**INDEX**

[Exploratory data analysis 2](#_Toc19112359)

[Shape of train and test data 2](#_Toc19112360)

[i. Univariate analysis 3](#_Toc19112361)

[Fare\_amount 3](#_Toc19112362)

[Passenger\_count 4](#_Toc19112363)

[Pickup\_datetime 7](#_Toc19112364)

[Feature engineering 9](#_Toc19112365)

[Pickup\_latitude/pickup\_longitude/drop\_off\_latitude/dropoff\_longitude 10](#_Toc19112366)

[Model creation 16](#_Toc19112367)

[i. Normalization 16](#_Toc19112368)

[ii. Boxcox transformation 16](#_Toc19112369)

[Chi-square analysis 17](#_Toc19112370)

[Model 17](#_Toc19112371)

[Model Tuning 18](#_Toc19112372)

[i. RandomForestRegressor 18](#_Toc19112373)

[ii. KNN 18](#_Toc19112374)

[iii. Linear regression 20](#_Toc19112375)

[K folds 21](#_Toc19112376)

[Objects for runnning the algorithm 21](#_Toc19112377)

[Fitting and evaluating models 21](#_Toc19112378)

[Results 21](#_Toc19112379)

[LinearRegression results analysis 23](#_Toc19112380)

[RandomForest feature importance 23](#_Toc19112381)

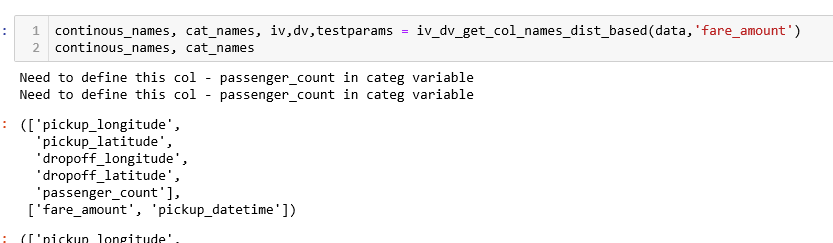
[Best model: 24](#_Toc19112382)

[Predicting the results 25](#_Toc19112383)

**REPORT :** This project is about predicting the fare of a cab ride, given the coordinates of pick/drop off, along with the pickup timestamp

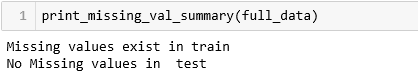
# Exploratory data analysis

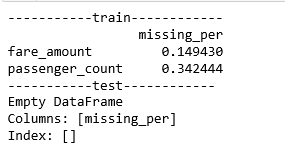
## Shape of train and test data



* Passenger\_count is given as numeric, but we will treat it as categorical, since a very few and discrete values exists
* Fare\_amount is numeric
* Pickup\_datetime is given as string –
* Had to remove a few mis formatted strings and convert to float
* Then parse it as a TimeStamp object and store

**Missing value analysis**

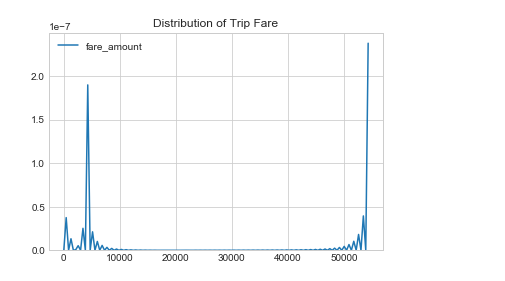




* Missing values in Fare\_amount are replaced with median, since it fares are heavily right skewed and median is not affected by outliers unlike mean
* Missing values in passenger\_count should be replaced by mode(since categorical variable), but then the result will be baised towards passenger count 1
  + Instead I checked the fare statistics in the subset dataframe where passenger count was missing and mapped out the records with same fare range and replaced the passenger count from those.
  + However the passenger count turned out to be 1 , even by this method

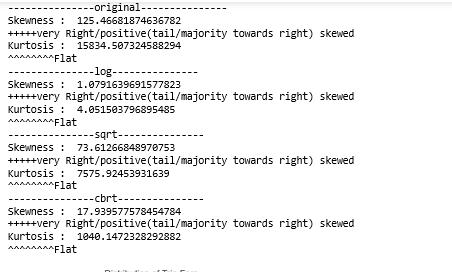
## Univariate analysis

### Fare\_amount



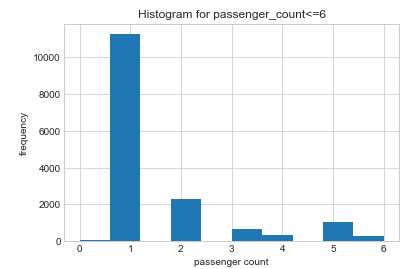
Fare\_Amount has outliers where fare is 40k,50k for the ride in city.

Below is the skew and kurtosis value for fare\_amount

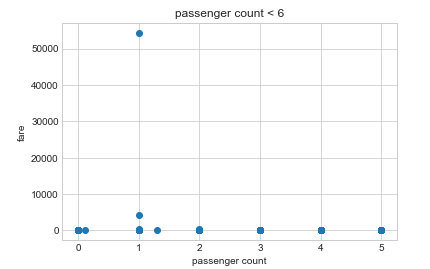


These can’t be explained by the predictors given to us, for eg for the same distance and same passenger\_Count, how can the fare be varying so highly? Unless there are other factors like

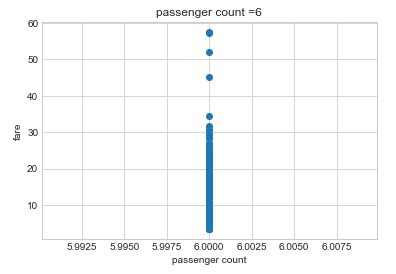
### Passenger\_count



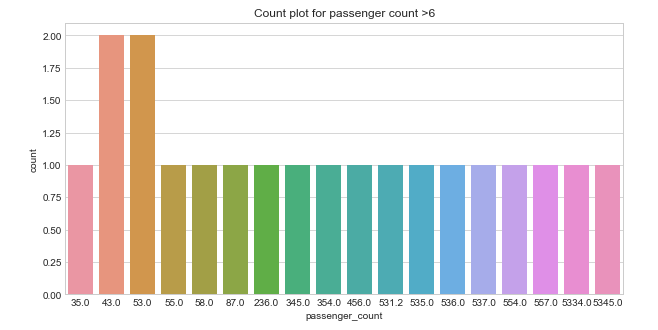
passenger\_Count =1 is the maximum occurring

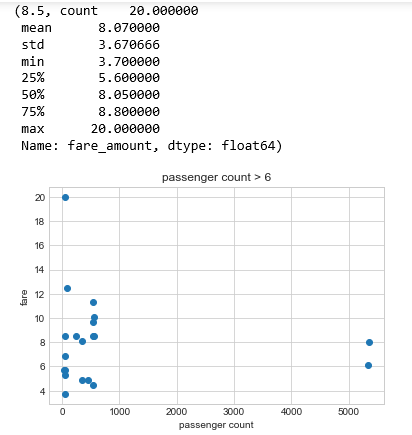


Fares are lying closely, except for a outlier (for passenger\_Count<=6)



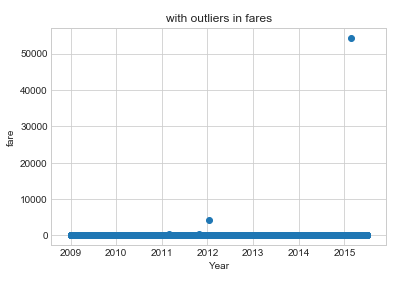
Few fares are high with passenger count=6



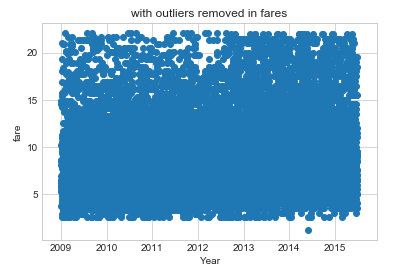


* In records where passenger count is above 6/10, the Fares are normally distributed,(see mean and median almost equal).
  + This suggests that we should not delete these records, but rather correct the passenger count (referring records with valid passenger count where fare is between 8-9, since here the mean exits around that range)

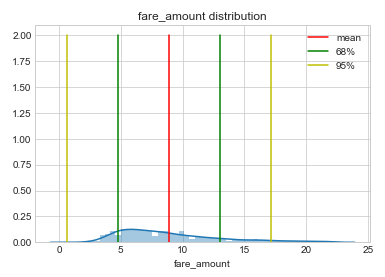
### Pickup\_datetime

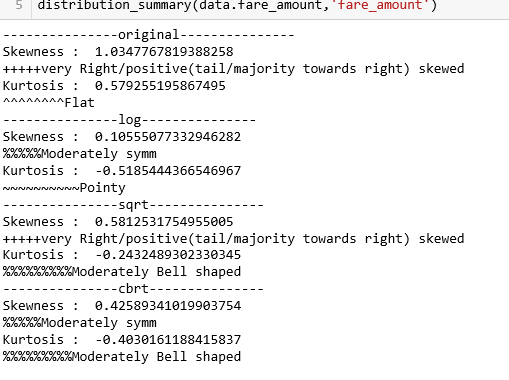


We see a flat line, with a few outliers

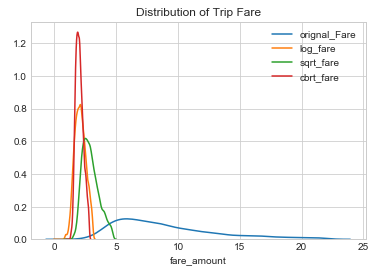


Now we can see it more clearly





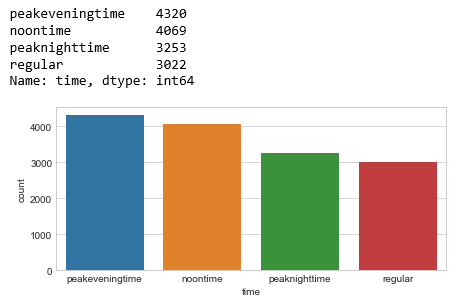
We can see fare\_amount is now flat and right skewed



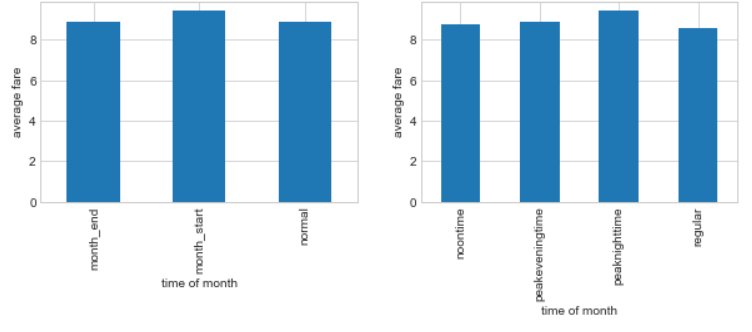
Cbrt gives the best distribution here

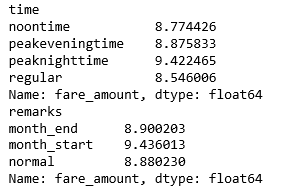
### Feature engineering

Derive new features from date timestamp, let us see



The data has maximum records for peakevening time( 5pm-9pm), followed by noon time(11am-4pm), peak night time(10pm-4am) and rest of the times are regular hours





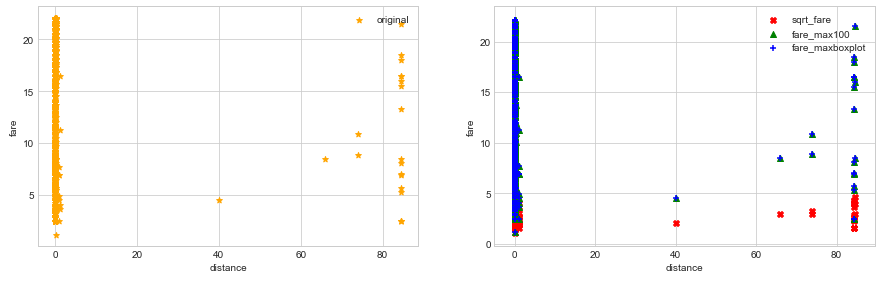
* Month\_start recorded the highest average fare, followed by month\_end, but overall they are very close, showing less significance on fare price
* Peak night time recorded highest average fare in a day as expected and it is quite significantly larger than others groups, showing night hours will impact the fares

### Pickup\_latitude/pickup\_longitude/drop\_off\_latitude/dropoff\_longitude

Manhattan vs Euclidian

Mahattan distance is calculated when we have a mix of conitinous and categorical variables, whereas eucudian distance is the shortest distance between 2 points and hence can be used when you have only continous variables

Below I have plotted original fare\_amount vs distance in 1st figure, and right hand side shows sqrt\_fare, fare\_max100(fare with cap 100) and fare\_maxboxplot(fares with max 22.5 according to boxplot)

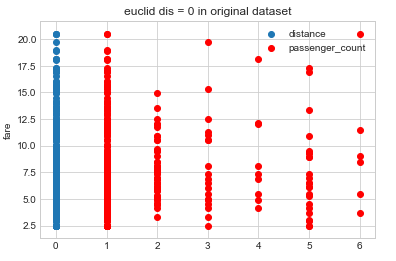


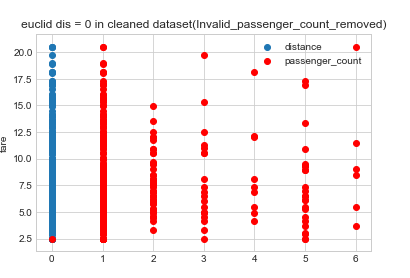
* With constant distance also the fares are increasing, there has to be a other factors which are not considered, like the Other independent factors like dropoff timestamp, waiting time, cab type(luxury/hatch back/suv).
* But since we dont have that as independent variable to access, we will ignore these observations as they will mislead our model

Now I have 2 datsets

* Original – with invalid passenger\_counts mapped to 1
* Cleaned- with invalid dataset removed

Now few records show distance of 0, let us see

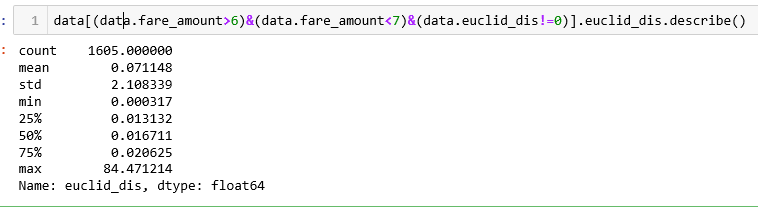




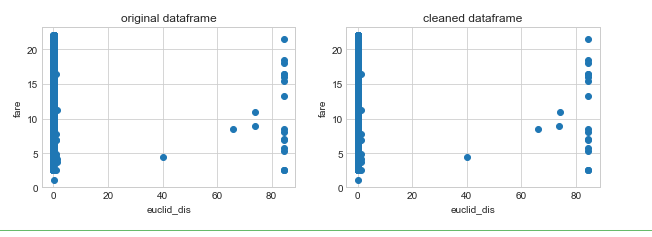
In cleaned dataset, passenger count =0 also

#### Impute Euclid\_dis now

Let us the statistics for euclid\_dis where fare is in around 6.90(mean of this datset with eculid\_dis=0)

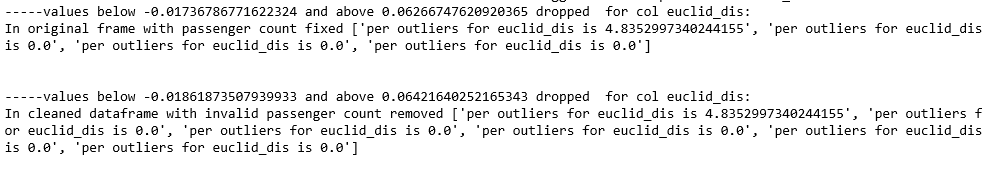


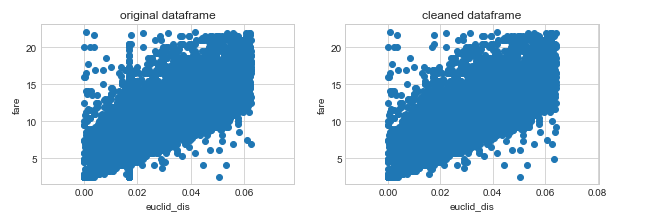
* Again this is right skewed since mean>median, I am using median value(0.017 rounded off) for the Euclid distance in these records
* In cleaned dataset, I deleted these records

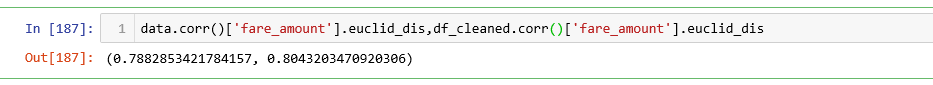


T he no of observations which are differing are very small in number and hence not much visible

There are outliers in distances, let us fix those



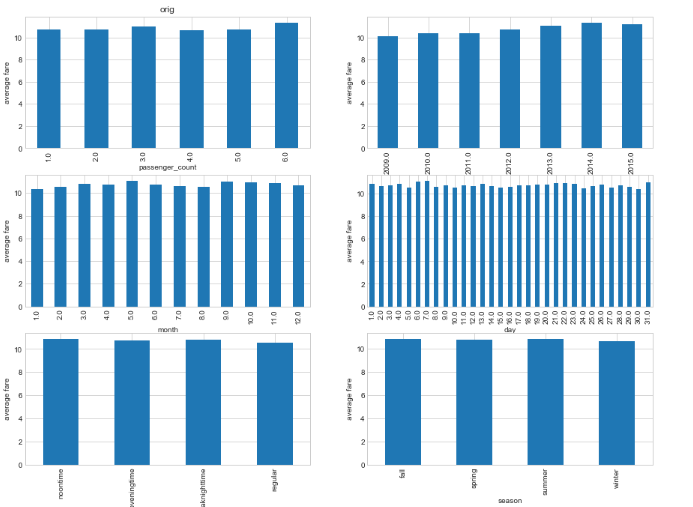




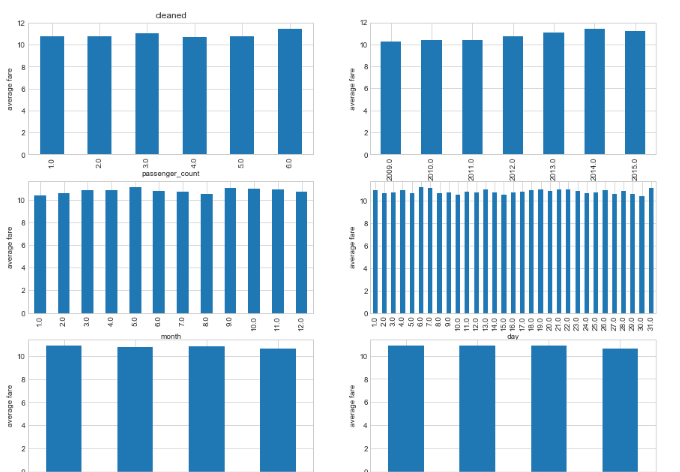
Cleaned dataframe shows more correlation with fare\_amount, so I am going to build a model with that.

#### Why higher fares for keeping distance as constant

Plot high fares(>6.9)

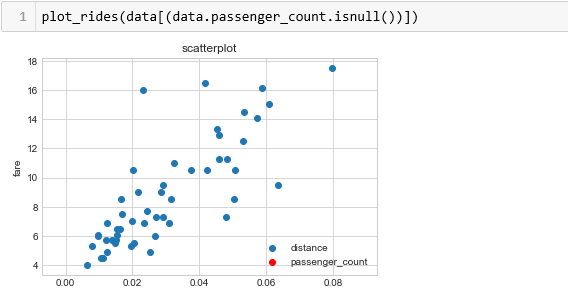
For original dataset

In cleaned dataset



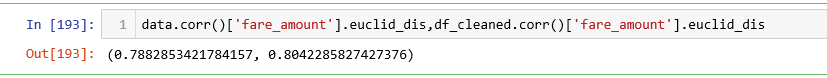
* cleaned dataframe gives better variations in time bar chart
* year effects the most
* month wise also fluctuting
* days also needs to be introspected
* passenger count 6 has highest fares
* season or time of day not effecting the high fares

#### few rides with passenger count as NA also



In original dataset, replaced these observations with passenger\_Count =1

In cleaned dataset, dropped these records



We will continue with cleaned dataframe, since the correlation is higher with fare\_amount

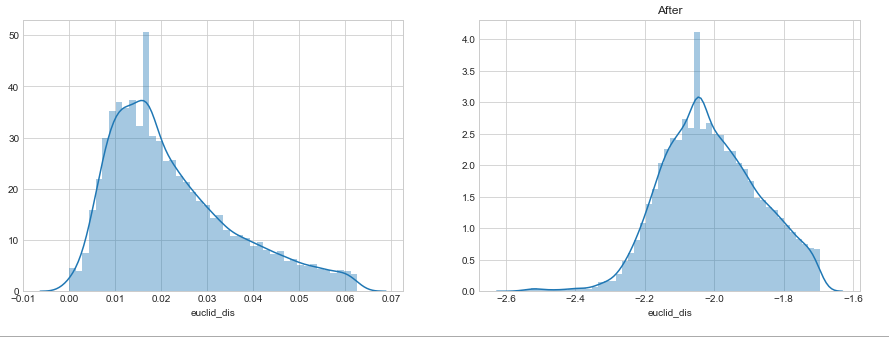
# Model creation

### Normalization

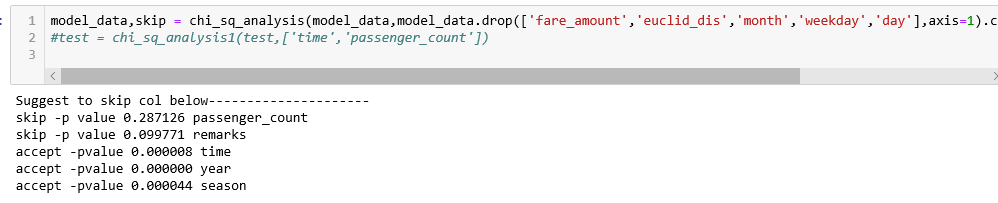
In one model I used Normalizing mont,day,Euclid\_dis for Linear model

### Boxcox transformation

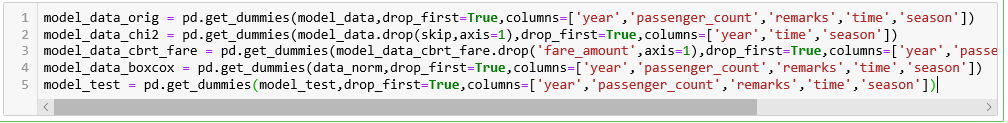
In one model, used boxcox method to normalize these fields



### Chi-square analysis



## Model



Created 4 dataframes for testing purpose

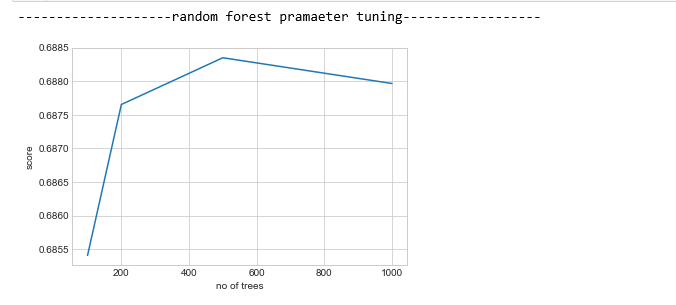
Model\_data\_chi2- skip the columna s per suggested by chi-sq analysis

Model\_data\_cbrt- has cbrt\_log instead of fare\_Amount

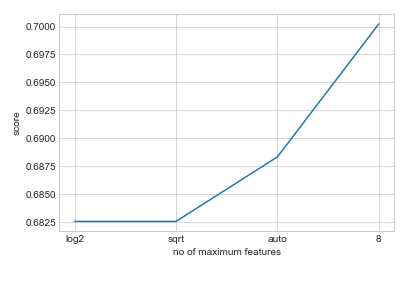
Model\_data\_boxcox- has dataset normalized with boxcox

## Model Tuning

### RandomForestRegressor



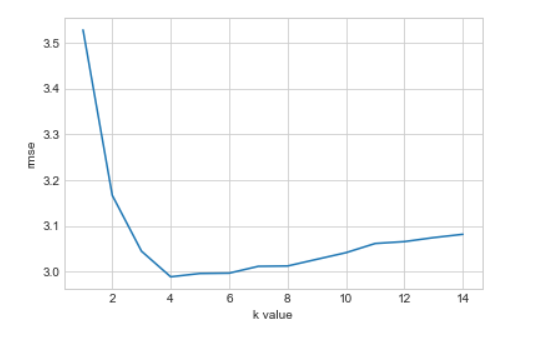
Performance is max with no of trees as 500



Performing most at no of max features as 8

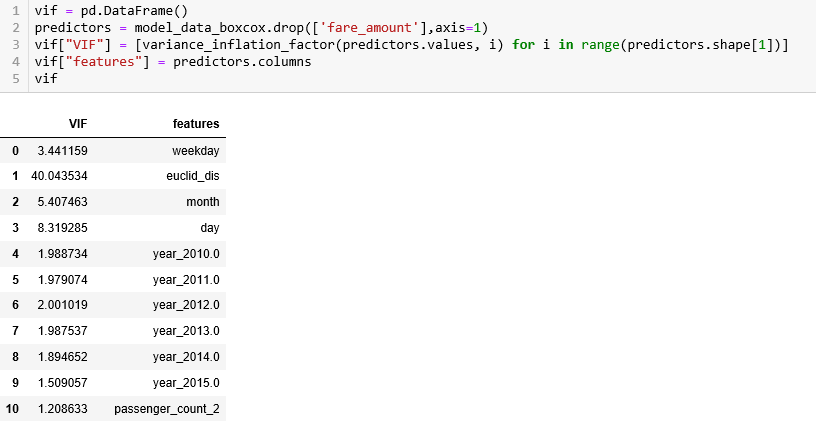
### KNN

For k =4, the rmse is the least and also that makes the elbow of the curve



### Linear regression

The vif is below 10 for model\_data\_orig****

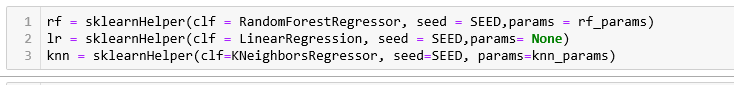


For model\_data\_boxcox, the variation inflation factor is 40, so going to not use this model for evaluation

## K folds

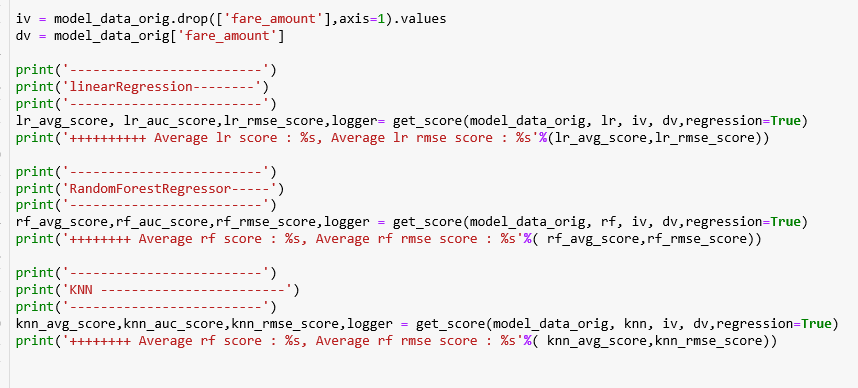
Used folds=5 to get the results

## Objects for runnning the algorithm

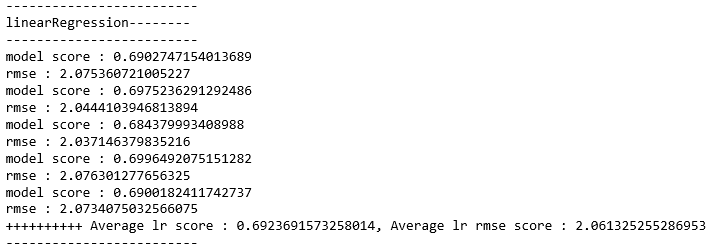


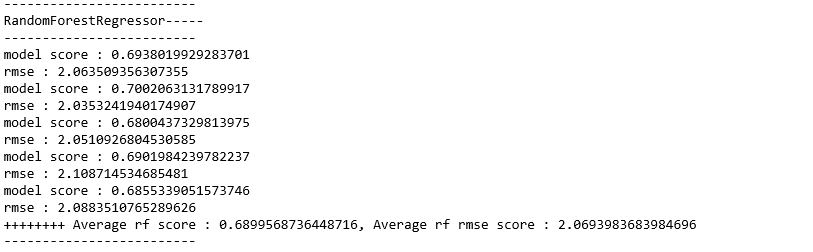
Using the skleanrhelper, I have created a wrapper class, to run the models

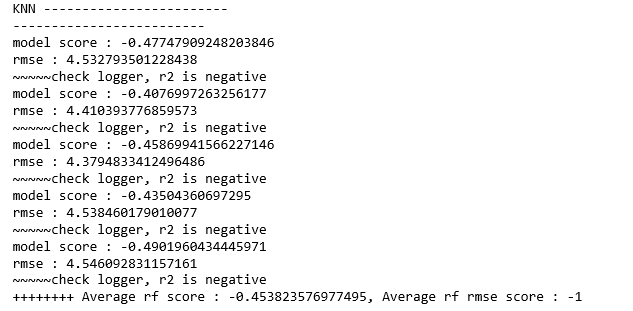
## Fitting and evaluating models



## Results

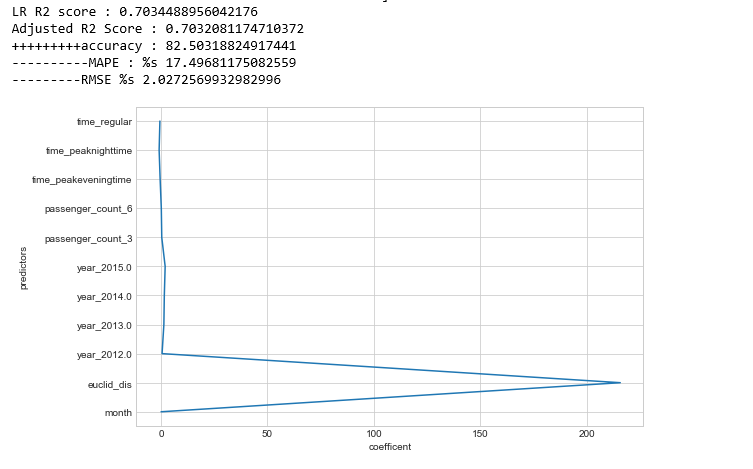






KNN performs very poor, score is negative, it is even worse thean mean fit line

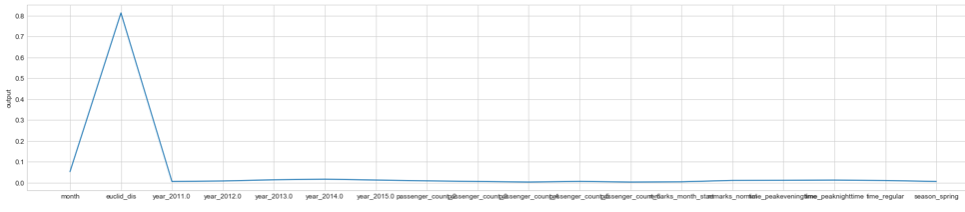
### LinearRegression results analysis



### 

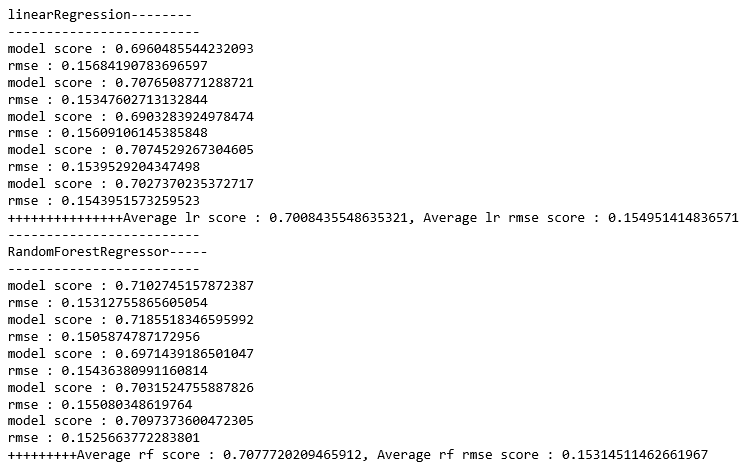
### RandomForest feature importance

Euclid\_dis shows the highest peak



### Best model:

Model\_fare\_cbrt\_fare gives the best result with linearregression



## Predicting the results

