Traffic Prediction

Presentation

IML Project

Group

- Tharakadatta D Hegde (B22ES007)
- Kiran S (B22ES001)
- Aaroh Kale (B22ME001)
- Biswas Guptha (B22BB018)
- *Maheswar Reddy(B22BB029)*

Dataset

Traffic Prediction Dataset

Traffic.csv

This dataset contains 48120 observations of the number of vehicles each hour in four different junctions:

DateTime

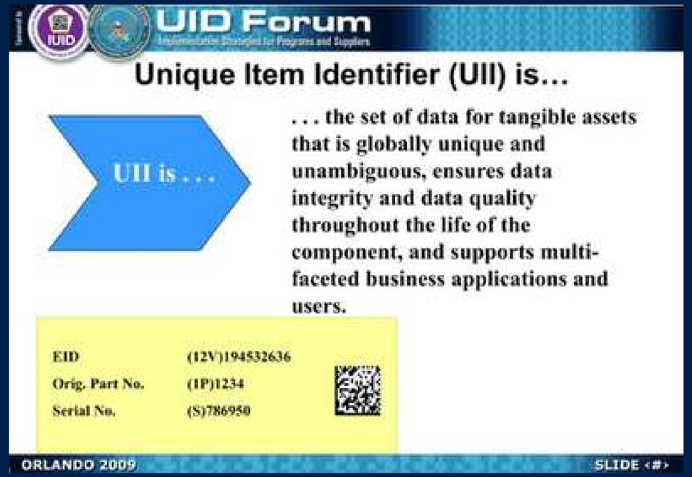
The DateTime feature provides the exact date and time of each recorded traffic count, allowing for detailed temporal analysis.

Junctions

The Junctions feature indicates the specific intersection where the traffic count took place, helping us understand traffic flow in different areas.

Vehicles

The Vehicles feature records the number of automobiles passing through the intersections in each given hour, offering insights into traffic volume.



ID

The The above 3 features stored in the form of a number.

Time series Data

Time series data is a collection of observations or measurements recorded or collected over time.

Time series analysis involves studying and modeling patterns, trends, and variations in data over time.

This type of analysis can be particularly relevant in domains like traffic monitoring

Data Preprocessing

Data Preprocessing

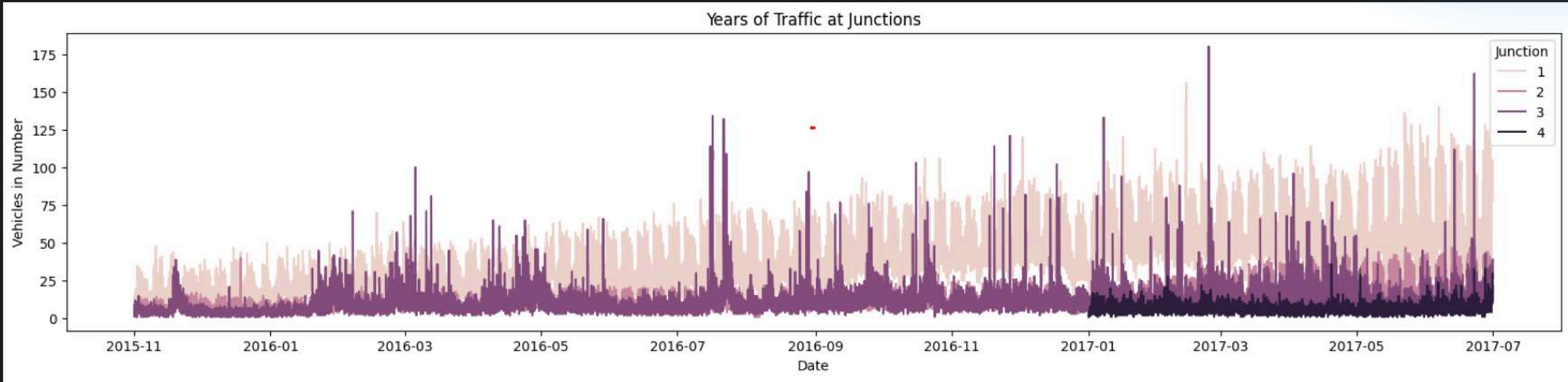
Check for any null values

split the DateTime column into year,month, date etc

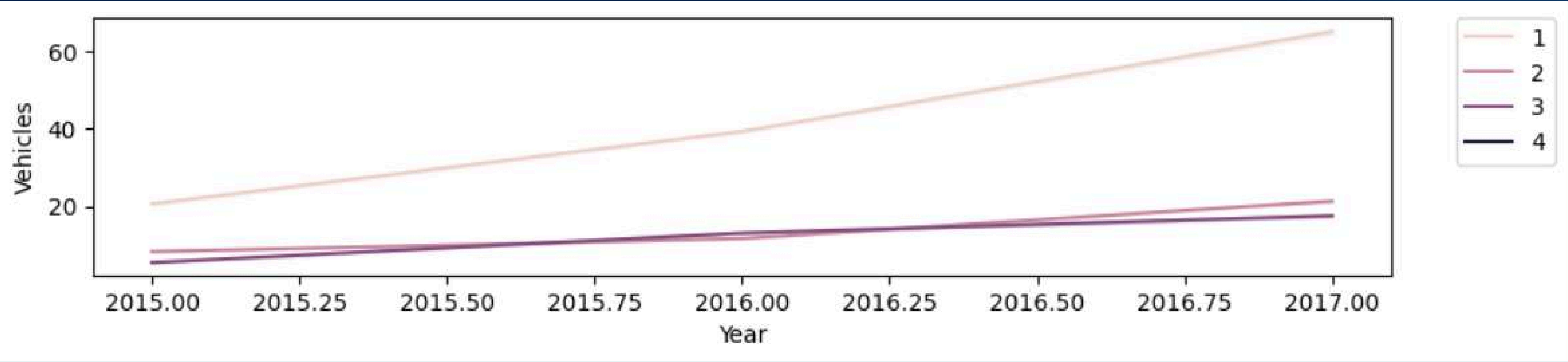
Z-Score Normalization

Analiysing the Data

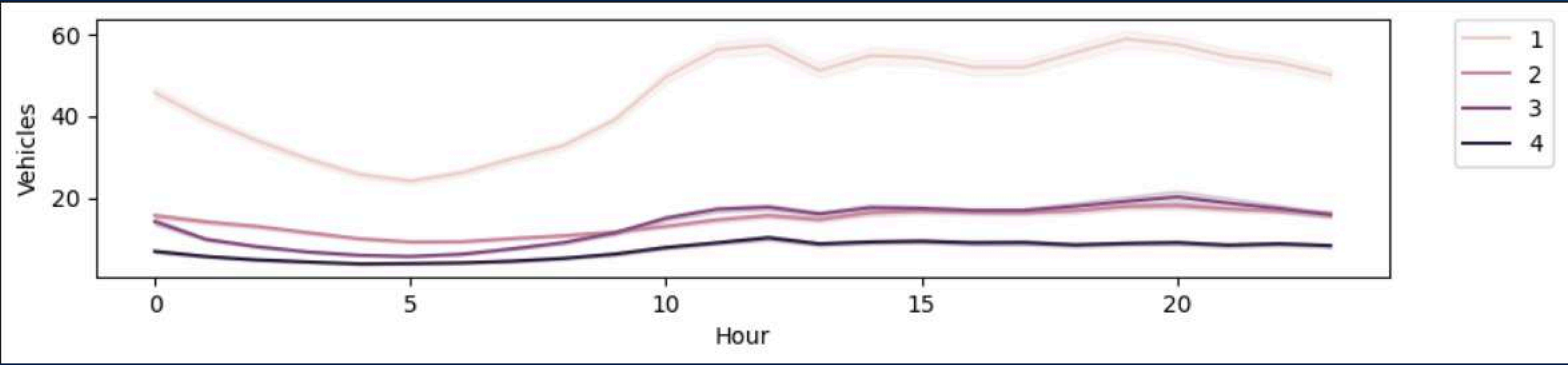
Vehicles at different Junctions



Vehicles at Junction over Time



Vehicles at Junction over Time

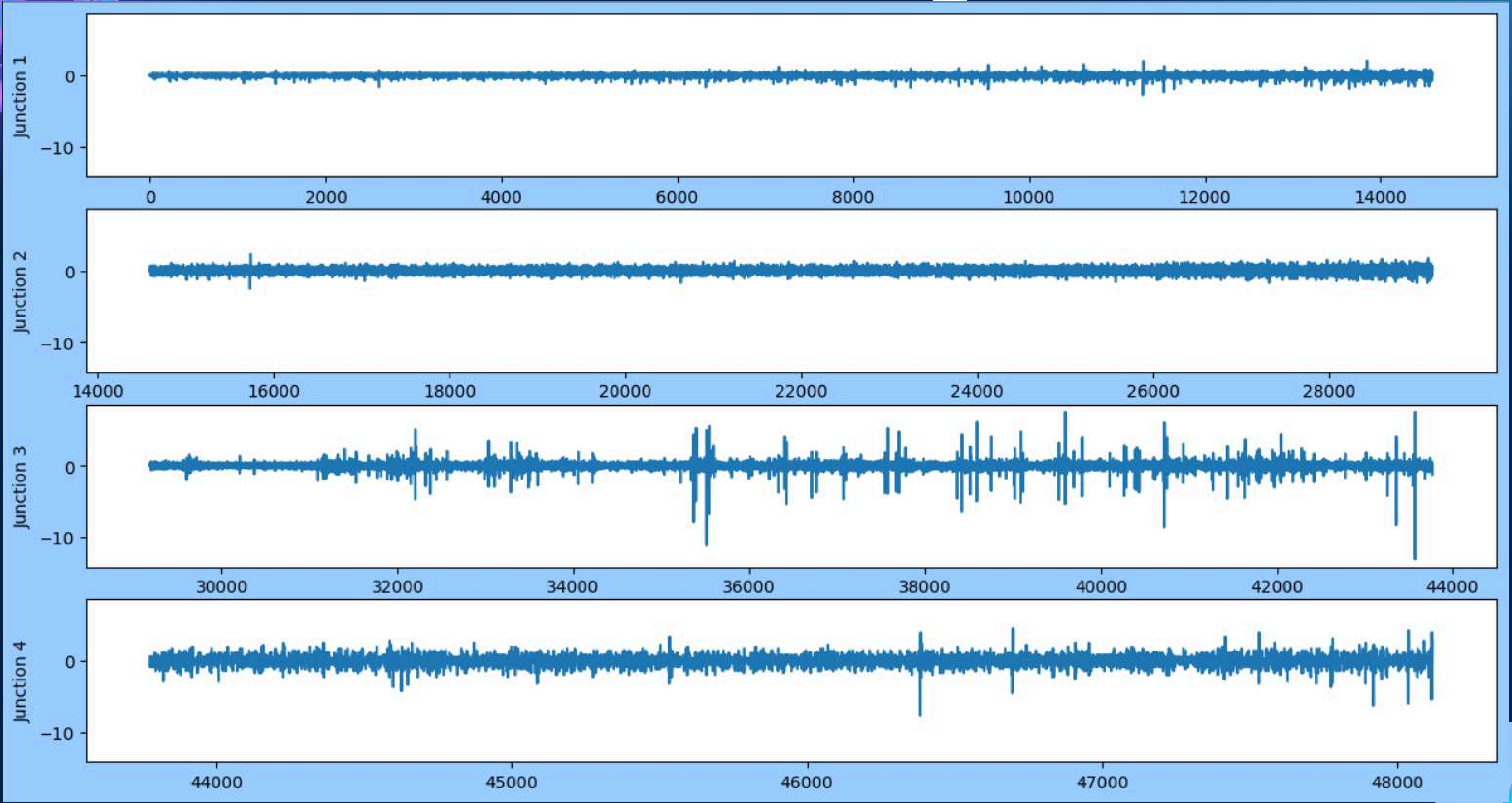


Differencing

Differencing is a common technique used in time series analysis to stabilize the mean and eliminate trends or seasonality.

The basic idea is to compute the difference between consecutive observations at a specific lag or interval

Differencing



Data Splitting

the null Values generated during the differencing were dropped

Data was split to Training and testing set using the usual Train_test_split function from Sklearn

Why Not RNNs?

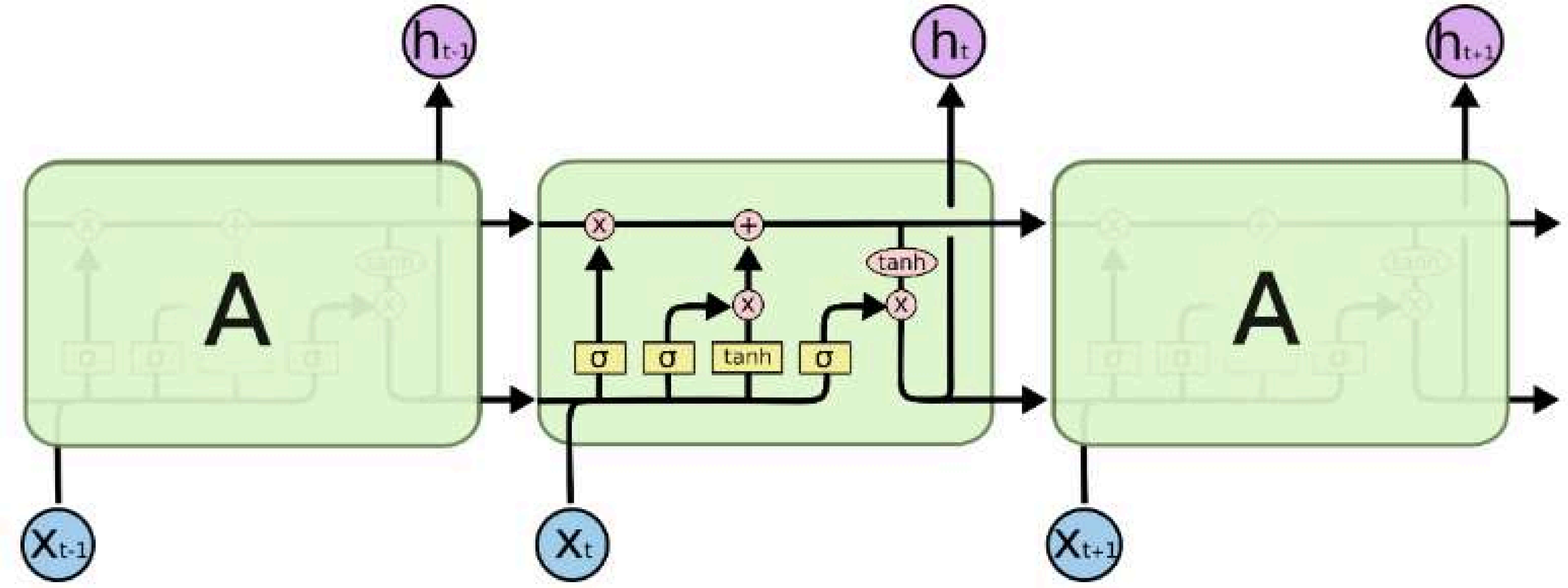
In traditional RNNs, during backpropagation, the gradients can become extremely small over long sequences, leading to vanishing gradients.

Conversely, RNNs can also suffer from exploding gradients, where the gradients become very large during training.

Traditional RNNs struggle to capture and remember information over long sequences.

Model Build

Long Short Term Memory (LSTM)



The repeating module in an LSTM contains four interacting layers.

Why LSTMs?

LSTMs are designed to store memory from a long time to remember information for long periods of time, and they do this by using a set of "gates" that control the flow of information

LSTMs are well-suited for capturing dependencies and patterns in sequential data.

LSTMs can capture non-linear relationships and complex temporal patterns in data.

Hyperparameter Tuning

The process of finding the optimal set of hyperparameters for a machine learning model is known as hyperparameter tuning

Model Performance

Avoiding Overfitting or Underfitting

Computational Efficiency

Hyperparameter Tuning

We created a function to find the best parameters by hyperparameter tuning using Random Search from scratch

The best parameters found were -

- Number of Layers: 2
- Number of Units in first layer: 110
- Batch Size: 60
- Dropout Rate: 0.2
- Learning Rate: 0.001

Fitting the Model with the Best Parameters

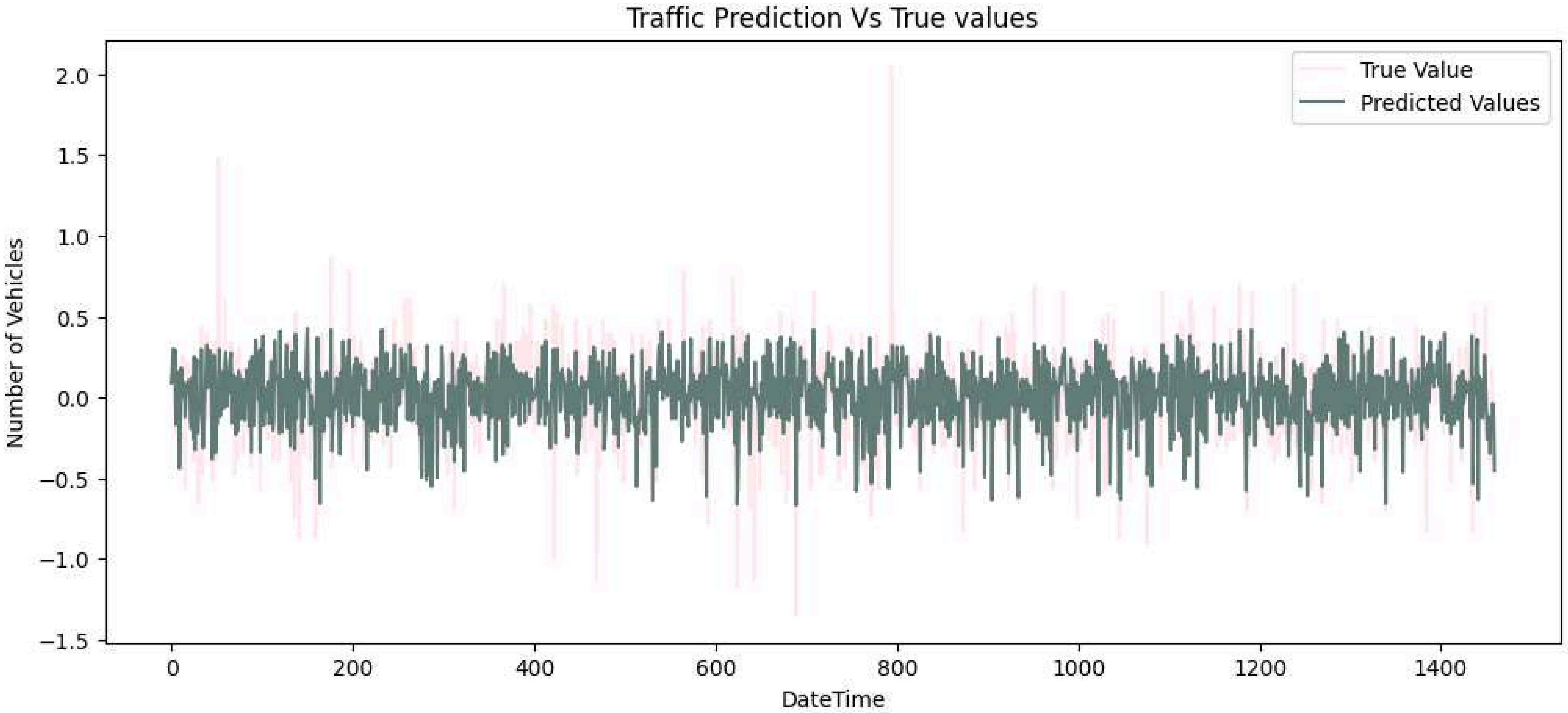
Results

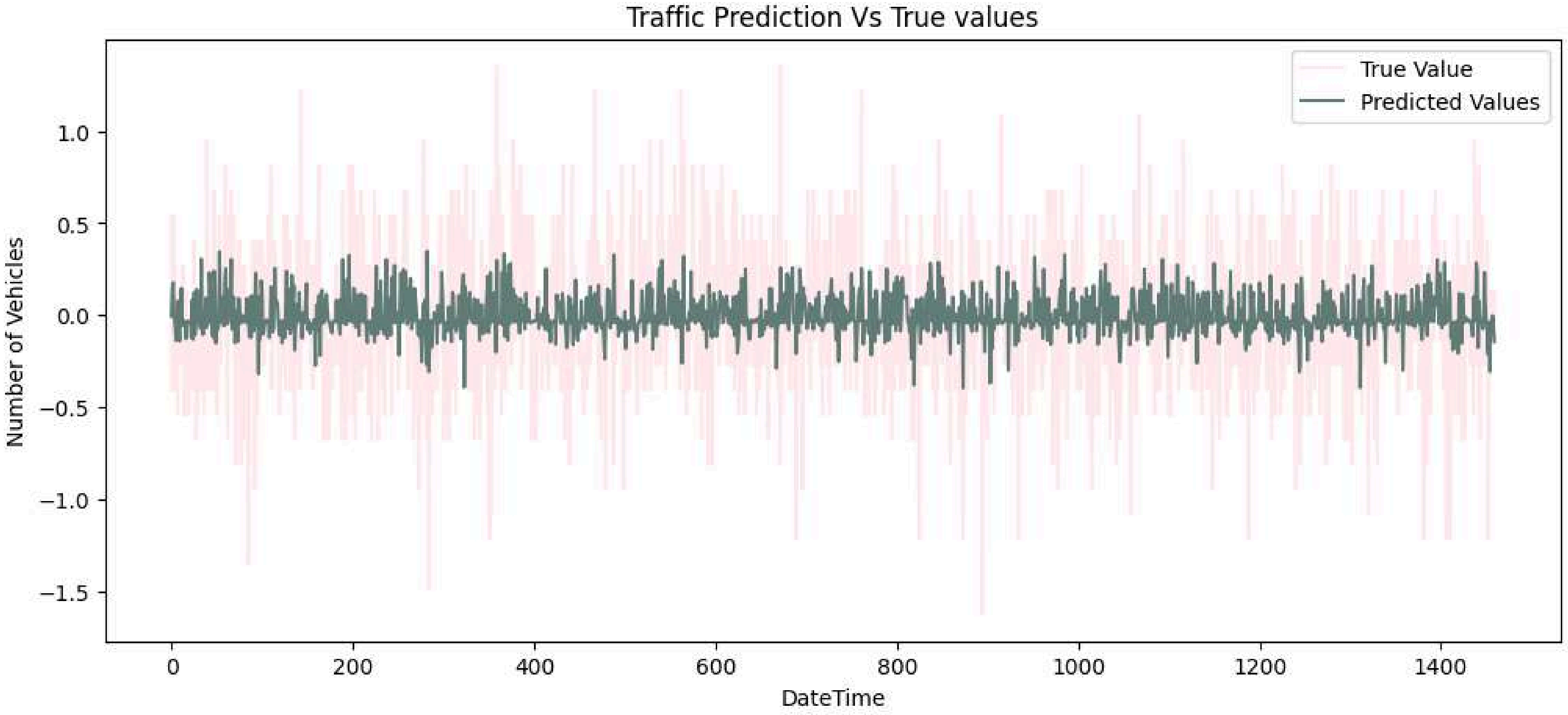
Junction 1

0.0434

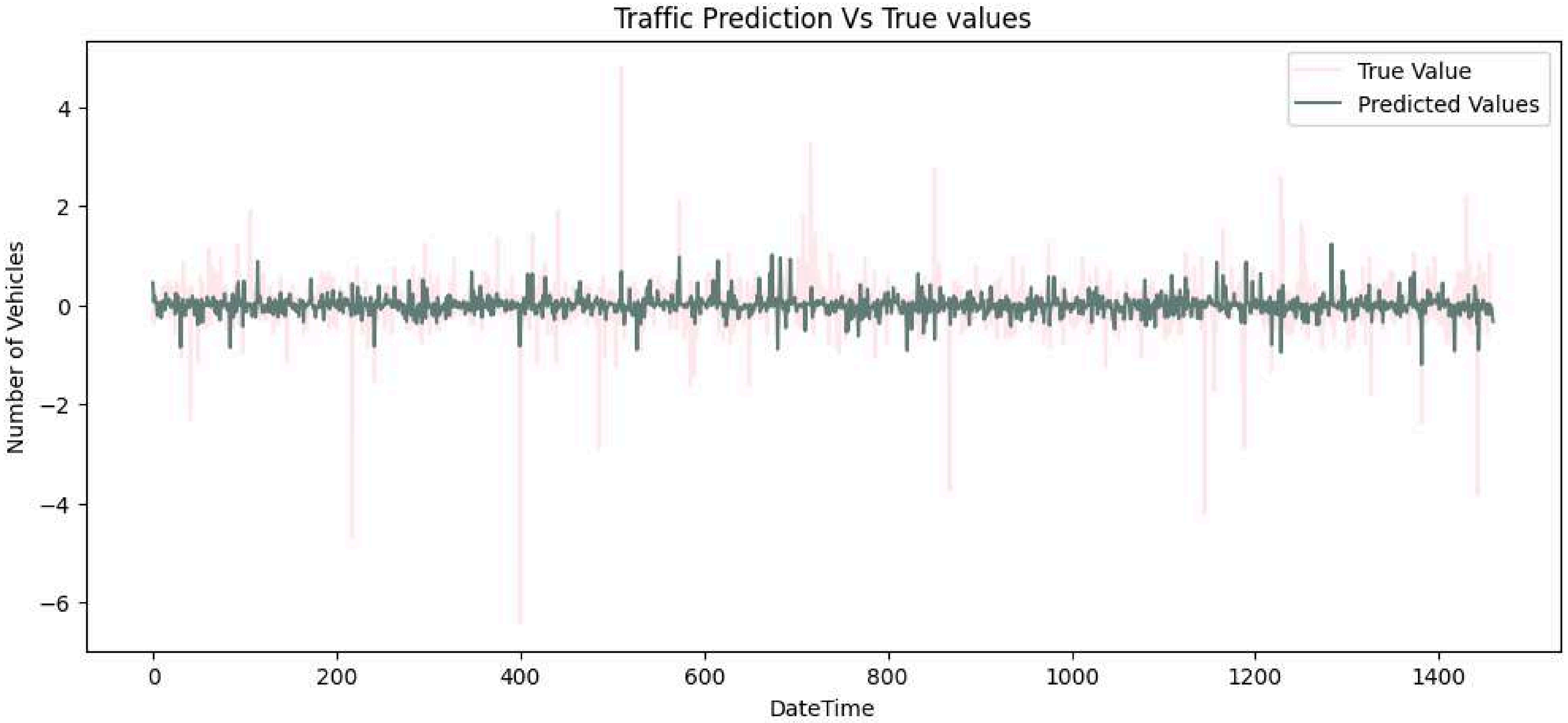
Junction 2

0.1583





Junction 3
0.2737



Junction 4
0.6570