Saint Peter's UNIVERSITY

Assignment 10 – Draft Manuscript

Vehicle Traffic, Provided by City of Aarhus in Denmark

Presented by: Amit Pandey

Guided by: Professor Sylvain Jaume

Course: DS670 - Capstone

Manuscript

We loaded the data into Zeppelin to clean and analyze it with the idea of stacking the data into the framework and clean it/investigate it. The idea was to invest as much energy as it could to set up the data into a strong base of a domain that could be valuable and simpler to do the model. This way it would be easier to analyze it simply. Thus, in view of that, the data initially has been stacked into Spark, which did not work out as efficient language for the data and the purpose of our model. We, therefore, changed it over the data outline into a SQL as it simple to deal with excessively numerous records. This turned out to be great to just analyze the data. However, we still had to deal with a prediction and analysis part. In this way, we inferred that SQL won't work, therefore we have attempted to stack the data into a PySpark, which is a competitor language amongst Spark and Python, and that where the PySpark originated from. This was a decent decision notwithstanding, because of the absence of data and information in how to deal with the programming part on the grounds that the linguistic structure must conform to both language Spark and PySpark, and because of the long run time that we have confronted on doing the stacking and the analysis, then we have chosen to do only python for loading, clean, analysis and prediction the data and the model.

1. **Collect and download the datasets into one folder.**
   Based on the data provided for Aarhus, we will download and collect all the datasets in one folder. As the data is available in the raw (CSV) and semantically annotated format using the Citypulse information model, we will use the raw (CSV) format to process and analyze the ata.

2. **Load data and combine all data sets in Zeppelin.**
   Using Zippelin, we will load and combine all the files (449) in the datasets to analyze the information. We have utilized numerous languages to the analysis. We have utilized: Spark, SQL PySpark and Python. The languages that we used to fit to the data were both PySpark and Python, in any case we have chosen to run with Python as it was anything but difficult to deal with a sentence structure for one language as opposed to taking care of the structure for two languages. What's more, the outcome for that part, the run time stacking into PySpark took twofold the run time stacking into Python. As Python was more proficiency in this part, why have chosen to finish with python.

   Due to the vast data set we have chosen to take a major measure of the data to test and fabricate the model on it, as taking every one of the data won't be conceivable because of the capacities that we have. It is more than 29 million records and that without mapping and other data.

3. **Clean the data.**

We'll did an analysis to see if the data needs to be cleaned and if there are any errors that may corrupt our analysis. We also wanted to make sure that the null values are set to zero and take care of missing values. We checked for common errors like: missing values, corrupted values, data range errors, etc. We looked through the file rows and columns and sample test values to see if the values make reasonable sense.

We have utilized the cleaning techniques to clean the data of N/A by either supplanting or barring from the example or notwithstanding expelling from the data if the record has an excessive number of missing qualities. What's more, it can't withdraw or supplanted. Likewise, we have examined the data to ensure that the data is not tainted for instance if the data has an immense out of the sudden spike on the chart that implies the data is ruined. Or, on the other hand if the data has some wrong esteem, for example, a period of 99 that implies it is ruined as there are no 99 clock time and the time goes just to 24 hours.

4. **Handling the outliers:**

In order to handle the outliers, what we have done here is we have rejected the anomaly from the data, as this will influence the data as we didn't evacuate it or erase it, as this can be utilized on further analysis. For instance, utilizing the outliers to anticipate the possibility of having an accident on this street. Nonetheless, on this case we truly need to ensure that the outliers are a result of an accident not due to absent or undermined values, or might be a result of street work in the city which cannot be utilized at this stage, with the goal that we have chosen to avoid the outliers from the data and account for the typical data to improve a creation.

5. **Split Datasets into Training (80%) and Testing Data (20%).**

As we are going to do some prediction, it we split our combined dataset into two different datasets: Training (80%) and Testing (20%). The Training dataset was used to build and test different statistical mode and the Testing dataset was used to evaluate (cross-validate) our model and assumption. We did the splitting and create the Test partition to provide us with a fair assessment of our prediction model build.

There was no single method for selecting the extent of training and testing data. Some people use 90/10 and some prefer 80/20. In any case, doing as such can cause bias the classification results. For selecting the right split, we used "N-Fold cross validation" and/or "K-fold cross validation. This will take out any bias out of our assumptions.

6. **Mapping the data:**
   We have mapped the data into the Meta Data to have more data about the record that the framework caught and comprehend the data

7. **Final discovery:**
   We did the time-series analysis using ARIMA (Autoregressive integrated moving average) model. ARIMA is a generalization of an autoregressive moving average (ARMA) model, both of these models are fitted to time series data either to better understand the data or to predict future points in the series (forecasting). ARIMA models are applied in some cases where data show evidence of non-stationarity, where an initial differencing step can be applied one or more times to eliminate the non-stationarity.

   We predicted the next 5 hours (data points) for the time-series using ARIMA and compared them with the actual. Here's the result of the model:

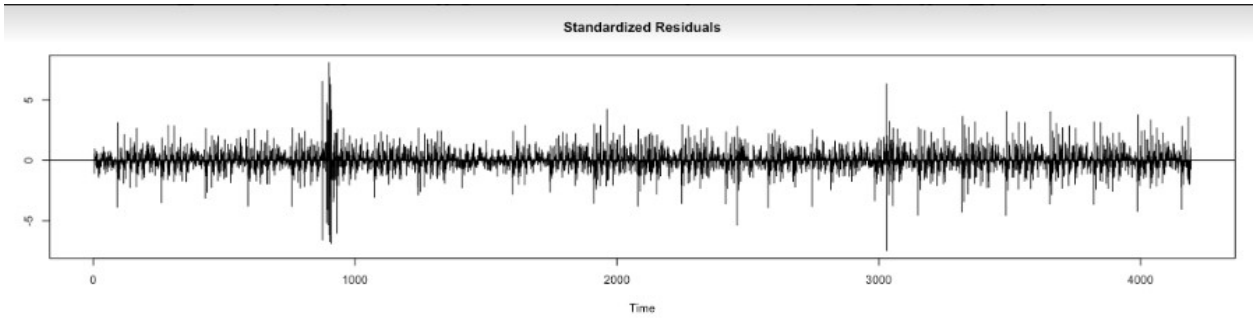

Chart 1:For the existing data, this is the residuals of the predicted fits.

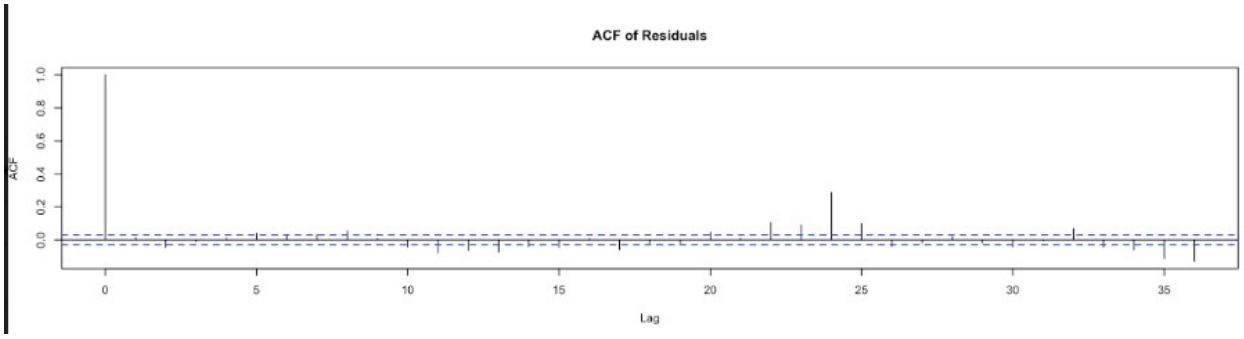

Chart 2: Autocorrelation function (ACF) of residuals shows the correlation of the residuals (as a time series) with its own lags.