Traffic volume prediction

Machine Learning | 30th April,2017

Team members:

Nitin Vasireddy	nxv160230
Pranathi Peri	pxp162530
Sangeeta Kadambala	sxk160731
Gautam Gunda	gxg161830

Table of Contents

Introduction 2

Problem Statement 2

Data Description 3

Methodology 5

1. Pre-processing techniques used 5

2. Solutions and Methods 7

3. Experimental Evaluation and Analysis 10

3.1 Using MSE and MAPE 10

3.2 Best Results: 10

Tools and Languages Used 10

Related Work 11

Contributions 11

Conclusion 11

Introduction

Tollgates form the major bottlenecks in traffic during rush hours. Long queues at tollgates during rush hour can overwhelm traffic management authorities. This can be avoided by following countermeasures like expediting the toll collection by opening more lanes during rush hours or streamlining the future traffic by adaptively tweaking traffic signals at upstream intersections. The prediction will allow the traffic management authorities to capitalize on big data & algorithms for fewer congestions at tollgates. These countermeasures can be deployed only when there is a reliable source of future rush hour prediction. For example, if heavy traffic is predicted in the next, then traffic regulators could open new lanes and/or divert traffic to other intersections.

As with most prediction problems dealing with critical traffic data, we needed to explore models that give hard figures as opposed to general categories of simply 'high' or 'low' volumes. We found neural network to give the best results, as it can model most accurately the heavily nonlinear data produced by the traffic and give actual volume figures as desired. While regression could achieve good results, neural net was noticeably better. This can be attributed to its strengths in dealing with nonlinear separation boundaries, strong correlation between features, ability to implicitly detect complex nonlinear relationships between dependent and independent variables, and its ability to detect all possible interactions between predictor variables.

Problem Statement

To predict average tollgate traffic volume at each tollgate.

For every 20-minute time window, we predict the entry and exit traffic volumes at tollgates a target area with tollgate numbers 1, 2 and 3 (fig 1). We predict the traffic volume of entries and exits to and from a tollgate separately.

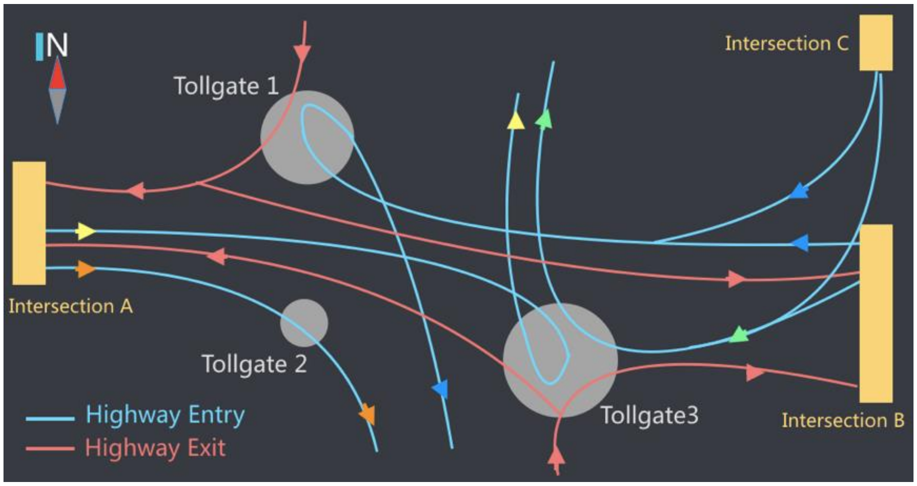


Fig 1. Road Network Topology of the Target Area

Data Description

The dataset we are using for traffic volume prediction is taken from KDD cup which can be found at KDD dataset Link .We are using two data sets "Weather data" and "Traffic Volume through the Tollgates". Traffic flow patterns vary due to different stochastic factors, such as weather conditions, time of the day, etc.

Following are the attributes and(or) features used for our model:

Field	Type	Description
Time	date time	The time when a vehicle passes the tollgate.
Tollgate_id	String	ID of the tollgate.
Direction	String	0: entry,1: exit
Vehicle model	Int	This number ranges from 0 to 7, which indicates the capacity of the vehicle (bigger the higher).
Has_etc	String	Does the vehicle use ETC (Electronic Toll Collection) device? 0: No ,1: Yes
Vehical_type	string	Vehicle type: 0-passenger vehicle,1-cargo vehicle
Date	Date	Date
Hour	Int	Hour
Pressure	Float	Air Pressure (hPa: Hundred Pa)
sea_pressure	Float	Sea Level Pressure (hPa: Hundred Pa)
Wind	Direction	Wind Direction (°)
Wind	Speed	Wind Speed (m/s)
Temperature	Float	Temperature (r)
rel_humidity	Float	Relative Humidity
Precipitation	Float	Precipitation (mm)

The data we were provided consisted of a training dataset of 543700 instances in volume data and 743 instances in weather data, with 15 features in all:

- 6 features define the information about the traffic volume:

  Time, Tollgate_id, Direction, Vehicle_model, Vehicle_type, Has_etc

- 9 features define the information about the weather:

  Date, Hour, Pressure, Sea_pressure, Wind_direction, Wind_speed, Temperature, Relative_humidity, Precipitation

Additionally, we were provided with a testing dataset 29442 instances in volume data and 57 instances in weather data, with the same features. We ran similar merging processes on this training data to test our model with.

We observed the plots of the features to eliminate any outliers or abnormality.

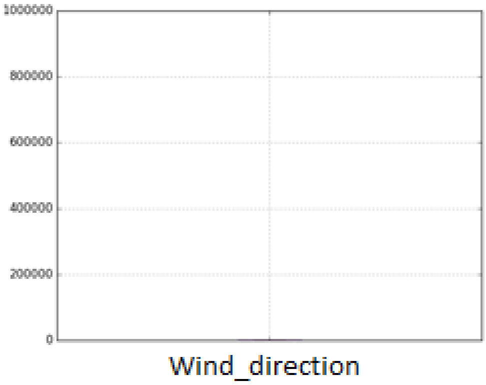


Fig 2. Boxplot of wind direction

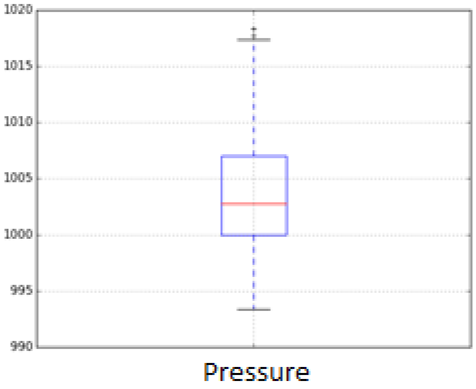


Fig 3. Boxplot of pressure

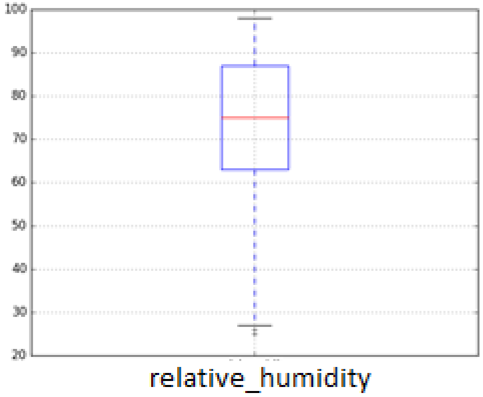


Fig 4. Boxplot of relative humidity

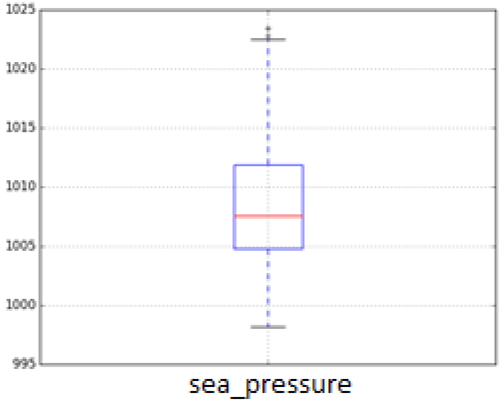


Fig 5. Boxplot of sea pressure

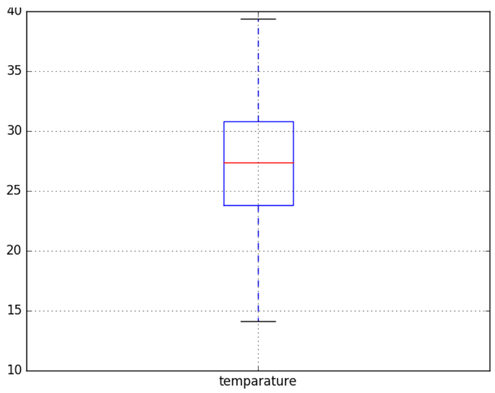


Fig 6. Boxplot of temperature

Methodology

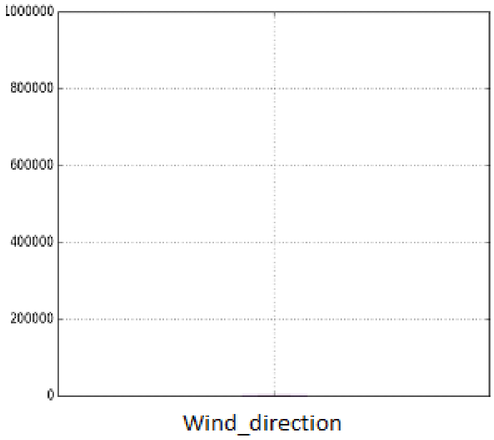
1. Pre-processing techniques used

- Decision to remove few features

Initially our dataset included 14 features but we removed a few of them as after a few experiments we found them not helpful in prediction. We removed 'Vehicle type', 'has etc', 'vehicle model' and used 'Traffic direction', 'tollgate id', 'precipitation', 'Wind direction', 'pressure', 'relative humidity', 'sea pressure', 'temperature' and 'wind speed'. We also used two derived features: 'Previous 20 minutes' volume' and 'previous day same time volume' for prediction.

- 4-nn for wind speed outliers

One of the features called 'wind direction' had outliers. We do not want to throw away the rows so, we used 4-nn to replace the outlying values using the values of four of its neighbors. We used time as the similarity metric.



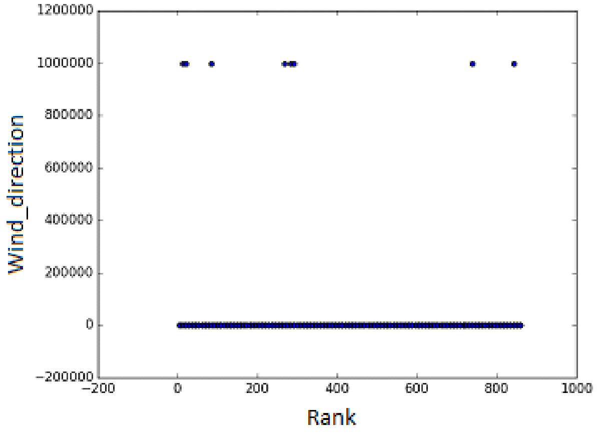


Fig 7. Wind direction plot before removing outliers

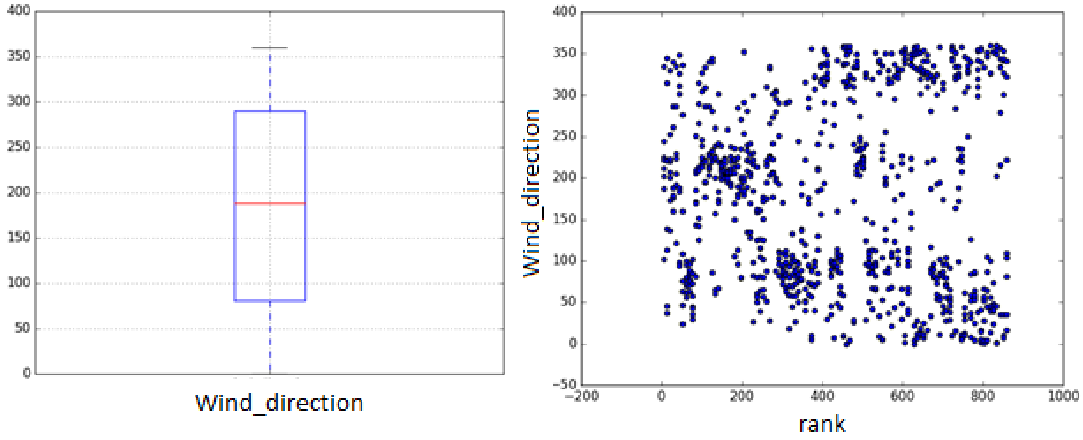
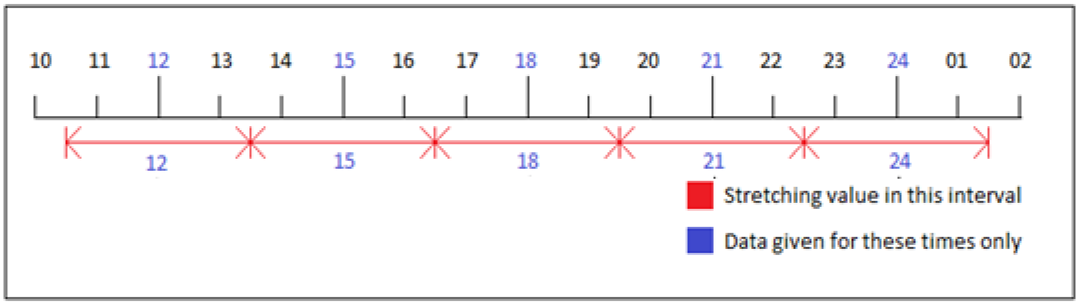


Fig 8. Wind direction plot after removing outliers

- Weather data continuous format by stretching out

The volume data and the weather data were not in sync. i.e. volume data was available for every minute and weather data was available for every 3 hours. This was resolved by smudging the data in a range of +/- 1.5 hours from the time the data was available. Below figure (fig. 4) shows this clearly:



10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
01
02
12
15
18
21
24
🟥 Stretching value in this interval
🟦 Data given for these times only

Fig 9. Stretching of weather data

- Mapped values for weather data to volume data

We stored the weather data in one RDD and volume data in another RDD. Using the date and hour as key, we merged the data in from both the RDDs and stored them in yet another RDD. Therefore, we ended up with a RDD containing the merged data.

- Standardization and Normalization of data

We used standardization on every feature data. The code for this was written manually using python to have better control. We normalized the output(volume) to match the output produced by neural nets.

2. Solutions and Methods

We predicted the traffic data using neural net by skewing the data appropriately for the required time of prediction. We employ a skewing of data so that we can couple all the required lags for any datum in the RDD.

Skewing involves the following:

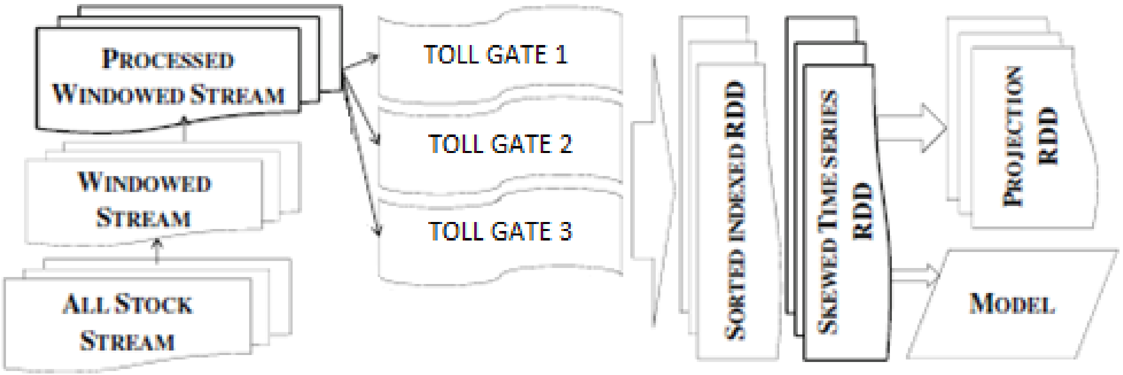


Fig 10. Steps involved to get skewed data

How the data looks after skewing:

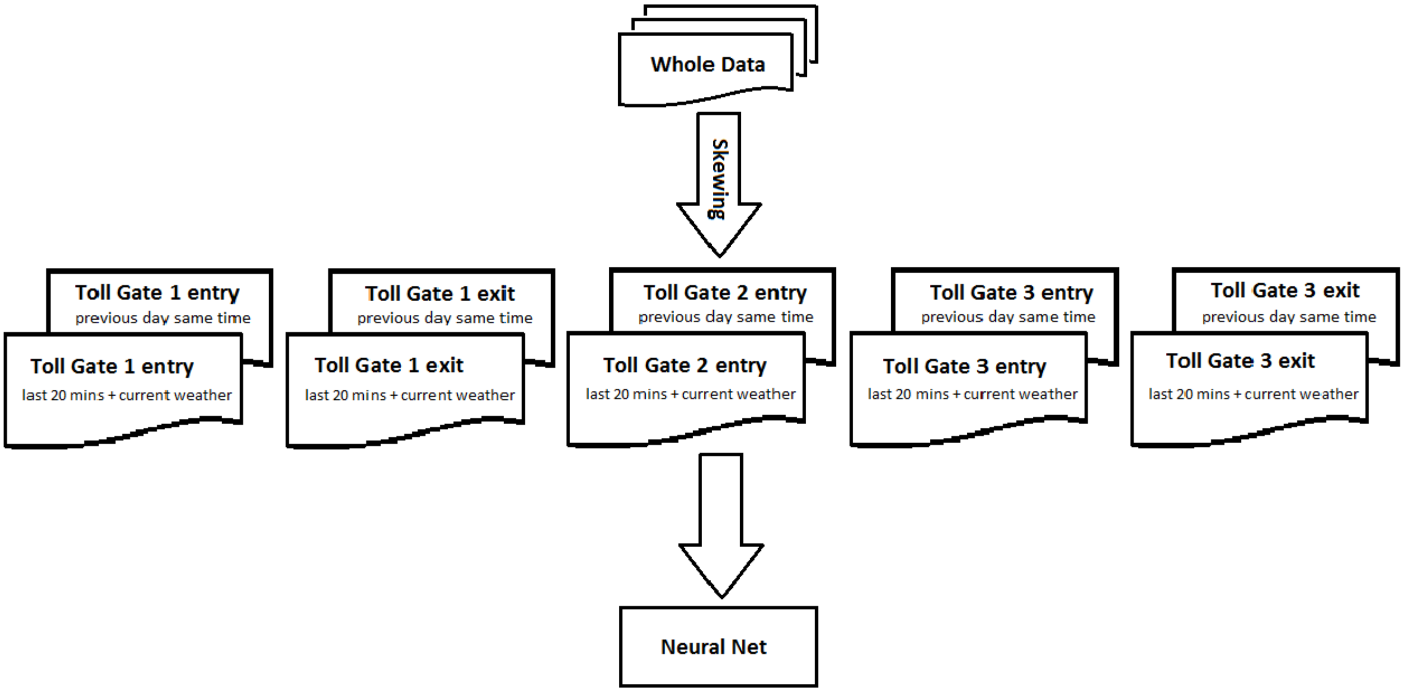
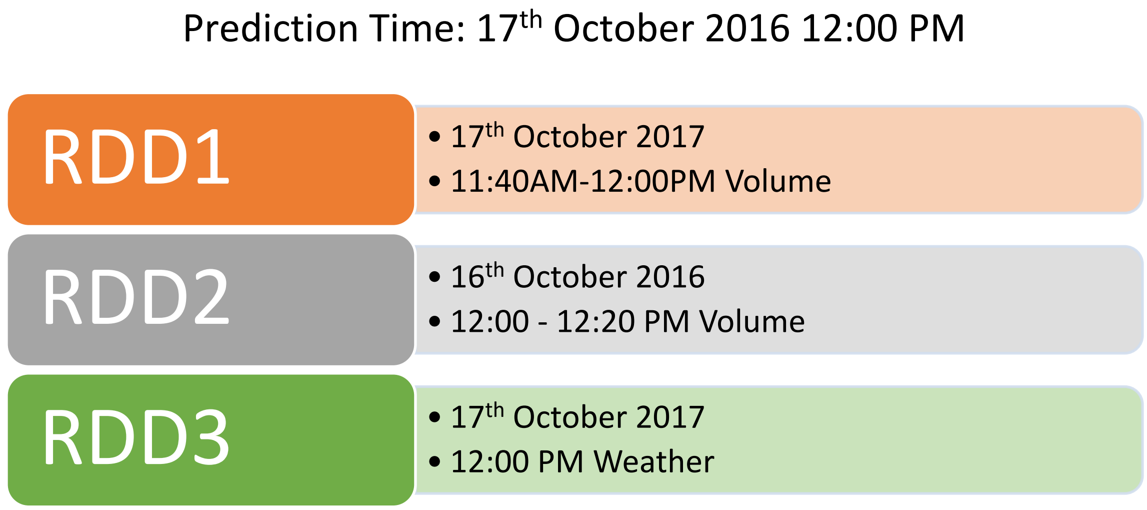


Fig 11. Skewed Data

We took the current day 20 minutes' data of the time before which we want to predict the volume. These data are stored in one RDD. In another RDD we store data of the same time (the time when we want to predict data) of the previous day. Example is shown below (fig 12).



Prediction Time: 17th October 2016 12:00 PM

RDD1
- 17th October 2017
- 11:40AM-12:00PM Volume

RDD2
- 16th October 2016
- 12:00 - 12:20 PM Volume

RDD3
- 17th October 2017
- 12:00 PM Weather

Fig 12. Distribution of data in RDD's for each toll gate entry & exit

The subset was chosen by sliding the data according to the window size of 20 mins. We tried 3 different activation functions all of them being sigmoid functions which return a continuous value:

1. Logistic or soft step: $f(x) = \frac{1}{1+e^{-x}}$
2. Tanh: $f(x) = \frac{2}{1+e^{-2x}} - 1$
3. ArcTan: $f(x) = tan^{-1}(x)$

The final output thus obtained represents the traffic volume at the desired interval. Below is a visual representation of our model:

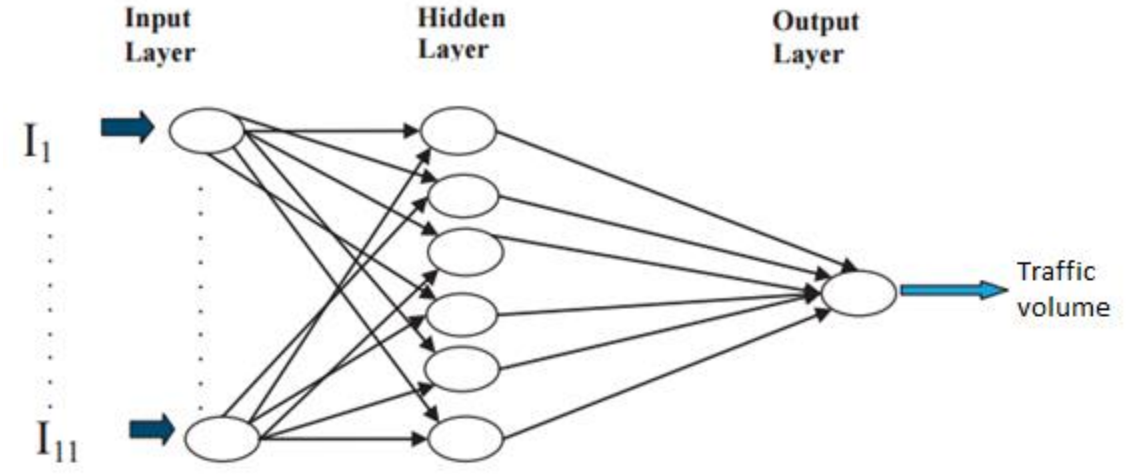


Fig 13. Neural Net Model

If C is the number of tollgate-direction pairs (1-entry, 1-exit, 2-entry, 3-entry and 3-exit), T be the number of time windows in the testing period, and fct and pct be the actual and predicted traffic volume for a specific tollgate-direction pair c during time window t. The MAPE for traffic volume prediction is defined as:

$$MAPE = \frac{1}{C}\sum_{c=1}^{C}\frac{1}{T}\sum_{t=1}^{T}\left|\frac{f_{ct}-p_{ct}}{f_{ct}}\right|$$

If C is the total number of instances in the tollgate-direction pairs the MSE for the same is defined as:

$$MSE = \frac{1}{2C}\sum_{c=1}^{C}(f_{ct}-p_{ct})^2$$

3. Experimental Evaluation and Analysis

3.1 Using MSE and MAPE

ACTIVATION FUNCTION	HIDDEN LAYERS	TRAIN ERROR	CROSS VALIDATION ERROR	TEST ERROR (MSE)	TEST ERROR (MAPE)
tanh	2(3,3)	0.015472	0.014092	0.041458	10.9241
tanh	3(3,3,3)	0.014978	0.013479	0.044819	10.4526
tanh	3(2,2,2)	0.014742	0.014743	0.045824	9.5962
tanh	16(3,3,3,3..)	0.019977	0.013405	0.041021	9.8213
tanh	64(3,3,3..)	0.022271	0.013648	0.04902	9.8286
sigmoid	2(3,3)	0.015582	0.013662	0.0442	10.6342
sigmoid	3(3,3,3)	0.015366	0.013505	0.045868	10.73728
sigmoid	64(3,3,3...)	0.014934	0.014389	0.042386	9.4678
sigmoid	3(2,2,2)	0.015246	0.014598	0.0419	10.7637
arctan	3(2,2,2)	0.015268	0.013946	0.048594	9.7745

Table 1. MSE and MAPE Results

3.2 Best Results:

We obtained the best results using the following parameters:

- tanh as the sigmoid function
- 16 hidden layers with 3 neurons in each

We first tested by doing a 80/20 split of training data which gave us an MSPE error of 0.013405. We then tried our model on the derived test data which gave an MSPE error of 0.041021.

Tools and Languages Used

To construct the model, we will use Spark engine with python. We used Python to code the neural net. We also used Hadoop distributed file system to store the data for use. We used:

- **pyspark.sql** module for creating DataFrame, register DataFrame as tables, execute SQL over tables.
- **matplotlib.pyplot** to visualize data.

Related Work

Related problems that we encountered in our research focused on revenue or stock prediction using machine learning methods, most specifically regression type problems. Following are the research papers which we referred to carry out our experiments on the problem statement:

[1] Estimating Traffic Intensity at Toll Gates Using Queueing Networks ((IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 5, No. 4, 2014)

[2] Traffic Congestion Analysis Using Highway O-D Tollgate Data(School of Electronics Engineering and Computer Science Peking University Beijing, China)

[3] Bulk Price Forecasting Using Spark over NSE Data Set(Vijay Krishna Menon1(&), Nithin Chekravarthi Vasireddy, Sai Aswin Jami, Viswa Teja Naveen Pedamallu, Varsha Sureshkumar, and K.P. Soman1)

[4] Short term traffic flow prediction for a non urban highway using Artificial Neural Network ((ELSEVIER)Kranti Kumara, M. Paridab, V.K. Katiyarc)

Contributions

Nithin & Pranathi: Worked on the 4-nn and stretching phases of Preprocessing, Coded the neural net and designed the skewing of data for feeding into neural net. Performed experiments with tanh as the activation function. Helped in writing the final report.

Sangeeta & Gautam: Worked on the mapping and standardization & normalization phase of preprocessing, helped in coding of neural net and design of the data skewing. Performed experiments with logistic (soft step) as the activation function. Wrote the final report.

Conclusion

By knowing the peak traffic hours, the traffic authorities can effectively manage the traffic at toll gates during that time. This way the major bottleneck in the traffic can be decreased to quite an extent which resolves the congestion of traffic at toll gates.

We provide a solution for a specific target area that has three toll gates. But this can be extended to predict the traffic volumes at tollgates in larger areas like cities or states where the toll gates will be represented by nodes in a graph. This solution can also be extended to control the traffic flow at every signal.