Machine Learning for supermarket sales prediction

1st Yao An Lee
Department of Mechanical and Aerospace Engineering, University of Florida, Gainesville, FL 32611 USA
danielyaoanlee@gmail.com

***Abstract*—Over the years, Machine Learning(ML) has been widely used to analyze the contribution between attributes and predict the profit from sales data in different industry. This study proposes several regression and classification tasks over supermarket sales dataset to analyze the relationship between attributes and effectively predict targets. In my study, I compare the performance between Linear and Lasso Regression [1] toward the prediction of gross income and unit price. Classification and feature interaction of gender and customer type is done by using Logistic Regression [2] and Polynomial Feature [3]. I also compare the performance between Logistic Regression, Decision Tree and Random Forest Classifier [4] regarding the classification of day of week. The metric to compare model performance is determined by the accuracy of classification, the coefficient of determination, confusion matrix and their Confidence interval. All models are tuned using hyperparameter tuning techniques to maximize performance of each model.**

***Index Terms*—Machine Learning, Linear Regression, Logistic Regression, Polynomial Feature, Decision Tree, Random Forest Classifier.**

I. Introduction

Machine learning has developed into a widely used framework for approximating statistical and algebraic relations over a given set of data for decades. The ability of algorithms to accurately predict outcomes and generalize relationships between data sets has been beneficial in many working environments including health-care, government, transportation, and more. Two most widely known methods for machine learning are supervised learning and unsupervised learning. Supervised learning algorithms learn and train with labeled data. A few applications for supervised learning are regression, classification, and prediction. Unlike supervised learning, unsupervised learning does not learn and train with labeled data. Instead, it learns the structure and similarities within the unlabeled data. Popular applications for unsupervised learning are clustering, classification and function approximation.

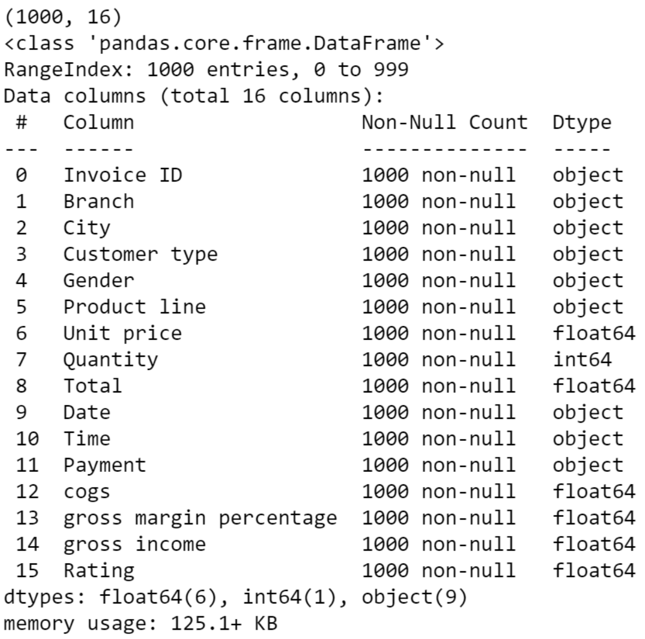
Contributions: The contributions of this study are summarized as follows:

- To create an end-to-end Machine Learning pipeline for sales data analysis and prediction.
- I propose Lasso Regression model for gross income and unit price prediction and analyze the most informative feature based on its trained parameter values. The performance evaluation is made on the basis of its coefficient of determination, mean squared error and confidence interval.
- I implement Polynomial Feature before the classification of gender and customer type to introduce the 2-nd order interaction term and explain the relationship between attributes.
- I propose three different algorithms to predict day of week and evaluate the performance based on accuracy of labels prediction and the computational efficiency.

II. Implementation

A. Data

The data set contains 1000 samples each for 16 attributes (Fig. 1). No data is missing so we can skip the imputer.



```
(1000, 16)
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1000 entries, 0 to 999
Data columns (total 16 columns):
 #   Column                   Non-Null Count  Dtype  
---  ------                   --------------  -----  
 0   Invoice ID               1000 non-null   object 
 1   Branch                   1000 non-null   object 
 2   City                     1000 non-null   object 
 3   Customer type            1000 non-null   object 
 4   Gender                   1000 non-null   object 
 5   Product line             1000 non-null   object 
 6   Unit price               1000 non-null   float64
 7   Quantity                 1000 non-null   int64  
 8   Total                    1000 non-null   float64
 9   Date                     1000 non-null   object 
 10  Time                     1000 non-null   object 
 11  Payment                  1000 non-null   object 
 12  cogs                     1000 non-null   float64
 13  gross margin percentage  1000 non-null   float64
 14  gross income             1000 non-null   float64
 15  Rating                   1000 non-null   float64
dtypes: float64(6), int64(1), object(9)
memory usage: 125.1+ KB
```

Fig. 1. Number of samples in each attribute.

The train and test split is considered to be 80:20 for all algorithms.

B. Lasso Regression

I use the open source sklearn [5] library to implement Lasso algorithm. In order to simplify the process of hyperparameter tuning, Lasso is implemented using pipeline.

C. Logistic Regression and Polynomial Features

Polynomial Features and Logistic Regression can be implemented using sklearn library. The relationship between attributes can be studied by adding interaction terms before implementing Logistic Regression. Interaction terms can be produced by using Polynomial Features and set interaction only to true.

D. Decision Tree and Random Forest Classifier

For implementing these two algorithms, we use open source sklearn library to set them as pipeline.

III. Experiment

A. Data preprocessing

Deeply look into the data, we can see that:

Attribute Invoice ID is just the identification number.

Attribute City represents the same thing as Branch, each City matches with one of the branch. Since Branch states the easier one, I decide to keep Branch.

Attribute Total is cogs times gross margin percentage, and gross margin percentage is the same in all rows. So Total is basically the same as cogs for model and there is no need to keep it.

Attribute cogs is Unit price times Quantity, and we can get gross income by multiply cogs with 0.05. Since gross income is one of our target, we decide to eliminate cogs.

Attribute gross margin percentage is the same in every rows. There is no need to keep it.

Attribute Rating barely contribute to the data, I decide to drop it.

After dropping these attributes, only seven categorical attributes and three numerical attributes remain. In order to reduce the computational expense and increase performance, a common way is to encode categorical attributes into numerical values and scale numerical attributes to eliminate the influence of outliers and unbalance scale. Two common encoding methods are Ordinal Encoding and One-Hot Encoding. Ordinal Encoding encodes categorical value into integers in numerical order based on the number of class in the specific attribute. Unlike Ordinal Encoding, One-Hot Encoding creates new (binary) columns, indicating the presence of each possible value from the original data, new binary variable is added for each unique categorical value. Two common numerical scalers are Min-Max Scaler and StandardScalr, Min-Max Scaler is widely used but very sensitive to outliers.

For categorical attributes like Date and Time, they require additional preprocessing implementation. Due to the fact that data only contains historical sales in three months, we should encode Date into day of week and Time into timeslots(morning, afternoon, evening, night) to increase the repeatability of data and relation between attributes.

Among these seven categorical attributes, two of them are binary terms. We should implement Ordinal Encoding rather than One-Hot Encoding to avoid increasing extra columns. The other four categorical attributes then implement One-Hot Encoding. For numerical attributes, Standardscaler is implemented in order to eliminate the influence of outliers.

When special circumstances such as the prediction of day of week. We change the encoding method from One-Hot Encoding to Ordinal Encoding to allow target labels lie in the same column.

B. Hyperparameter tuning for Lasso Regression and Logistic Regression

Lasso Regression requires only one hyperparameter to be tuned, which is α, the constant that multiplies the L1 term, controlling regularization strength.The hyperparameter is selected using Cross Validation(CV) Grid search between the range [1e-7, 1e-6, 1e-5, 1e-4, 1e-3, 1e-2, 0.1, 1, 2]. Similarly, Logistic Regression requires only two hyperparameter to be tuned, which is solver, algorithm to use in the optimization problem, and penalty. For solver, we Grid search between['newton-cg', 'lbfgs', 'sag', 'saga']; for penalty, we select ['l2', 'none']. The reason why I didn't select L1 as an option is due to the limitation of solver support.

C. Hyperparameter tuning for Decision Tree and Random Forest Classifier

Decision Tree and Random Forest Classifier have similar parameters, which Random Forest Classifier is just a meta estimator that fits a number of decision tree classifiers on various sub-samples of the dataset and uses averaging to improve the predictive accuracy and control over-fitting.

For Decision Tree, we can follow the following procedure: (i) criterion: ['entropy', 'gini']; (ii) number of maximum depth: [1, 2, 3, 4, 5]; (iii) number of minimum samples split: [2, 3, 4, 5]; (iv) number of minimum samples leaf: [1, 2, 3, 4].

Similarly, for Random Forest Classifier, procedure are: (i) criterion: ['entropy', 'gini']; (ii) number of maximum depth: [1, 2, 3, 4, 5]; (iii) number of minimum samples split: [2, 4, 6, 8]; (iv) number of minimum samples leaf: [1, 2, 3, 4] with an extra term:(v) number of estimators: [100, 150, 200].

The hyperparameter is selected using Cross Validation(CV) Grid search.

D. Correlation matrix

Correlation matrix is introduce to provide basic understanding of relation between features and target. By focusing on the attributes with high values, the results allow us to check if the coefficients of Lasso regression are correctly weight and excluded.

IV. Results and discussion

A. Prediction of gross income

Experiments show that high correlation between unit price, quantity and gross income correspond to the coefficients obtained by implementing lasso regression. Except unit price and quantity, other attributes do not contribute and are excluded in the model. The r2 score in test set for both linear algorithms are in the range of 95 percent confidence interval (Fig. 2) and are also close to the train set. However, by comparing two models, higher error, high coefficients and larger confidence interval for linear regression implies that lasso demonstrates a greater degree of precision and more unlikely to overfit. As a result, based on hyperparameter tuning, Lasso Regression with alpha=0.01 (Fig. 3) has test set r2 score = 0.903 and is the best hyperparameter setting in this problem.

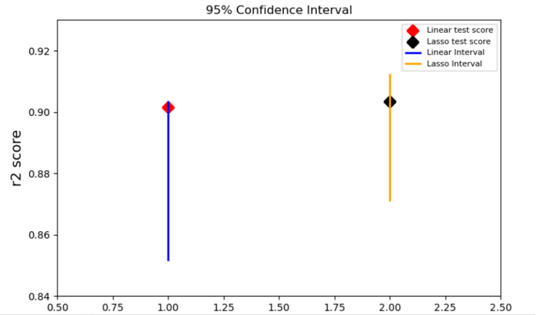


Fig. 2. Test score in 95 percent confidence interval.

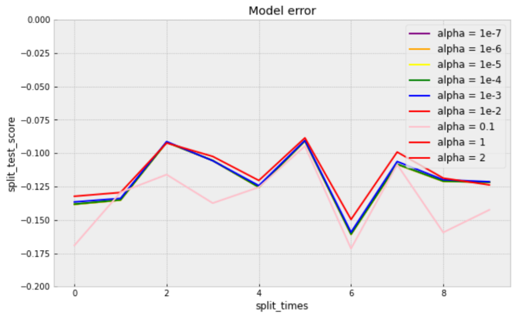


Fig. 3. Error in CV.

B. *Prediction of unit price*

Unlike prediction of gross income, the results of correlation matrix show that only one attribute is highly correlated with the target and we can then assume that some other attributes may not be excluded. The trained coefficients in lasso regression prove this assumption that more than one features are used in training. Despite the fact that most weights are small, they still contribute to prediction. Among these attributes, gross income, Gender and Branch B positively contribute to the model. Quantity, Tuesday, Afternoon and Credit card negatively contribute to the model. The rest attributes contribute nothing to the prediction.

The r2 score in test set is in the range of 95 percent confidence interval (Fig. 4) and is close to the train set. However, by comparing two models, higher error and larger confidence interval for linear regression implies that lasso regression demonstrates a greater degree of precision and be a better option to this problem. As a result, Lasso Regression with alpha=0.01 (Fig. 5) has test set r2 score = 0.802 and is the best hyperparameter setting in this problem.

C. *Attributes relation for gender classification*

Experiments introduce the second-order interactions when gender = male through polynomial features and logistic regression. The interactions help modeling the model and are presented by positive logistic regression coefficients in Fig 6. Among all these attributes and interactions, interactions between Fashion accessories and Cash has the highest parameter value and is the most informative attribute.

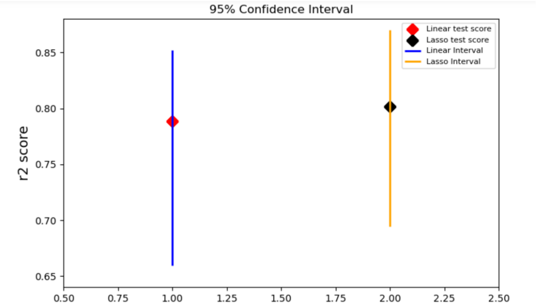


Fig. 4. Test score in 95 percent confidence interval.

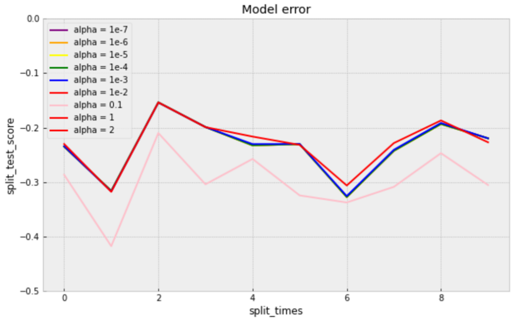
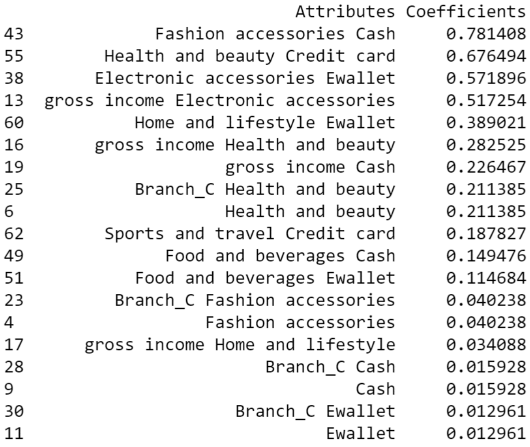


Fig. 5. Error in CV.

The logistic regression is hyperparameter tuned and the best setting is when penalty = 'none' and solver = 'newton-cg'. However, since gender is weakly correlated with attributes, there are huge gap between train and test accuracy score, which implies the algorithm is overfitting. Also, the large confidence interval demonstrates that the model does not provide a precise representation of the population mean.



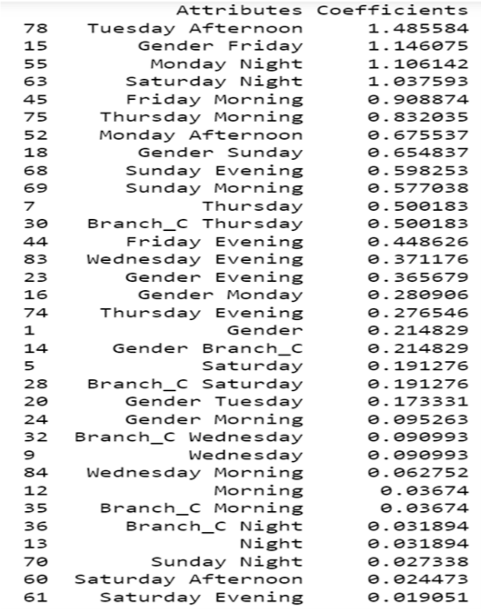
```
                          Attributes  Coefficients
43          Fashion accessories Cash      0.781408
55      Health and beauty Credit card     0.676494
38     Electronic accessories Ewallet     0.571896
13 gross income Electronic accessories    0.517254
60         Home and lifestyle Ewallet     0.389021
16        gross income Health and beauty  0.282525
19                  gross income Cash     0.226467
25        Branch_C Health and beauty      0.211385
6                  Health and beauty      0.211385
62       Sports and travel Credit card    0.187827
49             Food and beverages Cash    0.149476
51          Food and beverages Ewallet    0.114684
23        Branch_C Fashion accessories    0.040238
4                Fashion accessories      0.040238
17      gross income Home and lifestyle   0.034088
28                      Branch_C Cash     0.015928
9                                Cash     0.015928
30                   Branch_C Ewallet     0.012961
11                            Ewallet     0.012961
```

Fig. 6. Positive interactions when gender = male.

D. *Attributes relation for customer type classification*

Second-order interactions when customer type = normal are also introduced to help modeling the model. It is presented by positive logistic regression coefficients in Fig 7. Among all these attributes and interactions, interactions between Tuesday and Afternoon has the highest parameter value and is the most informative attribute. Other than that, (Gender = male, Friday), (Monday, Night) and (Saturday, Night) also have parameters higher than 1.

The logistic regression is hyperparameter tuned and the best setting is when penalty = 'none' and solver = 'newton-cg'. Similarly, customer type is also weakly correlated with attributes. As a result, huge difference between train and test accuracy score implies the algorithm is overfitting. Also, the large confidence interval demonstrates that the model does not provide a precise representation of the population mean.



```
              Attributes  Coefficients
78     Tuesday Afternoon      1.485584
15         Gender Friday      1.146075
55          Monday Night      1.106142
63        Saturday Night      1.037593
45        Friday Morning      0.908874
75      Thursday Morning      0.832035
52      Monday Afternoon      0.675537
18         Gender Sunday      0.654837
68        Sunday Evening      0.598253
69        Sunday Morning      0.577038
7               Thursday      0.500183
30     Branch_C Thursday      0.500183
44        Friday Evening      0.448626
83     Wednesday Evening      0.371176
23        Gender Evening      0.365679
16         Gender Monday      0.280906
74      Thursday Evening      0.276546
1                 Gender      0.214829
14       Gender Branch_C      0.214829
5               Saturday      0.191276
28     Branch_C Saturday      0.191276
20        Gender Tuesday      0.173331
24        Gender Morning      0.095263
32    Branch_C Wednesday      0.090993
9              Wednesday      0.090993
84     Wednesday Morning      0.062752
12               Morning       0.03674
35      Branch_C Morning       0.03674
36        Branch_C Night      0.031894
13                 Night      0.031894
70          Sunday Night      0.027338
60    Saturday Afternoon      0.024473
61      Saturday Evening      0.019051
```

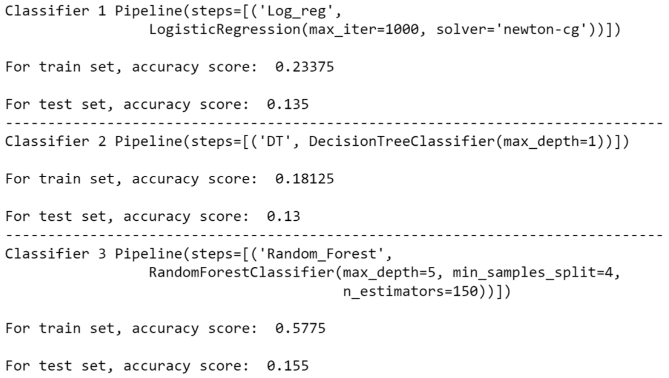
Fig. 7. Positive interactions when customer type = normal.

E. Prediction of day of week

Three classifiers are proposed to fit this model. By comparing the train and test accuracy score (Fig 8), we can conclude that all algorithms are all overfitting and have low accuracy. Although decision tree algorithm overfit the less, the result of confusion matrix implies that decision tree can not correctly classify several targets. For random forest classifier, test accuracy does not lie in 95 percent confidence interval, which is not suitable for this classification. As a result, the best algorithm for this problem is logistic regression with solver='newton-cg' (Fig 8).

V. Conclusion

My experiments show that lasso regression is a better option predicting gross income and unit price compared to linear regression. The introduction of interaction term help increase the accuracy and complexity for classifying the attributes with low correlation. By comparing different algorithms, logistic regression is a better option classifying day of week in supermarket sales dataset.



```
Classifier 1 Pipeline(steps=[('Log_reg',
                 LogisticRegression(max_iter=1000, solver='newton-cg'))])

For train set, accuracy score:  0.23375

For test set, accuracy score:  0.135
-----------------------------------------------------------------------------
Classifier 2 Pipeline(steps=[('DT', DecisionTreeClassifier(max_depth=1))])

For train set, accuracy score:  0.18125

For test set, accuracy score:  0.13
-----------------------------------------------------------------------------
Classifier 3 Pipeline(steps=[('Random_Forest',
                 RandomForestClassifier(max_depth=5, min_samples_split=4,
                                        n_estimators=150))])

For train set, accuracy score:  0.5775

For test set, accuracy score:  0.155
```

Fig. 8. Accuracy for three classifiers.

TABLE I
Experiments carried

Type	Content
Data Preprocessing	Drop data, Ordinal Encoding, One-Hot Encoding, StandardScaler, Train-Test-Split, Day Encoding, Timeslot Encoding
Hyperparameter Tuning	Cross-Validation Grid Search
Correlation Matrix	Corrrelated Value

In this project, I test with various different combination of algorithm tuning and preprocessing (Table 1) of the dataset available. Due to the weak correlation relation and limited data, several classifiers are not able to successfully fit the data. We should introduce more interaction term or increase the number of data to increase their performance. Testing with various different configuration shows that high validation accuracy is not achieved by a single technique or model architecture but by combination of algorithm tuning and preprocessing.

Acknowledgment

The authors would like to thank Dr. Catia Silva for providing this opportunity to experiment Machine Learning models.

References

[1] R. Muthukrishnan and R. Rohini, "LASSO: A feature selection technique in predictive modeling for machine learning," 2016 IEEE International Conference on Advances in Computer Applications (ICACA), 2016, pp. 18-20, doi: 10.1109/ICACA.2016.7887916.

[2] X. Zou, Y. Hu, Z. Tian and K. Shen, "Logistic Regression Model Optimization and Case Analysis," 2019 IEEE 7th International Conference on Computer Science and Network Technology (ICCSNT), 2019, pp. 135-139, doi: 10.1109/ICCSNT47585.2019.8962457.

[3] C. Zhao, L. Huang, L. Hu and Y. Yao, "Transient fingerprint feature extraction for WLAN cards based on polynomial fitting," 2011 6th International Conference on Computer Science and Education (ICCSE), 2011, pp. 1099-1102, doi: 10.1109/ICCSE.2011.6028826.

[4] S. Paul, P. Ranjan, S. Kumar and A. Kumar, "Disease Predictor Using Random Forest Classifier," 2022 International Conference for Advancement in Technology (ICONAT), 2022, pp. 1-4, doi: 10.1109/ICONAT53423.2022.9726023.

[5] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikitlearn: Machine learning in Python. Journal of Machine Learning Research, 12:2825–2830, 2011.