Machine Learning and Related Applications

Coursework – Technical Report

1. Synthetic features of machine learning

Synthetic data is data that is generated using machine learning algorithms that can mimic the natural real-world data. The benefits of using synthetic data are,

- Increasing the performance of a machine learning model by providing information that is not present in the original data.
- Reduce data requirement in scenarios where data is scarce.

Synthetic data can be generated using,

- Generative Adversarial Networks (GANs)
- Variational Autoencoders (VAEs)
- Autoregressive models

The generated synthetic data is mainly used in the industry to train neural networks and ML models. For example, data can also be used for fraud detection by generating fake data that is like real data and train the model to detect fake data.

2. Over-Forecasting and Under-Forecasting Error

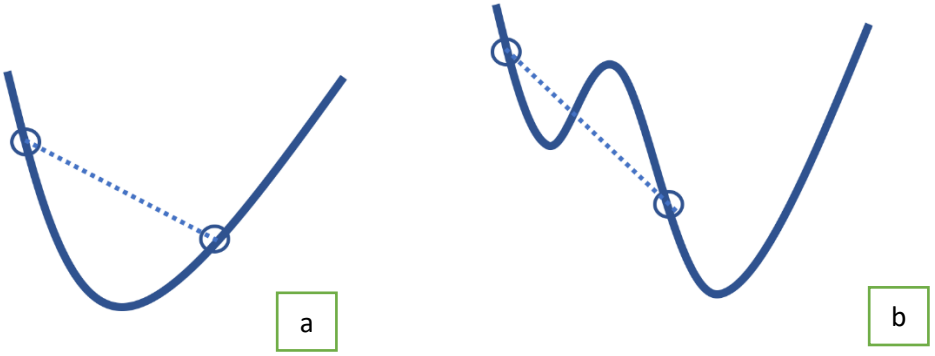
Over-Forecasting error occurs when the machine learning model predicts a value higher than the actual value. Under-Forecasting is when the model predicts a value lower than the actual value. These errors can be due to,

- Data Quality
- Model Complexity – More Complex the model, more likely it is to make errors.
- Data Noise

Example: A logistics company that uses a machine learning model to predict delivery times might over-forecast delivery times and end up with idle trucks. This could lead to increased costs. An under-Forecasting error can give false hopes to customers. A buffer stock can be used to meet the demand in the case of a under-forecasting.

3. Gradient descent algorithm and learning rate.

Gradient descent is an optimization algorithm that is used train machine learning algorithms by finding the minimum of a function. It works by iteratively taking steps in the direction of the negative gradient of the function. The gradient descent algorithm only works for functions that are differentiable and convex.

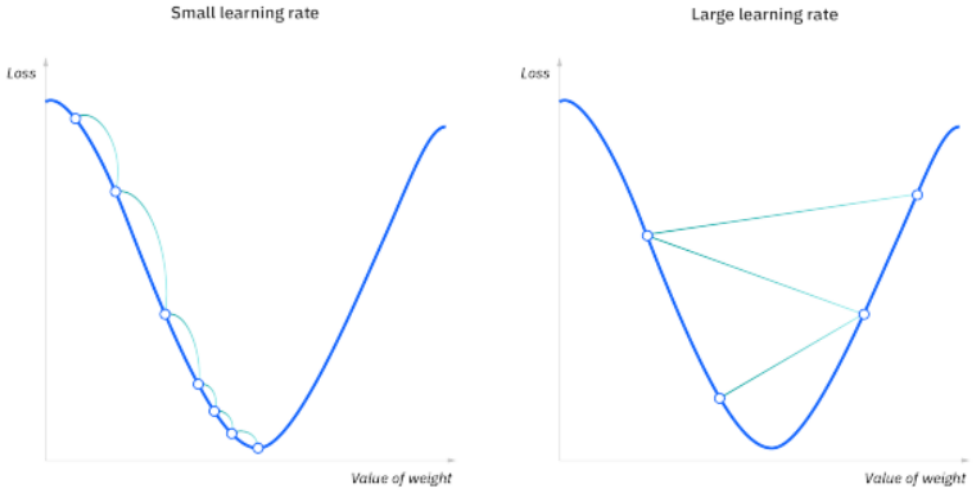


Graph b has a saddle point, and it is a real challenge to carry out a gradient descent on it since obtaining a global minima on it is not guaranteed.

There are three types of gradient descent algorithms.

a) Stochastic gradient descent -
b) Mini-batch gradient descent
c) Batch gradient descent

The learning rate is a hyperparameter that controls the size of the steps that are taken during gradient descent. A larger learning rate will cause the algorithm to take larger steps, which can lead to faster convergence. However, a larger learning rate can also cause the algorithm to overshoot the minimum of the function. A smaller learning rate will cause the algorithm to take smaller steps, which can lead to slower convergence. However, a smaller learning rate can also help the algorithm to avoid overshooting the minimum of the function.



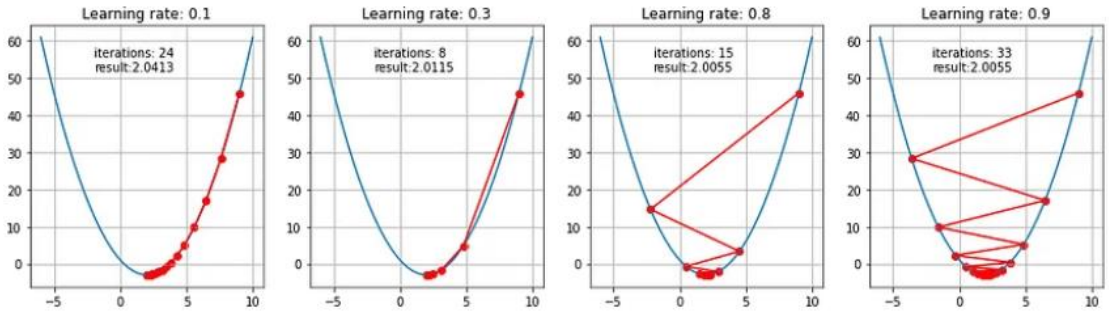
Code Snippets

```python
import numpy as np

def gradient_descent(start, gradient, learn_rate, max_iter, tol=0.01):
  steps = [start] # history tracking
  x = start
  n=0

#Obtain difference between actual and predicted values (Cost)
  for _ in range(max_iter):

    diff = learn_rate*gradient(x)
    if np.abs(diff)<tol:
      break
    x = x - diff
    steps.append(x) # history tracing
```



4. Market Basket analysis

This is a technique that is used mainly by retailers to discover relationships between various items. The algorithm works by looking at items that occur together often. A main part of market basket analysis is the determination of association rules, which give the relationship between the items in the market basket.

For example: If a customer buys baby food, then they are most likely to buy diapers and baby formula as well.

Market basket analysis is useful for:

a) Product placement – Place products that are purchased together near each other to push more customers to purchase them.
b) Targeted Marketing – Target marketing campaigns towards customers that are likely to buy certain products.
c) Customer Segmentation – Find groups of customers that have similar purchasing patterns and develop specialized marketing strategies for those groups.

Apriori Algorithm

This algorithm is used to determine the association rules. It works by first determining the single items that are purchased frequently. Then it expands to finding two items brought most frequently together. Likewise, it repeats this process until all frequent itemsets have been found. The apriori algorithm uses pruning techniques to reduce the number of possible itemsets.

(Code Snippet provided in the attached notebook folder)

5. Sales forecasting problem

1. Introduction

Sales forecasting has become a vital tool for organizations to efficiently plan and manage their operations in the always changing retail market. The purpose of this case study is to address the difficulty in estimating sales for the top five rapidly evolving categories in certain departments and retailers. I will pay particular attention to Stores A, B, C, D, and E's Grocery, Beverages, and Chilled divisions.

1.1 Statement of Purpose

This case study aims to provide a reliable machine learning system that can predict sales for the following week in the chosen categories and stores. My goal is to provide the chosen departments and stores with actionable insights to enhance inventory management, expedite operations, and make data-driven decisions by utilizing advanced analytical methodologies and historical sales data.

For merchants to maintain appropriate stock availability, reduce expenses related to excess inventory or stockouts, and improve customer happiness, accurate sales forecasting is crucial. This can help the chosen departments and stores efficiently meet consumer demand and enhance their overall business performance by projecting sales for the top 5 fast-moving categories.

The departments of Groceries, Beverages, and Chilled are well-known for their rapid sales growth and considerable revenue contributions. By concentrating on these categories, we can address the primary forces behind retail shop sales and offer insightful information that is useful for inventory planning, purchasing choices, and resource allocation.

Additionally, I want to be able to capture the variations in consumer behavior, tastes, and regional market dynamics by taking into account the five stores I've chosen (A, B, C, D, and E). This will allow for a more detailed analysis and forecasting that is individual to each store, producing predictions and plans that are more personalized and precise.

Patterns, trends, and seasonality that affect sales performance can be discovered through the use of machine learning algorithms and the analysis of historical sales data. This data will act as the basis for creating prediction models that can precisely anticipate sales for the future week. With the aid of the projections, the chosen departments and stores will be able to plan ahead with regard to inventory management, resource allocation, and strategy alignment with customer demand.

For a strong machine learning approach to anticipate sales for the top 5 fast-moving categories in particular departments and locations, check this case study's conclusion. I want to deliver precise and useful insights that will enable the departments and stores to optimize their operations, enhance inventory management, and increase their competitive advantage in the retail sector.

2. Methodology

The main aim of this case study was to analyze the factors affecting the selling price of items in 5 outlet types, spread across various item types in various departments and sub-departments. Analyzing the data given, it was noted that since the data was in the daily format, it needed to be converted to the weekly format. In order to answer some of the questions posed in the case study and also for the sake of convenience, the data was combined to form a master table, containing the variables of interest from the 3 separate datasets provided.

After carrying out some exploratory data analysis, the problem was defined as a timeseries analysis. This was done for the weekly data since the final prediction needed to be for the following week. It was understood that even though the data was in the form of weeks, if the weeks were given concurrent numbers, the problem can be simplified into predicting the sales of the next number.

Ex: If the data was given up to 48 weeks, then the prediction will need to be made only for the 49^{th} week.

However, since the time series analysis did not yield favorable results, a regression approach was considered using all of the available variables but not fully achieved due to time constraints.

3. Implementation

3.1 Data Preparation

3.1.1 Data Sources

The data was provided in the form of 3 .csv files namely:

- Item_info.csv
- Outlet_info.csv
- Transactions_info.csv

3.1.2 Features Construction

- A feature called “Week” was designed to capture the weekly sales data. There were two separate analyses carried out by obtaining the sum of the weekly sales data and also the average of the weekly sales data.
- Several columns were developed to capture the lag of the time series data.
- Data was broken down according to outlet code, weekday and item category.

3.1.3 Target Construction

The target variable in this case, price, was in the daily format and both the sum and the average daily sales were taken to calculate the weekly data. There were 3 missing dates which were filled with zero values in order to help the analysis further down the line.

3.1.4 Master table Construction

In order to extract all of the useful information from the given data, the tables needed to be merged to create a master table. Transaction table and outlet table were merged on ‘outlet_code’. The resulting table was merged with item table on ‘item_code’.

3.2 Model Development

After organizing the data into weekly data, the consecutive differences were obtained to ensure the linearity of the data and then the lag data was taken in order to subject the data into SARIMAX modeling down the line. This data was then scaled using the min-max scaler and split into testing and training datasets.

A function was created to fit the relevant models to the training dataset and evaluate the fit and also to plot the results with the training and testing data. The following approaches were used:

- Linear Regression
- Random Forest
- XG Boost
- LSTM

Since random forest showed the best performance, Hyper-Parameter tuning and a grid search with cross validation were performed to increase accuracy. SARIMAX modeling also was attempted but efforts were halted midway due to time restrictions. The data was also set up for a regression modeling using all of the available data but could not be completed due to the above-mentioned reasons.

3.3 Risks and Assumptions

- Since the dataset starts with data on a Saturday, each week was assumed to start on Friday.
- The days on which there was no information available were considered to have no data since the outlets island-wide were closed due to a public holiday and hence no sales happened on those days.

4. Findings and Conclusions

- Following are the top 5 fast moving items in each outlet type:

outlet_code	item_code	item_category	
A	898	Powdered Milk	5721.000000
	36808	Ambient Instant Noodles	20316.000000
	116836	Crackers	4173.000000
	117520	Fat Spread	5472.000000
	123307	Ambient Liquid Milk	26232.000000
B	36808	Ambient Instant Noodles	10203.000000
	42154	Rice	2912.754002
	102490	Rice	4237.218000
	117520	Fat Spread	2566.000000
	123307	Ambient Liquid Milk	11102.000000
C	898	Powdered Milk	10165.000000
	36808	Ambient Instant Noodles	12125.000000
	102490	Rice	3293.763008
	119554	Single Consumption RTD Beverages	7905.000000
	123307	Ambient Liquid Milk	18374.000000
D	36808	Ambient Instant Noodles	18006.000000
	96136	Rice	5480.386998
	102490	Rice	9713.550011
	117520	Fat Spread	5712.000000
	123307	Ambient Liquid Milk	18094.000000
E	898	Powdered Milk	1258.000000
	36808	Ambient Instant Noodles	3096.000000
	96136	Rice	976.547001
	102490	Rice	2130.203001
	123307	Ambient Liquid Milk	3816.000000

In the above result, there are two kinds of rice in each outlet type. This can be considered as two varieties of rice with different item codes.

- Top 5 fast moving item categories and the associated products

```
item_category                      item_code
Ambient Dessert Syrups & Toppings  102967        662.0
Ambient Instant Noodles            36808       63746.0
                                   1672        11849.0
                                   6163         9155.0
                                   91438        7805.0
                                                ...
Sweet Biscuits & Cookies Regular   111382       2898.0
Vinegar                            124396        876.0
Wheat                              40309         438.0
Whipping Cream                     7396         2342.0
                                   42658         194.0
Name: sales_qty, Length: 161, dtype: float64
```

- When Considering the Weekday, on each weekday, the most sold item by quantity was Powdered milk (Please refer the python files provided)

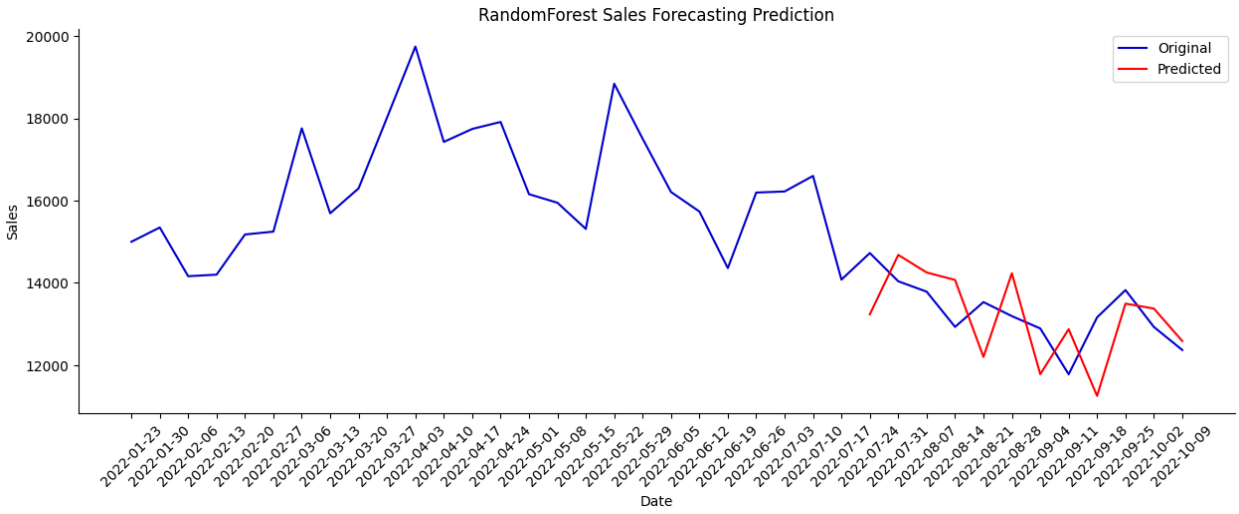
Weekday	item_code	item_category	sales_qty
Friday	898	Powdered Milk	3297.000000
	36808	Ambient Instant Noodles	8909.000000
	102490	Rice	3494.784006
	119554	Single Consumption RTD Beverages	2626.000000
	123307	Ambient Liquid Milk	10808.000000
Monday	898	Powdered Milk	2331.000000
	36808	Ambient Instant Noodles	9074.000000
	102490	Rice	2727.909001
	119554	Single Consumption RTD Beverages	2289.000000
	123307	Ambient Liquid Milk	8512.000000
Saturday	898	Powdered Milk	3375.000000
	36808	Ambient Instant Noodles	10482.000000

Weekday	item_code	item_category	sales_qty
	102490	Rice	4018.228007
	117520	Fat Spread	3028.000000
	123307	Ambient Liquid Milk	11755.000000
Sunday	898	Powdered Milk	4141.000000
	36808	Ambient Instant Noodles	9886.000000
	102490	Rice	3262.706003
	117520	Fat Spread	2712.000000
	123307	Ambient Liquid Milk	11243.000000
Thursday	898	Powdered Milk	2766.000000
	36808	Ambient Instant Noodles	8471.000000
	102490	Rice	3301.234005
	119554	Single Consumption RTD Beverages	2483.000000

- With regards to the regression and timeseries modeling, the following gives the summary of the performance of the regression models.

	index	RMSE	MAE	R2
0	LinearRegression	2505.758723	1817.836083	10.105101
1	LSTM	2023.061522	1547.546083	6.238727
2	XGBoost	1560.172310	1148.738750	3.305159
3	RandomForest	990.275685	892.638250	0.734428

As it can be seen above, the random forest regressor performs the best with the lowest R2 values compared to all other models.



The Random Forest regressor can perform better than other machine learning models due to its capability for managing high-dimensional data, tolerating missing values, and accurately capturing non-linear relationships between features and target variables. It makes use of the strength of ensemble learning, which decreases overfitting and increases generalization, by combining a number of decision trees. Due to its integrated feature selection approach, it also enhances model performance and does not require extensive feature engineering.

Even though hyper-parameter tuning and Grid search with cross validation were performed, they did not improve the accuracy of the predicting model.

In conclusion, it can be said that in order to predict the most accurate sales for the future, a random forest regressor can be used with the following parameters:

```
{'bootstrap': True,
 'ccp_alpha': 0.0,
 'criterion': 'squared_error',
 'max_depth': None,
 'max_features': 1.0,
 'max_leaf_nodes': None,
 'max_samples': None,
 'min_impurity_decrease': 0.0,
 'min_samples_leaf': 1,
 'min_samples_split': 2,
 'min_weight_fraction_leaf': 0.0,
 'n_estimators': 100,
 'n_jobs': None,
 'oob_score': False,
 'random_state': None,
 'verbose': 0,
 'warm_start': False}
```