CMPE 257 - Project

Herbarium and Store sales forecasting using Time Series

GROUP 14: Kashyap T, Mahesh Chandra Mareedu, Army P, Mehulkumar K, Wen-Hao Tseng

Herbarium : Identify novel plant species

- **About Data**
  - Number of images: 1.05 M
  - Number of classes : 15501
  - Data size :160 GB (train)
- **Challenges involved**
  - Cannot load data on to Kaggle RAM
  - Requires high processing GPU (Kaggle provides only 36hrs).
  - Classification models cannot be developed due to high dimensional target (15k+ classes).
  - Data Augmentation is not possible as it requires 2X space (160GB+160GB)
  - Neural network output layer becomes sparse due to high number of target variables
  - Alternate approach takes 9 hrs to run each time
    - Faiss + ResNET

Transfer learning with ResNet+faiss

- Removed last layer or ResNet and used it to generate embeddings of image each image of dimension 512.
- Similarly for all the images belonging to 15k+ classes ,we have composed image embeddings for 839772 training samples.
- We fed all the image embeddings to *faiss* (Facebook AI Similarity Search) model to create indexes (Paper)
- The index file stores information of all the image vectors with their corresponding image ids.

```
print("Shape of image embedding",vec.shape)
print(vec[0:10])
plt.imshow(img)
```

```
Shape of image embedding (512,)
[0.48710352 0.3259773  0.8979322  0.28265658 0.3696866  0.3621263
 2.0340765  1.2006575  0.949871   0.11798392]
<matplotlib.image.AxesImage at 0x7fe3137c1390>
```

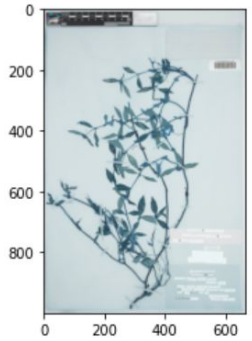
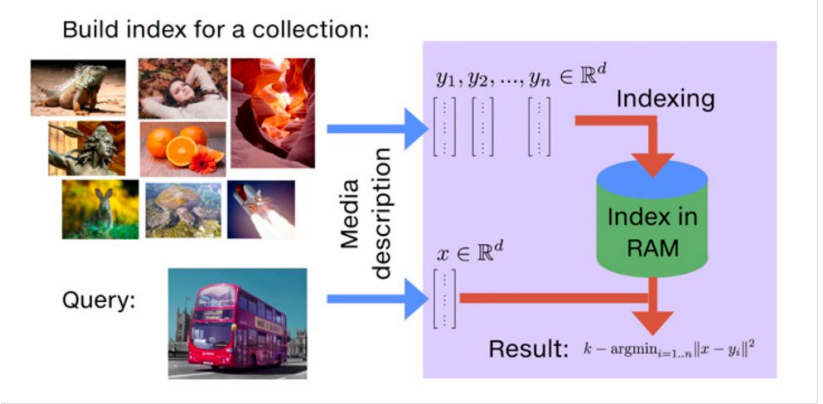
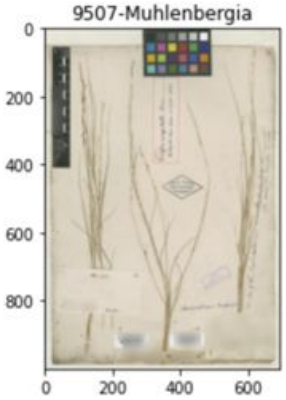


Image classification using Faiss

- Faiss clusters images embedding to P possible clusters.
- When ever we want find similar images for a given image, the module checks target image embeddings distance with the centroid of each clusters.
- The cluster centroid with minimum distance with target image is selected.
- Now, distance between the target image and images in that cluster are composed.
- The closes image ids are returned by the model.
- Runtime complexity comes down from O(N) => O(P)



Sample predictions



- As we are using non-annotated data, the cluster based classification model uses the non important features on images as well.
- The first two predictions seem right, but the remaining three have high cosine similarity due to the color block (a non important feature)

Sales Forecasting

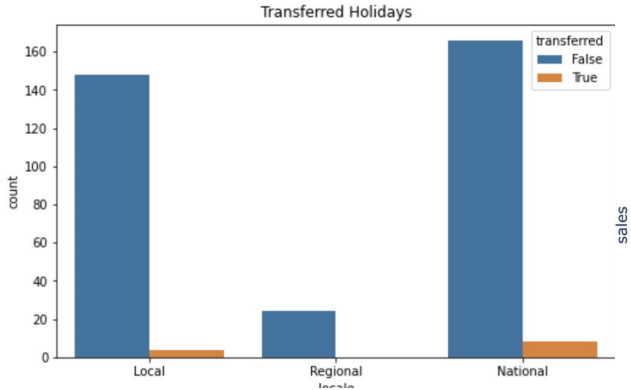
- Motivation - current market is all about predicting real time growth for the business
- Objective - Forecast store sales using time-series on data from a large Ecuadorian-based grocery retailer
- Evaluation - Submission evaluated based on RMSLE (Root Mean Squared Logarithmic Error)

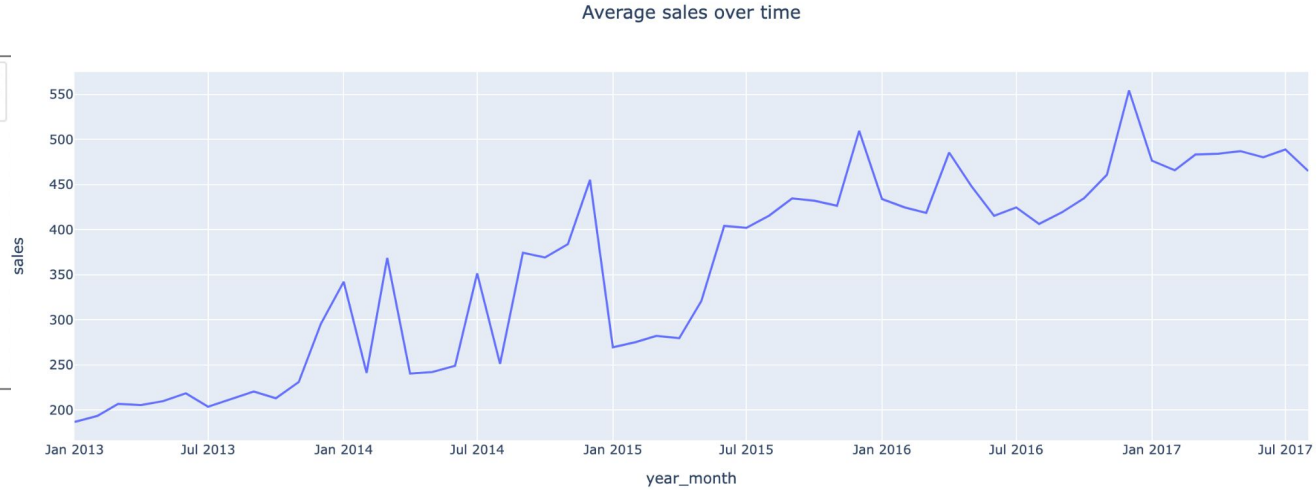
Source - https://www.analyticsvidhya.com/blog/tag/time-series-forecasting/

Dataset

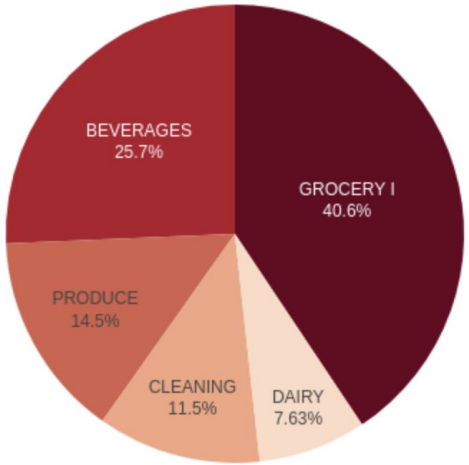
- Six files that contains information about
  - Holidays and events
  - Transactions
  - Stores
  - Oil prices over time
  - Train
  - Test
- Explored each of the dataset in deep to study the hidden patterns
- Examined the combination of each dataset with the other dataset to get more information on the data provided
- The data collected is from 2013-01-01 to 2017-08-31(combining both train and test)

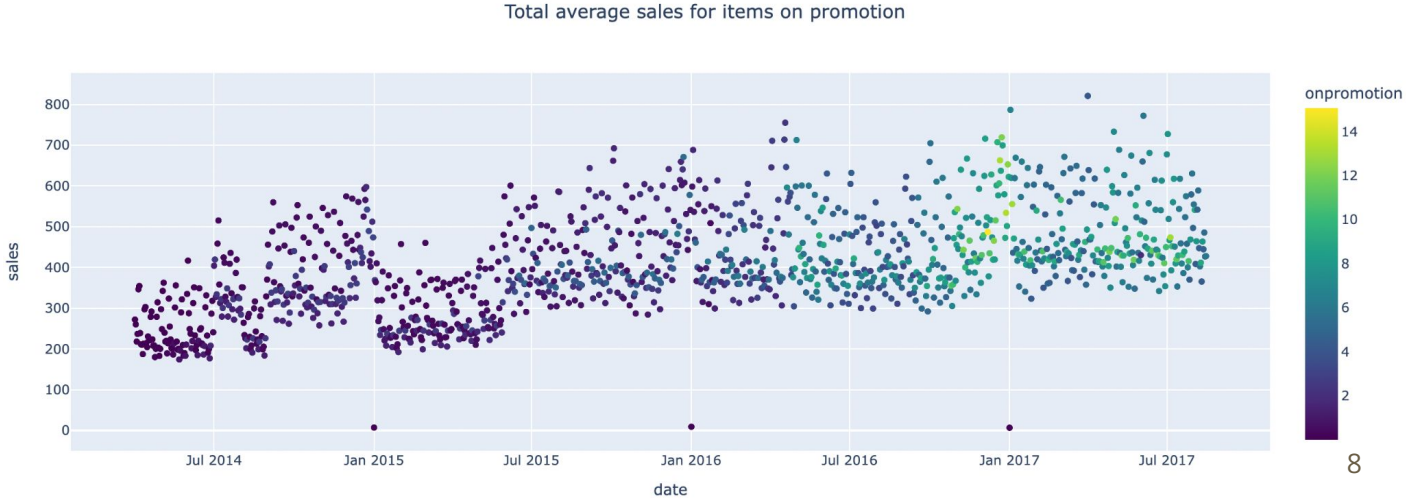
Data sources, cleansing, validation, transformation, visualization.

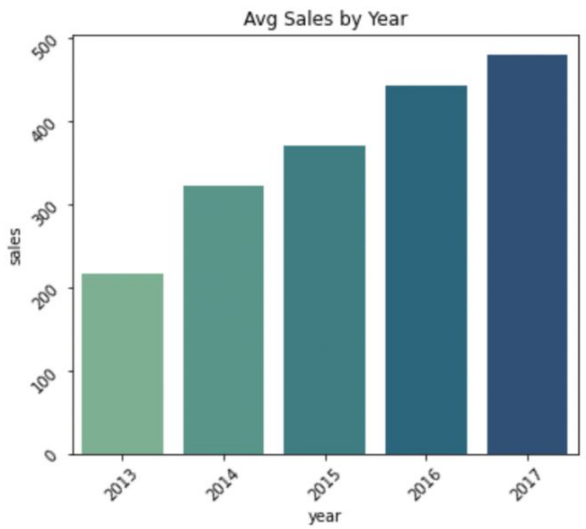


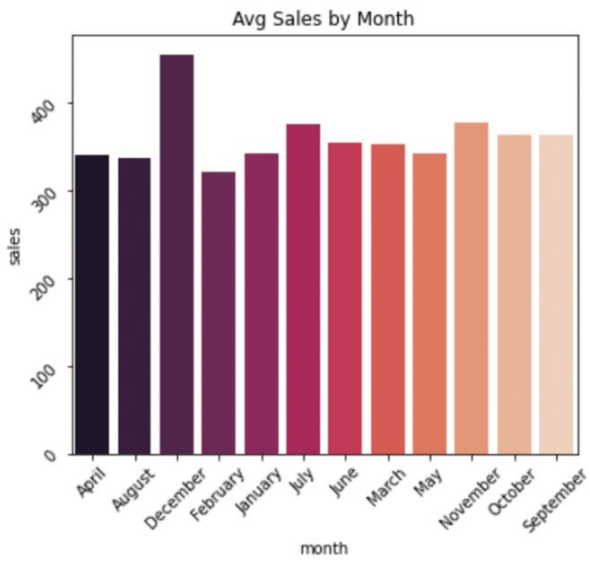


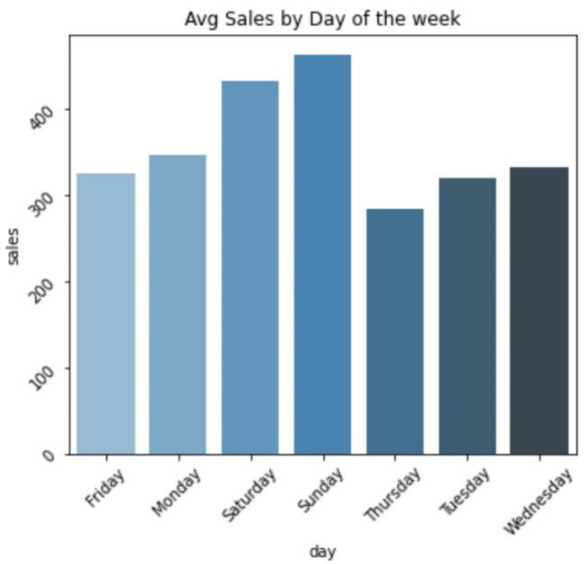
5 categories that make the most sales



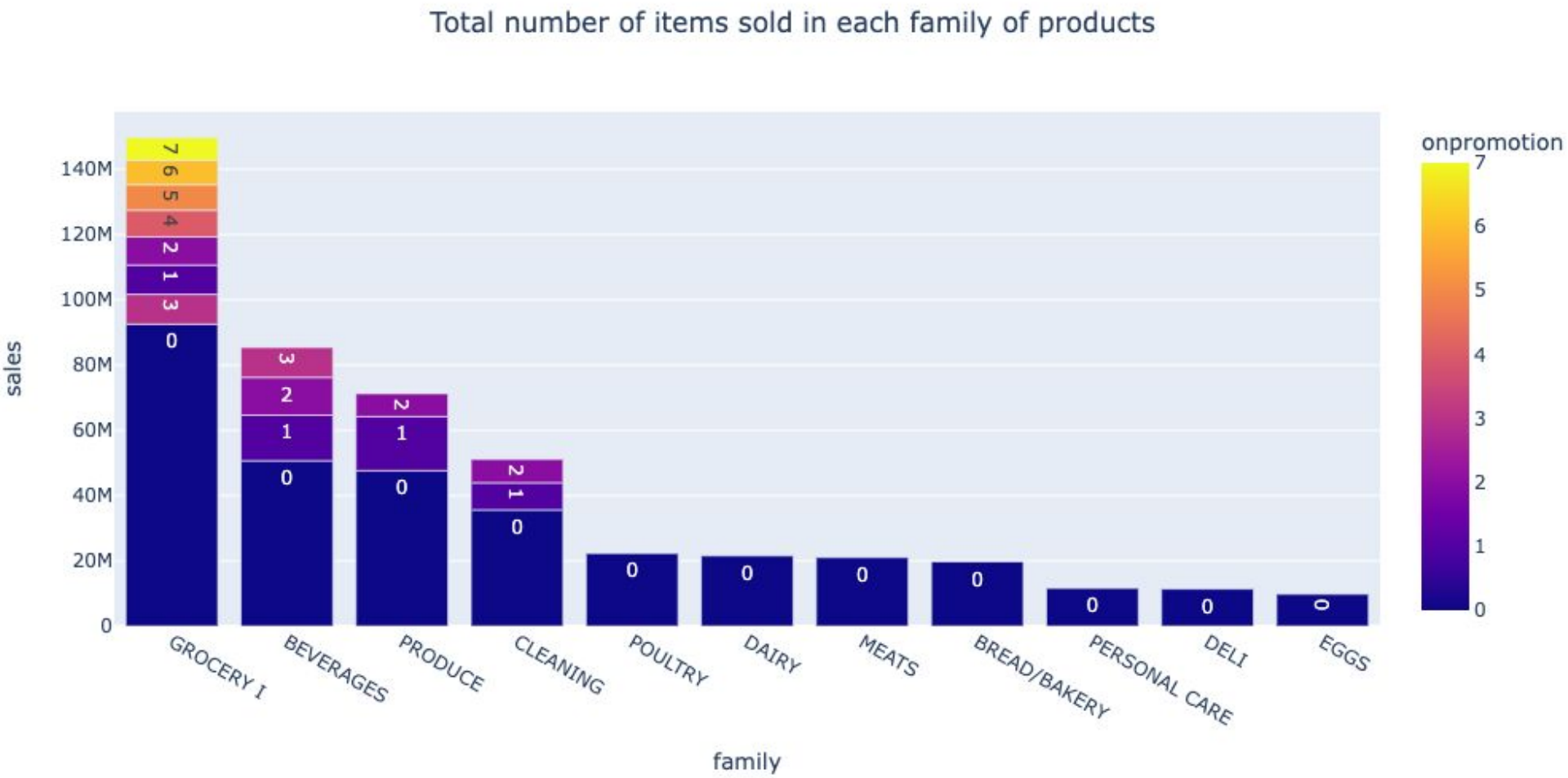


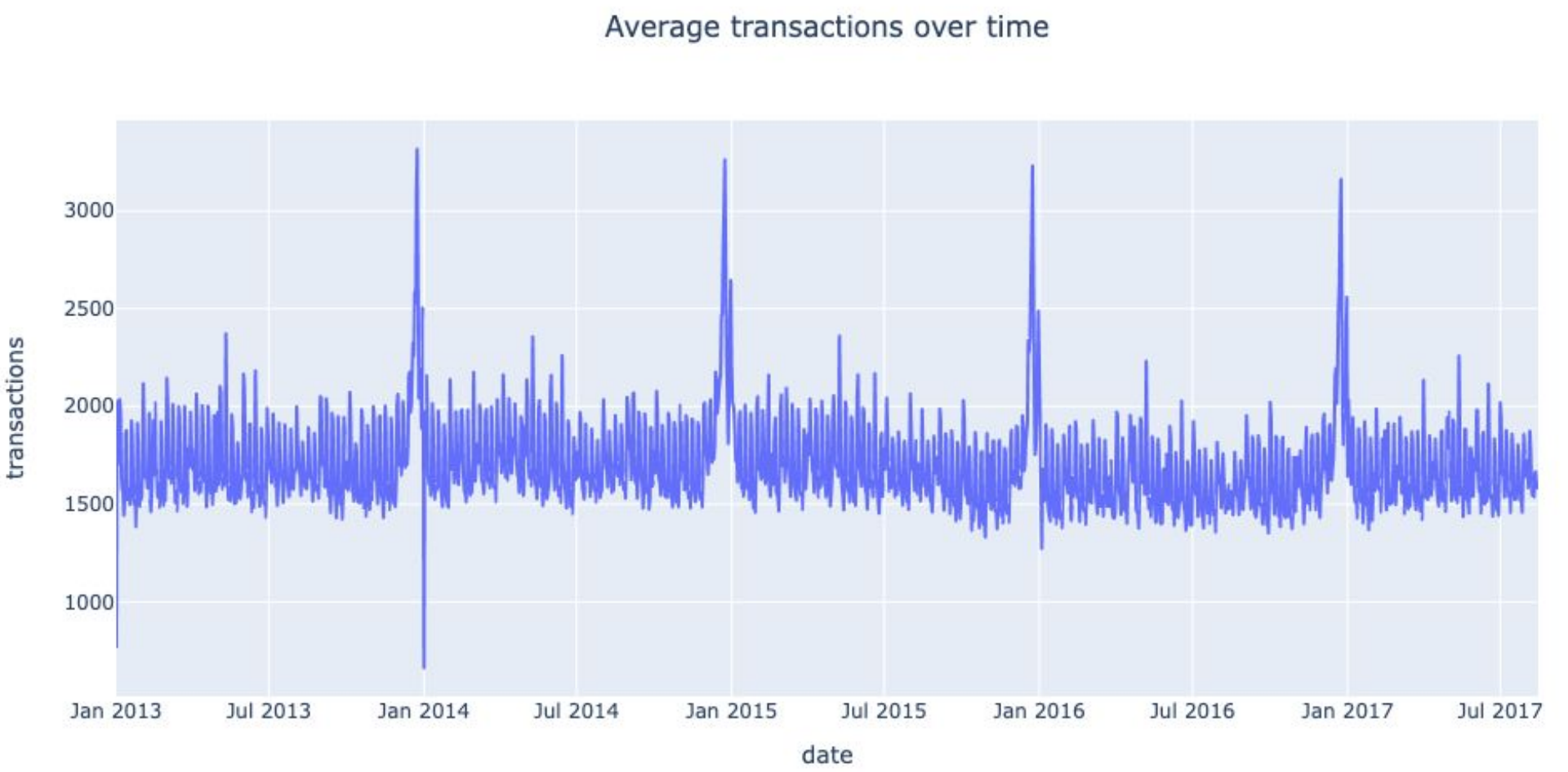






Oil prices over time


Total number of items sold in each family of products
sales
140M
120M
100M
80M
60M
40M
20M
0
GROCERY I
BEVERAGES
PRODUCE
CLEANING
POULTRY
DAIRY
MEATS
BREAD/BAKERY
PERSONAL CARE
DELI
EGGS
family
onpromotion
7
6
5
4
3
2
1
0

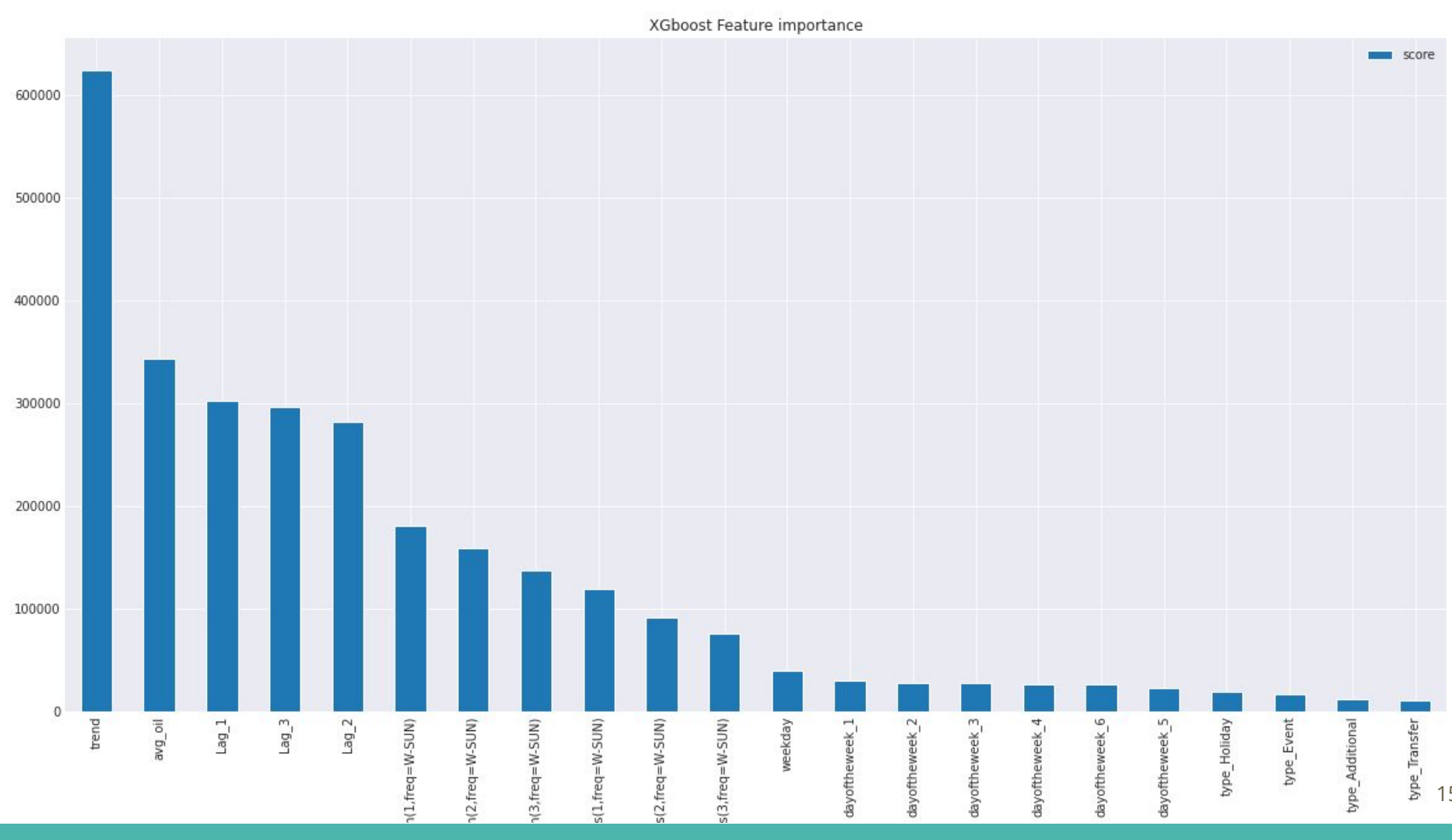

Average transactions over time
transactions
3000
2500
2000
1500
1000
Jan 2013
Jul 2013
Jan 2014
Jul 2014
Jan 2015
Jul 2015
Jan 2016
Jul 2016
Jan 2017
Jul 2017
date

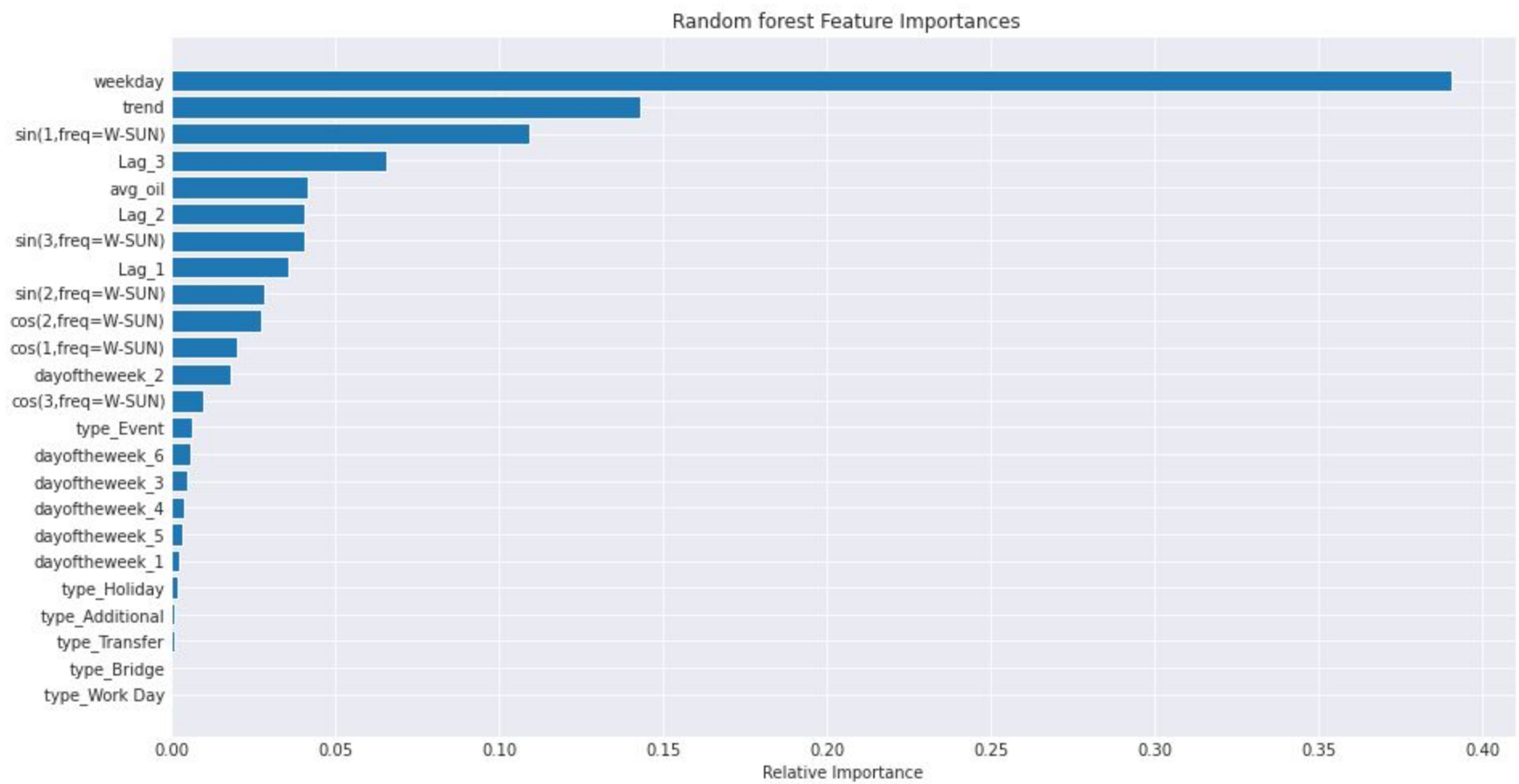

Total number of transactions for each store
transactions
store_nbr
type
A
D
B
C
E
4500
4000
3500
3000
2500
2000
1500
1000
500
0
10
20
30
40
50

Modeling

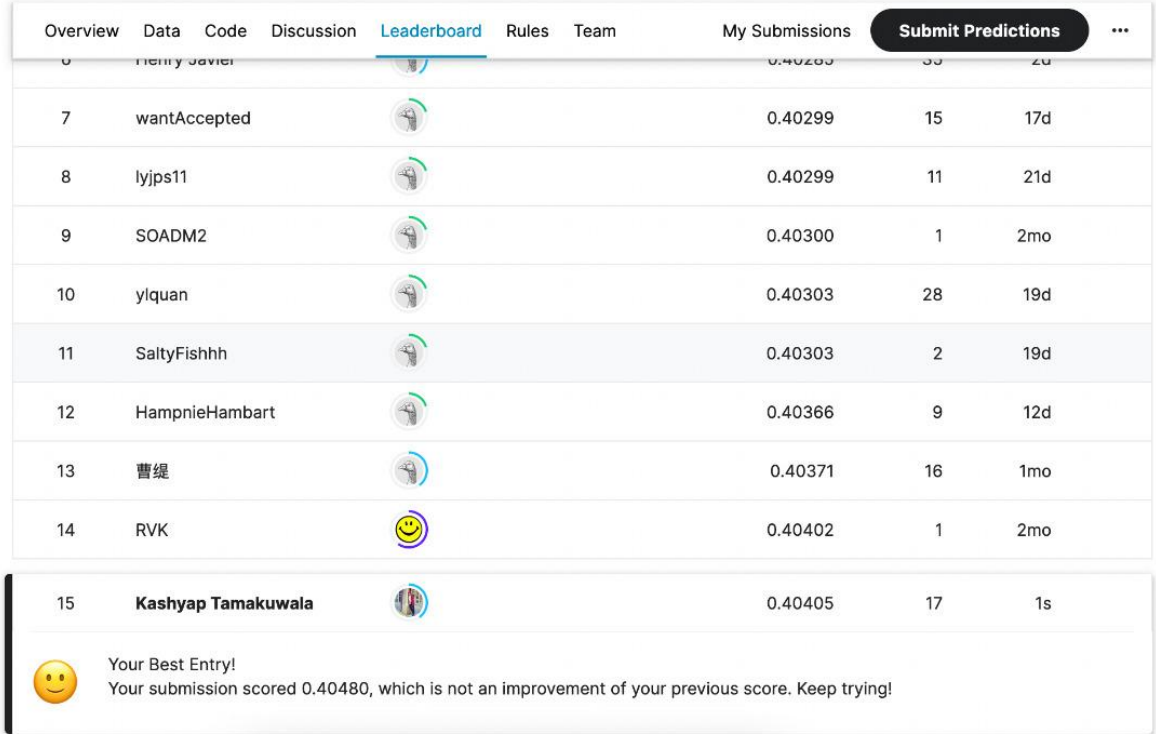
- Regression techniques
  - Linear Regression
  - Ridge
  - Lasso
  - Random Forest
  - SVR
  - XGboost
- To consider time dependency
  - ARIMA
  - SARIMA
- Blending and Custom Regressor (0.40408)
  - Used Linear model as base model
  - Built a custom regressor with Random Forest,Extra Tree Regressor, Ridge regression and svr.

Code demo


XGboost Feature importance
score
600000
500000
400000
300000
200000
100000
0
trend
avg_oil
Lag_1
Lag_3
Lag_2
n(1,freq=W-SUN)
n(2,freq=W-SUN)
n(3,freq=W-SUN)
s(1,freq=W-SUN)
s(2,freq=W-SUN)
s(3,freq=W-SUN)
weekday
dayoftheweek_1
dayoftheweek_2
dayoftheweek_3
dayoftheweek_4
dayoftheweek_6
dayoftheweek_5
type_Holiday
type_Event
type_Additional
type_Transfer


Random forest Feature Importances
weekday
trend
sin(1,freq=W-SUN)
Lag_3
avg_oil
Lag_2
sin(3,freq=W-SUN)
Lag_1
sin(2,freq=W-SUN)
cos(2,freq=W-SUN)
cos(1,freq=W-SUN)
dayoftheweek_2
cos(3,freq=W-SUN)
type_Event
dayoftheweek_6
dayoftheweek_3
dayoftheweek_4
dayoftheweek_5
dayoftheweek_1
type_Holiday
type_Additional
type_Transfer
type_Bridge
type_Work Day
0.00
0.05
0.10
0.15
0.20
0.25
0.30
0.35
0.40
Relative Importance

Kaggle Leaderboard



Total Participants: 900

Our team is in top: 2 %

App demo