Store Sale Analysis using Kaggle Dataset

FINAL PROJECT - LAMBTON COLLEGE

AML 2103 - DATA VISUALIZATION & EXPLORATORY DATA ANALYSIS

Bhanu Prakash Mahadevuni[1], Deeksha Naikap[1], Pramod Reddy Gurrala[1], Sai Varun Kollipara[1]

[1] – AIMT Group Student, Lambton College

Abstract: Sales are the only way to survive any store, so implementing advanced techniques that can analyze those sales would be helpful for many stores. So, machine learning models are implemented to get some business insights.

I. Introduction

Machine Learning is a process where the data is analyzed for performing some predictions based on the gathered business insights. The data is collected from an online public repository called Kaggle. The store sales are data that consists of multiple information on the location of stores, holiday details, type of groceries, and transaction details. The data is loaded into the project to gather results to increase sales or predict sales using a forecasting method.

II. Solution Approach

The Approach for this case is to consider the Store Sales Analysis data available on Kaggle and predict the future sales of products sold at Favorita stores in Ecuador by using necessary libraries and packages. We will analyze the trend and pattern in sales throughout the timeframe and utilize the XGBoost regression model to predict the expected sales for each product. Different visualization techniques will be used to retrieve better insights from the data.

III. Dataset Utilized

Data consists of 6 different parts in CSV format.

Train.csv – The training data contains store_nbr, family, on promotion, and target sales.

Test.csv – Test data has the same features as the training data. The prediction will be made on the target sales for the dates in this file.

Stores.csv – It contains store metadata, including city, state, type, and cluster.

Oil.csv - Consists of information on the daily oil price. Includes values during both the train and test data timeframes.

Holidays_events.csv – It contains metadata for holidays and events.

```
df_oil = pd.read_csv("oil.csv", index_col='date')
df_stores = pd.read_csv("stores.csv", index_col='store_nbr')
df_test = pd.read_csv("test.csv")
df_train = pd.read_csv("train.csv")
df_transactions = pd.read_csv("transactions.csv")
df_holidays = pd.read_csv("holidays_events.csv")
```

Fig 1: Data Frames for all the CSV files

```
print('Number of train samples: ', df_train.shape)
print('Number of test samples: ', df_test.shape)
print('Number of store data: ', df_stores.shape)
print('Number of Holiday data: ', df_holidays.shape)
print('Number of Oil Price data: ', df_oil.shape)
print('Number of transactions data: ', df_transactions.shape)
```

```
Number of train samples:  (3000888, 6)
Number of test samples:  (28512, 5)
Number of store data:  (54, 4)
Number of Holiday data:  (350, 6)
Number of Oil Price data:  (1218, 1)
Number of transactions data:  (83488, 3)
```

Fig 2: Shape of all the Data Frames

```
df_train.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3000888 entries, 0 to 3000887
Data columns (total 6 columns):
 #   Column       Dtype
---  ------       -----
 0   id           int64
 1   date         object
 2   store_nbr    int64
 3   family       object
 4   sales        float64
 5   onpromotion  int64
dtypes: float64(1), int64(3), object(2)
memory usage: 137.4+ MB
```

Fig 3: Collect Information on the Data Frames

IV. Steps Involved

These are the steps involved in the project study:

- Importing the Packages, Functions, and Libraries
- Importing and Loading the Dataset
- Data Pre-processing
- Exploratory Analysis
- Data Visualization
- Data Splitting
- Model Building

V. Exploratory Data Analysis

In General, this phase prepares the data for machine learning or deep learning models.

```
df_train.date = pd.to_datetime(df_train.date)
df_test.date= pd.to_datetime(df_test.date)
```

Fig 4: Change the date data type as DateTime in both train and test datasets.

Missing values: Missing values are not available values, which would be meaningful if observed. Missing values can be anything from missing sequence, incomplete feature, files missing, information incomplete, data entry error, etc. Most datasets in the real world contain missing values.

```
print(df_train.isna().sum())
print(df_test.isna().sum())
```

```
id             0
date           0
store_nbr      0
family         0
sales          0
onpromotion    0
year           0
month          0
dayofmonth     0
dayofweek      0
dayname        0
dtype: int64
id             0
date           0
store_nbr      0
family         0
onpromotion    0
year           0
month          0
dayofmonth     0
dayofweek      0
dayname        0
dtype: int64
```

Fig 5: Checking for Missing Values

Duplication identification: Datasets containing duplicates may contaminate training data with the test data or vice versa and effects the accuracy and performance of the machine learning model.

```
duplicate_records = df_train[df_train.duplicated()]
print("number of dupicate records:", duplicate_records.shape)
```

```
number of dupicate records: (0, 11)
```

```
duplicate_records = df_test[df_test.duplicated()]
print("number of dupicate records:", duplicate_records.shape)
```

```
number of dupicate records: (0, 10)
```

Fig 6: Checking for Suplicates

VI. Data Visualization

Data visualization translates the information into a pictorial context, making it easier for humans to understand the data and derive insights from it.

In Python, there are several plotting libraries Matplotlib, Seaborn, Plotly, etc. There are different types of visualization in Python. Some of them are: Histogram, Scatterplot, Bar chart, Pie chart, Boxplot, Heatmap and Line chart.

In this section we are going to carry out various studies of the data obtained above, using various graphs. We will focus on seeing:

- The shops with the highest percentage of sales
- The types of products most sold.
- The sales of each cluster.
- The sales history for each of the months of the year.
- The percentages of sales per quarter of the year.
- Average sales per week.

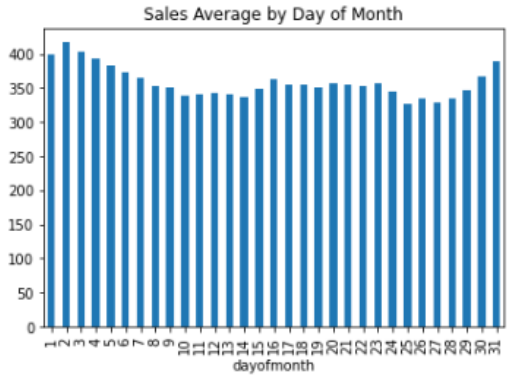

Fig 7: Bar Graph for Sales

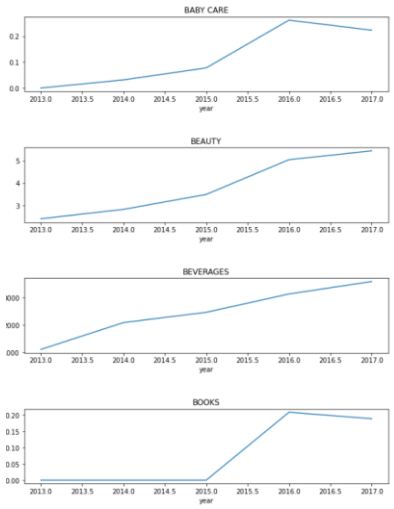

Fig 8: Line Graph on Yearly Basis

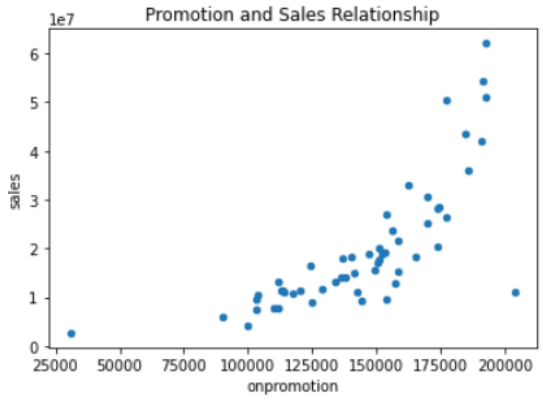

Fig 9: Scatter Plot for Sales on Promotion

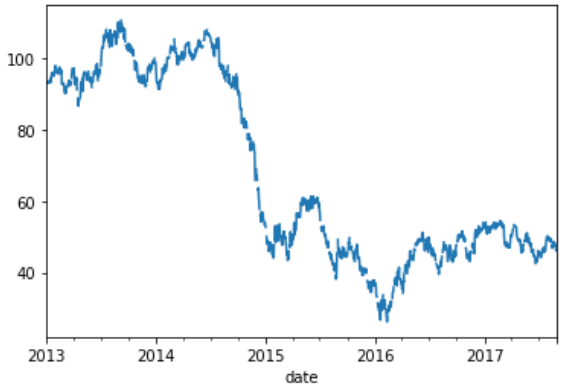

Fig 10: Oil trend Chart

VII. Data Splitting

Data Splitting is a stage where the data is decided and assigned to multiple variables that will be used in the future for training and testing the model. (This is a primary step for every machine/deep learning model building). The data can be split into training and Testing.

```
FEATURES = ['store_nbr','family','onpromotion', 'year', 'month',
       'dayofmonth', 'dayofweek','dcoilotico', 'city', 'state',
       'type', 'cluster', 'national_holiday', 'national_event',
       'national_workday', 'weekend', 'local_holiday', 'regional_holiday','lag_16','lag_17','lag_18','lag_19','lag_20','lag_21','lag_22','lag_28','lag_31','lag_365']
TARGET =['sales']
```

Fig 11: Target Variables

```
from sklearn.model_selection import train_test_split
X_train,X_val,y_train,y_val = train_test_split(df_train,df_train[TARGET],test_size=0.05,shuffle=False)
```

Fig 12: Data Splitting for Model

VIII. Build Model

The foremost step for every machine learning / deep learning is to build a model based on the data. We have used the below Linear regression and XGBoost algorithms to build the model.

```
linear = LinearRegression()
model = linear.fit(X_train[FEATURES],y_train)
```

Fig 13: Linear Regression Model

```
xgb = XGBRegressor(n_estimators=500)
model_XGB = xgb.fit(X_train[FEATURES], y_train,
        eval_set=[(X_train[FEATURES],y_train),(X_val[FEATURES], y_val)],
       verbose=False,early_stopping_rounds=10)
```

Fig 14: XGBoost Model

```
MSLE:  0.03227693103877899
XGBRegressor(base_score=0.5, booster='gbtree', callbacks=None,
             colsample_bylevel=1, colsample_bynode=1, colsample_bytree=1,
             early_stopping_rounds=None, enable_categorical=False,
             eval_metric=None, gamma=0, gpu_id=-1, grow_policy='depthwise',
             importance_type=None, interaction_constraints='',
             learning_rate=0.300000012, max_bin=256, max_cat_to_onehot=4,
             max_delta_step=0, max_depth=6, max_leaves=0, min_child_weight=1,
             missing=nan, monotone_constraints='()', n_estimators=500, n_jobs=0,
             num_parallel_tree=1, predictor='auto', random_state=0, reg_alpha=0,
             reg_lambda=1, ...)
```

Fig 15: Model Evaluation

IX. Conclusion

We have identified some graphs that can show the trends for the datasets. Visualization is best for understanding the data and their relationship. There are multiple ways to visualize data. It can often take a while to figure out what's going on with your forecast. Visualizing the results of time series can be done intuitively and effectively with the following plots.

These plots allow you to see, for a single store, anywhere from 1 to 33 families plotted, with 3 different fits (train, validation, forecast). When you do runs that take a long time, you'll usually save/commit and then go away. When you come back, all these plots are saved along with your results so you can visually get a sense of how your latest changes impacted your hybrid model's accuracy.

X. Future Developments

The insights from the results can be implemented in multiple stores to analyze the report from the machine learning models. Based on the performance, the scale of the implantation can be increased.

XI. References

[1]. Loola Bokonda, Ouazzani Touhami Khadija, Nissrine Souissi, "Predictive analysis using machine learning: Review of trends and methods", Conference: International Symposium on Advanced Electrical and Communication Technologies - IEEE ISAECT, vol. 2, pp. 329-340, 2020. DOI: 10.1109/ISAECT50560.2020.9523703

[2]. S M Nazmuz Sakib, "Restaurant Sales Prediction Using Machine Learning", 2021, DOI: 10.31224/osf.io/wa927

[3]. Marko Bohanec, Mirjana Kljajic Borstnar, Marko Robnik-Sikonja, "Explaining machine learning models in sales predictions", 2016, DOI: 10.1016/j.eswa.2016.11.010

[4]. Victor Roman, "How To Develop a Machine Learning Model From Scratch", Towards Data Science, 2018.

[5].Dataset - https://www.kaggle.com/competitions/store-sales-time-series-forecasting/data

[6].Linear Regression in python - https://realpython.com/linear-regression-in-python/

[7].XGBoost for Regression - https://machinelearningmastery.com/xgboost-for-regression/

[8]. Python Plotting With Matplotlib (Guide) - https://realpython.com/python-matplotlib-guide/

[9]. Seaborn W3Schools - https://www.w3schools.com/python/numpy/numpy_random_seaborn.asp

[10]. Exploratory Data Analysis in Python — A Step-by-Step Process - https://towardsdatascience.com/exploratory-data-analysis-in-python-a-step-by-step-process-d0dfa6bf94ee

[11].Dealing With Missing Values in Python – A Complete Guide - https://www.analyticsvidhya.com/blog/2021/05/dealing-with-missing-values-in-python-a-complete-guide/

[12]. A Complete Guide to Data Visualization in Python With Libraries, Chart, Graphs & More - https://www.simplilearn.com/tutorials/python-tutorial/data-visualization-in-python#:~:text=Python%20offers%20several%20plotting%20libraries,most%20simple%20and%20effective%20way