Strategic Sales Prediction: Time series Analysis using Machine Learning

Team Members:
Meduri Sai Krishna - saikrishnameduri@my.unt.edu
Anumolu Yaswanth Kumar - yaswanthkumaranumolu@my.unt.edu
Mandalapu Nagasai - nagasaimandalapu@my.unt.edu
Vennelaganti Goutham - gouthamvennelaganti@my.unt.edu

Abstract

Using a Kaggle dataset from 2012 to 2017, we focused on sales data along with store-specific features and categorical markers such as holidays. We used XGBoost and Linear Regression models, together with time series analysis, to predict sales trends, focusing on the effect of seasons and the relationship between sales and specific events. Our main objective was on displaying various patterns, analyzing seasonal sales impacts, and performing correlation analysis between sales and specific occurrences. The primary objective of the project was to generate actionable data that would help retailers with proactive planning strategies and enhanced sales techniques.

Problem Specification

The problem statement involves using machine learning algorithms and time series analysis to accurately predict sales patterns for retailers. It involves forecasting various sales trends, identifying the impact of different seasons on sales, and comprehending relationships between sales data and certain events or occasions. The goal is to provide actionable information that will enable retailers to optimize their strategy, improve planning, and increase overall sales success.

Design & Milestones

Data Collection and Preprocessing:

- Obtained a dataset from Kaggle containing sales information from 2012 to 2017.
- Preprocessing the data, including handling missing values, encoding categorical variables.

Exploratory Data Analysis (EDA):

- To understand the distribution of sales, seasonal trends, and associations between variables, carefully examine the dataset.
- Use statistical methods and visualizations to identify the key factors and how they affect sales.

Feature Engineering:

- Provide modeling-relevant features, including category encodings and time-based features.
- Assemble the dataset for training the model by choosing appropriate input attributes.

Model Selection and Training:

To capture complex patterns, use effective machine learning models such as XGBoost and Linear Regression. Use these to prepare and predicting future sales and transaction patterns.

Hybrid Model Development:

To improve predicted accuracy, implement and deploy a hybrid ensemble model that builds on the abilities o multiple models (e.g., XGBoost, Linear Regression).

Forecasting and Evaluation:

Predict sales trends using trained models, compare accuracy to actual sales data. Evaluate estimates, accounting for inconsistencies and fine-tuning models as required.

Insights Generation:

Establish conclusions from the model's forecasts that focuses on sales patterns, seasonal effects, and relationships with particular events. Aggregate results for retailers to easily interpret.

Data Specification

- We are using 6 csv files that contain different data
- **Train.csv and Test.csv**

Contains information of sales, product family, date and store numbers. Where sales column has a total of sales for product category at a specific store and date.

- **Store.csv**

Contains data with columns city, state, store type etc

- **Transactions.csv**

Contains data of sales, will be used to understand trends in sales.

- **Holidays and Events.csv**

Contains important data about seasonal sales, requires more data preprocessing.

- **Daily Oil Price.csv**

Contains information which will be used to see how oil prices effected other product sales negatively or positively.

- We did not require any preprocessing like filtering etc., since we are working with mostly numerical and categorical data. We just wanted to convert and transform the dates columns into pandas date format. We utilized all these data files in order to provide comprehensive analysis.
- We specifically utilized Train.csv and Transactions.csv to perform the predictions, where we focused on the predicting target variable “sales” and “transactions”.

Features and Data types:

```
print(holiday_events_df.info())
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 350 entries, 0 to 349
Data columns (total 6 columns):
 #   Column       Non-Null Count  Dtype
---  ------       --------------  -----
 0   date         350 non-null    datetime64[ns]
 1   type         350 non-null    object
 2   locale       350 non-null    object
 3   locale_name  350 non-null    object
 4   description  350 non-null    object
 5   transferred  350 non-null    bool
dtypes: bool(1), datetime64[ns](1), object(4)
memory usage: 14.1+ KB
```

```
print(oil_df.info())
<class 'pandas.core.frame.DataFrame'>
DatetimeIndex: 1218 entries, 2013-01-01 to 2017-08-31
Data columns (total 1 columns):
 #   Column      Non-Null Count  Dtype
---  ------      --------------  -----
 0   dcoilwtico  1175 non-null   float64
dtypes: float64(1)
memory usage: 19.0 KB
```

```
print(train_df.info())
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3000888 entries, 0 to 3000887
Data columns (total 6 columns):
 #   Column       Dtype
---  ------       -----
 0   id           int64
 1   date         datetime64[ns]
 2   store_nbr    int64
 3   family       object
 4   sales        float64
 5   onpromotion  int64
dtypes: datetime64[ns](1), float64(1), int64(3), object(1)
memory usage: 137.4+ MB
```

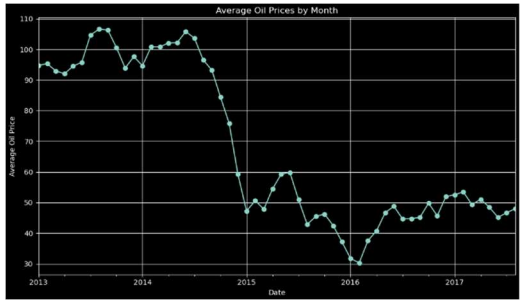
```
print(stores_df.info())
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 54 entries, 0 to 53
Data columns (total 5 columns):
 #   Column      Non-Null Count  Dtype
---  ------      --------------  -----
 0   store_nbr   54 non-null     int64
 1   city        54 non-null     object
 2   state       54 non-null     object
 3   type        54 non-null     object
 4   cluster     54 non-null     int64
dtypes: int64(2), object(3)
memory usage: 2.2+ KB
```

```
print(test_df.info())
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 28512 entries, 0 to 28511
Data columns (total 5 columns):
 #   Column       Non-Null Count  Dtype
---  ------       --------------  -----
 0   id           28512 non-null  int64
 1   date         28512 non-null  datetime64[ns]
 2   store_nbr    28512 non-null  int64
 3   family       28512 non-null  object
 4   onpromotion  28512 non-null  int64
dtypes: datetime64[ns](1), int64(3), object(1)
memory usage: 1.1+ MB
```

```
print(transactions_df.info())
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 83488 entries, 0 to 83487
Data columns (total 3 columns):
 #   Column        Non-Null Count  Dtype
---  ------        --------------  -----
 0   date          83488 non-null  datetime64[ns]
 1   store_nbr     83488 non-null  int64
 2   transactions  83488 non-null  int64
dtypes: datetime64[ns](1), int64(2)
memory usage: 1.9 MB
```

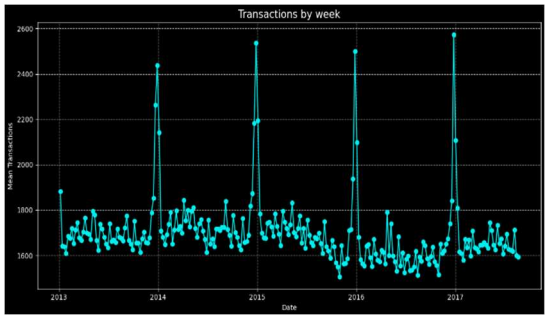
Results and Interpretation

The following plot illustrates the average distribution of oil prices by month from 2013 to 2017.



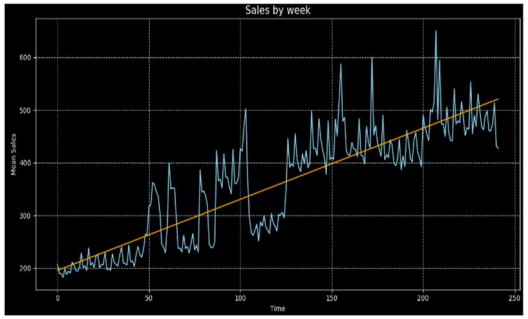
Interpretation:
This visualization shows that average prices for oil has been decreased from 2015 to 2016 as lowest and started rising.

The following plot illustrates the Transactions frequency by Week.



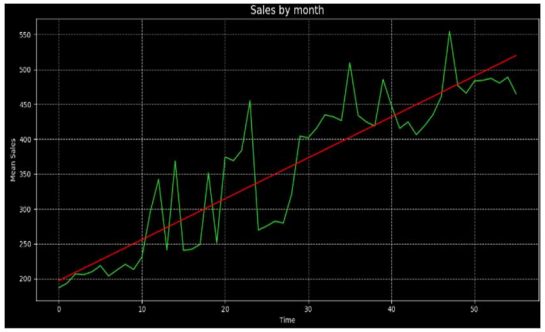
Interpretation:
This visualization shows that transactions are immensely increasing at the end of the year probably because of black Friday or Christmas holiday season.

The following plot illustrates the Sales frequency by Week with a Linear Regression Line fit.



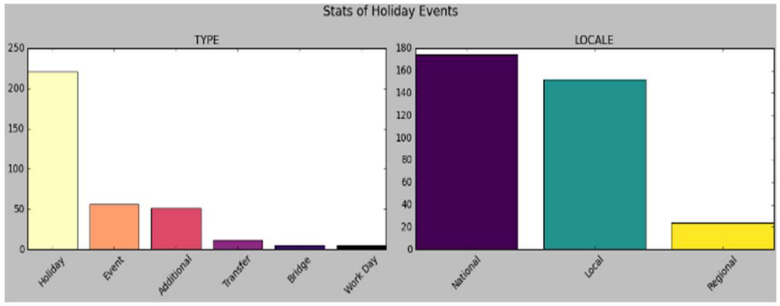
Interpretation:
This visualization shows sales by week, where the sales substantially increase on weekends, we also fit linear regression line.

The following plot illustrates the Sales frequency by Month with a linear regression Line.



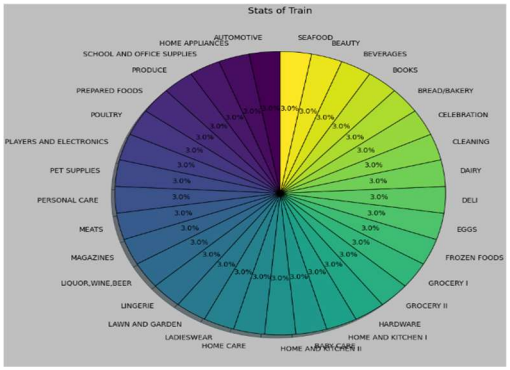
Interpretation:
This visualization shows sales by month, where the sales substantially increase at end of the year and decline in summer, we also fit linear regression line.

Statistics of Holiday events By type and Location whether its by national, locally or regional.



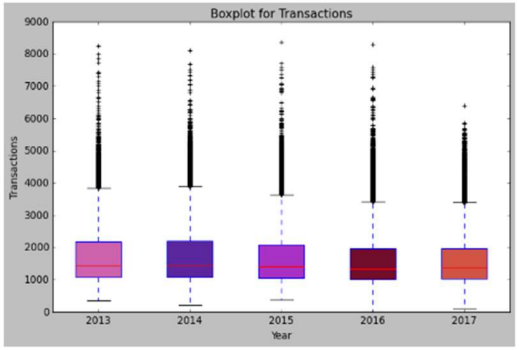
Interpretation:
This visualization shows that holiday events have most sales both nationally and locally dominating other types substantially.

The following pie chart shows the distribution of types of products in Train.csv



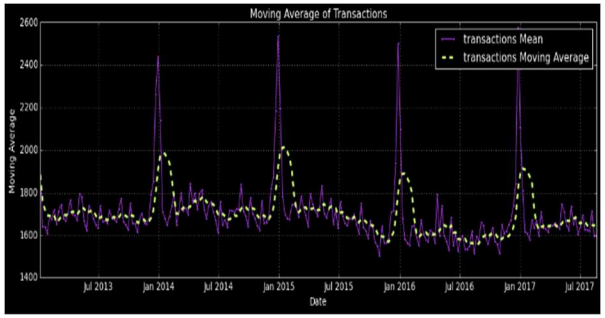
Interpretation:
This visualization shows distribution of different product families across the family column, the representation of %, is not actual percentage of the products, it is formatted so that we can visually inspect what product families are in the data.

Box plot of transactions across the years



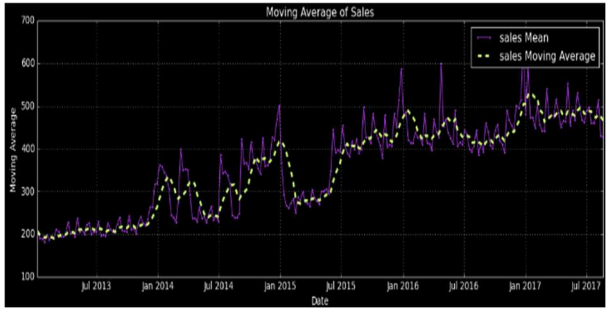
Interpretation:
This visualization shows distribution of transactions using box plots, we can see that year 2014 have most transactions.

Moving Average Identification of transactions



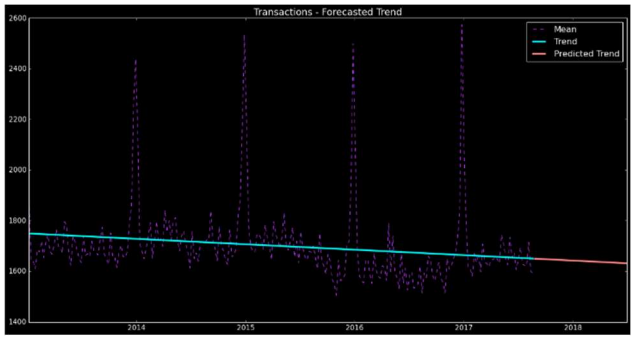
Interpretation:
This visualization shows moving average of transactions using green dotted line, we can see that how the average transaction is spread over time.

Moving Average Identification of Sales



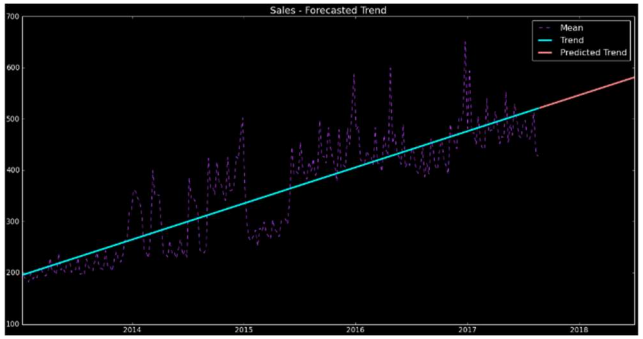
Interpretation:
This visualization shows moving average of sales using green dotted line, we can see that how the average sales is spread over time.

Transaction Prediction using Linear Regression



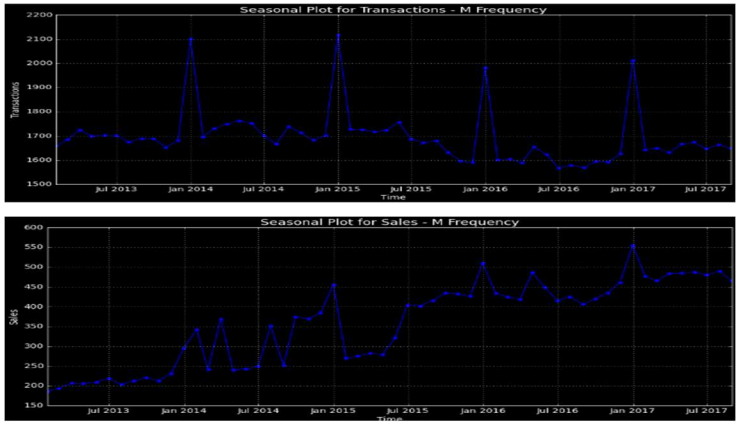
Interpretation:
This visualization shows the predicted trend in Transactions for upcoming 45 future steps using Linear regression. We can observe that it is steady and moving towards slightly decreasing number of transactions than usual

Sales Prediction using Linear Regression

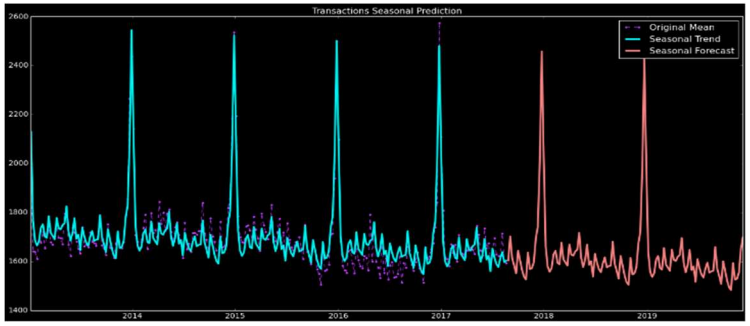


Interpretation:
This visualization shows the predicted trend in Sales for upcoming 45 future steps using Linear regression. We can observe that it is steady and Increasing number of Sales. No drastic increase.

Seasonal monthly frequency of transactions and sales across the years.

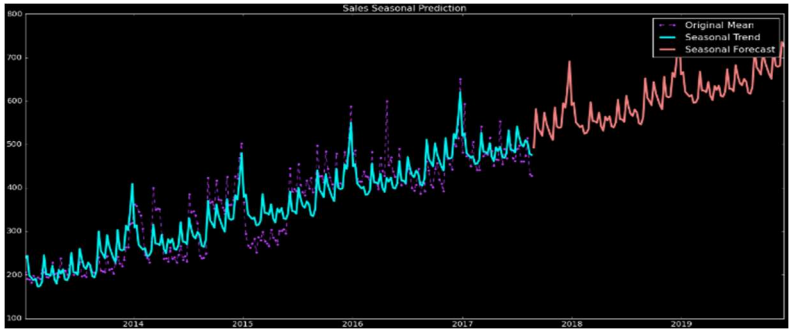


Seasonal Transaction Prediction using Linear Regression



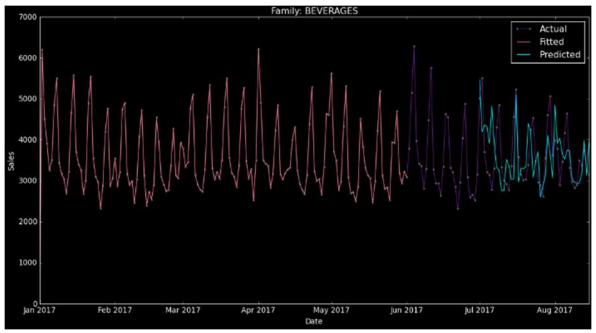
Interpretation:
This visualization shows the predicted Seasonal trend in Transactions for upcoming 120 future steps using Linear regression. We can observe that it is steady decrease but high number of sales in holiday seasons at the end of each years.

Sales Prediction using Linear Regression



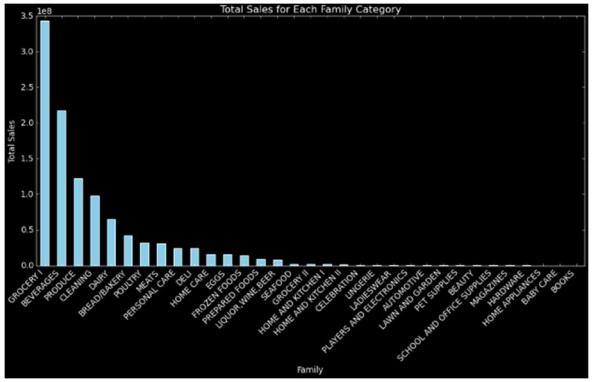
Interpretation:
This visualization shows the predicted Seasonal trend in Sales for upcoming 120 future steps using Linear regression. We can observe that it is steady increase in number of sales towards end of the year at holiday seasons.

Sales Prediction using XGBoost on BEVERGAES product family.



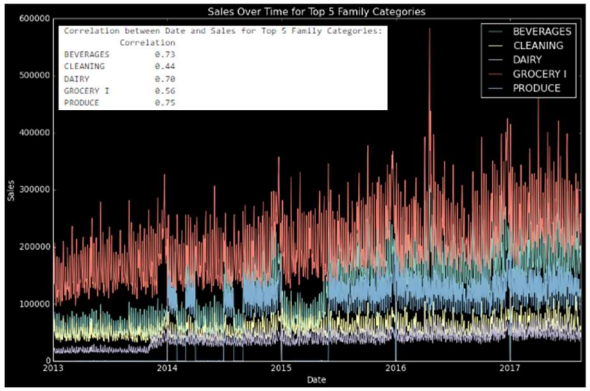
Interpretation:
This visualization shows that predicted trend in Sales for upcoming 30 days using XGBoost. We can observe that it is steady acceptable. But we can see in comparison Linear regression is best fit for our data since it is effective in predicting continours variables.

Total sales for each product family



Interpretation:
This visualization shows the distribution of sales for each product family types. We can observe that there is domination of one class that is GROCERY I and BEVERAGES indicating how frequently they are bought.

Sales Prediction using Linear Regression



Interpretation: This visualization shows the Correlation between the top product family types and dates indicating increase in sales of every type over the years.

Used Libraries

- pandas: Data manipulation and analysis.
- xgboost: optimized gradient boosting.
- sklearn.linear_model: linear regression model.
- sklearn.preprocessing: Tool for data preprocessing.
- statsmodels.tsa.deterministic: time series analysis.
- matplotlib.pyplot: data visualization and graphical analysis.
- XGBRegressor from xgboost: regression tasks, gradient boosting for predictive modeling.
- LinearRegression from sklearn.linear_model: linear relationship modeling between variables in regression problems.
- LogisticRegression from sklearn.linear_model: Binary classification tasks, predicting the probability of binary outcomes.

Challenges

- Model Complexity: Managing complex relationships in sales data and choosing appropriate models to efficiently capture trends which are complex.
- Seasonality and Trends: Considering variation by seasons and identifying long-term patterns that affect the accuracy of projections for sales.
- Feature Engineering: It might be challenging to extract appropriate characteristics and encode categorical data for the ideal model performance.
- Model Interpretability: Maintaining a balance between interpretability and model complexity will ensure clear information for business choices.

Future Direction

- Advanced Ensemble Techniques: investigating innovative ensemble techniques to improve prediction accuracy even more.
- Feature Engineering Enhancement: Experimenting with deeper feature engineering techniques which helps to obtain more in-depth insights.
- Incorporating External Data: Using other data sources for building a sales projection model which is more accurate.
- Advanced Time Series Models: Examining deep learning models or advanced time series developed for sales forecasting.
- Interactive Visualization: Creating user friendly visualizations for retailers for investigating sales data.

References and Related projects

[1] https://www.kaggle.com/datasets/shivan118/big-mart-sales-prediction-datasets

[2] Intelligent Sales Prediction Using Machine Learning Techniques

[3] Pavlyshenko, B.M. Machine-Learning Models for Sales Time Series Forecasting. Data 2019, 4, 15. https://doi.org/10.3390/data4010015

[4] Kramar, V.; Alchakov, V. Time-Series Forecasting of Seasonal Data Using Machine Learning Methods. Algorithms 2023, 16, 248. https://doi.org/10.3390/a16050248

[5] Zhang, X., Kim, T. A hybrid attention and time series network for enterprise sales forecasting under digital management and edge computing. J Cloud Comp 12, 13 (2023). https://doi.org/10.1186/s13677-023-00390-1

We have utilized all the information available in the form of famous papers and projects addressing this topic, we have made our design choices and decisions based on influence, mainly we utilized the following Kaggle competition to observe how each top contestant is handling this competition and have learned how they handled different challenges.

[6] https://www.kaggle.com/competitions/store-sales-time-series-forecasting/code?competitionId=29781&sortBy=voteCount

Contribution and Roles:

- Meduri Sai Krishna – Data collection and preparation.
- Anumolu Yaswanth Kumar – Exploratory data analysis.
- Mandalapu Nagasai – Model development and tuning.
- Vennelaganti Goutham – Model testing/evaluation and visualization.

Repository

- We will be our archived version of the project in a zip file along with data, so it can be reproducible.