PSTAT 195/ CS 190D

Factors that Influence Customers' Satisfaction Level

Member: Yuehan Guo and Shuyun Tang

Professor: Kate Kharitonova and Alex Franks

Abstract

In our final project, we analyzed a real 100k+ real-time local online orders dataset in Brazil in 2019, and explored potential factors involved in e-shopping process that could influence customers' satisfaction levels. In doing this, we used *Numpy, Pandas*, and *datetime* for processing data; *Altair, Seaborn, Matplot* for visualization; and logistic regression, random forest, and NLP for fitting our classification dataframe (influential factors) and predicting our target dataframe (review score). We found out that delivery time and product details have a huge impact on customers' satisfaction level. Shorter delivery period and more detailed product description could lead to better comments and review scores. By comparing different models, we've also found that if e-shops want to improve their service, they should pay attention to review comments prior to all other factors mentioned above.

1. Introduction

Nowadays, online shopping is becoming more and more popular. People could shop whatever they want at home with a few fingers clicks and need not to consider all those bothersome transportations to get to the shopping mall and the energy and time spent with the in-store assistants. People could even spend more time on comparing different products and deciding what to buy. Driven by this trend, it is very important for eshop owners to explore what is really valued by customers in order to make more profit. Besides those eshop owners, other readers who are online shopper can gain insightful, experienced reports from most of the other online shoppers, so they can avoid some fraud information in the eshops.

The primary goal of our project is to explore different influencing factors (namely product price, delivery length, payment method, quantity of product photos, and the geographic position of customers) on customer's e-purchase behaviors and satisfaction, so that we could predict customers' purchase preference and the trend of ecommerce.

Shuyun is an experienced accessories e-shop owner, and he would like to get some inspiration on how to improve his e-shop to make more benefit by analyzing the customer data; and Yuehan is an experienced online shopper who is very familiar with different factors involved in online shopping that could potentially affect customers' behavior and preference. Hence, this data exploration is especially meaningful and interesting for us.

This data we use is a real 100k+ real-time local online orders dataset in Brazil in 2019. It provides lots of clean, insightful data about those influencing factors listed above and thus it is appropriate for us to use.

2. Questions of Interest

We want to explore the review score and other influential factors' relationships and patterns. These influential factors include delivery time, delivery time difference (actual-estimated), products' photos, description length in words, review comments. By estimating the review scores based on these factors, we want to get an idea of what factor could influence customers' satisfaction level in which way so that we could find potential ways to improve our e-shop and provide customers better experience.

3. Data and Methods

The dataset we use is from *https://www.kaggle.com/jainaashish/orders-merged*, and it does not have a specific license. The publisher does not provide detailed information either. However, with our analysis, we found out that most of the buyers were located in Brazil and that this was a 100k+ real-time local online orders dataset in Brazil in 2019. This dataset provides a lot of clean, insightful data about factors that influence customers' satisfaction level; thus, it is a very informative dataset. The packages we used for visualization, including pair plot, correlation matrix, line chart, violin plot, are *Altair, Seaborn, Matplo*t. The packages we used for processing the data and features engineering are *Numpy, Pandas,* and *datetime* (convert timestamp). We will use *Sklearn* to conduct our machine learning parts. The logistic regression, random forest, NLP will fit our classification dataframe (influential factors) to predict our target dataframe (review score). All the packages and libraries listed above can be referred to in the citation part.

There are a lot of variables in the original dataset, so we've created a new dataframe with only the columns that we are interested in, namely **'review_score'**, which reflects customers' satisfaction level, **'order_purchase_timestamp'**, which represents the time when the customers placed their orders, **'order_estimated_delivery_date'**, which represents the estimated delivery date of the product, **'product_description_lenghth'**, which is the description length of the product on the e-shops' website, **'product_photos_qty'**, which shows the amount of photos the sellers provide the customer of a specific product, **'price'** , which is the purchase price, and **'freight value'**, which represents the shipping fee. In order to better conduct our analysis, we've also added some new features based on the current columns. '**del_time'** and '**est_del_time'** are the datetime conversion of the 'order_delivered_customer_date' and 'order_estimated_delivery_date' columns. '**timediff'** is the delivery time minus the estimated delivery time and gets converted into the days difference to integers. '**deliver_time'** is the delivery time minus the purchase time and gets converted into integers. '**purchase_weekday'** is the weekday of the purchase time, in which 0 is Monday and 6 is Sunday. '**Late**' is a categorical variable that indicates 0 if the corresponding 'timediff' is less than 0, indicating that the actual delivery time is earlier than the estimated delivery time, and vice-versa. Also, in order to make our model more accurate, we will sort out the review scores that were lower or equal than 2 into the negative reviews and denote it by "0"; and the review scores higher than 2 into the positive reviews, denoted by "1".

All the data were supposed to be generated by the e-shop platform, and there should be no data entered by a human after our observation to the data source. There might be some potential sources of error on the other columns such as the length/weight of the products, but our dataframe will not address those potential columns here. We could also see some noticeable large or small values in delivery time column. The reasons are probably custom regulation or the pre-order sales.

Since all the records were given precise, the precision of our analysis should be guaranteed. Also, we likely have no measurement distortion since the original dataset did not distort the ecommerce system and structure under study. The customers' purchase behaviors and reviews won't be affected by the observation. The most questionable part might be the relevance, since our records come from different individuals with different backgrounds. Also, the whole dataset is collected from Brazil, which might be different to the dominant e-commerce system in Asia. Sample size and selection will result in lots of time cost since we need to consider and extract the useful variables from more than 30 columns. Originally, there were 2728 missing values, i.e. some of the columns includes Nah. After our process, there is no missing value anymore.

The primary ethical considerations are the privacy of each customer, especially their payment information. Thus, when generating the processed dataset, we will exclude those sensitive columns to protect their information. Another consideration is the customers' locations. Since the original dataset contains their exact address and zip code, it is not safe to use them in our final project without the instructions of the original publisher, so we exclude those columns in our processed dataset. There is no specific group in our processed dataset being over-represented or underrepresented, but the locations are in Brazil, which might possibly make our exploration lack of diversity due to cultural differences.

4. Exploratory Data Analysis

Pair plot & Correlation Matrix
Our ultimate goal is to explore how to make potential customers more satisfied. In order to do this, we need to analyze the potential factors which could possibly bring us a better understanding in marketing. By creating a pair plot, we could get a brief glance of some of the potentially interesting relationships. With the help of Seaborn and Matplot, pair plot illustrates a pairwise relationships in the dataset, which can provide us an overview of the correlations of all the variables. From Figure 1(Bigger version in Appendix Figure 1), we could see some noticeable patterns between the review score and delivery time, product description length, product photo quantity and price, delivery time and freight value.

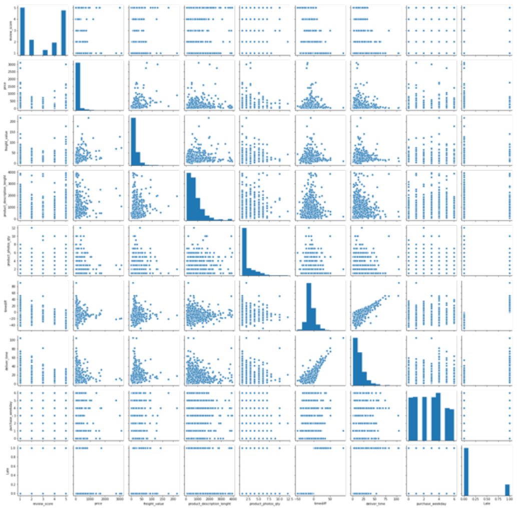


Figure 1: Pair Plot

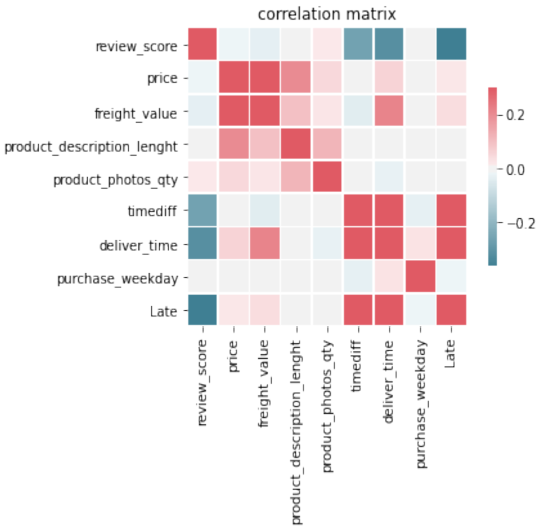


Figure 2: Correlation Matrix

In order to have a closer look into the relationships among these variables, we then conducted a correlation matrix to make further explorations. Correlation matrix is a table showing the correlation coefficients among all the variables, so it could give us a more vivid visualization of those factors and review scores.

From our observations, we find that there exist negative relationships between the timediff, delivery time, and Late versus the review score. This could be due to the obvious reason that most of the buyers do not want to wait for too long for the product. We also see that there are positive relationships among freight value, price, product description length and product photo quantities. This might be due to the fact that more valuable goods usually have more detailed information provided by the e-shops.

Review Score vs Delivery Time (Violin Plot)
We would like to separately visualize our interested relationships. Firstly, we created two violin plots visualizing the relationship of review score vs delivery time (Figure 3) and review score vs delivery time difference (Figure 4).

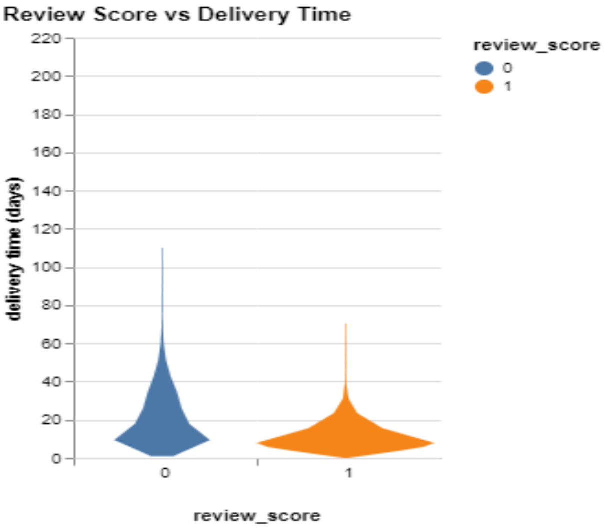


Figure 3: Review Score vs Delivery Time

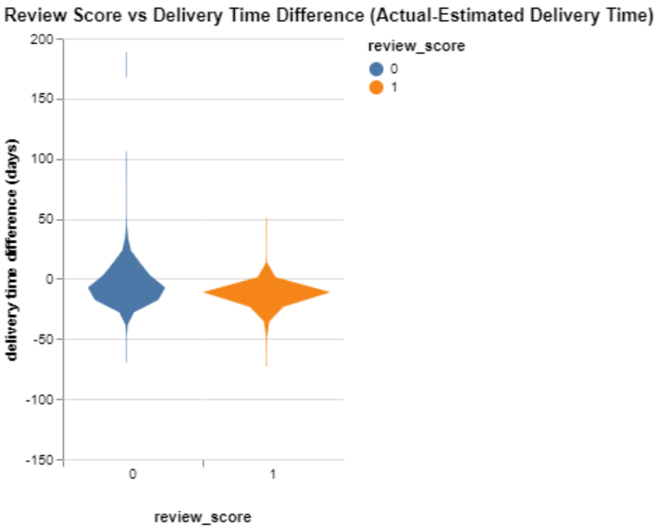


Figure 4: Review Score vs Delivery Time Diff.

We find out that the bad reviews (score 0) has a greater density when the delivery time is long. Also, we find out that good reviews (score 1) has a larger proportion in 0 to -50 delivery time difference, which means that if customers receive their goods earlier than the estimated delivery time, they will tend to give good reviews. Hence the delivery time clearly has a negative relationship to the review score, so does the delivery time difference.

Review Scores vs Photo Quantities & Review Score vs Description Length (Line Plot)
Then we explored the relationships between review scores and photo quantities, and review score vs description length using line plot. Line chart displays information as a series of data points connected by straight line segments. It is a good, straightforward fit because it can illustrate the photo and description counts grouped by review score. From Figure 5 and the details below, we can discover that the review score 0 usually has less product photos and shorter description,

which could probably because the customers are misled by the fraud or incomplete product photos and information.

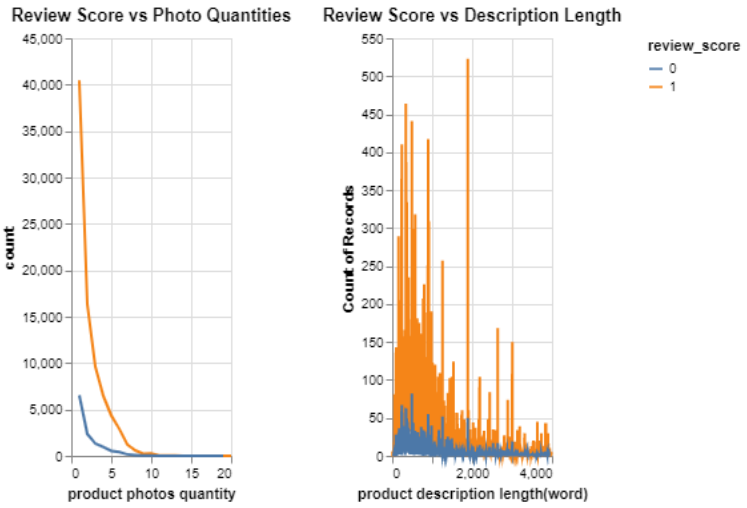


Figure 5: Photo Quantities and Description Length

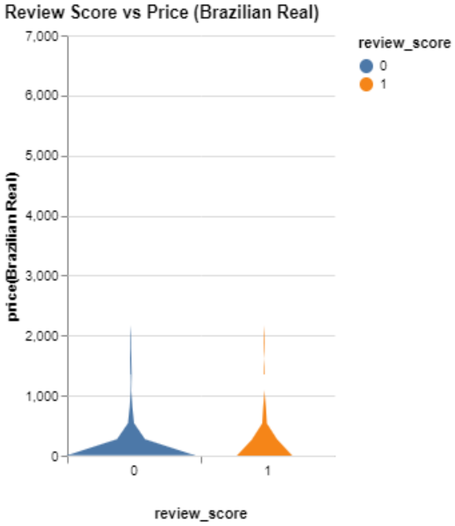


Figure 6: Review Score vs Price

Review Score vs Payment Value (Violin Plot)
Next, we explored the relationship between Payment Values (Product Price) and Review Scores by creating a violin plot again. From Figure 6, we could see that when the price is low, there are more reviews, resulting in more good reviews and bad reviews. While with higher prices, the amount of both kinds of reviews are much fewer. However, when the price is very low, there is much more bad reviews than good reviews. This is possibly because cheaper goods might have poorer quality which could lead to lower review scores.

5. Applying Machine Learning (Sklearn)

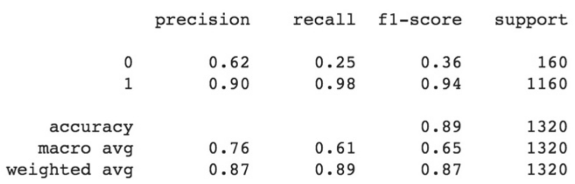
Sampling, partitioning and balancing the data
Due to the extremely large sample size which is more than 90000 rows, it will take a lot of time and computer memory to analyze all of them. Also, the positive reviews are significantly more than the negative reviews (people are nice). We will sample and partition the dataset into the classification columns and target columns before we put it into our model.

Actually, we started with sampling the unbalanced dataset, however, we figured out some problems with it, which we will present later in the report. After figuring out the problems, we immediately balanced our samples. In our final process, we took 4000 random samples: 2000 from those review score higher than 2 (good reviews) and 2000 from those lower or equal than 2 (bad reviews), so that it could make the sample set more balanced and easier to analyze. Then we standardized our classification data frame so that all the columns could have the same weight during the machine learning process.

Logistic Regression about the Unbalanced Sample and Balanced Sample
The reason why we use logistic regression is that the review score is a categorical variable rather than continues numerical variables. Thus, we need to apply the logistic regression to examine how well those classification columns (df_classification) could predict the target column (review score). The packages are used from Sklearn.

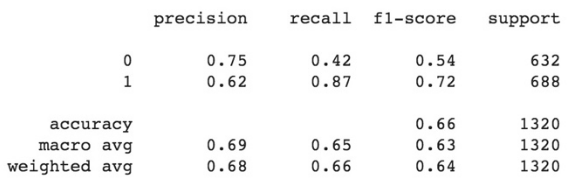
After our first attempt to conduct logistic regression using the unbalanced sample, we were actually pretty happy with the result at first, because the accuracy for the unbalanced data was 0.89 (weighted avg is 0.87) (See Table 1). It is actually significantly higher than the balanced sample. This might because of the facts that the original ratio (a higher proportion of the give positive review scores) is a better fit to the model and that the balanced data amplifies the uncertainty of those give bad review scores.

Table 1: Logistic Regression Results (Unbalanced)



```
              precision    recall  f1-score   support

           0       0.62      0.25      0.36       160
           1       0.90      0.98      0.94      1160

    accuracy                           0.89      1320
   macro avg       0.76      0.61      0.65      1320
weighted avg       0.87      0.89      0.87      1320
```

Table 2: Logistic Regression Results (Balanced)

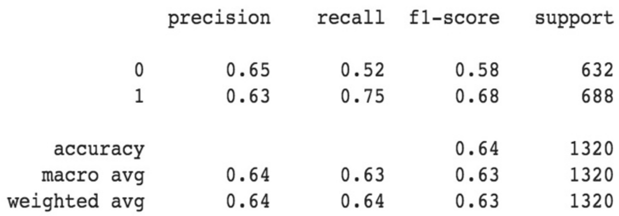


```
              precision    recall  f1-score   support

           0       0.75      0.42      0.54       632
           1       0.62      0.87      0.72       688

    accuracy                           0.66      1320
   macro avg       0.69      0.65      0.63      1320
weighted avg       0.68      0.66      0.64      1320
```

However, from the results above, we discovered that the “0” review score had a very low accuracy (which was 0.62). Since there were too many positive reviews, the accuracy of the bad reviews is undermined. Thus, we used the balanced sample to re-check the model's real accuracy. From the results in Table 2, we could see that the accuracy of our logistic model is actually very low. We have observed two possible reasons: 1. there are lots of factors such as the weekdays that are completely uncorrelated to the review score; 2. the target column itself is hard to predict, because review scores are very subjective and there are a lot of people who never leave their review scores so the systems will automatically assign the score 5.

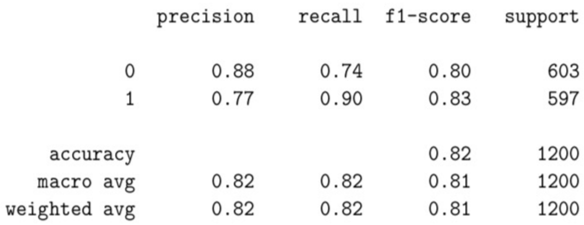
Random Forest about the balanced sample
Since the results in logistic regression were not ideal, we then conducted the random forest model. In short, it is a model that consists of a large number of individual decision trees that operate as an ensemble. Each individual tree in the random forest spits out a class prediction and the class with the most votes becomes our model’s prediction. The reason why we use it is because it could handle non-linear parameters efficiently: non-linear parameters don't affect the performance of a Random Forest (unlike curve-based algorithms). Hence, if there is a high non-linearity between the independent variables, Random Forest may outperform as compared to other curve-based algorithms. Random Forest is usually robust to outliers and can handle them automatically. It can also reduce overfitting that exists in the decision tree model. However, as seen in Table 3, the result is still bad when using the balanced sample. For customers who left bad reviews (0 score), there existed too many subjective reasons such as their preferences and expectations. Even their emotions would affect their feedbacks. Thus, it is very hard for the model to predict such subjective category.

Table 3: Random Forrest Results



```
              precision    recall  f1-score   support

           0       0.65      0.52      0.58       632
           1       0.63      0.75      0.68       688

    accuracy                           0.64      1320
   macro avg       0.64      0.63      0.63      1320
weighted avg       0.64      0.64      0.63      1320
```

Table 4: NLP Results



```
              precision    recall  f1-score   support

           0       0.88      0.74      0.80       603
           1       0.77      0.90      0.83       597

    accuracy                           0.82      1200
   macro avg       0.82      0.82      0.81      1200
weighted avg       0.82      0.82      0.81      1200
```

NLP For Review Comments on the balanced sample

Reflecting on the two previous results, we think that there are many factors that could influence review scores, like systematically rate “5” or randomly giving good review scores, so we decide to exclude some of the influencing factors and take only the valuable reviews which could genuinely reflect the satisfaction level of customers. In doing this, we add another variable to our dataframe: “**review_comment_message**”, which contains customers’ comments. We will use the Natural Language Processing to analyze the correlation of the strings in review comment and the review scores. The reason we use NLP is that it does a good job in translating the human readable language into computer readable language. Here, it transforms the customers’ comment messages from strings to TF-IDF scores. We then apply the Naive Bayes to classify the good/ bad reviews in the training model. Naive Bayes applies similar method to predict the probability of different class based on various attributes. Its algorithm is mostly used in text classification. Then we get our trained model that predicts the review scores below (Table 4).

We find the result to be quite surprising since we get relatively better accuracy in both bad review scores and good review scores. The reason is probably that the review comments have a closer relationship to the review scores. It is obvious that people would leave their unsatisfied comment if they receive a bad shopping experience. On the other hand, if people forget to leave the positive comment when they receive a good shopping experience, the system will automatically assign the default good review comment to the eshop. That’s why on both sides the NLP works well.

6. Results, Discussion and Future Work

Key Results and Conclusion

We partially achieved our goal to analyze the customers' shopping behaviors (feedbacks) based on those influential factors. After cleaning, processing, and re-designing our features and target data frame, we addressed our questions of interest. The delivery length and late delivery have a clear negative correlation to the review scores. It informs the e-sellers that they should improve their shipping service and make sure to deliver the products as soon as possible. Another feature that exerts a huge impact on customers’ satisfaction level is photo quantity and description. After analyzing the dataset, we found that products with more photos and descriptions tend to have better reviews. This indicates that sellers should provide more product photos and product descriptions in order to fully display the product and avoid misleading potential customers. The payment amount also affects the overall reviews because cheaper goods usually have worse quality controls, leading to lower review scores. However, we should see this problem critically. For example, for shops whose target customers have high consumption, they could consider only producing good quality stuffs and reduce the production and selling of low-price-poor-quality product in order to maintain the shop’s reputation. Yet for shops whose target customer have low consumption level, they usually have larger audience, so they could still make profits even if there might be a risk of receiving more bad reviews.

We also find that if e-shops want to improve they service, they should pay attention to review comments prior to all other factors discussed above. There is a tight connection between review comment and review score. According to our model fit results, review comment could be an important predictor to review score, because it has a high accuracy, especially to bad reviews. E-shops should carefully read through the comments and improve what is mentioned in the bad comments.

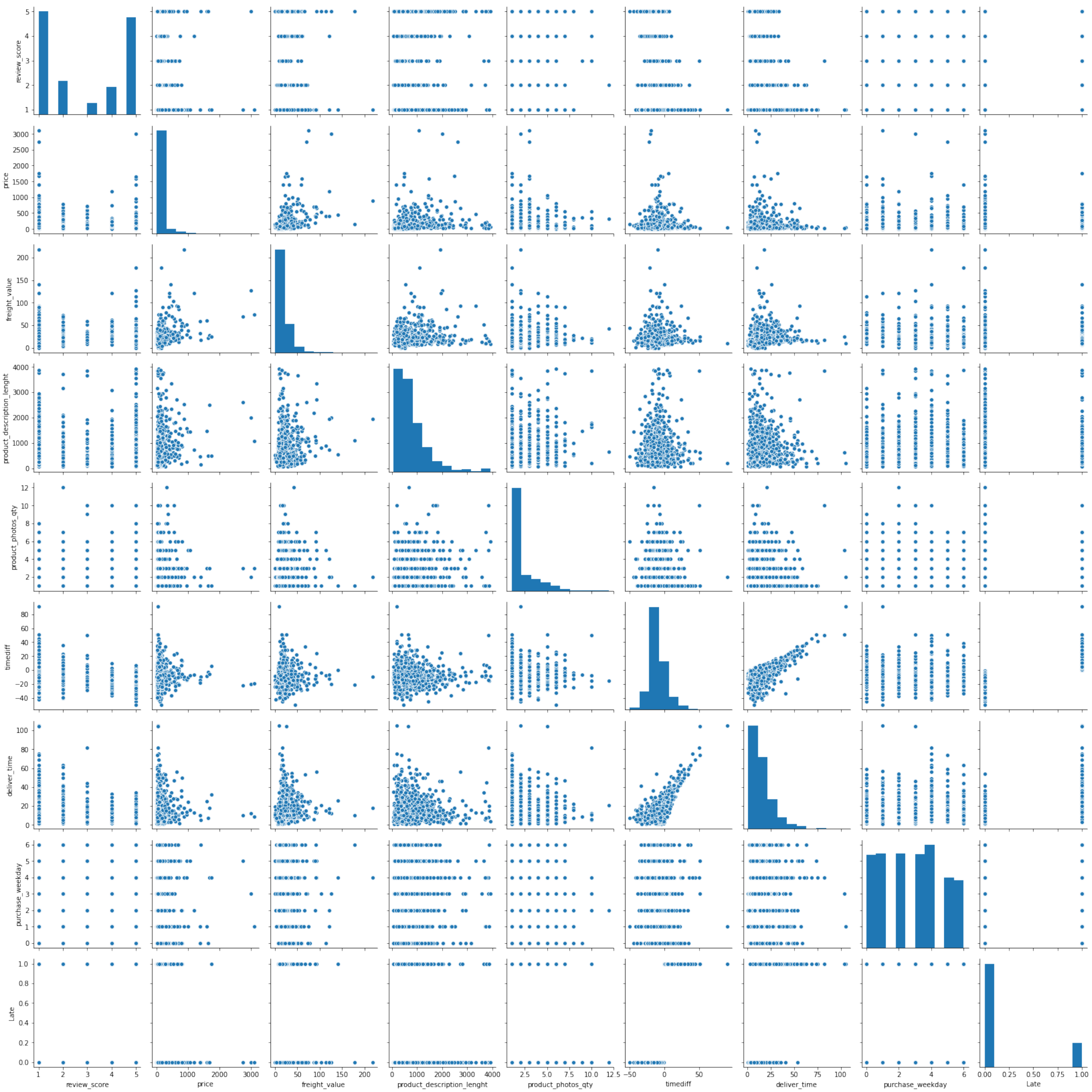
Discussion

At the beginning, we spent a lot of time designing and cleaning our ideal dataset due to the original dataset's complexity. Throughout our approaching, we made some struggles choosing the right models to fit our datasets. We started with logistic regression. Due to the fact that linear regression only handles continuous numerical variables and that multi-categorical logistic regression will amplify the weight of the review score 5 which is a large proportion of the whole dataset, sorting out the review scores into two categories (positive and negative) provides a better fit to the logistic regression. Since our results were not ideal, we tried the Random Forest model, which ensures the outliers and large variance samples to have smaller effect on the classification process by making collective decisions by each decision tree. After getting another frustrating result, we realized that the predictors that we used might be not so proper. Review score is very subjective, yet the factors that we previously used were all pretty objective, hence there might be a significant inaccuracy in the prediction. Therefore, we introduced another variable: review comments, which is also a subjective factor. After applying Natural Language Processing, we finally got a satisfying model to predict the review score.

Future Work

In the future, we will collect more datasets from different regions since Asian E-commerce usually has different structures from Brazil's. We will also try to analyze their review comments with the help of Natural Language Processing. Also, when preparing the random sampling, we will adjust different categories' ratio since balanced and unbalanced dataset often has different outcomes in the models.

Appendix



Appendix Figure 1: Pair Plot

References

Aashish Jain. (2019). Ecommerce_orders [100k realtime orders from an E-commerce]. Retrieved from http://www.kaggle.com/jainaashish/orders-merged

Hunter, J. D. (2007). Matplotlib: A 2D graphics environment. Computing in Science &amp; Engineering, 9(3), 90–95.

Oliphant, T. E. (2006). A guide to NumPy (Vol. 1). Trelgol Publishing USA.

McKinney, W., & others. (2010). Data structures for statistical computing in python. In *Proceedings of the 9th Python in Science Conference* (Vol. 445, pp. 51–56).

Pedregosa, F., Varoquaux, Ga"el, Gramfort, A., Michel, V., Thirion, B., Grisel, O., … others. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, *12*(Oct), 2825–2830.

Michael Waskom, Olga Botvinnik, Joel Ostblom, Maoz Gelbart, Saulius Lukauskas, Paul Hobson, … Brian. (2020, April 26). mwaskom/seaborn: v0.10.1 (April 2020) (Version v0.10.1). Zenodo. Retrieved from http://doi.org/10.5281/zenodo .3767070

Bird, Steven and Klein, Ewan and Loper, Edward. (2009). Natural Language Processing with Python (Vol. 1). O'Reilly Media, Inc.