**Solution for Traffic Flow Prediction**

**Problem Statement**

The goal for this dataset is to forecast the spatio-temporal traffic volume based on the historical traffic volume and other features in neighbouring locations. Specifically, the traffic volume is measured every 15 minutes at 36 sensor locations along two major highways in Northern Virginia/Washington D.C. capital region. The 47 features include: 1) the historical sequence of traffic volume sensed during the 10 most recent sample points (10 features), 2) weekday (7 features), 3) hour of day (24 features), 4) road direction (4 features), 5) number of lanes (1 feature), and 6) name of the road (1 feature). The goal is to predict the traffic volume 15 minutes into the future for all sensor locations. With a given road network, we know the spatial connectivity between sensor locations.

**Data**

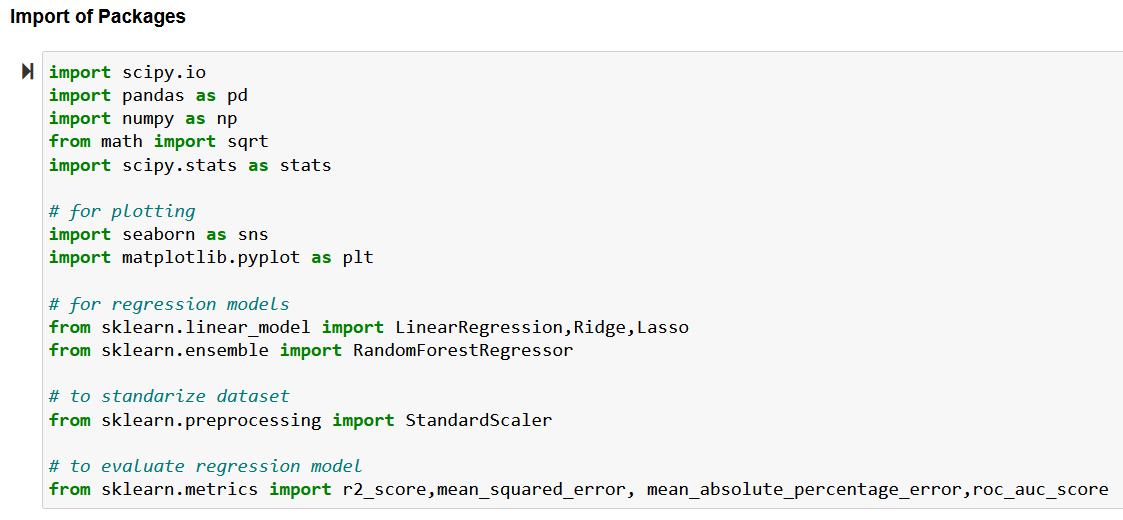
Data (in .mat format) to be used for this problem can be found in below link:

mason.gmu.edu/~lzhao9/pages/dataset\_pages/datasets/spatial/traffic\_dataset.mat

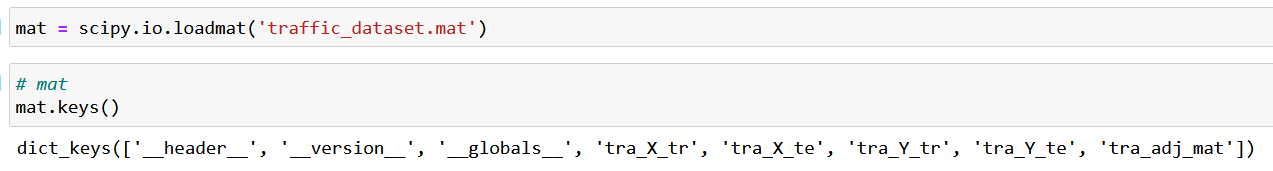
Below are further details of this data:

|  |  |  |  |
| --- | --- | --- | --- |
| **Variable Name** | **Type** | **Size** | **Description** |
| tra\_X\_te | array of matrices | 1\*840 | test set input data: traffic indices for 840 contiguous quarter-hours  each element is a 36\*48 matrix: 36 spatial locations by 48 features |
| tra\_X\_tr | array of matrices | 1\*1261 | training set input data: traffic indices for 1261 contiouous quarter-hours  each element is a 36\*48 matrix: 36 spatial locations by 48 features |
| tra\_Y\_te | array of matrices | 36\*840 | test set output data: traffic flowfor 36 locations in 840 contiguous quarter-hours from 2017-01-02 00:00 |
| tra\_Y\_tr | array of matrices | 36\*1261 | training set output data: traffic flowfor 36 locations in 1261 contiguous quarter-hours until 2017-02-01 00:15 |
| tra\_adj\_mat | squared matrix | 36\*36 | adjacency matrix denoting the spatial connectivity of traffic network among 36 locations |

**Approach, Analysis & Solution**

**Import of Required Python Packages**:🡪 

**Loading of DataSets**:🡪



**Formatting of DataSets**:🡪

Original Dataset in matlab format is first converted to lists followed by Data frame as below:

|  |  |  |  |
| --- | --- | --- | --- |
| Original Dataset  Components of ‘traffic\_dataset.mat’ | Python List | Data Frame |  |
| tra\_X\_tr | list\_tra\_X\_tr | df\_train\_ds (45396\*51) | quarter\_hour, spatial\_location, traffic\_flow are added to 48 features(feature\_0 to feature\_47) |
| tra\_Y\_tr | list\_tra\_Y\_tr |
| tra\_X\_te | list\_tra\_X\_te | df\_test\_ds  (30240\*51) |
| tra\_Y\_te | list\_tra\_Y\_te |
| tra\_adj\_mat | list\_tra\_adj\_mat |  |  |

**Feature Engineering**:🡪

Types of variables:🡪

There are total of 51 numeric variables of which 49 are floating & 2 are integers.

They are further classified as below:

|  |  |  |
| --- | --- | --- |
| **Type of variable** | **Feature** | **Example** |
| Discrete Variables | those that contain a finite and small number ( < 20) of distinct values | There are 38 discrete variables from feature\_10 till feature\_47  Most of they having allowed values as [0, 1]. |
| Continuous Variables | those that contain more than a fixed number ( >= 20) of distinct values | There are 13 continuous variables from feature\_0 till feature\_09 & quarter\_hour, spatial\_location, traffic\_flow. |

**Analysis of Variables**:🡪(with help of plotting variables wrt mean of target variable(traffic\_flow)

|  |  |  |  |
| --- | --- | --- | --- |
| **Set of variables** | **Features** | **Example** | Trends & Properties |
| historical sequence of traffic volume | sensed during 10 most recent sample points (10 features) | feature\_0 to feature\_9 | It's clear that strong correlation exists among these 10 historical sequence of traffic volume (feature\_0 to 9). Approach to combine (mean) these 10 features to a single feature can be followed. |
|  | | | |
| week day | 7 features for each of 7 days of a week | feature\_10 to feature\_16 | 5 of week day (7 features) tend to affect traffic flow in similar manner to some small extent. May be these 5 days are weekdays when traffic pressure is high & remaining 2 days are weekends when traffic pressure is low due to holidays. |
|  | | | |
| hour of day | 24 features representing each of 24 hours | feature\_17 to feature\_41 (excluding feature\_39) | From above plot, no trend can be found for 24 Hours (feature\_17 to 40).  feature\_39 (23 rd Hour) also contains value of 3, 4 & 5 as compared to 0,1 for other hours. Value of 3, 4 & 5 (values are absurd) might be outlier/noise for feature\_39. We can drop feature\_39 as it contains noise & does not signify hour feature. Instead, we will include feature\_41 as hour feature. |
|  | | | |
| road direction | 4 features | feature\_42 to feature\_45 | No trend/relation found among 4 road direction. |
|  | | | |
| number of lanes | 1 feature | feature\_46 | Distinct Values: 2 [0. 1.]  Data Type: float64 |
| # name of road | 1 feature | 'feature\_47' | Distinct Values: 2 [0. 1.]  Data Type: float64 |
|  |  |  |  |

**Types of problems within the variables**:🡪

|  |  |
| --- | --- |
| Type | Observations |
| Missing Values | No of Columns with missing values = 0 |
| Duplicate Rows | No of Duplicate Rows = 0 |
| Outliers and distributions | Boxplot is used to visualise outliers in continuous variables & histograms to get an idea of its distribution. |
|  | Outliers in continuous variables:  The majority of the continuous variables seem to contain outliers. In addition, the majority of the variables are not normally distributed. As we are planning to build linear regression, we need to tackle these to improve the model performance. |
|  | Outliers in discrete variables:  Will call outliers those values that are present in less than 5 % of total observations. Most of discrete variables show values that are shared by a tiny proportion of total observations in dataset. |
|  |  |

**Linear Model Assumptions**:🡪

|  |  |  |
| --- | --- | --- |
| **Property** | **Significance** | **Observations** |
| Linearity | Mean values of the outcome variable for each increment of predictor(s) lie along a straight line. We evaluate linear assumption with plots. | all continuous dependent variables (except quarter\_hour, spatial location) are linearly dependent on target variable.  Not all discrete dependent variables are linearly dependent on target variables |
|  | | |
| No perfect multicollinearity | here should be no perfect linear relationship between two or more of the predictors. | As feature\_0 to feature\_9 are highly correlated among all continuous variables, we can derive only one feature (taking mean of them) & then drop 10 features from final dataset.  As no features are highly correlated among all discrete variables, we can keep all of them in final dataset |
|  | | |
| Normally Distributed Errors | Residuals are random, normally distributed with a mean of 0. | We see that residuals show a fairly normal distribution centred at 0. |
| Errors are not perfectly normally distributed. By visual inspection, there is some skew towards both ends with a few higher unusual values. | | |
| From Q-Q plot, we see more easily how the residuals deviate from the red line towards the ends of the distribution, thus, they are not completely normally distributed. | | |
| Homoscedasticity | Homoscedasticity implies that at each level of the predictor variable(s), the variance of the residual terms should be constant. So we need to plot the residuals against the variables. | The residuals seem fairly homogeneously distributed across the values of spatial\_location & quarter\_hour. |
|  | | |
|  |  |  |

**Fine Tuning of Training & Testing DataSet**:🡪



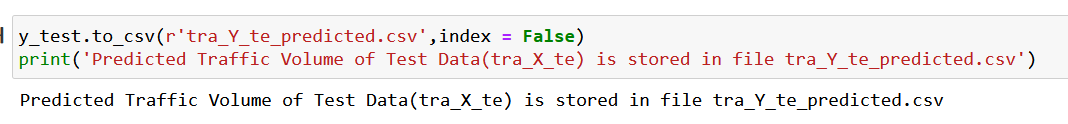
Feature Scaling:🡪 Scaled all features using Standardisation for linear models using below formula:

X\_new = (X - mean)/Std

**Regression Modelling**:🡪

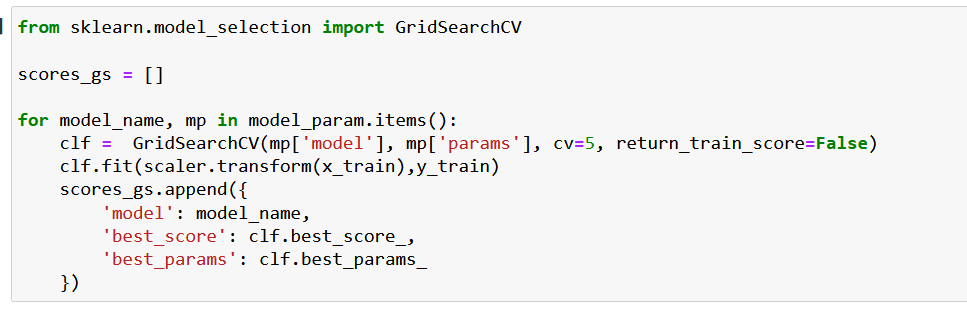
|  |  |
| --- | --- |
| **Model Type** | **Linear Regression** |
| Model Training & Prediction:  Performance: | |
| **Model Type** | **Ridge Regression** |
| Model Training & Prediction:  Performance:    Observations: Model performance can be optimized by fine-tuning hyper-parameter alpha. | |
| **Model Type** | **Lasso Regression** |
| Model Training & Prediction:  Performance:    Observations: Model performance can be optimized by fine-tuning hyper-parameter alpha. | |
| **Model Type** | **Random Forest** |
| Model Training & Prediction:  Performance:    Observations: Model performance can be optimized by fine-tuning hyper-parameter n\_estimators. | |

**Final File containing Predicted Traffic Volume**:🡪

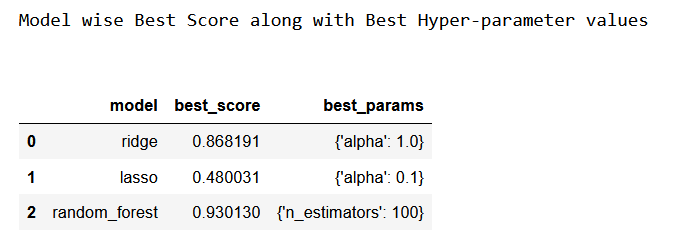


**Hyper-parameter Tuning**:🡪

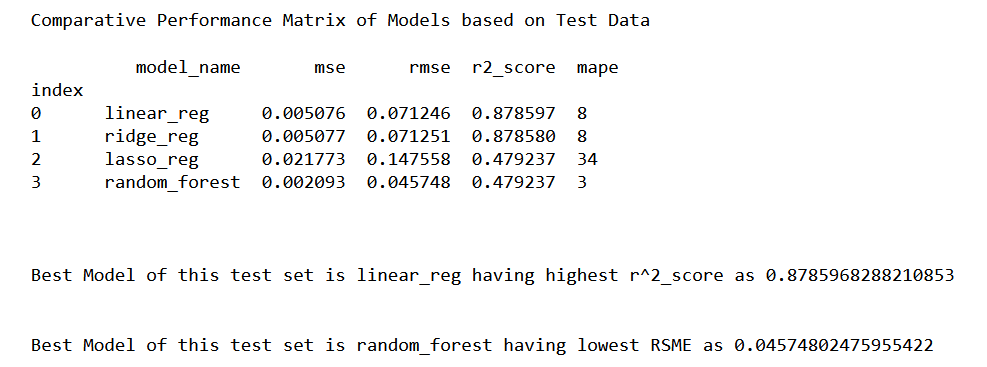
Declaration of Different Models along with their hyper-parameters & Set-up on Grid Search :

Summary of Models along with best Score & Hyper-parameters:



**Comparative Performance Matrix of Models**:🡪



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Feature Engineering + Performance Tuning** | **Performance Measures** | **Linear Regression** | **Ridge Regression** | **Lasso Regression** | **Random Forest** |
| Standard\_Scaling | mse,  rmse,  r^2 | 0.002064 0.045434 0.950629 | 0.002064 0.045434 0.950629 | 0.042102 0.205188  -0.006972 | 0.001468 0.038312  -0.006972 |
| Feature\_0 to feature\_9 is replaced by only feature\_0 + Standard\_Scaling | mse,  rmse,  r^2 | 0.009731 0.098644 0.767267 | 0.009731 0.098645 0.767263 | 0.042102  0.205188  -0.006972 | 0.003591 0.059925  -0.006972 |
| Feature\_0 to feature\_9 is replaced by its avg + Standard\_Scaling | mse,  rmse,  r^2 | 0.005044 0.071023 0.879356 | 0.005044 0.071019 0.879367 | 0.042102  0.205188  -0.006972 | 0.002066 0.045455  -0.006972 |
| Feature\_0 to feature\_9 is replaced by its avg + Standard\_Scaling + omission of feature\_39 | mse,  rmse,  r^2 | 0.005076 0.071246 0.878597 | 0.005077 0.071251 0.878580 | 0.042102  0.205188  -0.006972 | 0.002075 0.045556  -0.006972 |
| Hyper-parameter tuning on top of previous conditions | mse,  rmse,  r^2,  mape | 0.005076 0.071246 0.878597  8 | 0.005077 0.071251 0.878580  8 | 0.021773  0.147558  0.479237  34 | 0.002069 0.045490 0.479237  1 |

|  |  |  |  |
| --- | --- | --- | --- |
| Group No | Name | Student Email id | % Contribution |
| 74 | TANISH KHANNA | 2021sc04236@wilp.bits-pilani.ac.in | Equal (100%) |
| 74 | AVISHEK KUMAR | 2021sc04238@wilp.bits-pilani.ac.in | Equal (100%) |
| 74 | SOUMYADIPTA MAITI | 2021sc04237@wilp.bits-pilani.ac.in | Equal (100%) |