrm{W}}$	The proposed weighted importance score for column j	-
$\boldsymbol{S_i}$	The i^{th} cluster	-
$\mathbf{U}$	The unitary matrix with columns being left singular vectors of **A**	-
$\mathbf{V}$	The unitary matrix with columns being right singular vectors of **A**	-
$\mathbf{v}_{\mathbf{j}}^{\xi}$	The j^{th} coordinate of ξ^{th} right singular vector	-
$\mathbf{X}$	The relationship matrix	-
x	The data point	-
$\boldsymbol{\Sigma}$	The rectangular diagonal matrix with non-negative diagonal entries being the singular values of matrix **A**	-
$\boldsymbol{\mu_i}$	The mean (centroid) of points in cluster $\boldsymbol{S_i}$	-
$\lVert.\rVert_{\mathrm{F}}$	The Frobenius norm of a matrix	-

References

1. Chindanur, N.B.; Sure, P. Low-dimensional models for traffic data processing using graph Fourier transform. *Comput. Sci. Eng.* **2018**, *20*, 24–37. [CrossRef]
2. Mitrovic, N.; Asif, M.T.; Rasheed, U.; Dauwels, J.; Jaillet, P. CUR decomposition for compression and compressed sensing of large-scale traffic data. In Proceedings of the 16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013), The Hague, The Netherlands, 6–9 October 2013; pp. 1475–1480.
3. Wold, S.; Esbensen, K.; Geladi, P. Principal component analysis. *Chemom. Intell. Lab. Syst.* **1987**, *2*, 37–52. [CrossRef]
4. Djukic, T.; Flötteröd, G.; Van Lint, H.; Hoogendoorn, S. Efficient real time OD matrix estimation based on Principal Component Analysis. In Proceedings of the 2012 15th International IEEE Conference on Intelligent Transportation Systems, Anchorage, AK, USA, 16–19 September 2012; pp. 115–121.
5. Djukic, T.; Van Lint, J.; Hoogendoorn, S. Application of principal component analysis to predict dynamic origin–destination matrices. *Transp. Res. Rec.* **2012**, *2283*, 81–89. [CrossRef]
6. Jin, X.; Zhang, Y.; Yao, D. Simultaneously prediction of network traffic flow based on PCA-SVR. In Proceedings of the 4th International Symposium on Neural Networks, ISNN 2007, Nanjing, China, 3–7 June 2007; pp. 1022–1031.

7. Li, Q.; Jianming, H.; Yi, Z. A flow volumes data compression approach for traffic network based on principal component analysis. In Proceedings of the 2007 IEEE Intelligent Transportation Systems Conference, Bellevue, WA, USA, 30 September–3 October 2007; pp. 125–130.
8. Asif, M.T.; Kannan, S.; Dauwels, J.; Jaillet, P. Data compression techniques for urban traffic data. In Proceedings of the 2013 IEEE Symposium on Computational Intelligence in Vehicles and Transportation Systems (CIVTS), Singapore, 16–19 April 2013; pp. 44–49.
9. Mahoney, M.W.; Drineas, P. CUR matrix decompositions for improved data analysis. *Proc. Natl. Acad. Sci. USA* **2009**, *106*, 697–702. [CrossRef] [PubMed]
10. Mitrovic, N.; Asif, M.T.; Dauwels, J.; Jaillet, P. Low-Dimensional Models for Compressed Sensing and Prediction of Large-Scale Traffic Data. *IEEE Trans. Intell. Transp. Syst.* **2015**, *16*, 2949–2954. [CrossRef]
11. Golub, G.H.; Reinsch, C. Singular value decomposition and least squares solutions. In *Handbook for Automatic Computation: Volume II: Linear Algebra*; Springer: Berlin/Heidelberg, Germany, 1971; pp. 134–151.
12. Couras, M.F.; de Pinho, P.H.; Favier, G.; da Costa, J.P.; Zarzoso, V.; de Almeida, A.L. Multidimensional CX Decomposition of Tensors. In Proceedings of the 2019 Workshop on Communication Networks and Power Systems (WCNPS), Brasilia, Brazil, 3–4 October 2019; pp. 1–4.
13. Han, L.; Huang, Y.S. Short-term traffic flow prediction of road network based on deep learning. *IET Intell. Transp. Syst.* **2020**, *14*, 495–503. [CrossRef]
14. Feng, S.; Zhang, Y.; Li, L. A comparison study for traffic flow data compression. In Proceedings of the 2016 12th World Congress on Intelligent Control and Automation (WCICA), Guilin, China, 12–15 June 2016; pp. 977–982.
15. Agarwal, S.; Regentova, E.E.; Kachroo, P.; Verma, H. Multidimensional compression of ITS data using wavelet-based compression techniques. *IEEE Trans. Intell. Transp. Syst.* **2016**, *18*, 1907–1917. [CrossRef]
16. Sandryhaila, A.; Moura, J.M. Big data analysis with signal processing on graphs: Representation and processing of massive data sets with irregular structure. *IEEE Signal Process. Mag.* **2014**, *31*, 80–90. [CrossRef]
17. Nguyen, T.T.; Krishnakumari, P.; Calvert, S.C.; Vu, H.L.; Van Lint, H. Feature extraction and clustering analysis of highway congestion. *Transp. Res. Part C Emerg. Technol.* **2019**, *100*, 238–258. [CrossRef]
18. Cheng, Z.; Wang, W.; Lu, J.; Xing, X. Classifying the traffic state of urban expressways: A machine-learning approach. *Transp. Res. Part A Policy Pract.* **2020**, *137*, 411–428. [CrossRef]
19. Chen, T.; Shi, X.; Wong, Y.D. A lane-changing risk profile analysis method based on time-series clustering. *Phys. A Stat. Mech. Its Appl.* **2021**, *565*, 125567. [CrossRef]
20. Golub, G.; Kahan, W. Calculating the singular values and pseudo-inverse of a matrix. *J. Soc. Ind. Appl. Math. Ser. B Numer. Anal.* **1965**, *2*, 205–224. [CrossRef]
21. Drineas, P.; Mahoney, M.W.; Muthukrishnan, S. Relative-Error CUR Matrix Decompositions. *SIAM J. Matrix Anal. Appl.* **2008**, *30*, 844–881. [CrossRef]
22. Dauwels, J.; Srinivasan, K.; Ramasubba, R.M.; Cichocki, A. Multi-channel EEG compression based on matrix and tensor decompositions. In Proceedings of the 2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Prague, Czech Republic, 22–27 May 2011; pp. 629–632.
23. Boutsidis, C.; Mahoney, M.W.; Drineas, P. An improved approximation algorithm for the column subset selection problem. In Proceedings of the Twentieth Annual ACM-SIAM Symposium on Discrete Algorithms, New York, NY, USA, 4–6 January 2009; pp. 968–977.
24. Sun, J.; Xie, Y.; Zhang, H.; Faloutsos, C. Less is more: Compact matrix decomposition for large sparse graphs. In Proceedings of the 2007 SIAM International Conference on Data Mining, Minneapolis, MN, USA, 26–28 April 2007; pp. 366–377.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.