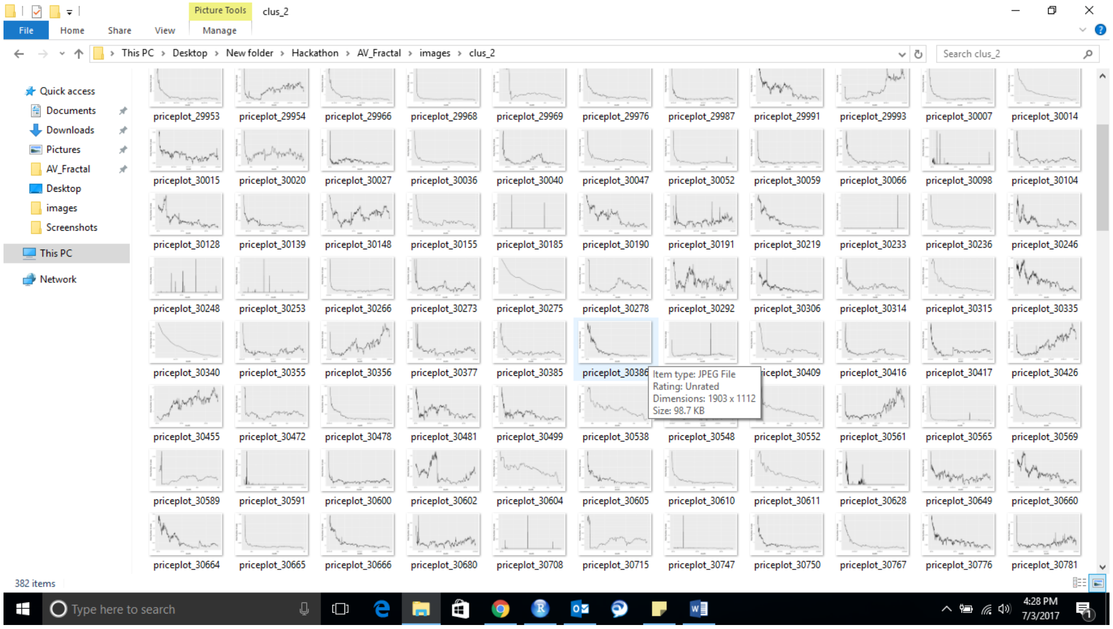
ANALYTICS VIDHYA

FRACTAL ANALYTICS HIRING HACKATHON

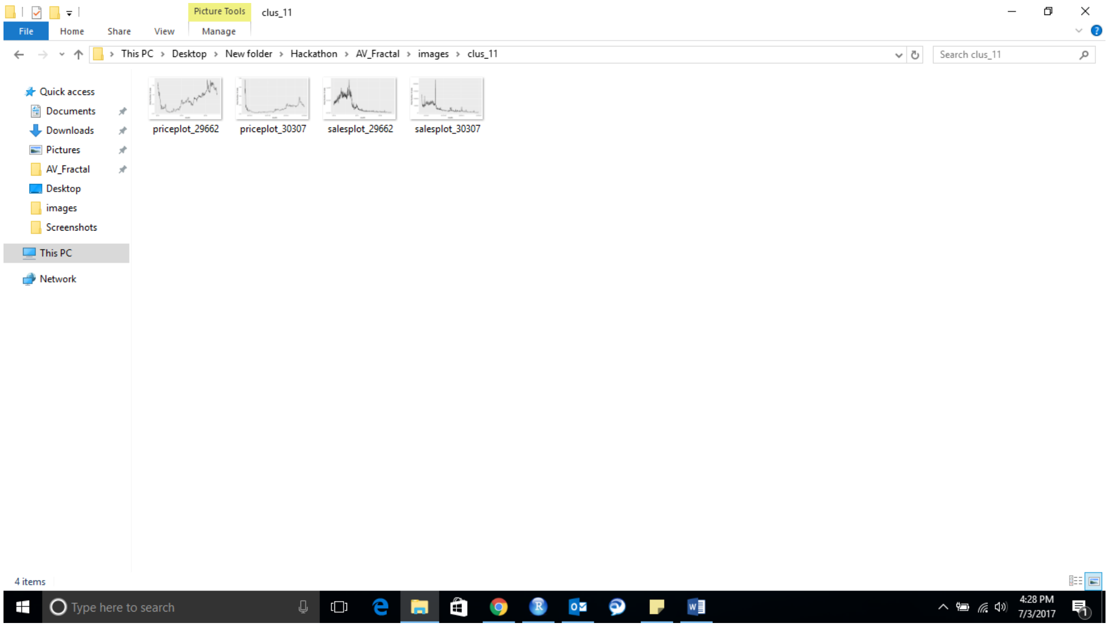
HARSHIT SAXENA

Approach for the challenge:

- Programming Language: **R**, Visualization: **ggplot2**
- We need to build a model to predict **Price** and **Number_Of_Sales** for next 6 months (Huge Number) for all the Items.
- To find out any relation in distributions of each item represented as Item_ID.
- Blindly predicting/forecasting on all data gives random and very bad results which is a proof of not having a same behavior for all Items >> data cannot be combined to build one forecasting or linear model.
- Next step was to cluster all the items so that each cluster has its own distribution/behavior for **Price** and **Number_Of_Sales** and separate models can be created for the same. This also gave bad variations in each cluster. Screenshot below shows that:


Picture Tools
clus_2
File
Home
Share
View
Manage
This PC › Desktop › New folder › Hackathon › AV_Fractal › images › clus_2
Search clus_2
Quick access
Documents
Downloads
Pictures
AV_Fractal
Desktop
images
Screenshots
This PC
Network
priceplot_29953 priceplot_29954 priceplot_29966 priceplot_29968 priceplot_29969 priceplot_29976 priceplot_29987 priceplot_29991 priceplot_29993 priceplot_30007 priceplot_30014
priceplot_30015 priceplot_30020 priceplot_30027 priceplot_30036 priceplot_30040 priceplot_30047 priceplot_30052 priceplot_30059 priceplot_30066 priceplot_30098 priceplot_30104
priceplot_30128 priceplot_30139 priceplot_30148 priceplot_30155 priceplot_30185 priceplot_30190 priceplot_30191 priceplot_30219 priceplot_30233 priceplot_30236 priceplot_30246
priceplot_30248 priceplot_30253 priceplot_30266 priceplot_30273 priceplot_30275 priceplot_30278 priceplot_30292 priceplot_30306 priceplot_30314 priceplot_30315 priceplot_30335
priceplot_30340 priceplot_30355 priceplot_30356 priceplot_30377 priceplot_30385 priceplot_30386 iceplot_30409 priceplot_30416 priceplot_30417 priceplot_30426
Item type: JPEG File
Rating: Unrated
Dimensions: 1903 x 1112
Size: 98.7 KB
priceplot_30455 priceplot_30472 priceplot_30478 priceplot_30481 priceplot_30499 priceplot_30538 priceplot_30548 priceplot_30552 priceplot_30561 priceplot_30565 priceplot_30569
priceplot_30589 priceplot_30591 priceplot_30600 priceplot_30602 priceplot_30604 priceplot_30605 priceplot_30610 priceplot_30611 priceplot_30628 priceplot_30649 priceplot_30660
priceplot_30664 priceplot_30665 priceplot_30666 priceplot_30680 priceplot_30708 priceplot_30715 priceplot_30747 priceplot_30750 priceplot_30767 priceplot_30776 priceplot_30781
382 items
Type here to search
4:28 PM
7/3/2017

[All the images show **Price** distribution for all the items in one cluster: cluster 2]

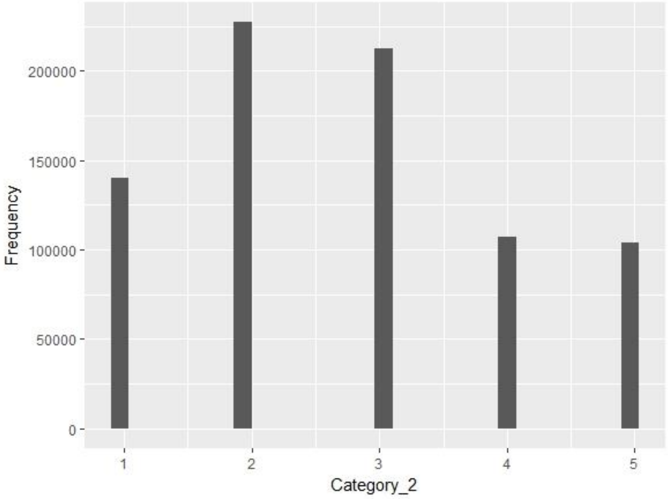


[All the images show **Price** and **Number_Of_Sales** distribution for all the items in one cluster: cluster 11]

- These clusters didn't perform well so I approached for one vs one approach and build separate models for each Item_ID.
- Model Selections is given below under algorithm selection section.

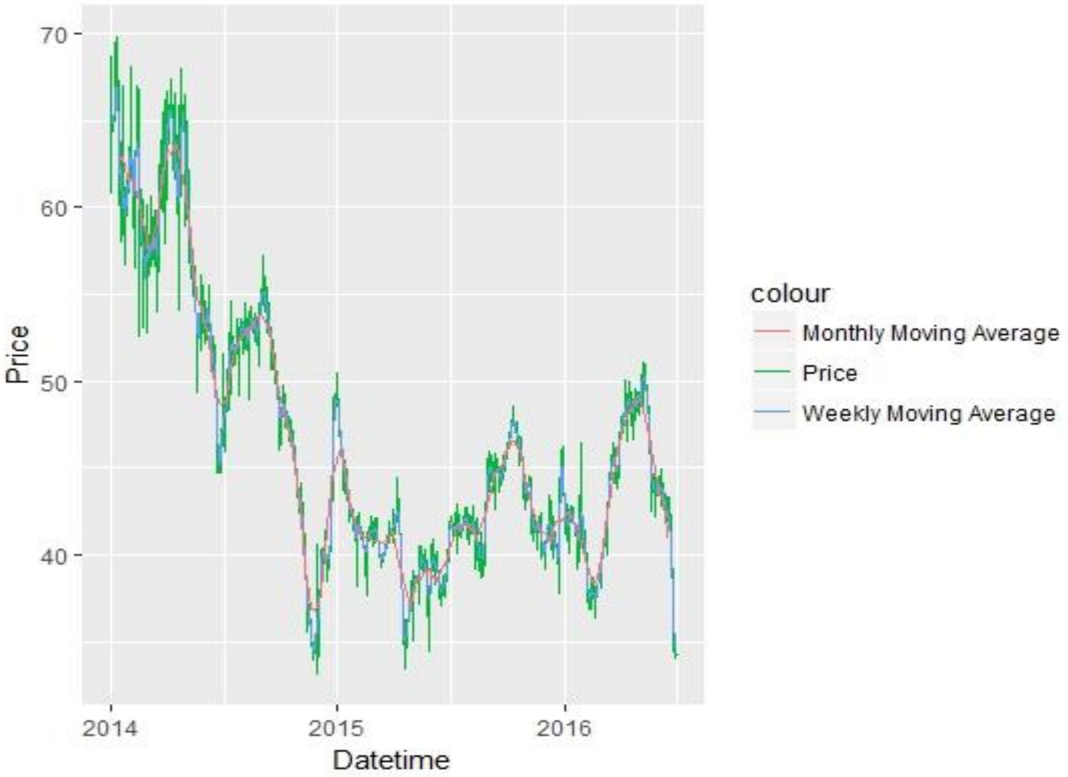
Data preprocessing/ cleaning steps:

- Imputed Missing values in Category_2 with value = 2

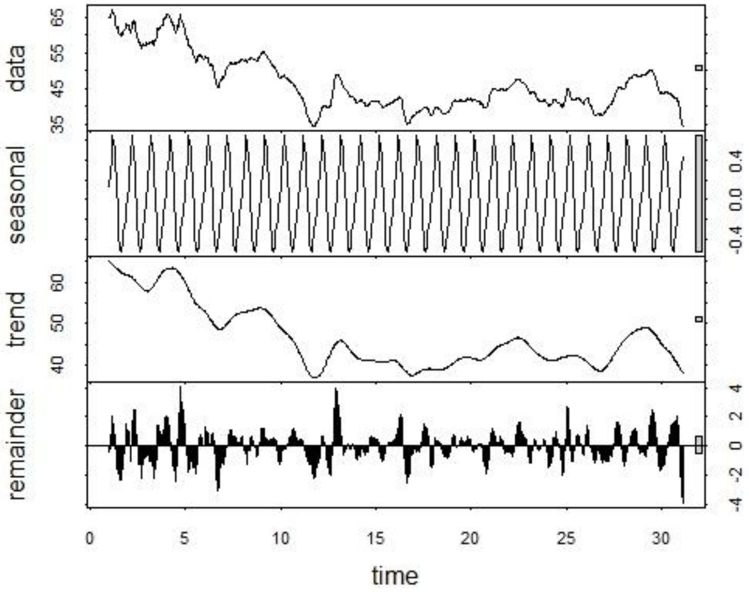


- Creating **Price & Number_Of_Sales** a time series object and removing outliers from the distribution using tsclean () function.

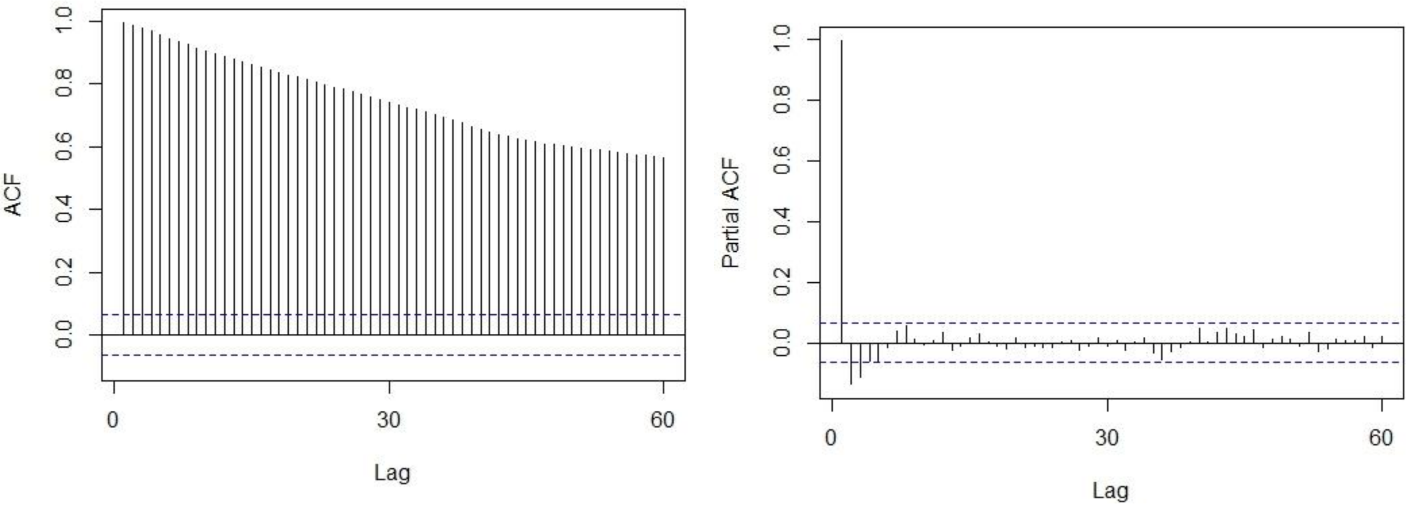
- Created features like year, month, date, weekday, weekend, diff_days (difference of days from day 1 which is 01-01-2014)
- Looked onto distribution using ACF and PACF plots which tells pattern with every 7 days.



[Distribution of Price for Item_ID = "30375" with weekly and monthly moving averages]

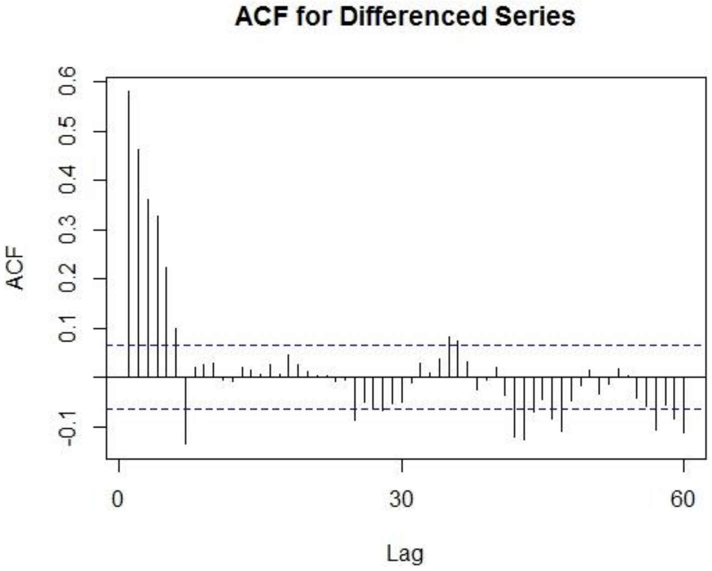


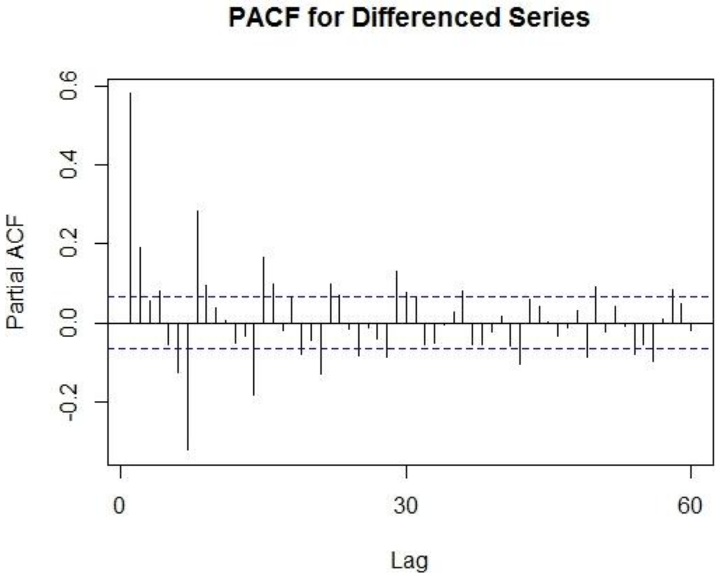
[checking for trend, seasonality in our distribution]



[ACF and PACF plots for our price distribution]

- We need to make our series stationary for ARIMA models and with differencing of 1 gives us stationary series from Dickey-Fuller Test.





- We can see PACF clearly showing 7th lag out of blue line which tells weekly pattern.

Algorithm Selection:

- Firstly, I tried ARIMA on each item which gave good results. It was making sense to me as well but what about smaller datasets like Item_ID = “30908” has just five entries.

- So, I introduced simple exponential smoothing forecasts for smaller datasets which further improved the results.
- Then I introduced linear regression for datasets which gave very bad results on leaderboard dataset
- Even XGBoost didn’t improve my score because of less data and large forecasting range of 6 months.
- Then I did parameter pruning and found SES (simple exponential smoothing) to outperform ARIMA models and used it for final submission.

HARSHIT SAXENA

LEADERBOARD RANK – 19

I have attached code also in the mail.