Store Sales - Time Series Forecasting

Data Mining Final Project

Yu-Ting Huang	Chieh-Hsiang Hsu	Sen-Yun Ku	Po-Yen Chu	Pin-Cheng Chen
B10303029	B09704079	B09704061	B10704031	B10701219

1 Introduction

1.1 Experimental Background

Our group will participate in a competition hosted by Kaggle: *Store Sales - Time Series Forecasting*, and utilize the dataset provided by Kaggle to employ various machine learning models for predicting the sales volume of retail products.

By forecasting sales volume, product waste can be minimized, and customer satisfaction may also be enhanced by ensuring sufficient inventory levels. These benefits are advantageous for the store to predict future inventory needs.

1.2 Experimental Objectives

In the project, we use datasets with time series from *Corporación Favorita*, a large retailer in Ecuador. The aim is to utilize 4 and 2/3 years of data to predict the total sales volume of various product categories within each store over the next 16 days.

The evaluation metric is Root Mean Squared Logarithmic Error (RMSLE), which is calculated as:

$$\sqrt{\frac{1}{n}\sum_{i=1}^{n}(\log(1+p_i)-\log(1+a_i))^2}$$

where:

- n is the total number of samples
- p_i is the predicted value
- a_i is the actual value

2 Data Description

2.1 Dataset

The dataset utilized in this study comprises real-world data sourced from six primary files: `train.csv`, `test.csv`, `stores.csv`, `transactions.csv`, `oil.csv`, and `holiday_events.csv`.

2.1.1 Data Format

`train.csv` The columns and their respective descriptions are as follows:

- `id`: Number of the training data
- `date`: Sales date
- `store_nbr`: Number of the store
- `family`: Product category
- `onpromotion`: Quantity of promotion products from a specific category on a specific day at a specific store
- `sales`: The sales of a specific category on a specific day at a specific store

`test.csv` The columns are the same as in `train.csv`, except there is no sales column in the data.

`stores.csv` The fields include:

- `store_nbr`: Number of the store
- `city`: Store city
- `state`: Store state
- `type`: Store type
- `cluster`: Store cluster

`transactions.csv` The fields include:

- `date`: Sales date (training data period)
- `store_nbr`: Number of the store
- `transactions`: The sales on a specific day at a specific store

`oil.csv` The fields include:

- `date`: Sales date
- `dcoilwtico`: Oil prices in US dollars

holiday_events.csv The fields include:

- `date`: Sales date
- `type`: Event type
- `locale`: Event scale (including national and regional)
- `locale_name`: Region name
- `description`: holiday or event description
- `transferred`: whether the vacation is transferred

2.1.2 Data Volume

`train.csv`

- Total number of samples: 3,000,888
- Total number of stores: 54
- Total number of categories: 33
- Time span: 1st January 2013, to 15th August 2017

`test.csv`

- Total number of samples: 28,512
- Time span: 16th August 2017, to 31th August 2017

2.2 Explorational Data Analysis

2.2.1 Correlation between daily store sales and oil prices

In **Figure 1**, it is observed that while oil prices have been falling since 2014, store sales have been increasing over time. The overall correlation coefficient during the training data period is -0.71, suggesting that oil prices can be an important feature in predicting store sales.

2.2.2 Correlation between daily store sales and promotions

In **Figure 2**, a positive correlation is observed between the quantity of promotion items for a specific category and the median of daily store sales, indicating a stimulating effect on sales.

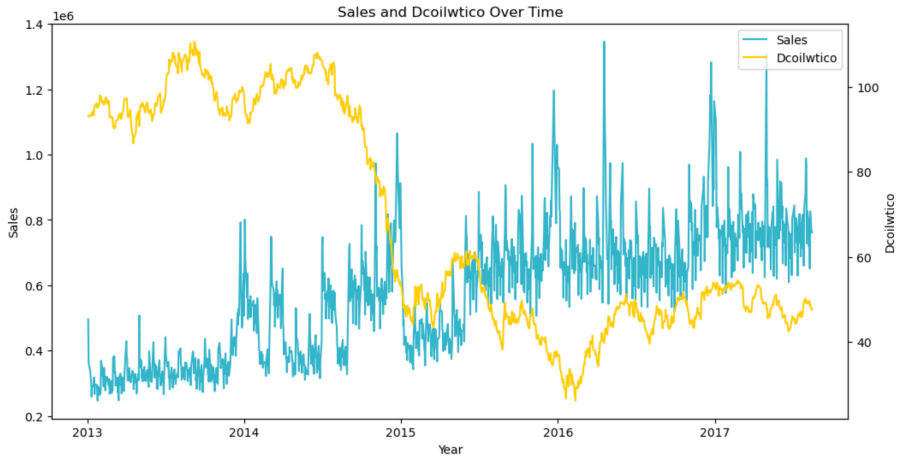

Figure 1: Trend of store sales and oil prices

Figure 2: Stores classified by cluster and their sales

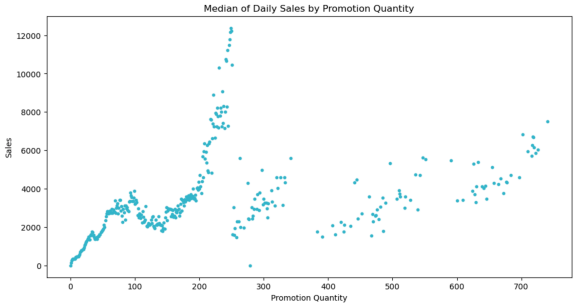

Figure 3: The influence of promotions on store sales

2.2.3 Correlation between daily store sales and store clusters

In Figure 3, the dataset provides 17 clusters for 54 stores; however, it fails to capture similar daily store sales situations due to the wide distribution within some clusters.

2.3 Data Preprocessing

2.3.1 Data Scheme

Because there are totally 6 datasets included in the project, we would like to utilize data scheme to visualize the relationship between them. In the following data prepossessing section, we would also merge the datasets as shown in **Figure 4**.

2.3.2 Data Reduction and Feature Engineering

After knowing the relationship between datasets, we further reduced data volume and improved data quality through Data Reduction and Feature Engineering.

1. **data reduction**: Only included `2015/08/15 to 2017/08/15` data for the following study.

2. **min-max normalization**: Applied min-max normalization to `"onpromotion", "dcoilwtico" and "transactions" columns`.

3. **date information extraction**: Extracted the `"year", "month",`

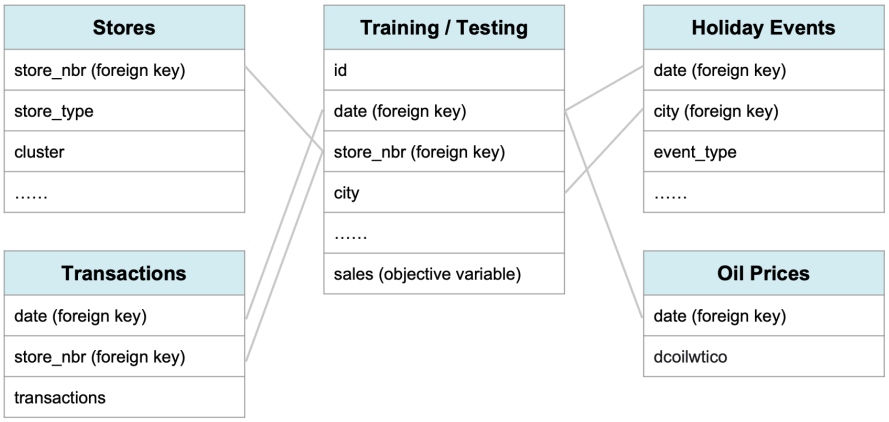

Figure 4: Data Scheme

`"day" and "dayofweek" columns` from the "date" column; Also, we mapped the `"date" column` to the holidays and the events date, generating the `"isHoliday" and "isEvent" columns`.

4. **oil price imputation**: On weekends or holidays, oil price information wasn't provided. To avoid inconsistency, we used interpolation to fill the missing value.

5. **word embedding**: Utilized Bert to convert the `"family" and "description" columns` into word vectors.

6. **spatial information extraction**: Mapped the `"city" column` to a geographic dictionary, getting the `"longitude" and "latitude"` columns.

7. **one-hot encoding**: Converted the `"categorical" columns` into one-hot encoding format.

8. **transaction data imputation**: The `"transaction" column is only available in the training data` ; thus, averaging the data in the previous two years for supplementing.

3 Methodology

Our primary goal is to develop an accurate model to predict retail sales for Corporación Favorita supermarkets using the provided dataset. We will employ a variety of methodologies to ensure comprehensive analysis and robust model performance. The methods are broadly categorized into three types: Linear Regression Models, Ensemble Learning Models, and Time Series Models.

3.1 Linear Regression Models

In this section, we explore the utilization of various linear regression models to predict retail sales for the Corporación Favorita supermarkets.

3.1.1 Multi-index Linear Model

In a multi-index linear model, each data point at a specific time (e.g., sales of a particular product in a specific store) is treated as an independent observation. This method involves running multiple linear regression models simultaneously within

a single model framework.

This method offers simplicity in implementation and allows for direct interpretation of coefficients, which helps in understanding the impact of each feature on sales. However, it may not capture nonlinear relationships and complex interactions between features, potentially limiting its predictive power.

3.1.2 Moving Average (MA)

The Moving Average model smooths out short-term fluctuations and highlights longer-term trends or cycles by averaging sales over a specific window.

This simplicity makes it effective in reducing noise and identifying trends. However, the choice of window size is crucial and may require experimentation. Moreover, the model may lag behind actual trends, especially if sudden changes occur, limiting its responsiveness to new data.

3.1.3 Exponential Smoothing

Exponential Smoothing assigns exponentially decreasing weights to past observations, making it more responsive to changes compared to Moving Average.

This model adapts more quickly to recent changes in the data and can capture trends and seasonality when extended. The critical challenge lies in selecting the appropriate smoothing parameter, which significantly impacts the model's performance. Additionally, while it adapts faster, it may still require extensions for handling more complex patterns effectively.

3.1.4 ARIMA Models

Autoregressive Integrated Moving Average (ARIMA) combines autoregression (AR), differencing (I), and moving average (MA) components to handle a wide range of time series patterns.

It is capable of modeling various patterns, including trends and seasonality, making it a robust choice for time series analysis. Despite its advantages, ARIMA requires careful parameter selection and tuning, which can be complex and time-consuming. It is also more computationally intensive compared to simpler models, necessitating significant resources for optimal performance.

3.2 Ensemble Models

In this section, we explore the utilization of various ensemble models to predict retail sales for the Corporación Favorita supermarkets.

3.2.1 Random Forest

Random Forest is an ensemble learning method that constructs multiple decision trees during training and merges their outputs to improve accuracy and control overfitting. It is known for its robustness and effectiveness in handling large datasets with higher dimensionality.

3.2.2 XGBoost

XGBoost (Extreme Gradient Boosting) is a powerful and efficient gradient boosting framework that is widely used for its speed and performance in handling large-scale datasets. Known for its scalability and robust performance, XGBoost is a popular choice in machine learning competitions and practical applications alike.

3.2.3 LightGBM

LightGBM (Light Gradient Boosting Machine) is a highly efficient gradient

boosting framework that uses tree-based learning algorithms, optimized for speed and memory usage. It is particularly well-suited for handling large-scale data and high-dimensional features, making it a popular choice for machine learning tasks.

3.2.4 CatBoost

CatBoost (Categorical Boosting) is a gradient boosting library developed by Yandex, designed to handle categorical features automatically without extensive preprocessing. It offers robust performance with fewer hyperparameters and is particularly effective for datasets with a mix of numerical and categorical data.

3.3 Multi-Index LSTM Model

Considering our dataset is composed of many time-series features, we proposed the Multi-Index LSTM Model to try to capture those characteristics.

3.3.1 Multi-Index Transformation

The column *'sales'* to predict is actually been classified by *'family'* and *'store_nbr'*. Therefore, it would not be a time-series data unless it is converted to multi-index. After setting *'date', 'family', 'store_nbr'* to index, we unstack *'family', 'store_nbr'* to get the dataset. In the reconstructed dataset, the indices are *'date'* and the number of variables are *N_Unique(family) * N_Unique(store_nbr) * N(original_variables)*, while the number of value to be predicted (*'sales'*) are *N_Unique(family) * N_Unique(store_nbr)*.

3.3.2 Create Sequence

To better capture the characteristics of time-series data, we create time sequences according to *timesteps*. The input shape of the LSTM model will be $(timesteps, number\_of\_variables)$. Specifically, given ith row to be predicted, we use row $[i - timesteps, i]$ of X to predict.

3.3.3 Model Construction

The model is composed of multiple LSTM layers and Dropout Layers, and finally a Dense Layer as the output layer. The summary is showned in **Table 1**, while the optimizer is Adam and the loss function is Mean Squared Logarithmic Error (MSLE).

4 Experiment

The evaluation metric is, as mentioned above, Root Mean Squared Logarithmic Error (RMLSE). The performances shown in this section is calculated on the test dataset given by Kaggle.

4.1 Linear Regression Models

Subsequently, we conducted sales volume predictions, and the results are as **Table 2**.

Among them, the performance of moving average stands out as the best; however, the disparities among the results of the four models are relatively minor. Overall, the performance across all models is not very good. Potential reasons for poor performance of linear regression models are as follows:

- **Non-linear Relationships**

  Linear models may struggle to capture complex non-linear relationships and interactions between features.

- **Computational Complexity**

Layer (type)	Output Shape	Param #
input_layer (InputLayer)	(None, 32, 30294)	0
lstm (LSTM)	(None, 32, 1782)	228,644,856
dropout (Dropout)	(None, 32, 1782)	0
lstm_1 (LSTM)	(None, 1782)	25,411,320
dropout_1 (Dropout)	(None, 1782)	0
dense (Dense)	(None, 1782)	3,177,306

Table 1: Multi-Index LSTM Model Summary

model	Multi-index	Moving Average	Exponential Smoothing	ARIMA
rmsle	2.706	2.643	3.732	3.943

Table 2: Linear Regression Models Performance

More complex models, such as ARIMA, can be computationally intensive and require significant resources for optimal performance.

- **Parameter Sensitivity**

  Models like Exponential Smoothing and ARIMA require careful parameter selection, which can be challenging and impact performance.

4.2 Ensemble Models

After conducting sales volume predictions, the results are as **Table 3**.

The performance of Random Forest stands out as the best; however, the disparities among the results of the four models are relatively minor. Overall, the performance across all models is not very good. Ensemble models may exhibit less-than-optimal performance in competitions centered around time series sales forecasting for several reasons:

- **Lack of Adaptability to Temporal Patterns**

  Time series data often exhibit temporal patterns, such as seasonality, trends, and cyclical variations. Some ensemble models may struggle to capture and adapt to these complex temporal patterns effectively.

- **Limited Incorporation of Time Dependencies**

  Ensemble models typically combine multiple base models independently without explicitly considering the time dependencies present in the data. This can lead to suboptimal performance when predicting time series data, where the sequential order of observations is critical.

- **Difficulty in Handling Dynamic Changes**

  Time series data often involve dynamic changes over time, such as sudden shifts in demand patterns or external factors affecting sales. Ensemble models may have difficulty quickly adapting to these dynamic changes and updating their predictions accordingly.

model	Random Forest	Xgboost	LightGBM	Catboost
rmsle	2.173	2.258	2.363	2.302

Table 3: Ensemble Models Performance

4.3 Multi-Index LSTM Model

A Single-Index LSTM model is constructed to be the benchmark, which is composed of *N_Unique(family) * N_Unique(store_nbr)* LSTM models, predicting sales of *family* and *store_nbr* independently. The performances is shown in **Table 4**.

The performance of Multi-Index LSTM is significantly better than Single-Index. Furthermore, it performs better than other models in this project, but it did not beat most competitors on Kaggle. We can conclude some keypoints to interpret the result:

- **Multi-Index is trained in a global scope**

  Comparing to Single-Index, the model get a peek of other families/stores' variables, which can better reflect the overall market condition.

- **Considered Time Dependencies**

  Comparing to other proposed models, creating time-sequence data to train in LSTM heavily focus on the time dependencies nature of the task, which might be the reason of this model outperforms other proposed models.

- **Feature Extraction may be insufficient**

  Most method posted on Kaggle use time-series-based models, which is similar to this model. Relatively unideal performance may result from insufficient interpretable features.

5 Conclusion & Future Work

In all the methods proposed in this paper, Multi-Index LSTM outperforms other models, inferring that the response variable are highly dependent to time. However, the result does not beat most Kaggle competitors, therefore, some improvements can be made in future work. Meanwhile, some real-world suggestions for Favorita supermarket are also provided in this section.

5.1 Prediction Method Improvements

Overall, our prediction methods are unable to beat most Kaggle competitors. There are several improvements to be made in the future:

- **Extract Time-based Features**

  Since time-series model outperforms other models, it is highly possible that time-based features are more interpretable. In most implementation of Kaggle competitors, they extract a lot of time-based features like inferring seasons by holidays. Those features are relatively insufficient in our methods comparing to others.

- **Increase Computing Power & Memory**

model	Single-Index LSTM	Multi-Index LSTM
rmsle	2.5171	1.25308

Table 4: LSTM Models Performance

In the Multi-Index LSTM method, many features are dropped (including *month*, *day*, *BERT* related...etc) due to lack of sufficient memory while some of these features may be helpful. Alternatively, we can also investigate importance of each feature and optimize the result given limited memory or computing power.

5.2 Real World Recommendation

Through this competition, we aim to demonstrate the practical applications of machine learning and data analysis in the retail industry. Our goal is not only to achieve accurate predictive results but also to provide feasible solutions for Favorita supermarkets. By leveraging these advanced technologies, we hope to assist Favorita in enhancing its operational efficiency, optimizing inventory management, reducing costs, and ultimately achieving better business performance. This initiative will showcase the potential of data-driven decision-making in transforming retail operations and driving sustainable growth.

5.2.1 Application

A successful predictive model will bring significant business benefits to Favorita supermarkets and has numerous applications across various domains.

- **Inventory Optimization**

  By accurately predicting sales volume, supermarkets can better manage inventory, avoiding overstock and stockout situations, thereby reducing operational costs. This helps to improve supermarket efficiency, ensure that goods supply meets demand, and further enhance competitiveness. By optimizing inventory management processes, supermarkets can effectively utilize funds and resources, increase inventory turnover, reduce holding costs, and ensure an adequate supply of goods when needed, thereby increasing sales and profitability.

- **Food Waste Reduction**

  Accurate sales forecasting can help supermarkets avoid over-purchasing, thereby reducing food waste. This not only reduces costs but also helps to achieve environmental goals and enhance the company's corporate social responsibility image. By reducing food waste, supermarkets can effectively reduce their negative impact on the environment and promote sustainable development.

- **Increased Customer Satisfaction**

  Ensuring popular items are not out of stock can enhance the shopping experience and loyalty of customers. This will result in higher customer retention rates and better reputation, ultimately leading to higher sales. By improving customer satisfaction, supermarkets can establish a good brand image,

attract more customers, and promote word-of-mouth marketing, further enhancing market competitiveness.

5.2.2 Future Work

Building on the successful experience of the model, the team aims to expand the prediction scope to more stores and different product categories, thereby broadening the impact and utility of the predictive analytics. This expansion will allow for the capture of a wider array of sales patterns and trends, enhancing the overall accuracy and applicability of the model across the enterprise.

Additionally, the team can continuously monitor the predictive performance of the model, making dynamic adjustments and optimizations based on actual situations. This proactive approach will ensure that the model remains efficient and accurate over time, adapting to any changes in the market or operational conditions. Moreover, the plan is to transfer the technology and training gained from this competition to other data analysis and prediction projects within the company. By doing so, the overall data analysis capabilities of the enterprise will be elevated, fostering a culture of data-driven decision-making and empowering various departments to leverage advanced analytics for improved business outcomes.

References

[1] Alexis Cook, DanB, inversion, Ryan Holbrook. (2021). *Store Sales - Time Series Forecasting. Kaggle.* `https://www.kaggle.com/competitions/store-sales-time-series-forecasting/overview`.