

Online Competitive Algorithms and Reinforcement Learning for Traffic Management

August 2026

A Research Report from the Pacific Southwest Region University Transportation Center

Yazmin Elizabet Martinez, University of Nevada, Las Vegas

Wolfgang Bein, University of Nevada, Las Vegas



TECHNICAL REPORT DOCUMENTATION PAGE

1. Report No. PSR-24-47	2. Government Accession No. N/A	3. Recipient's Catalog No. N/A	
4. Title and Subtitle Online Competitive Algorithms and Reinforcement Learning for Traffic Management		5. Report Date 08-18-2026	
		6. Performing Organization Code N/A	
7. Author(s) Yazmin Elizabet Martinez, ORCID 0000-0003-1738-383X Wolfgang Bein, 0000-0001-7159-9880		8. Performing Organization Report No. TBD	
9. Performing Organization Name and Address METRANS Transportation Center University of Southern California University Park Campus, RGL 216 Los Angeles, CA 90089-0626		10. Work Unit No. N/A	
		11. Contract or Grant No. USC sub# SCON-00005219 USDOT # 69A3552348309	
12. Sponsoring Agency Name and Address U.S. Department of Transportation Office of the Assistant Secretary for Research and Technology 1200 New Jersey Avenue, SE, Washington, DC 20590		13. Type of Report and Period Covered Final report (Oct. 1, 2024 – Jul. 31, 2026)	
		14. Sponsoring Agency Code USDOT OST-R	
15. Supplementary Notes Research outputs can be found on this webpage: https://yazmin.cs.unlv.edu/ https://doi.org/10.25554/y5xx-8x81			
16. Abstract Traffic signals with fixed-time signal timing require retiming often because of traffic volume changes. Traffic signals with a fixed-time timings are configured with historical traffic data observations. The traffic signal retiming projects can take several years due to the major undertaking by transportation agencies. This seldom signal retiming can lead to higher vehicle delays. Although vehicle detection is a common technology, many intersections in urban areas are not setup with detection sensors because of their added installation and upkeep costs. In the Las Vegas Valley, there are fixed time and actuated (sensor-activated) traffic signals that we analyze for traffic congestion and to identify suboptimal signals. We built geospatial maps to perform exploratory visual analysis and to generate a list of the most monthly recurrent congested intersections in 2025. We also built an Autoencoder to find anomalous signals that are most likely experiencing congestion and identify important features that influence that congestion. Lastly, we present preliminary results from a traffic signal simulation that reveals that smaller cycle times only worsen average vehicle wait times by a small percentage from the optimal cycle length. The impact of our research has the potential to improve delays for motorist and to aid transportation agencies with signal retiming efforts.			
17. Key Words Road Traffic Congestion Prediction, MLP, Deep Learning, Geospatial Visualization, Machine Learning, Autoencoder, Feature Selection, Dimensionality Reduction, Anomaly Detection, and Batch Scheduling		18. Distribution Statement No restrictions.	
19. Security Classif. (of this report) Unclassified	20. Security Classif. (of this page) Unclassified	21. No. of Pages 36	22. Price N/A

Form DOT F 1700.7 (8-72)

Reproduction of completed page authorized

Contents

Acknowledgements	6
Abstract	7
Executive Summary	8
Introduction	9
Literature Review	10
Deep Learning and Autoencoders	10
Methods/Methodology	12
Process Datasets to Generate Maps and Working Dataset for Autoencoder	15
Process Working Dataset with an Autoencoder	18
Batch Scheduling – Traffic Signal Simulation	20
Results and Future Work	21
Anomaly Detection	21
Conclusion	22
References	23
Data Management Plan	25
Appendix A	26
Appendix B	27
Appendix C	28
Appendix D	29
Appendix E	30
Appendix F	31
Appendix G	32
Appendix H	33
Appendix I	34
Appendix J	35
Appendix K	36
Appendix L	37

About the Pacific Southwest Region University Transportation Center

The Pacific Southwest Region University Transportation Center (UTC) is the Region 9 University Transportation Center funded under the US Department of Transportation's University Transportation Centers Program. Established in 2016, the Pacific Southwest Region UTC (PSR) is led by the University of Southern California and includes seven partners: Long Beach State University; University of California, Davis; University of California, Irvine; University of California, Los Angeles; University of Hawaii; Northern Arizona University; Pima Community College.

The Pacific Southwest Region UTC conducts an integrated, multidisciplinary program of research, education and technology transfer aimed at *improving the mobility of people and goods throughout the region*. Our program is organized around four themes: 1) technology to address transportation problems and improve mobility; 2) improving mobility for vulnerable populations; 3) Improving resilience and protecting the environment; and 4) managing mobility in high growth areas.

U.S. Department of Transportation (USDOT) Disclaimer

The contents of this report reflect the views of the authors, who are responsible for the facts and the accuracy of the information presented herein. This document is disseminated in the interest of information exchange. The report is funded, partially or entirely, by a grant from the U.S. Department of Transportation's University Transportation Centers Program. However, the U.S. Government assumes no liability for the contents or use thereof.

Disclosure

Principal Investigator Wolfgang Bein conducted this research titled, "Online Competitive Algorithms and Reinforcement Learning for Traffic Management" at the Department of Computer Science, Howard R. Hughes College of Engineering, University of Nevada, Las Vegas. The research took place from October 1st, 2024 to July 31st, 2026 and was funded by a grant from the U.S. Department of Transportation in the amount of \$57,175. The research was conducted as part of the Pacific Southwest Region University Transportation Center research program.

Acknowledgements

The Center of Information Technology and Algorithms acknowledges the feedback from University of Nevada, Las Vegas Transportation Research Center, in particular professor Shashi Nambisan.

Abstract

Traffic signals with fixed-time signal timing require retiming often because of traffic volume changes. Traffic signals with a fixed-time timings are configured with historical traffic data observations. The traffic signal retiming projects can take several years due to the major undertaking by transportation agencies. This infrequent signal retiming can lead to higher vehicle delays. Although vehicle detection is a common technology, many intersections in urban areas are not setup with detection sensors because of their added installation and upkeep costs. In the Las Vegas Valley, there are fixed time and actuated (sensor-activated) traffic signals that we analyze for traffic congestion and to identify suboptimal signals. We built geospatial maps to perform exploratory visual analysis and to generate a list of intersections with the most recurrent monthly congestion in 2025. We also built an Autoencoder to find anomalous signals that are most likely experiencing congestion and identify important features that influence that congestion. Lastly, we present preliminary results from a traffic signal simulation that reveals that smaller cycle times only worsen average vehicle wait times by a small percentage from the optimal cycle length. The impact of our research has the potential to improve delays for motorist and to aid transportation agencies with signal retiming efforts.

Executive Summary

The management of traffic have a significant impact on delays experienced by motorists. Traffic signal cycle times in urban centers are generally long, leading to inefficiencies in traffic flow and increased environmental impact. Traffic flow coordination relies on partially outdated technology that is not based in contemporary algorithmic models (Dion et al., 2004). As an example, a number of papers show that the well-known Webster formula, now almost 70 years old and widely used to estimate the minimum delay optimal traffic signal cycle length, overestimates the cycle length for high degrees of traffic saturation – a degree of saturation now common in most urban areas. Though transportation has been studied from many different angles, the algorithmic focus is still recent and new, see survey (Karami and Kashef, 2020) and survey (Qadri, 2020) for traffic regulation at intersections.

In the Las Vegas Valley, the Regional Transportation Commission of Southern Nevada (RTC) follows an approach based in traditional transportation science, which tries to synchronize lights along corridors, and which use long cycle length, i.e., when a car encounters a red light, the wait can be substantial. We propose to use computer science approaches, namely deep learning and online competitive analysis to substantially improve the state-of-the-art. In fact, as multi-modal traffic systems, ride-share systems and autonomous vehicles are becoming more prevalent vehicle traffic becomes more of a distributed system resembling internet traffic. With the use of deep learning techniques, a system is envisioned that can analyze the large trove of data now available. The system will mine for systemic inefficiencies and then give algorithmic solutions to eliminate such inefficiencies. The automatic collection of data by the RTC has allowed us to perform our study using an open-source dataset on the traffic signal metrics in the Las Vegas Valley (RTC FAST, n.d.).

When vehicles accumulate at an intersection, this sequence of vehicles forms a platoon. Vehicles in a platoon all experience the same stopped delay and are subject to deceleration and acceleration delay at that intersection. The situation is like an area of computer science research called “batch scheduling.” We propose to study the problem of traffic delays at traffic signals/intersection in the framework of batching. Batching problems, both offline as well as online have been studied extensively, including contributions to domain made by Professor Wolfgang Bein, a researcher for this study.

We completed a multi-step solution to our investigation. The first step was to create a visualization map to explore traffic congestion. The second step was to create an Autoencoder, a machine learning approach, to automate the process of finding traffic signals that can potentially be retimed. Lastly, we ran multiple traffic signal simulations and found that reducing signal cycle times minimally affects the average vehicle wait time. The simulation makes use of the competitive ratio theory, a theory in computer science.

The final project data output is in the form of twelve geospatial maps that are publicly available, refer to the data management section in this document and Appendices A-L.

Introduction

Road traffic congestion affects our daily lives whether we drive or take bus transit. However, the complexity of road traffic congestion and its causes are active areas of research due to a need to find plausible solutions to reduce vehicle delay. Given the abundance of traffic signal performance data collected by transportation agencies, we, researchers, have an opportunity to analyze traffic signals to identify improvements for traffic signal cycle length and green times. In this report, we detail three efforts to improve vehicle traffic delays.

The RTC has publicized that making traffic signal changes is a major undertaking. In a small case study from our previous work (Martinez and Bein, 2025b), we noted the following re-timing information we found with regards to the Las Vegas Valley that shows the multiyear plan and effort to retime traffic signals. “The RTC published a Traffic Signal Re-Timing Activities report (Regional Transportation Commission of Southern Nevada, 2024b) from June 2024 that lists the progress for a signal retiming project, along with a map of the signals to be re-timed and the arterial performance metrics that the RTC staff monitors. This most recent RTC retiming project was publicized (Regional Transportation Commission of Southern Nevada, 2024a) and is estimated to take about two years to complete. The re-timing of the signals in the Las Vegas valley began in 2022 and is estimated to be completed in the fall 2024. The timing changes to the signals were done by region of southwest, southeast, northwest, and northeast part of the valley.” Moreover, the frequency of the traffic signals being retimed or optimized can take years due to the major undertaking by transportation agencies.

Traffic signal retiming is only part of our study. We start our study by creating a tool to visually explore the congestion in the Las Vegas Valley for an entire year, 2025. Part of our hypothesis is that traffic signals experiencing recurrent congestion may be candidates for retiming. However, once we find the congested intersections and corresponding signals, our next step is to come up with an automatic method to identify the affected traffic signals. We use a machine learning algorithm, an Autoencoder, to learn the most important characteristics of a congested intersection. This is accomplished by training the Autoencoder to learn from unlabeled data, in a semi-supervised approach. Our goal for our research output and methods is to automate the process of finding suboptimal traffic signals so that retiming can take place on a more frequent basis.

Although vehicle detection is a common technology, many intersections in urban areas do not use that technology because of the added costs for installation and upkeep. In Las Vegas, there is a combination of fixed time and actuated (sensor-activated) traffic signals. In our study we analyze the intersections in the Las Vegas Valley for congestion and to identify suboptimal traffic signals. Our analysis is twofold, first, we built a spatial map or maps that allowed us to perform a visual analysis of traffic congestion in the Las Vegas Valley to establish which intersections are experiencing the most congestion and in a recurrent manner. Second,

we built an Autoencoder to automatically identify traffic signals that are most likely experiencing congestion due to suboptimal timing, more specifically traffic signal cycle time length.

We also collaborated with an undergraduate student to run multiple traffic signal simulations. We set out to find observations on short and long traffic signal cycle lengths. The simulation results revealed that smaller cycle times only worsen the average vehicle wait times by a small percentage from the average wait times associated with an optimal cycle length. The impact of our research has the potential to improve delays for motorist and to aid transportation agencies with signal retiming efforts.

Literature Review

Deep Learning and Autoencoders

An Autoencoder (AE) is an artificial neural network that can be trained to learn patterns of unlabeled data in an unsupervised learning setting. Unsupervised learning is a type of machine/deep learning that does not require costly human labeling of data. Regarding unsupervised learning, in a review article on Deep Learning, LeCun et al. 2015 predicted that unsupervised learning would be far more important in the long term given that human learning is largely unsupervised. Given the large amount of road traffic data are automatically collected through various forms (sensors, GPS, etc.), we naturally leaned on machine learning and unsupervised learning to analyze road traffic congestion and possible causes.

An AE can be structured in numerous ways, some network architectures include Undercomplete/Feedforward AE (the most basic form), Variational AE, Sparse AE, Convolutional AE, Denoising AE, and Deep AE. An Autoencoder learns the most important characteristics from the input to reconstruct it without creating an exact replica, the identity function. In a typical feedforward AE, for example, as seen in Figure 1, inner layers decrease as they approach the middle layer (bottleneck/latent space). In this first part, the AE is the encoder where the input is compressed and produces a lower dimension representation of the input. The second part of the Autoencoder is the decoder where the output is generated using the most important characteristic learned in the latent space (bottleneck part of the AE).

Traditional Autoencoder Architecture

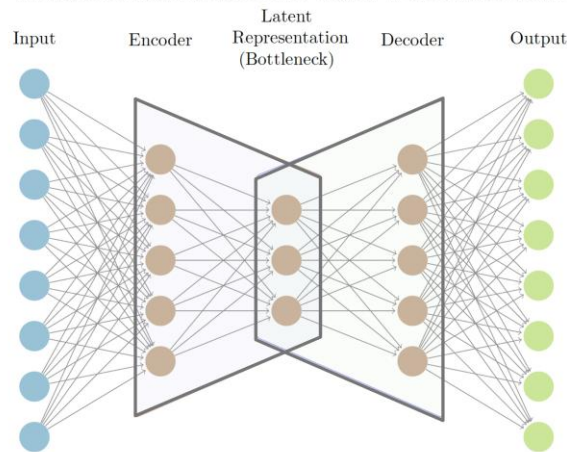


Figure 1. Undercomplete/Feedforward Autoencoder Architecture

Deep Learning is possible with Autoencoders by means of stacking. Stacked Autoencoders require less data to be learned effectively. However, one of the drawbacks of Stacked Autoencoders is that training them using Backpropagation causes problems with the gradient being too small and leads to a local minimum, vanishing gradient per original authors (Hinton and Salakhutdinov, 2006) and mentioned by (Murphy, 2012). The Deep Autoencoder's learning can be improved by using a recommended approach by Hinton and Salakhutdinov 2006 of using Restricted Boltzmann Machines (RBM) and training them greedily to get initial weights that can be fed into the Deep/Stacked Autoencoder as initial weights. The RBM is a two-layer bipartite architecture that is a Neural Network that learns the probability distribution of a set of inputs in an unsupervised manner. Common tasks that RBMs can do are feature extraction and recommendations. We will investigate these Autoencoder improvements as we continue into future work.

Methods/Methodology

The Regional Transportation Commission of Southern Nevada (RTC) Freeway and Arterial System of Transportation (FAST) division oversees traffic control, including managing traffic signals. RTC manages more than 1,168 arterial traffic signals (Regional Transportation Commission of Southern Nevada, n.d.). According to a public meeting report, the RTC uses their Freeway and Arterial System of Transportation's (FAST) automatic vehicle location (AVL) data for journey time data to conduct traffic studies on corridors (Regional Transportation Commission of Southern Nevada, 2021). AVL is a technology to track vehicle speed, among other travel metrics using satellite communications. The journey or trip travel data is an open dataset made available by the RTC that we will use in this study. The open dataset includes signal datapoints from Signal Performance Metrics but can be downloaded from the webpage dropdown as "Status at Signal" (RTC FAST, n.d.). The dataset includes granular trip data collected by the RTC maintenance vehicles or other city utility vehicles. For this study we use twelve datasets for visual analysis and exploration for the months for year 2025. Additionally, we use the same data to prepare a working dataset for the Autoencoder.

Primary datasets

Regarding the traffic signal data that appears in our primary dataset, the signal data consist of traffic signals owned by entities in the Las Vegas Valley: City of Las Vegas, City of North Las Vegas, City of Henderson and Clark County unincorporated areas in the valley. We excluded traffic signals located in Mesquite and Boulder City that appear in the primary dataset. The 2025 dataset distribution among the entities is noted in the Figure 2. In Figure 2, the x-axis labels correspond to the entity/owner of the traffic signals along with the signal identification listed in the RTC Traffic Signal at Status dataset.

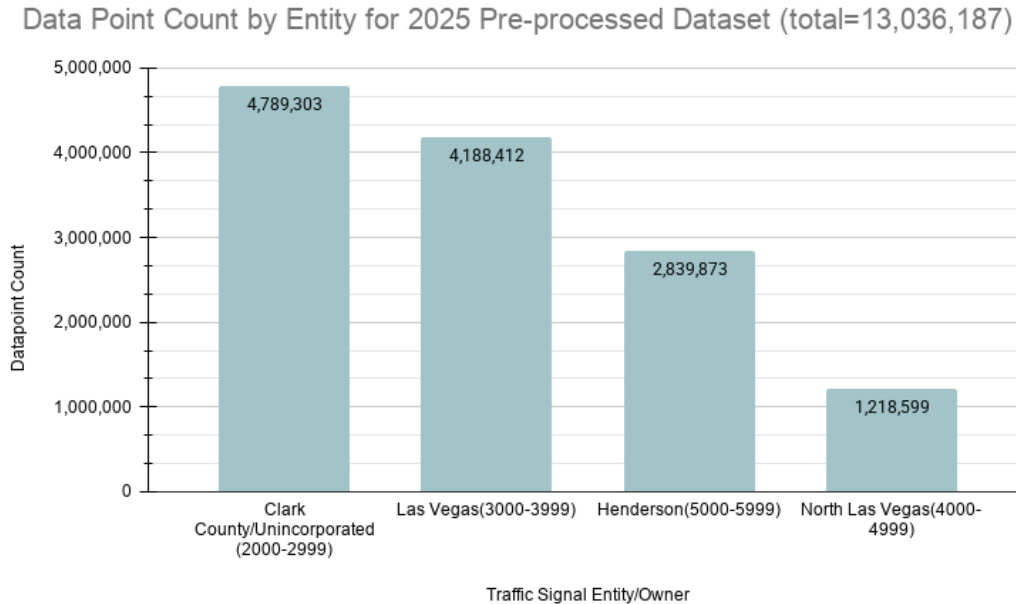


Figure 2. Data Points Collected by Entity/Owner in the Pre-processed Dataset for 2025

After cleaning the dataset, at the pre-processing step, the resulting working dataset in Figure 2 shows that we were working with thirteen million data points or records. The Figure 2 bar chart shows that the traffic signals that make up 69% of the data points captured belong to traffic signals located in the Clark County / Unincorporated and the City of Las Vegas. While the cities of Henderson and North Las Vegas traffic signals make up 31% of the data points collected. The distribution of the data is important to note when we evaluate, through a visual exploration, the congestion measures of the Las Vegas Valley intersections on the 12 geospatial maps we generated (one for each month in 2025).

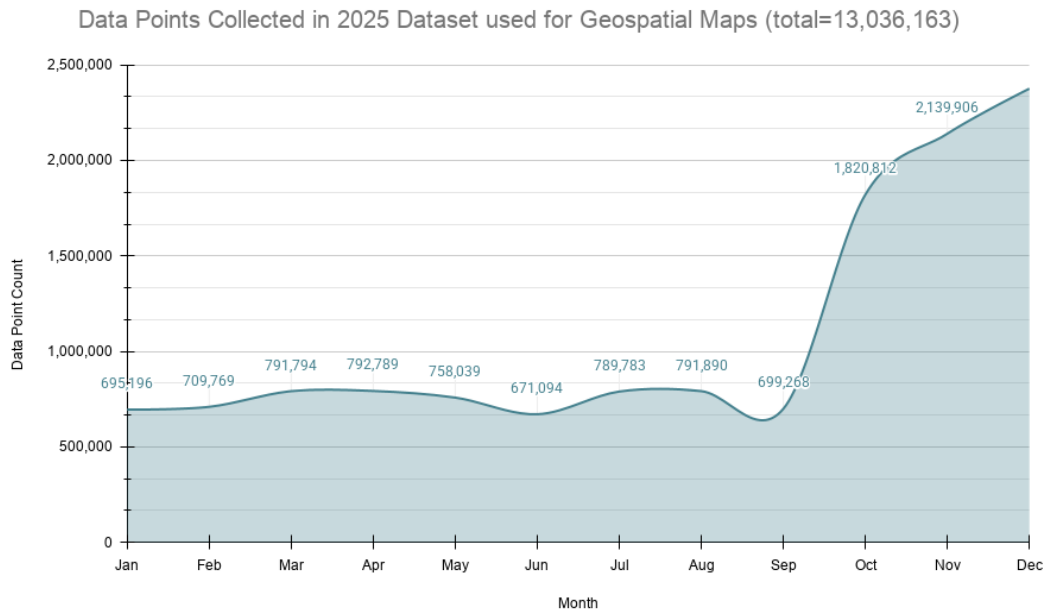


Figure 3. Data Points Collected in the Dataset for 2025 used for Geospatial Maps

Figure 3 shows the monthly breakdown of the traffic signal data collected for 2025. This area chart shows that RTC collected an unusually high amount of data the last three months of the year compared to earlier months. Additionally, the number of datapoints from Figure 2 versus Figure 3 differs by 24. This reduction was due to the validation we performed where we identified 24 datapoints or rows of data in the CSV file (generated by the Geospatial Maps step) that were included incorrectly in the original dataset download from RTC Signal Performance Metrics webpage (RTC FAST, n.d.).

Pre-processing Primary Dataset with Python Programs

Our paper on geospatial maps (Martinez and Bein, 2025b) provides a precise explanation of the steps involved in the pre-processing of the downloaded dataset, Traffic Signal Performance Metrics. “The first step in pre-processing the datasets was to download CSV files from the RTC download website for each month” “We downloaded the data for one month into multiple CSV files because the download failed when entering a month’s date range.” We had a total of 210 CSV files to preprocess for 12 months. “Next, we loaded the data and programmatically fixed the number of columns from 27 to 26 because there were some records that had an extra column due to an error in the data. The error was that those records had the text “Valve Truck” listed under the ‘SignalID’ column. So, we removed this text and shifted the rest of the data back one column over. Additionally, we removed any rows that were missing data in any of the 26 columns. “

Secondary datasets

For the secondary datasets, we downloaded weather data such as temperature and precipitation for every day of the year 2025. We prepared a list of holiday dates, and we collected RTC traffic incidents that arrived via email notifications. However, we did not get to incorporate these into the maps or our Autoencoder. These datasets along with traffic signal cycle lengths will be incorporated in the future work of Yazmin E. Martinez's PhD thesis.

Process Datasets to Generate Maps and Working Dataset for Autoencoder

After preprocessing the primary and secondary datasets, we generated twelve geospatial maps for each month in the year 2025. The purpose for this step was to visually explore the congestion in the Las Vegas Valley, refer to Appendices A-L and our research output webpage (Martinez and Bein, 2026b). These maps are interactive and were designed with Human-Computer Interaction principals along with Cartography (science of making maps) practices as listed in one of our published papers (Martinez and Bein, 2025b). The maps are available on our UNLV research website (Martinez and Bein, 2026b). Additionally, upon generating the geospatial maps, we can filter by the signal performance index (SPI) category. SPI is the congestion measure we used in our method for generating a visualization of the congestion using the RTC Signal Performance Metric open dataset (RTC FAST, n.d.). For a full explanation of our map generation method, implementation details, and the explanation of the congestion measure refer to our published paper from 2025 (Martinez and Bein, 2025b). The maps in the Appendix A-L correspond to aggregated congestion values at each intersection or signal listed in the primary datasets for the months of January through December.

The map images in the Appendix A-L show the resulting 2D Geospatial dot maps of road traffic congestion in Las Vegas Valley for January and December 2025. The maps show obvious differences in road traffic congestion at intersections/signals. The maps were designed with interactivity including HCI principals to allow for exploration and understanding of the road congestion. The maps make “use of the user’s visual sense to present information regarding the level of congestion. For example, using color dots to represent congestion levels (SPI) or categories at intersections. Another example is that the map shows a more dynamic gradient color scheme for each day block in the calendar view” (Martinez and Bein, 2025b). The SPI categories that we are concerned with include heavy congestion and mild congestion. Heavy congestion (dot color=Red, SPI range=0-25) represents low average speed and poor road traffic state. Mild congestion (dot color= Orange, SPI range=26-50) represents lower average speed and road traffic state bit weak (Martinez and Bein, 2025b).

Congestion Measure SPI

Regarding the congestion measure, we fully explain the reason we chose to use Speed Performance Index in the following excerpt (Martinez and Bein, 2025b).

“To analyze road traffic congestion, we can observe the vehicle travel speed; however, travel speed alone is not an indicator of a congested road or intersection. A survey on road traffic congestion measures by (Afrin and Yoto, 2020) lists 11 congestion measure approaches. Each approach has advantages and disadvantages; however, the data required to calculate the congestion measure should also be considered. According to (Afrin and Yoto, 2020), the congestion measure speed performance index (SPI) is advantageous because it is easy to comprehend; however, it fails to consider traffic nonrecurring conditions (i.e. work zones, weather, traffic incidents, etc.). Afrin and Yoto 2020 defined road traffic congestion into two categories, recurring and nonrecurring. Recurring traffic conditions include bottlenecks and capacity along with inadequate traffic controllers (i.e. poorly timed signals)”

$$SPI_{signal} = \left(\frac{v}{v_{max}} \right)_{avg} \times 100 \quad (1)$$

“Equation (1) is the speed performance index (SPI) [is] calculated at the signal level. The SPI is a congestion measure defined by the ratio of vehicle speed and the maximum permissible speed which we interpret as the speed limit. The ratio is a value included in the original dataset (RTC FAST, n.d.) as ‘LinkSpeedRatio’ which we averaged (or take the mean of) and multiplied by 100 for every signal ID in the range of 2000 to 5999 for each month. Each 1000 numbers are assigned to a different entity/owner of the traffic signal (i.e. Clark County, City of Las Vegas, etc.).

After visually inspecting the congestion measure SPI throughout the Las Vegas Valley for each month in 2025 (Appendix A-L), we generated a list that we can manipulate to generate a graph that would show the traffic intersections that experience recurring mild and heavy congestion six months or more in the year 2025. Additionally, the following Figure 4 shows intersections where the datapoints collected on each traffic signal were one hundred or more and the congestion measure was in the congested category ($SPI < 50$). This yearly dot graph allows us to quickly identify a candidate signal list of intersections that we can investigate for timing issues.

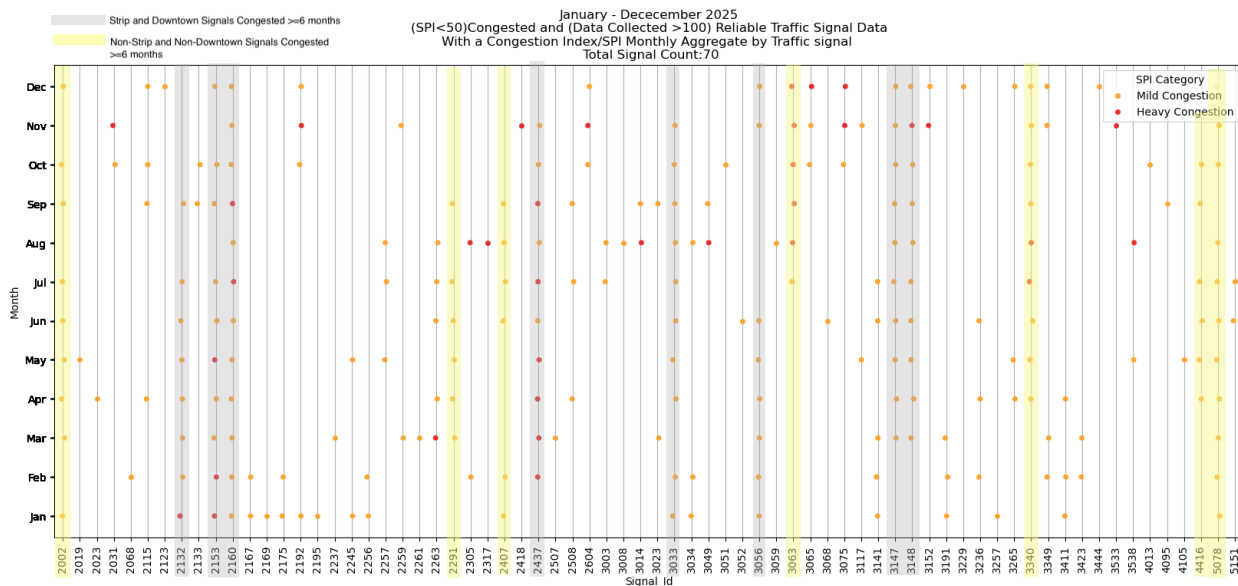


Figure 4. Dot Graph of the Top 20 Most Congested (Mild or Heavy) Traffic Signals each Month in 2025

To generate the graph in Figure 4, which is called a strip plot (Seaborn, 2026), we preprocessed the RTC Signal Metrics Dataset for each month in the year 2025. Furthermore, we developed a Python program to generate interactive geospatial maps with details of the congestion in the Las Vegas Valley including calculating the aggregated SPI congestion measure, which is an index ranging from 0 to 100 to denote the level or severity of the road traffic congestion (Martinez and Bein, 2025b). The same Python program also generated a monthly dataset for each of the months in 2025 that we used to generate Figure 4. However, we had to do further processing to each monthly dataset. For example, with the month of January, we sorted the aggregated SPI in ascending order for the 1,137 traffic signals with SPI values ranging from 25 to 151. Note that we disregarded any aggregate SPI values for each signal beyond 100. 17 traffic signals with unordinary metrics where vehicles had a speed that surpassed speed the limit so much that the ratio or the SPI went beyond the max categorical measure of 100.

We repeated this process for all twelve months for the year 2025.

The following are the process steps in detail.

1. Generated 12 Maps and comma-separated values (CSV) Files
 - Note: when generating working datasets in CSV format, discard any traffic signals that have less than the 100 datapoints deemed necessary for reliable analysis of signal congestion.
2. Imported 12 CSVs into Python Script
3. For each Month
 - a. Sorted by 'Speed Performance Index (SPI)' Column

- b. Added two columns for month values: 'month_name' and 'month_num'
 - c. Deleted all rows after the 20th row
4. Combined all the months traffic signals into one list
5. Generated a list of all the traffic signals with recurring or non-recurring congestion measures SPI along with other corresponding data.
 - a. This step yielded 241 datapoints with some traffic signals repeating if they were congested for multiple months in 2025.
6. Generated a list of unique traffic signals and the corresponding months that they were congested SPI < 50.
7. Generated a Seaborn Stripplot to show the 70-traffic signal congestion measure observations with color coded vertically oriented strips as a representation for the traffic signal congestion distribution throughout the year. The colored dot orange represents mild congestion (SPI<50 and SPI>25) and the red dot represents heavy congestion (SPI<25) (seaborn.stripplot, 2026).

Table 1. Fifteen signal congested signals ≥ 6 months and located on the strip/downtown or non-strip/non-downtown areas. The highlighted 7 signals rows represented the areas with least pedestrian traffic that we are interested in analyzing further.

Signal ID	Entity Or City Owner	Primary and Secondary Roads	No. Months Congested
2002	Clark County/Unincorporated	Flamingo Rd and Nellis Blvd	9
2132	Clark County/Unincorporated	Flamingo Rd and Las Vegas Blvd	8
2153	Clark County/Unincorporated	Las Vegas Blvd and Harmon Ave	10
2160	Clark County/Unincorporate	Las Vegas Blvd and Bellagio Dr-Paris	12
2291	Clark County/Unincorporated	Boulder Hwy and Flamingo Blvd	6
2407	Clark County/Unincorporated	Flamingo Rd and Grand Canyon Dr	6
2437	Clark County/Unincorporated	Sands Ave and Palazzo-Wynn Gate	10
3033	Las Vegas	Main St and Bonneville Ave	10
3056	Las Vegas	Casino Center Blvd and Fremont St	8
3063	Las Vegas	Charleston Blvd and Decatur Blvd	6
3147	Las Vegas	Fremont St and Sixth St	10
3148	Las Vegas	Grand Central Pkwy Bonneville-Alta	10
3340	Las Vegas	Las Vegas Blvd and US 95 NB	9
4416	North Las Vegas	Cheyenne Ave and N 5th St	6
5078	Henderson	Horizon Dr and Horizon Ridge Pkwy	11
17 Signals Total			

Process Working Dataset with an Autoencoder

We use a feedforward Autoencoder (simple form) to process the working dataset. The Autoencoder layers are configured in the manner listed in Figure 5. The hashtag in the Figure represents the number of input values / rows in the data / data points.

Autoencoder configuration

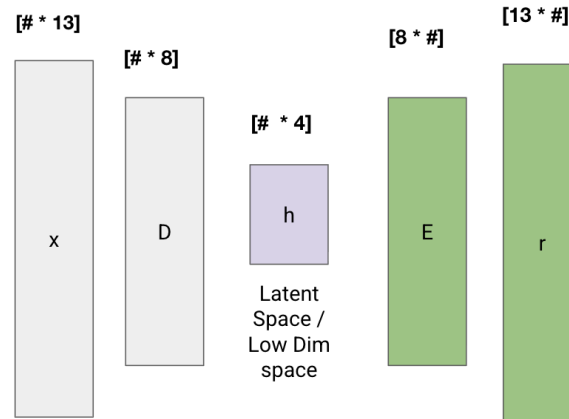


Figure 5. Layers of the Feedforward Autoencoder

Dimensionality reduction and Feature Extraction

While some common uses of Autoencoders in machine learning include finding anomalies, noise removal, and generating new data, the traditional use for Autoencoders was for dimensionality reduction or feature learning (Goodfellow et al., 2016). An Autoencoder compresses and reconstructs data allowing dimensionality reduction which leads to feature extraction. We focused on feature extraction to understand which dataset features / columns impacted the Autoencoder being trained on anomalous congested signal data. Given the characteristics of our primary dataset, we chose to use Autoencoders because they work well with non-linearly separable data.

Dimensionality reduction

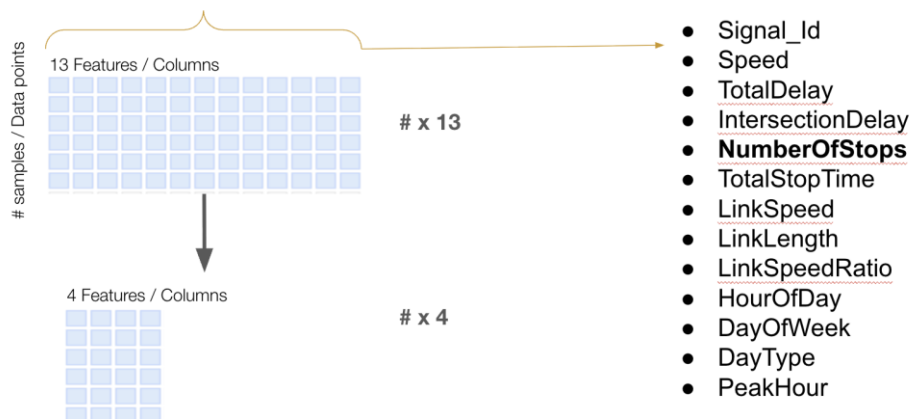


Figure 6. Visual of Dimensionality Reduction from Autoencoder Input to Latent Space

Anomaly Detection

The implementation of the Autoencoder consisted of a Python program using the TensorFlow Keras open-source library for deep learning. To setup the Autoencoder, we separate anomalous data with a congestion measure Speed Performance Index < 10 (ratio overall range is 0:100) which falls under the range of heavy congestion. We then trained the Autoencoder with normal data; we considered this data to be all datapoints that had a SPI > 10 . The idea was that when Autoencoder saw new data, the reconstruction error will be high for anomalous data since the Autoencoder was trained on normal data.

Signal Cycle Time	20 cars AWT in sec	20 cars CR	40 cars AWT in sec	40 cars CR	60 cars AWT in sec	60 cars CR
20 sec	208.00 sec	1.51	408.00 sec	1.62	608.00 sec	1.68
40 sec	146.75 sec	1.06	279.88 sec	1.11	413.00 sec	1.14
60 sec	138.00 sec	1.00	258.00 sec	1.02	378.00 sec	1.04
80 sec	138.25 sec	1.00	252.38 sec	1.00	366.50 sec	1.01
100 sec	141.25 sec	1.02	251.75 sec	1.00	362.25 sec	1.00
120 sec	144.75 sec	1.05	253.25 sec	1.01	361.75 sec	1.00
140 sec	151.75 sec	1.10	254.25 sec	1.01	363.00 sec	1.00
160 sec	156.75 sec	1.14	259.88 sec	1.03	363.00 sec	1.00
180 sec	159.75 sec	1.16	264.38 sec	1.05	369.00 sec	1.02
200 sec	160.75 sec	1.16	265.88 sec	1.06	371.00 sec	1.03
220 sec	165.50 sec	1.20	270.12 sec	1.07	374.75 sec	1.04
240 sec	175.50 sec	1.27	278.62 sec	1.11	381.75 sec	1.06

Figure 7. Traffic Signal Simulation Results for Vehicle Overhead 5 seconds and Batch Penalty of 3 seconds. The image was created by Finnagin Wantland and we were given permission to use it, we added the highlighted rows and arrow annotations (Wantland et al., 2026).

Batch Scheduling – Traffic Signal Simulation

We collaborated on a traffic signal simulation (Wantland et al., 2026). As a result, we observed that a shorter cycle length that is de-suboptimal differs only slightly in average vehicle wait time when compared to a longer optimal cycle length (Figure 7). The computational simulation calculates the average wait time (AWT) a queue of cars at a traffic signal had with an overhead of 5 seconds per vehicle. We define the vehicle overhead as a vehicle time delay that the vehicle takes to get through the intersection. The simulation also considers a batch penalty of 3 seconds. The batch penalty is the time lost when a platoon or a batch of vehicles initiates travel. For example, the last column of in Figure 7 shows the simulation results when we tested queuing 60 vehicles at a time over the span of an hour and setting the traffic signal cycle time 120 seconds/2 minutes. The Traffic signal cycle length is a combination of green and red times, namely 60 seconds each in the example demonstrated in Figure 7. The example shows a AWT of 361.75 seconds when the cycle time length is set to 120 seconds; the AWT only worsen by 4% to 378 seconds when we shorten the signal cycle time in half. Note that the CR is the

computational ratio which defined as the AWT value over the optimal ATW (in red text) value. The optimal AWT is the best average waiting time that we observed in the simulation.

Results and Future Work

Anomaly Detection

To calculate the accuracy of the Autoencoder performing anomaly detection, we used the MAE loss (Mean Absolute Error) to find the differences between reconstructed output and original input. Mean Absolute Error is preferred for Autoencoders. To get the top four important features identified by the Autoencoder, we analyzed the encoder part of the Autoencoder to get the weights and used the NumPy corrcoef function to calculate the Pearson product-moment correlation coefficient matrix.

The results in Table 2 show the accuracy of the Mean Absolute Error which indicates how well the Autoencoder found traffic signals with heavy congestion ($SPI < 10$). The table also lists the top four most important input features that influence the Autoencoder's learning. The month of August is the only month that had a notable accuracy of 72%. This could be explained by looking at the August geospatial map, Appendix H. There are more signals with red dots (represent an aggregated average SPI value for heavy congestion) than all the other months. This indicates that the performance of the Autoencoder may depend on having more anomalous data points that indicate extreme heavy congestion. Furthermore, some of the future work includes making improvements to the Autoencoder such using Restricted Boltzmann Machines to get initial weights for the Autoencoder rather than using random weights. Additionally, we plan to experiment with other Autoencoder architectures beyond the basic form of the Feedforward Autoencoder. Since the final goal for the Autoencoder is to use it to automatically identify traffic signals that are good candidates for retiming, we also plan to gather more data on the traffic signal parameters themselves, such as signal cycle length, green/yellow/red time intervals, among others. This additional data will allow the Autoencoder to reveal if signal parameters features have an influence on finding heavy congested intersections.

Table 2. Autoencoder Results

Month	Accuracy	Precision	Recall	Top 4 Important Features identified by Autoencoder
January	0.2085	0.01784	0.8788	LinkSpeed, IntersectionDelay, NumberOfStops, Signal_Id
February	0.3488	0.0353	0.0353	NumberOfStops, Signal_Id, LinkLength, Speed
March	0.35688	0.0293	0.90469	TotalStopTime, Speed, LinkSpeed, Signal_Id
April	0.14208	0.02036	0.9073	NumberOfStops, Speed, IntersectionDelay, TotalStopTime
May	0.4915	0.0680	0.9332	Signal_Id, NumberOfStops, IntersectionDelay, LinkSpeed
June	0.3629	0.0277	0.88024	LinkLength, IntersectionDelay, TotalStopTime, Speed
July	0.3942	0.07002	0.8772	Signal_Id, Speed, NumberOfStops, LinkLength
August	0.7172	0.2557	0.9449	Signal_Id, NumberOfStops, TotalStopTime, LinkLength
September	0.16176	0.0274	0.9097	TotalStopTime, Signal_Id, Speed, LinkLength
October	0.12482	0.0212	0.9340	LinkSpeed, LinkLength, NumberOfStops, Speed
November	0.4035	0.0718	0.9522	TotalDelay, IntersectionDelay, LinkLength, Signal_Id
December	0.08571	0.0263	0.9471	NumberOfStops, Signal_Id, IntersectionDelay, LinkLength

Conclusion

We set out to investigate road traffic congestion with the goal of finding traffic intersections affected by signal timing. We developed a tool that generated spatial maps based on an open data set containing Las Vegas Valley Signal Performance Metrics. We created an Autoencoder to aid in automatically finding the congested intersections and for identifying feature inputs that helped the Autoencoder learn. Lastly, we computed an average wait time for vehicle delay and found that shorter cycle lengths only impacted the vehicle average wait time by a low percentage from an optimal average wait time. Our work includes analysis of large amounts of traffic data. In future work we will use the analysis, tools, and traffic simulation data to continue our investigation and to validate that recurring congested traffic signals are candidates for retiming.

References

- Afrin, T., & Yodo, N. (2020). A survey of road traffic congestion measures towards a sustainable and resilient transportation system. *Sustainability*, 12(11), 4660.
- Dion, F., Rakha, H., & Kang, Y. S. (2004). Comparison of delay estimates at under-saturated and over-saturated pre-timed signalized intersections. *Transportation Research Part B: Methodological*, 38(2), 99-122.
- Goodfellow, I., Bengio, Y., Courville, A., & Bengio, Y. (2016). *Deep learning* (Vol. 1, No. 2, pp. 1-800). Cambridge: MIT press.
- Hinton, G. E., & Salakhutdinov, R. R. (2006). Reducing the dimensionality of data with neural networks. *science*, 313(5786), 504-507.
- LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *nature*, 521(7553), 436-444.
- Martinez, Y. E., & Bein, W. (2026a, In Press). BatchTrafficSim: A Batch Scheduling Traffic Simulator for Traffic Signal Optimization. In *International Conference on Information Technology-New Generations (ITNG)*.
- Martinez, Y.E., Bein, W. (2026b). Research Output Webpage. (Accessed August 14 2026).URL <https://yazmin.cs.unlv.edu>
- Martinez, Y. E., & Bein, W. (2025a, April). Methods for Identifying Issues with Traffic Signal Timing Parameters and Potential Adjustments: A Survey. In *International Conference on Information Technology-New Generations (ITNG)* (pp. 721-726). Cham: Springer Nature Switzerland.
- Martinez, Y. E., & Bein, W. (2025b, January). Road Traffic Congestion Exploration and Visualization Using Open Spatial-Temporal Data and HCI Principles. In *2025 IEEE 15th Annual Computing and Communication Workshop and Conference (CCWC)* (pp. 00286-00291). IEEE.
- Martinez, Y.E., Bein, W. (2025c). Methods for Identifying Issues with Traffic Signal Timing Parameters and Potential Adjustments: A Survey. In: Latifi, S. (eds) *The 22nd International Conference on Information Technology-New Generations (ITNG 2025)*. ITNG 2025. Advances in Intelligent Systems and Computing, vol 1463. Springer, Cham. https://doi.org/10.1007/978-3-031-89063-5_62
- Murphy, K. P. (2012). Machine learning: A probabilistic perspective. MIT press.
- Qadri, S. S. S. M., Gökçe, M. A., & Öner, E. (2020). State-of-art review of traffic signal control methods: challenges and opportunities. *European transport research review*, 12(1), 55.

- Regional Transportation Commission of Southern Nevada. (2024a). How fast tmc optimizes signal timing to keep southern nevada moving (Accessed December 13, 2024). URL <https://www.rtcnv.com/news/how-fast-tmc-optimizes-signal-timing-to-keep-southern-nevada-moving/>
- Regional Transportation Commission of Southern Nevada. (2024b). Traffic signal re-timing activities (Accessed December 13, 2024). URL <https://www.rtcnv.com/about/wp-content/uploads/sites/3/2024/08/MINUTES-BACKUP-ITEM-04-Traffic-Signal-Retiming-Activities.pdf>
- Regional Transportation Commission of Southern Nevada. 2021. Notice and Agenda of Public Meeting for Freeway and Arterial System of Transportation Operations Management Committee. Retrieved August 13, 2026. from <https://assets.rtcnv.com/wp-content/uploads/sites/3/2021/05/26105229/FAST-OMC-2021-06-03.pdf>
- Regional Transportation Commission of Southern Nevada. (n.d.). About Traffic Management, <https://www.rtcnv.com/traffic-cams/about-traffic-management/>, Date Retrieved August 13, 2026.
- RTC FAST. (n.d.). Signal Performance Metrics. (n.d.). Retrieved February 1, 2026, from <https://challenger.nvfast.org/SPM/DataDownload.aspx>
- Seaborn. (n.d.). seaborn.stripplot. (n.d.). Retrieved August 9, 2026, from <https://seaborn.pydata.org/generated/seaborn.stripplot.html>
- Karami, Z., & Kashef, R. (2020). Smart transportation planning: Data, models, and algorithms. *Transportation Engineering*, 2, 100013.
- Wantland, F., Bein, W., Martinez, Y. E. (2026). Traffic Signal Simulation Findings [Unpublished manuscript]. Department of Computer Science, University of Nevada, Las Vegas.

Data Management Plan

Products of Research

The dataset used for our study includes traffic signal performance metric data provided by the Regional Transportation Commission of Southern Nevada's FAST division. The signal data consist of traffic signals owned by entities in the Las Vegas Valley: city of Las Vegas, city of North Las Vegas, city of Henderson and Clark County unincorporated areas in the valley. The dataset consistent of 13,036,163 datapoints/rows with 26 characteristics/features. We had a total of 210 CSV files that we downloaded from the RTC FAST website that included data for all 12 months in 2025.

Data Format and Content

The 210 CSV files containing the raw data can be downloaded from <https://challenger.nvfast.org/SPM/DataDownload.aspx>.

Data Access and Sharing

The public can access twelve geospatial maps on <https://yazmin.cs.unlv.edu/>. These maps are included in Appendix A-L, but the web access allows for an interactive exploration of the Las Vegas Valley traffic congestion at an aggregate monthly level and a daily granular level.

UNLV-developed course materials, software, and select research reports (other than those with copyright) will be available from a publicly accessible web page, with no authentication requirements or need to get permission. Materials will be distributed under a Creative Commons license.

Creative Commons license materials will be made available through a publicly accessible website, which will be backed up regularly using facilities. Data will be hosted by a dedicated server each at with back-up and mirroring service which supports high availability and resilience to failure.

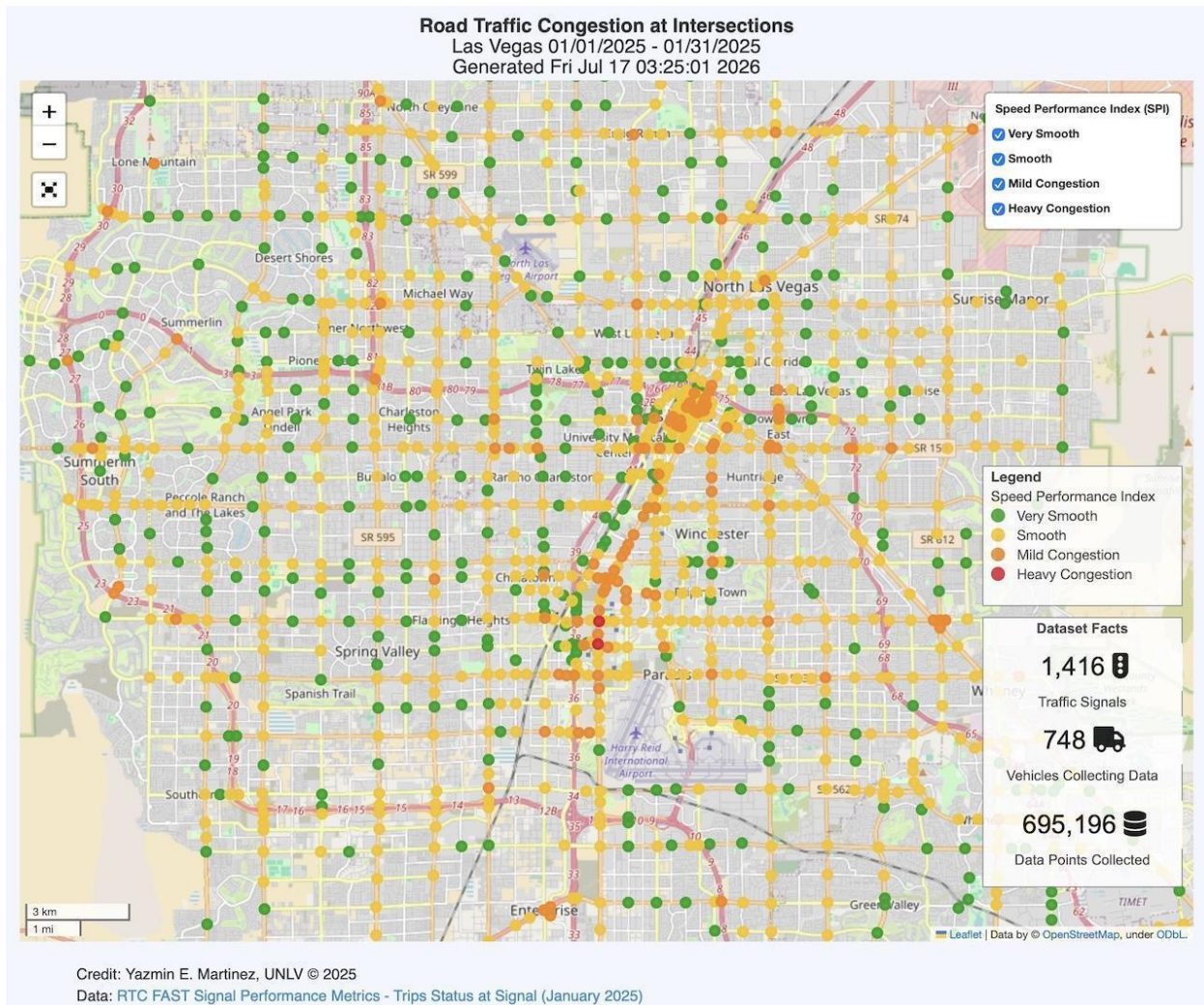
Reuse and Redistribution

UNLV-developed course materials, both lecture and online, will be made available under Creative Commons license (see <http://creativecommons.org/licenses/by-nc-sa/3.0/>). This license allows anyone to use and adapt the work as long as it is for noncommercial use and appropriate attributions are clear, and is referred to in general as the "Attribution-NonCommercial-ShareAlike" Creative Commons License. Similarly software will be licensed under the GNU General Public License. Other published information will be in the form of conference and journal papers, and re-distribution rights will be determined by the publishing venue (e.g., ACM copyright policy for SIGACT). The PIs will seek to publish in venues with an Open-Choice option, i.e. Springer Verlag.

Access is at <https://yazmin.cs.unlv.edu/>

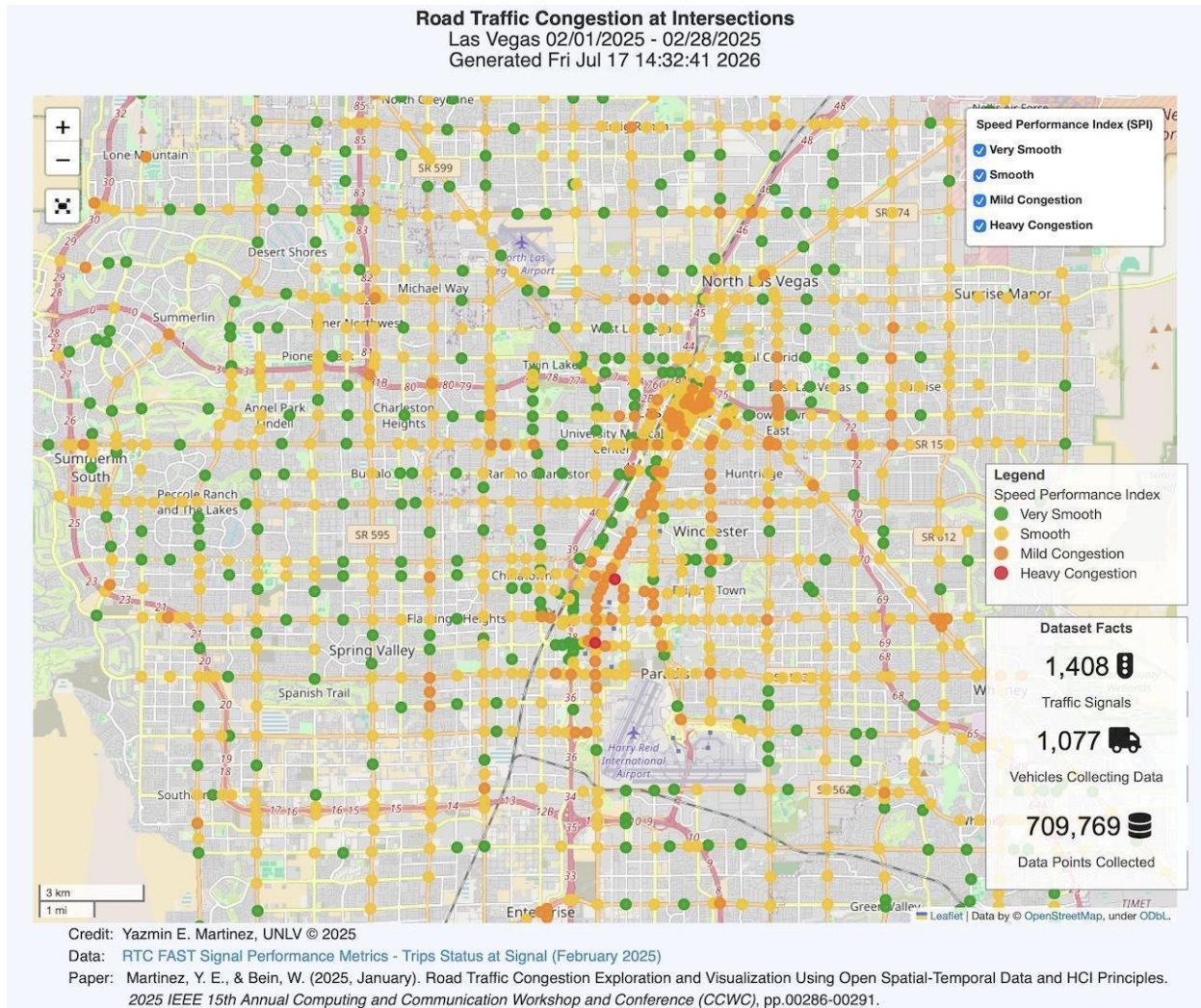
Appendix A

This is the January 2025 Geospatial Map Showing Aggregated Congestion Measure SPI.



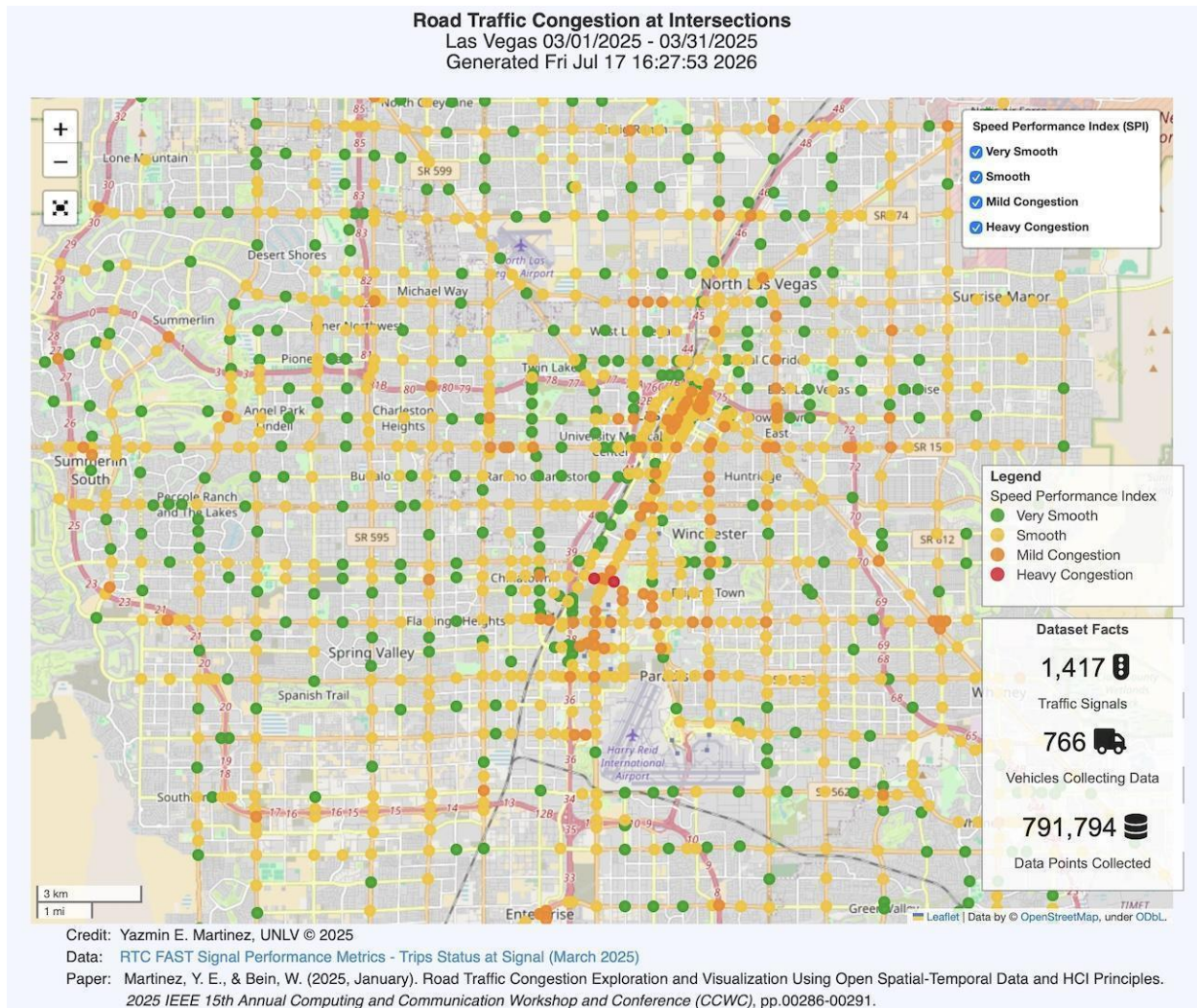
Appendix B

This is the February 2025 Geospatial Map Showing Aggregated Congestion Measure SPI.



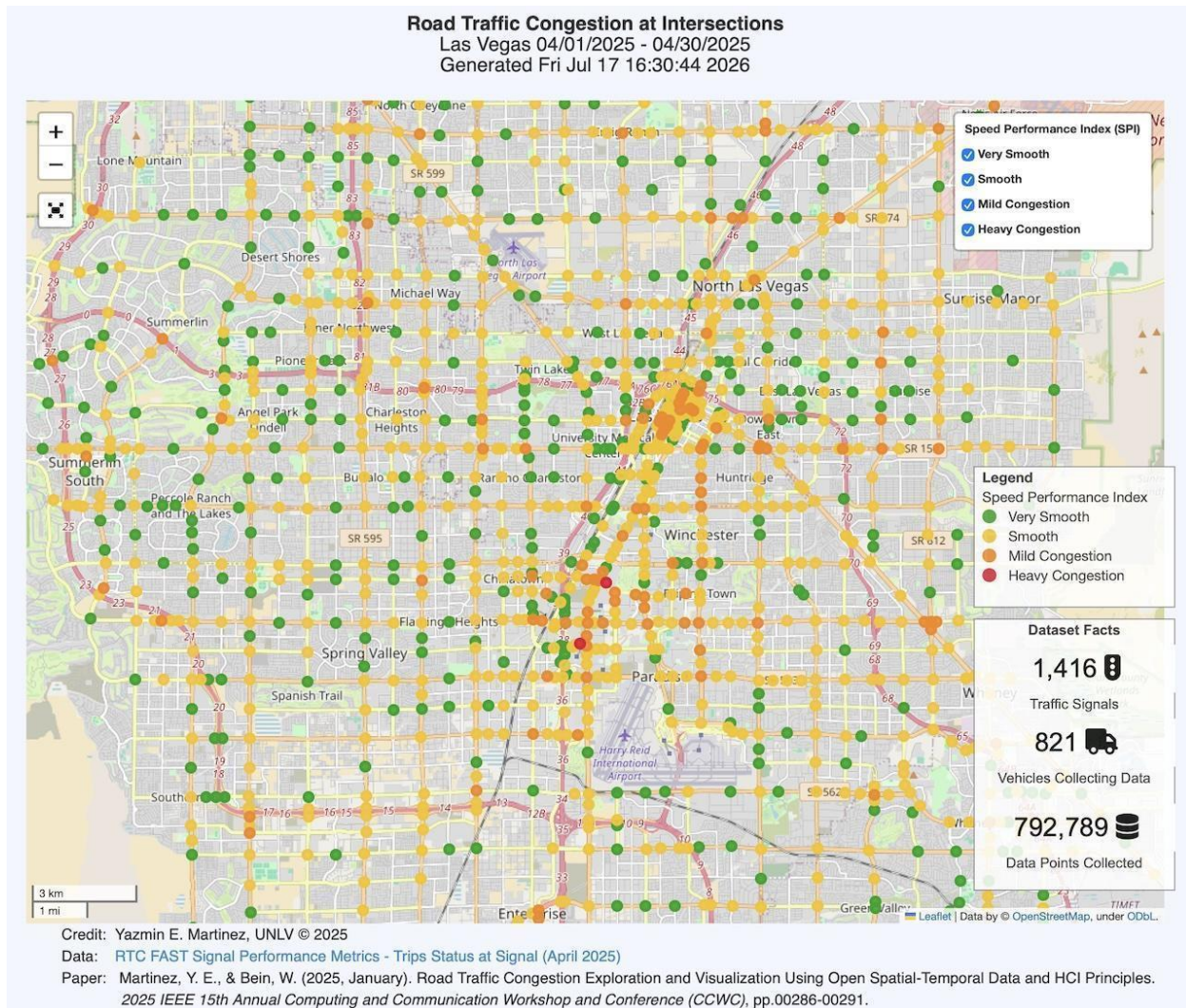
Appendix C

This is the March 2025 Geospatial Map Showing Aggregated Congestion Measure SPI.



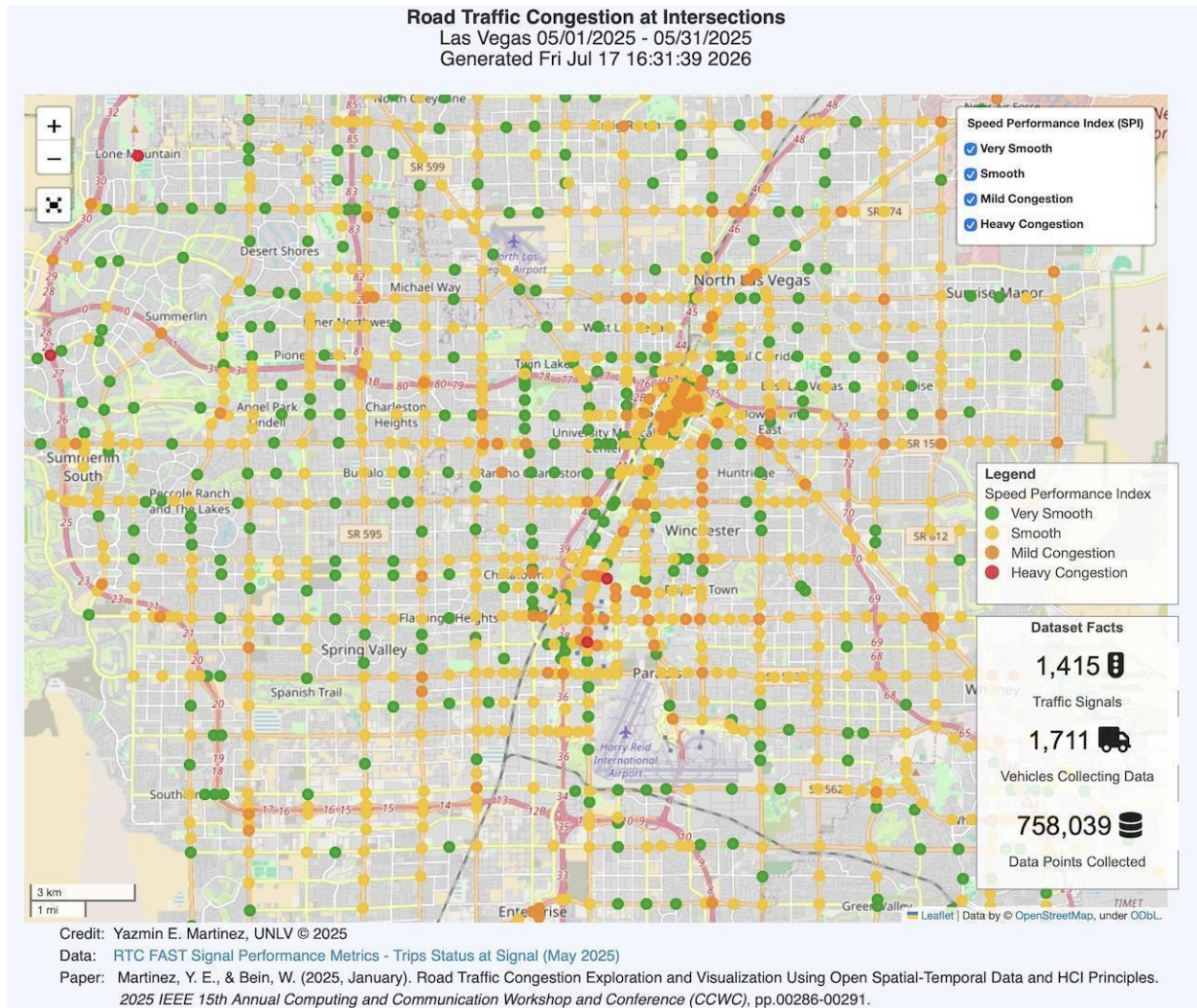
Appendix D

This is the April 2025 Geospatial Map Showing Aggregated Congestion Measure SPI.



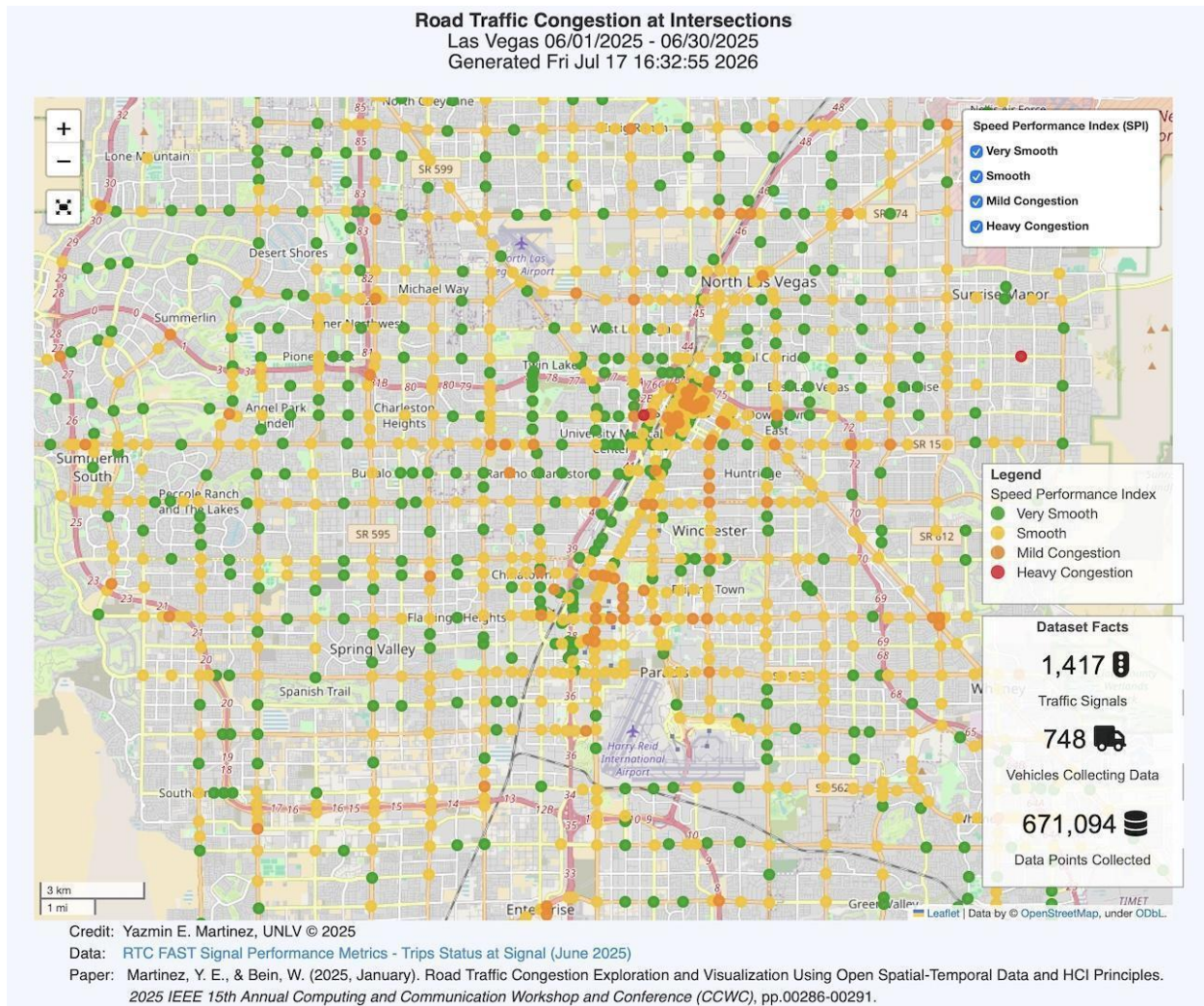
Appendix E

This is the May 2025 Geospatial Map Showing Aggregated Congestion Measure SPI.



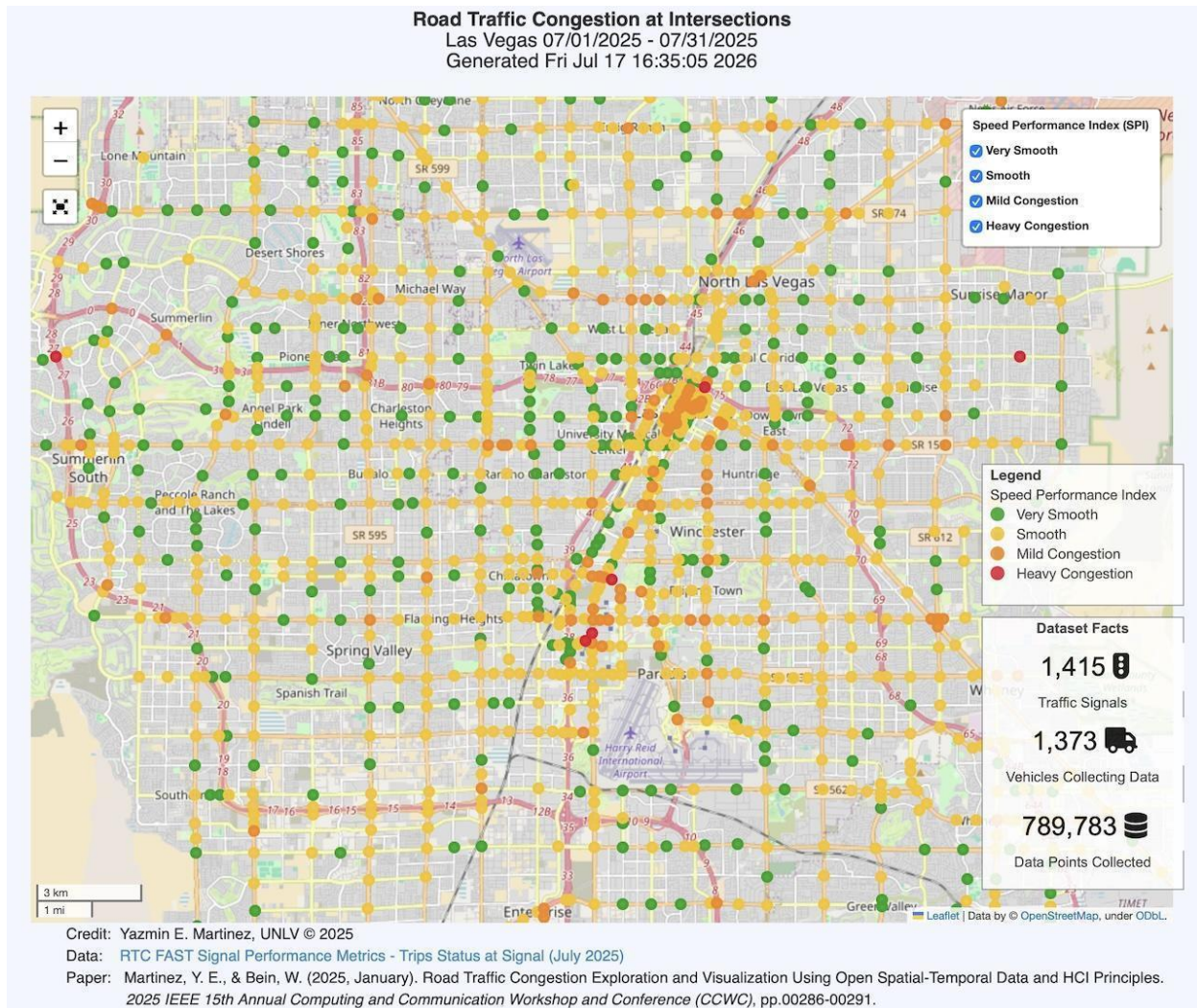
Appendix F

This is the June 2025 Geospatial Map Showing Aggregated Congestion Measure SPI.



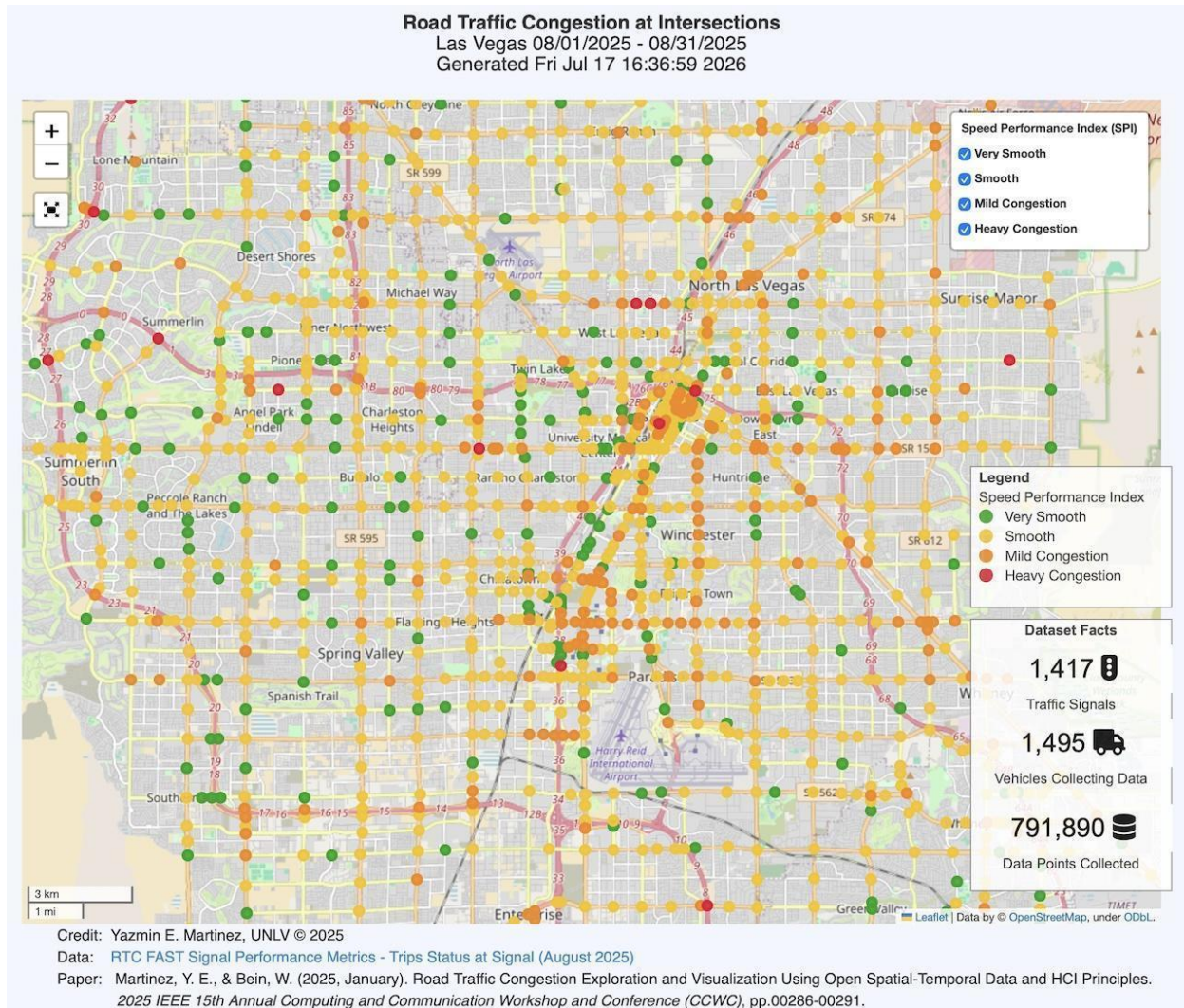
Appendix G

This is the July 2025 Geospatial Map Showing Aggregated Congestion Measure SPI.



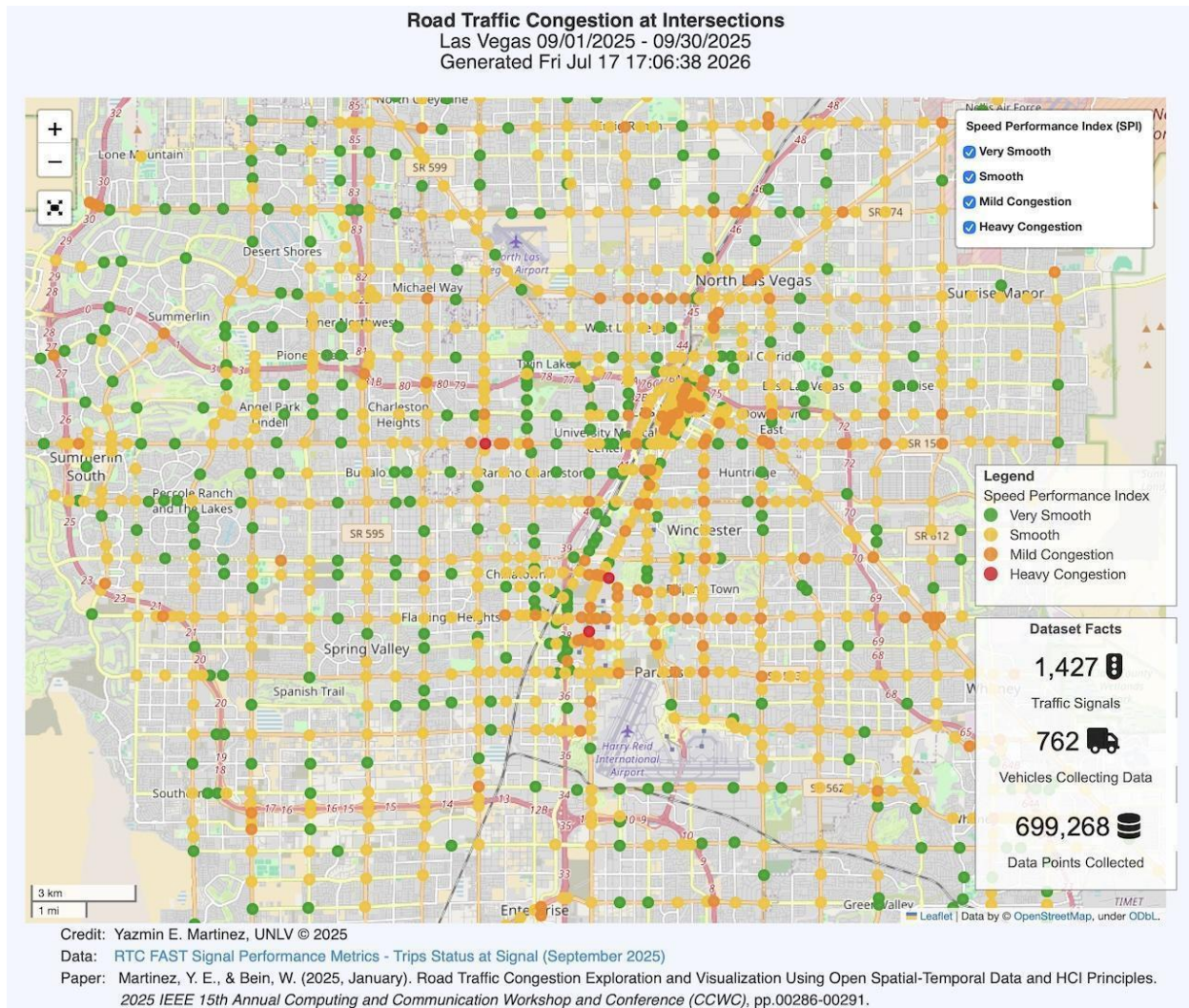
Appendix H

This is the August 2025 Geospatial Map Showing Aggregated Congestion Measure SPI.



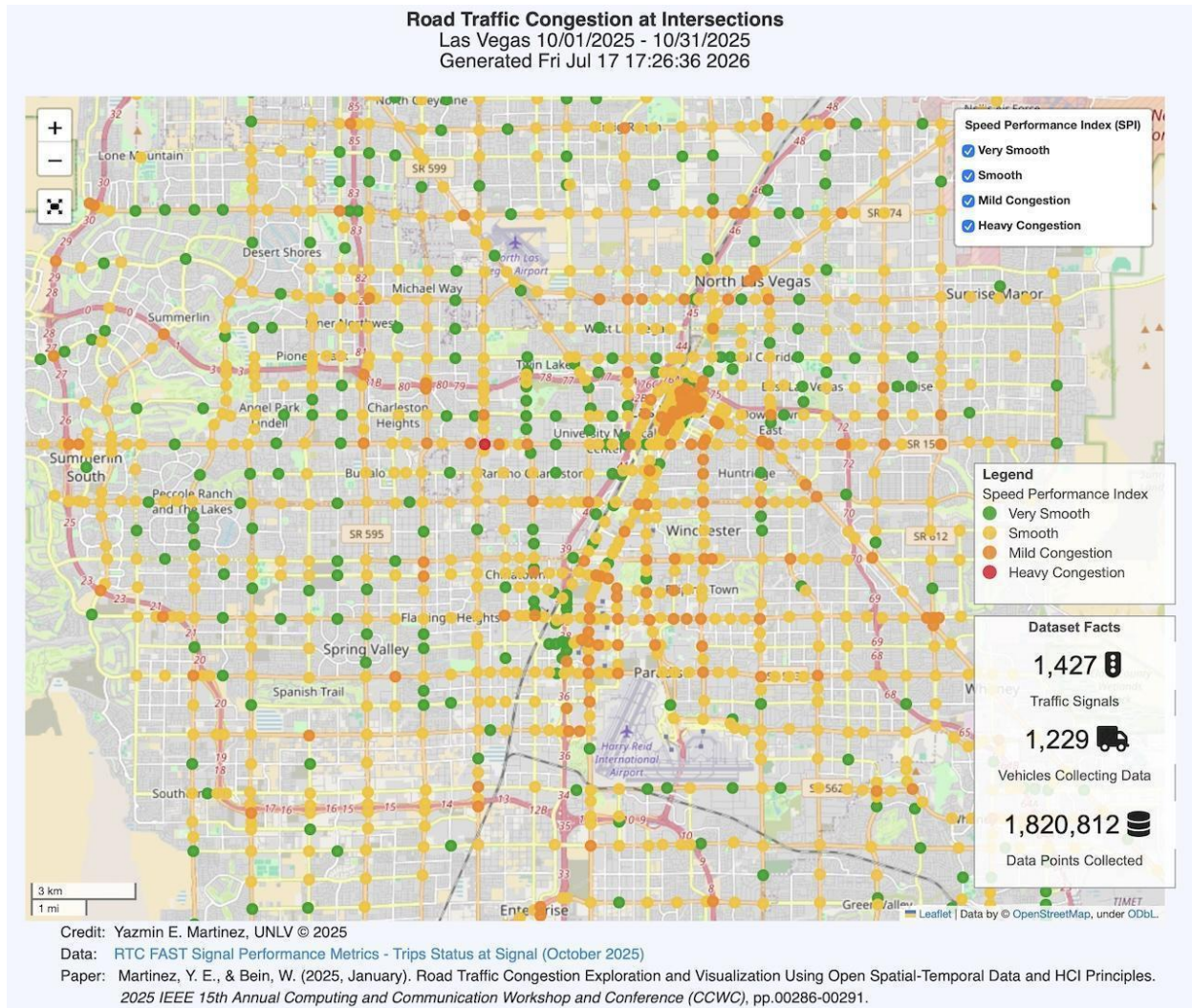
Appendix I

This is the September 2025 Geospatial Map Showing Aggregated Congestion Measure SPI.



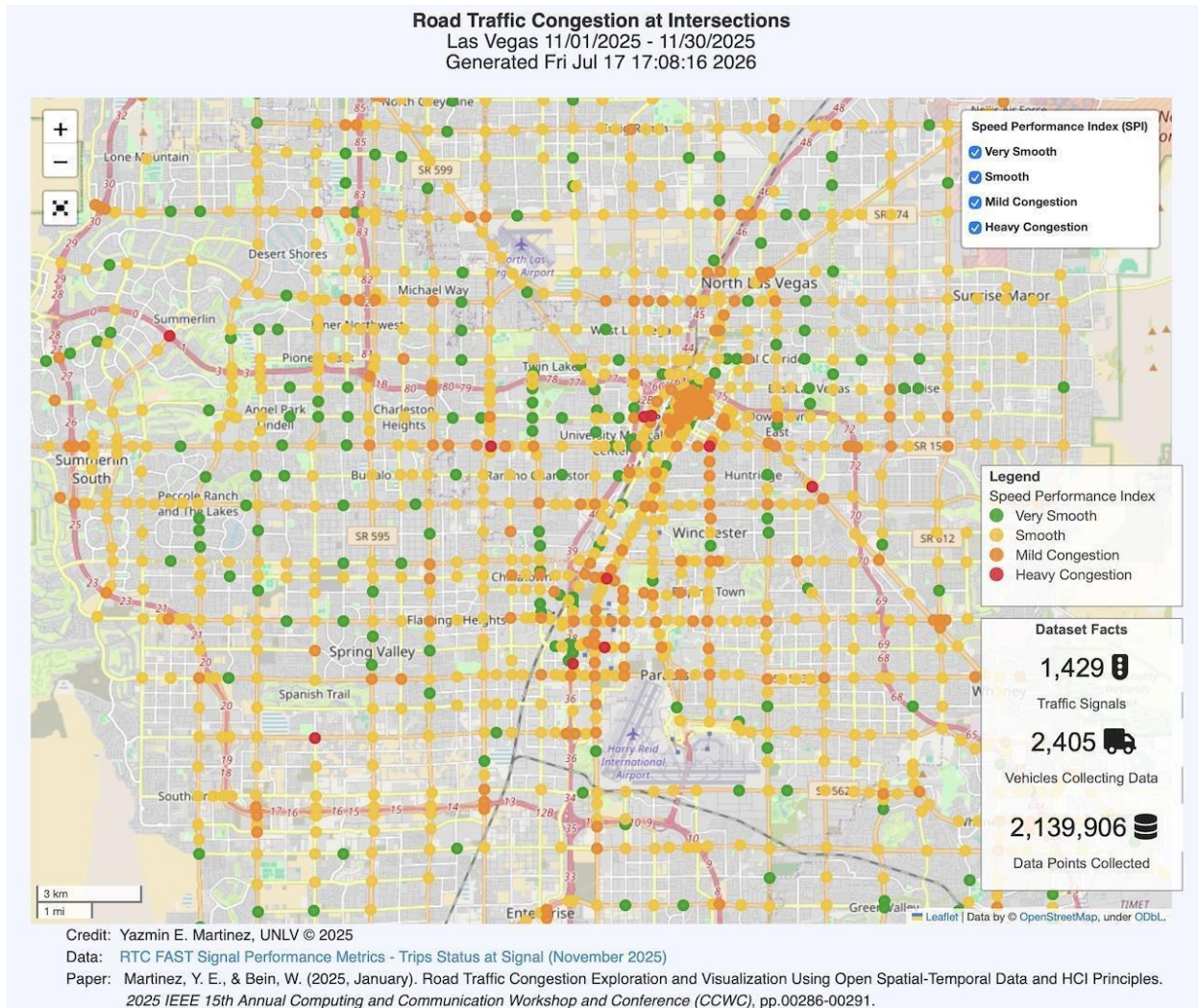
Appendix J

This is the October 2025 Geospatial Map Showing Aggregated Congestion Measure SPI.



Appendix K

This is the November 2025 Geospatial Map Showing Aggregated Congestion Measure SPI.



Appendix L

This is the December 2025 Geospatial Map Showing Aggregated Congestion Measure SPI.

