Starbucks Product Rewards: An exploration through Population Segmentation, Principle Component Analysis, and Predictive Customer Spending

Udacity Machine Learning Engineer Capstone

Scott Zetterstrom

14 January 2020

Table of Contents

1. Project Overview: This project seeks to use machine learning algorithms to determine the effectiveness of a Starbucks product reward offer. The project will determine the main variance in the data set and then classify who activated which of four different types of offers.

2. Problem Statement: Starbucks is interested in determining who should receive offers on a particular product they offer. They want to make sure the people who receive the offer are ones who will use it and not have it go to people who would get the product regardless of the offer. The solution will be primarily looking toward binary classification.

3. Metrics:
    a. The three evaluation metrics for binary classification are:
        i. AUC – Area Under Curve since the classes are not balanced.
        ii. Precision
        iii. Recall
    b. For the linear spending prediction:
        i. Accuracy

4. Data Exploration:
    a. Summary of Input Data:
        i.

Data	Type	Size
Gender	Categorical	14825
Age	Numeric	17000
Id - profile	String/hash	17000
Became_member_on	Date	17000
Income	Numeric	14825
Reward	Numeric	10 – types
Channels	List	4 - types
Difficulty	Numeric	10 - types
Duration	Numeric	10 - types
offer_type	String	10 - types
Id - portfolio	String/hash	10
Person	String/hash	306534
Event	String	138953
Value	Dictionary	306534
Time	numeric	306534

Table 1. Dataset and Description

b. Class Balance:

i. There are a few classes to check to see if they are balanced and are summarized in Table 2 below.

Class 1	Class 2	Balanced/Not Balance
Spent Money - 138953	Did Not Spend Money - 167581	Classes are not balanced, but not overly (45%/55%)
Triggered Offer - 33579	Did Not Trigger Offer - 272955	Not Balanced (11%/89%)

Table 2. Class Balance in Dataset

5. Exploratory Visualization:
   a. The dataset was explored through a variety of visualizations to better grasp the data. These figures are presented below:

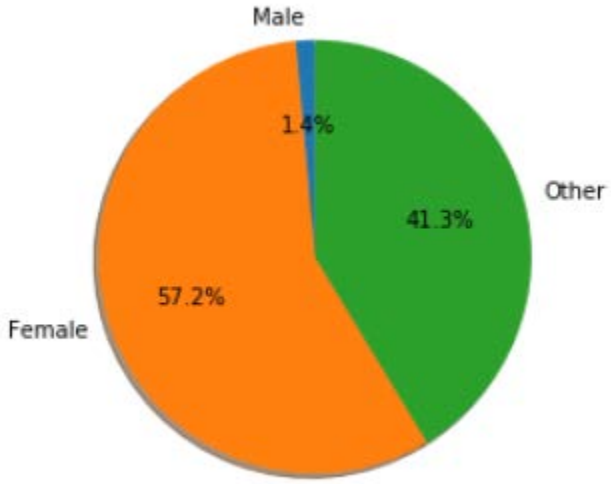

Figure 1. Male to Female to Other Chart

   b. From Figure 1 you can tell that majority of the dataset is female. This may be true or it is possible that the majority of males did not enter their gender. The general population has a 51%/48%/1% ratio (need reference). This will skew the data toward females and is a clear limitation of this dataset.

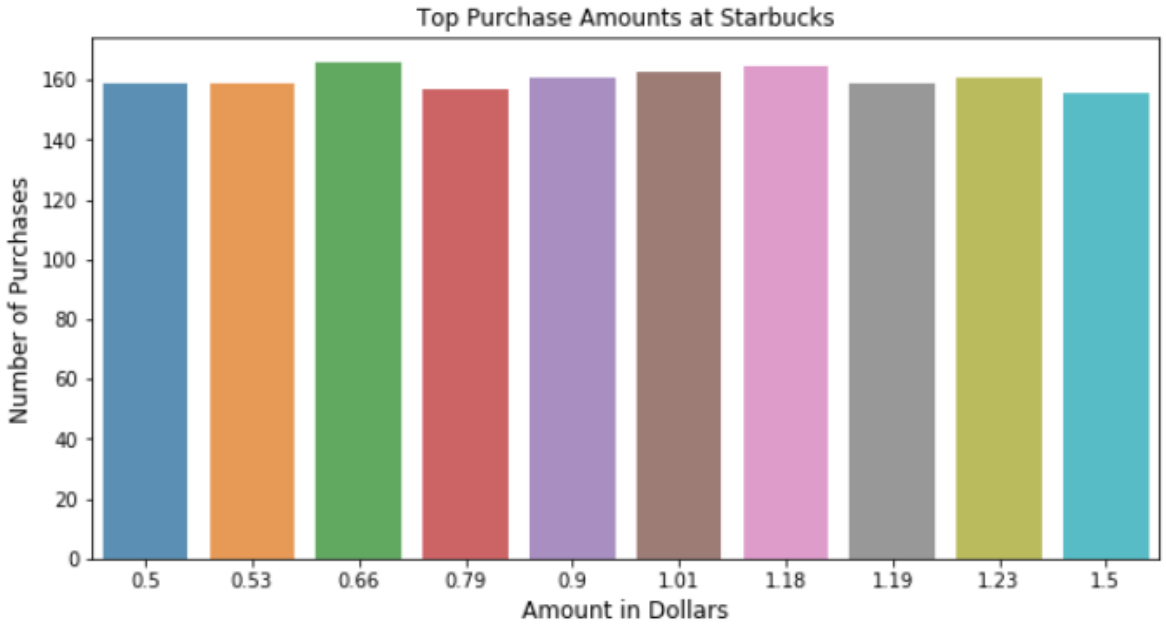

Figure 2. Top Ten Purchase Amounts at Starbucks

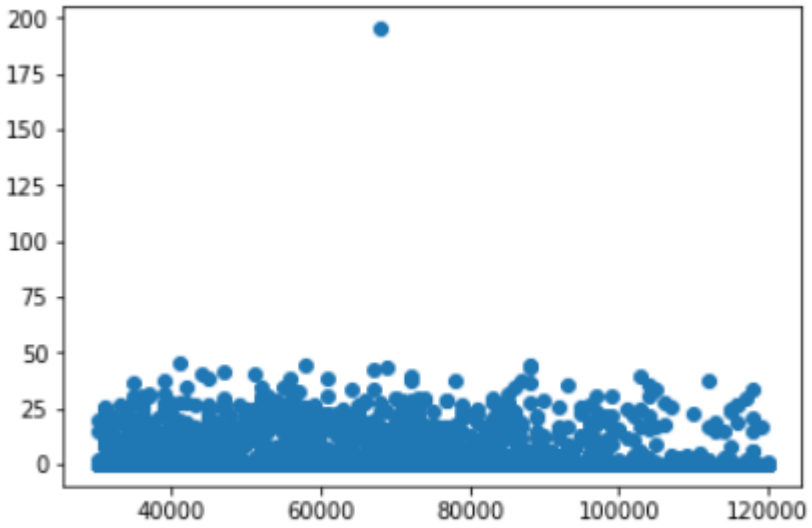

Figure 3. Income vs Amount Spent

Figure 4. Amount Rewarded

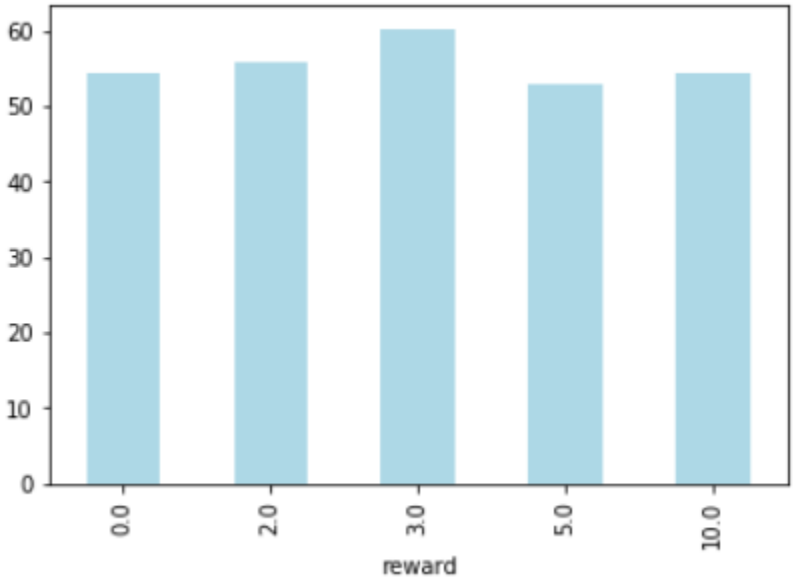

Figure 5. Reward vs Age

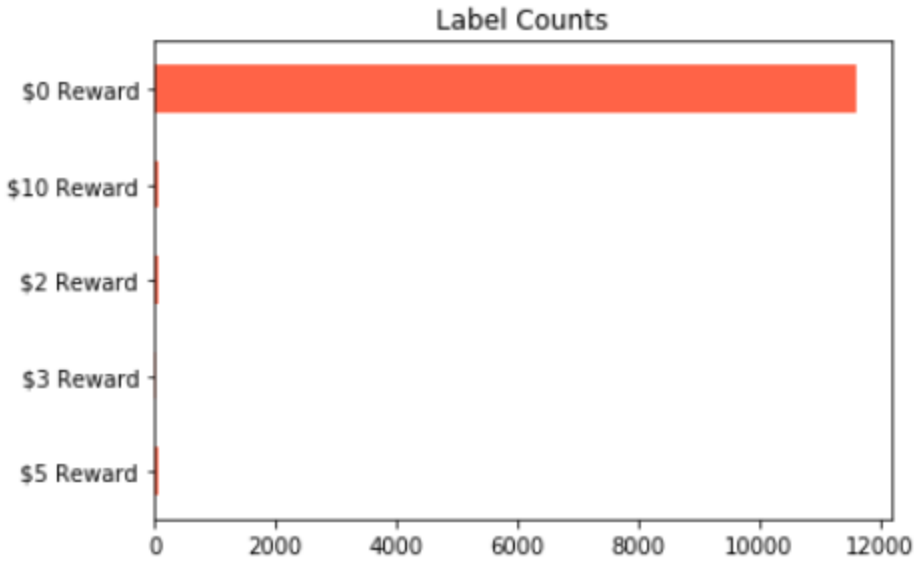

Figure 6. Count of Rewards – Highlights the class Imbalance

6. Algorithms and Techniques
    a. Two Algorithms were utilized, PCA and the Linear Learner.
        i. PCA seeks to look at the variations in the data and what data provide the most variance in the model. (Powerl, n.d.)

        ii. Linear Learner is a supervised learning algorithm used for regression or classification. In this case we are using it for classification. (Linear Learner Algorithmn, n.d.)
7. Benchmark
    a. The benchmark model is a simple logistics regression model. This model looked at a small subset of the data: age, income, gender and seek to determine who spends money at Starbucks. The head of that data frame is below.

    b. The head of the benchmark model data is below. Data was preprocessed to first remove unneeded data and then cleaned. Dummy variables were created for gender, NaN for reward were replaced with 0.0, and then all NaNs were dropped from the frame before processing.

	age	income	F	M	O	reward
1	55.0	112000.0	1	0	0	0.0
3	75.0	100000.0	1	0	0	0.0
5	68.0	70000.0	0	1	0	0.0
8	65.0	53000.0	0	1	0	0.0
12	58.0	51000.0	0	1	0	0.0

Figure 7. Benchmark Model Data frame Head

c. Algorithm: Linear Learner using binary classification was used to determine from this small subset who received and did not receive a reward. The model used a ml.c4.xlarge instance type, 15 epochs, model selection criteria of precision at target recall, a target recall of 0.9, and since the classes are highly unbalanced, a positive_example_weight_mult set to 'balanced'.

d. Metrics: After using the creating the model and then loading the test data. The results were as follows (This output modeled after the Udacity "XXX" case study output):

```
Metrics for simple, LinearLearner.

prediction (col)   0.0  1.0
actual (row)
0.0               2901    2
1.0                  0   62

Recall:     1.000
Precision:  0.969
Accuracy:   0.999
```

Figure 8. Benchmark Model Metrics

e. Benchmark Conclusions: This small data frame did an excellent of classifying who would and would not receive a reward. It only misidentified 2 people who not actually receive a reward. There are many limitations on this data such as the limited data frame, and a relatively small sample size of only 2,965. Therefore there would be limited utility in actually deploying this model for use.

8. Data Preprocessing
    a. In order to utilize machine learning algorithms the data give required extensive preprocessing. First the JSON files were read into notebook. Then columns with the same name were renamed in order to avoid confusion. Then the files were concated into a single data frame called “complete_df”. Following this, dummy variables were created for gender, event, and offer type. Next the json_normalize function was used on the value column to separate that dictionary into amount, reward, and offer_id. It should be noted that this function worked most of the time, however at certain times the “from_dictionary” had to be used. Other than rerunning the code several times, no source could be identified as the cause. Finally unneeded columns were dropped (person, id, gender, become_member_on, offer id, offer_id, channels, value, event, and customer_id).

    b. Next was cleaning the data before processing. In this step, the fillna() function was used on reward, amount, income, difficulty, duration, and reward type. In this step all NaNs were set to 0.0

    c. Then since we know that an age of 118 is a NaN value, if both the age equaled 118 and the income was 0, those rows were dropped.

    d. As a check of the data a few functions were implemented and visualized. First the value counts for the offer received were looked at. This indicated 11043 received the offer, and 3782 did not. Then the median income per reward was looked at using the groupby() function. This is below in figure 7. This looked reasonable and in line with the rest of the data.

	income
reward	
0.0	65409.426032
2.0	69487.804878
3.0	63716.981132
5.0	63010.000000
10.0	64493.506494

Figure 9. Reward vs Mean Income

9. Implementation
    a. After the data frame was ready it was processed for use in a Principal Component Analysis (PCA) algorithm. First, the components needed to be scaled/normalized to be between 0 and 1. From SKlearn MinMaxScaler was implemented.
    b. After this the data was transformed in a record set for use in the PCA. Once the model was complete a visualization was performed (code was used from Udacity Population Segmentation Case Study).

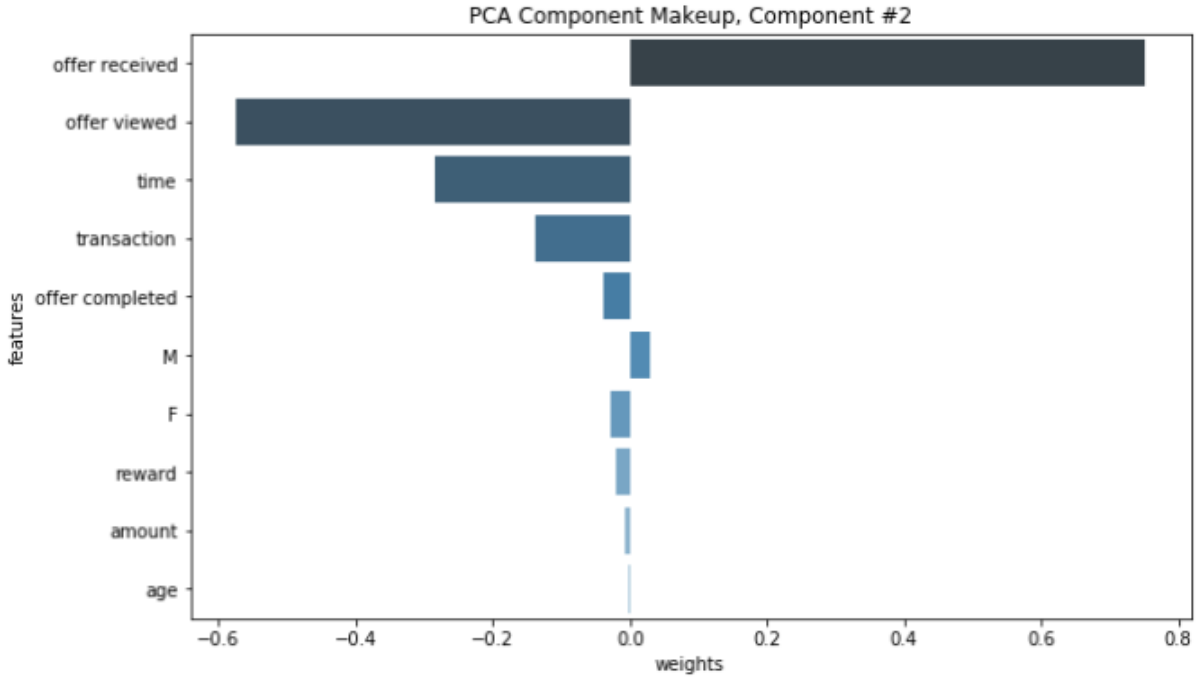

Figure `0. PCA Component Makeup

    c. After these top 7 components were feed into a multi-classification algorithm using Linear Learner. First, some small preprocessing was done so the classes, the rewards, were sequential. They were then mapped to a dictionary for post processing. The results are posted below. Since the classes are extremely unbalanced, the algorithm was run twice. The first results, run in the unbalanced mode are below. Even though the accuracy was 98.5%, this is mostly a function of the underlying classes and not having a highly accurate model.

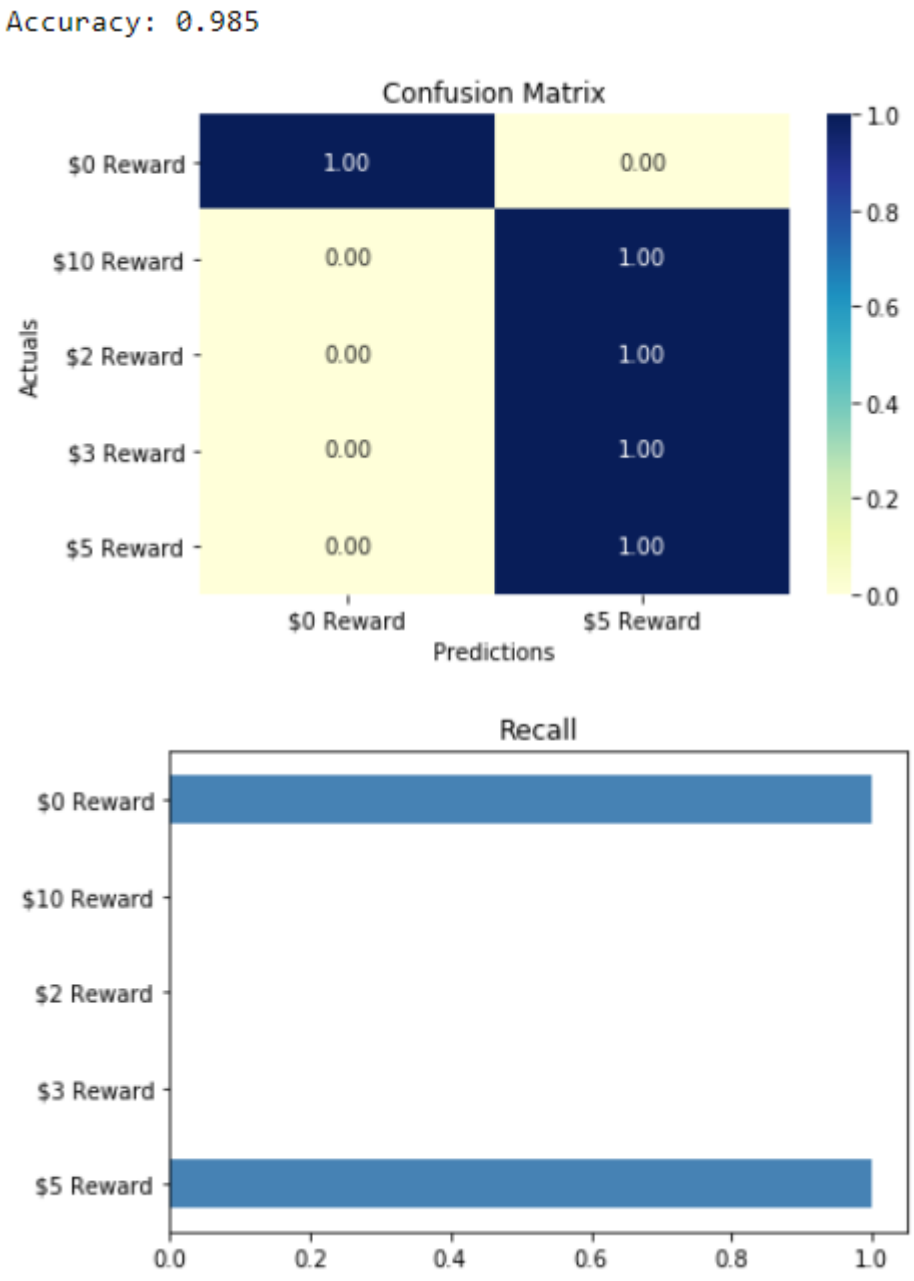

Figure 11. Accuracy and Confusion Matrix from Linear Learner Unbalanced

10. Refinement
    a. To refine the model, the main challenge is the class imbalance. Linear Learner does offer a hyperparameter that helps balance the classes. The results are displayed below.

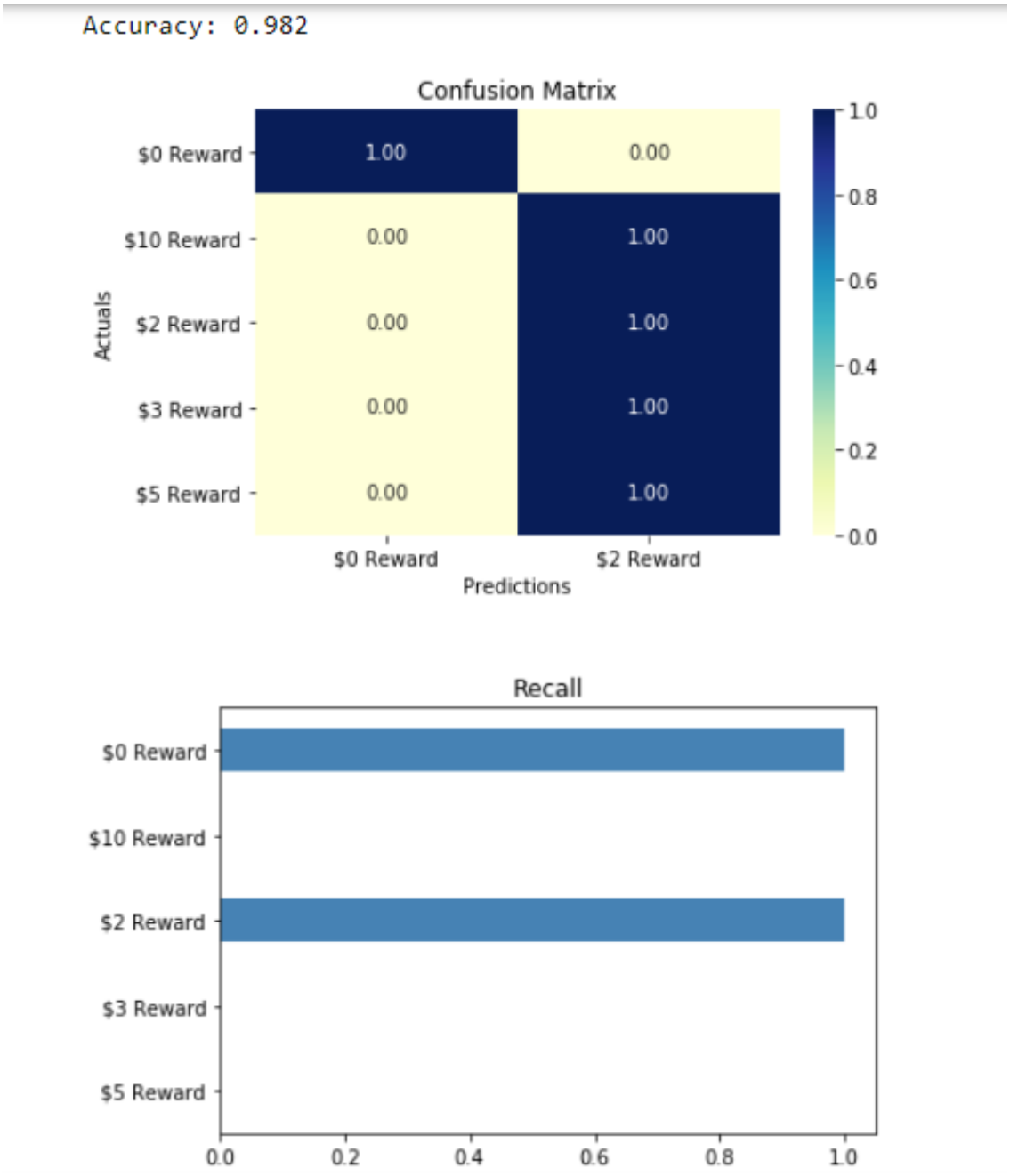

Figure 12. Accuracy and Confusion Matrix from Linear Learner Balanced

11. Model Evaluation and Validation
    a. The accuracy metric for the models were previously discussed and will be summarized here.
        i. Benchmark – 99%
        ii. Multiclass Unbalance – 98.5%
        iii. Multiclass balance – 98.2%
    b. For recall
        i. Benchmark – 100%
        ii. Multiclass Unbalance
            1. 100% for $0 and $5, 0% for all other classes
        iii. Multiclass Balance
            1. 100% for $0 and $2, 0% for all other classes

    c. Looking at these metrics, we know that the model is improving and is more robust. However, all these models need to further refinement and more clean data. The class imbalance is too large to produce a useful model at this stage.

12. Justification
    a. The most useful model used in this analysis was the PCA. From that we can see that the most important factors in those we received a reward or did not were first, was the offer received. In the cleaned dataset there were over 3000 people who did not even receive the offer. This could be do many factors, such as wrong email addresses, phone numbers, or other factors. While it represents a small set of the data, it was the most important factor.
    b. The next factor was whether the offer was viewed. This makes sense, some way to further increase viewership are push notifications on devices. If someone has to go into the app to see the offer, they may never see it. Another data point that may be useful is looking at app usage, how many times is the app opened.
    c. The conclusion here is that a customer receiving the offer and then viewing it should be the primary concern of the marketing team when utilizing this offer.
    d. For the classification problem, the model proved to be of little use for several reasons. The first is the class imbalance. Second is that data set was highly skewed toward females, and the range of income for those who utilized the offers was centralized for the dataset.
        i. From this you would conclude that all offers would go to females in their late 50s who makes around 60-65,000 a year. You could have learned that from the data visualizations alone.
        ii. Another conclusion is that use of offers tend to tail off at higher income levels. That group starting at around 100,000 of annual income did not spend as much as those who make a lower amount. Therefore offering rewards to those whose income is over 100,000 is not beneficial to the business.

e. Further data collection is recommended focusing on the areas outlined below.
    i. App usage
    ii. Accurate Gender representation (data is skewed from actual population statistics)
    iii. Accurate age
    iv. Accurate income

Works Cited

Linear Learner Algorithmn. (n.d.). Retrieved from AWS Blog: https://docs.aws.amazon.com/sagemaker/latest/dg/linear-learner.html

Powerl, V. (n.d.). *Principle Component Analysis Explained Visually*. Retrieved from Setosa : http://setosa.io/ev/principal-component-analysis/