Assessment 02 - Exploration into Company Refund Policy

Data Analysis by Alexander Raouf

Problem Statement

In recent discussions with customer associate reps there has been a lot of discussion about the amount of time spent processing refund transactions. Newer customer associate reps claim it feels like their entire shift is spent taking and processing requests for refunds. Additionally reps with more experience have claimed that they are receiving a larger number of requests for refunds than ever before. To best serve our customers, and best serve the business, we need to determine whether the company refund policy is problematic. We will review whether the data agrees with what our customer reps are claiming. Are refunds increasing? Is it causing a problem for our business, and if so, should the policy be changed?

Proposed Solution

After performing analysis on the data, I have concluded that the amount of refunds coming in for each sale we make is actually decreasing. **With this discovery I suggest the company leaves the refund policy as is, as I do not believe it to be a problem.** As sales increase, so will the need to hire additional associate service reps. The additional employees can help process refund requests much quicker ensuring that the reps have a more balanced work day.

Detailed Report

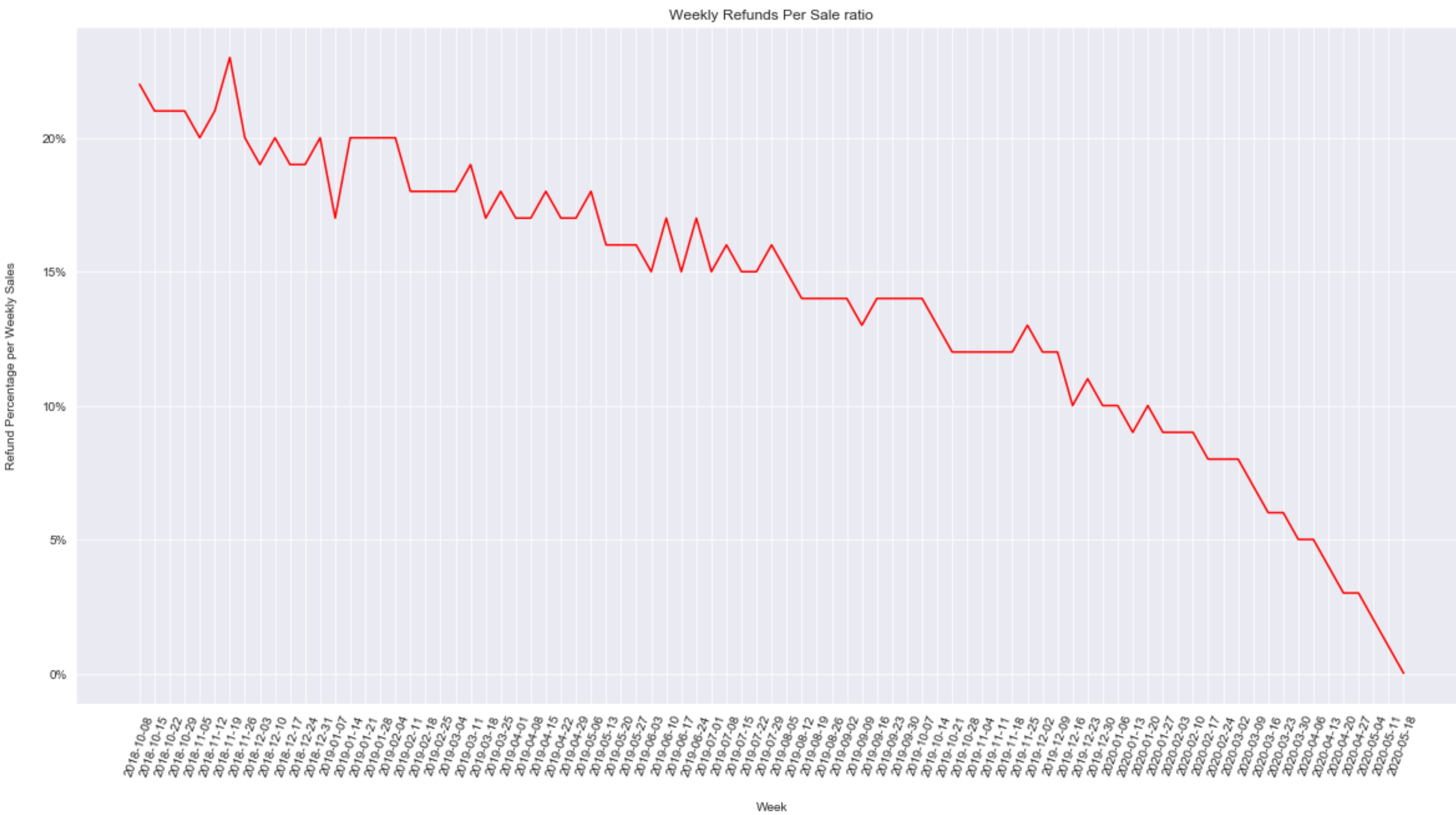
As total sales increase we expect the amount of total refunds to also increase if all else remains constant. Therefore total amount in refunds is not an accurate metric to measure whether refunds are actually increasing. Instead I suggest a normalized metric that can accurately measure whether refunds are increasing in relation to sales. This can be achieved by dividing the total refunds by the total sales. This 'refund ratio' essentially tells us for each sale, what percentage of that sale is on average going to be refunded.

As our company operates in New York under the Eastern Standard Timezone, for this analysis we have converted all of our sales and refund times from UTC to EST.
In additional we have chosen to perform our analysis on sales made between October 8th, 2018 and May 18th, 2020.

Weekly Report

For our weekly report I have grouped our data based on 7 day intervals. That means for every 7 days in our dataset we group all of the sales that have occurred within that 7 day window based on the sale date.

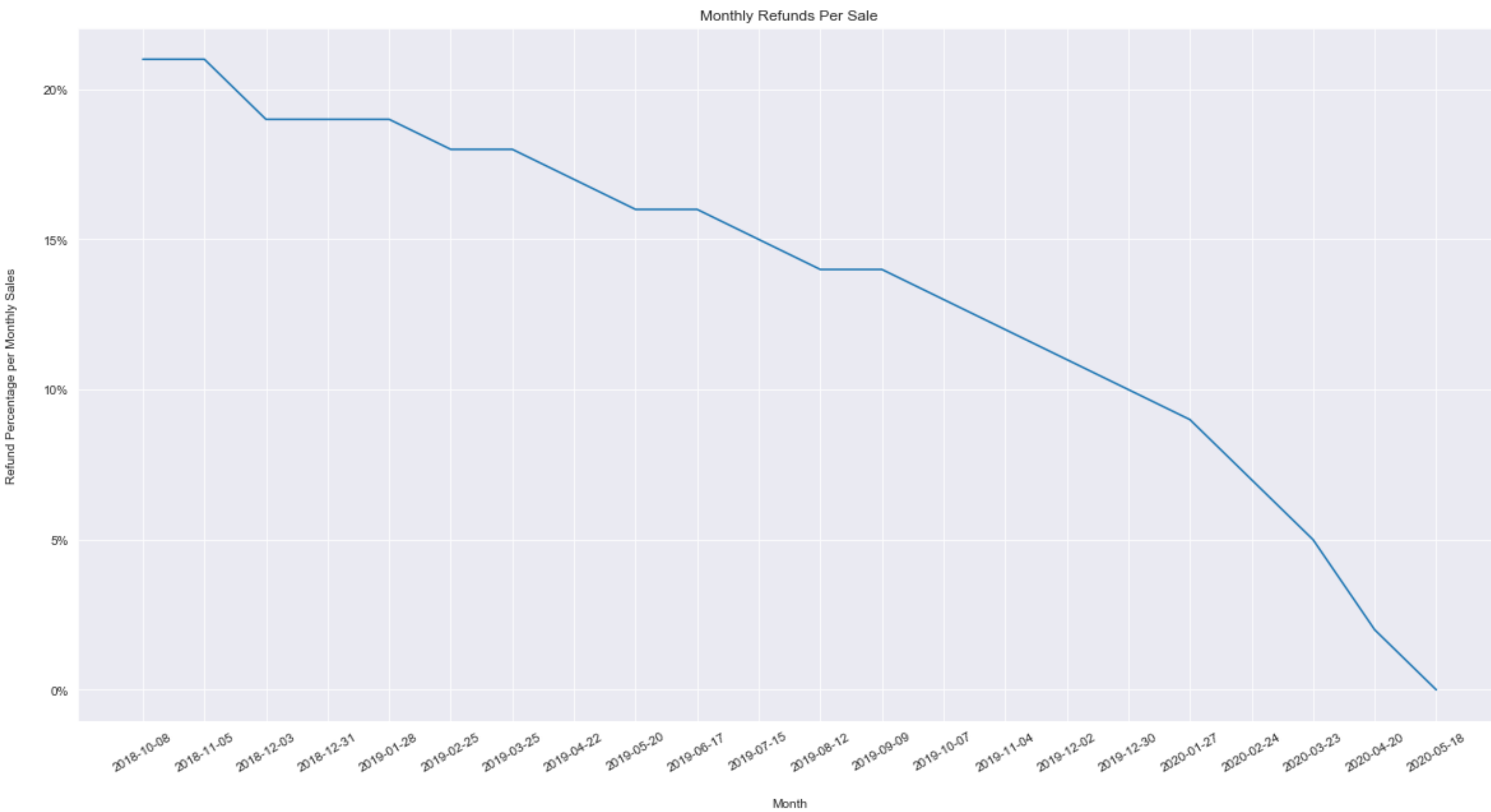
As we can see from the visualization below titled *Weekly Refunds Per Sale Ratio*, about 22% of all sales in the first week in our dataset were refunded. By the first week of January the amount of refunds were down to 10% of sales. On the last week of our data refunds are down to 1% of sales.



Monthly Report

For our monthly report I have grouped our data based on 28 day intervals. That means for every 28 days in our dataset we group all of the sales that have occurred within that 28 day window based on the sale date.

We can see from our analysis that the refund ratio has decreased by **90%** (from 21% refunds to 2% refunds) from October 2018 to April 2020. About 21% of the money we made from sales in the month of October 2018 was refunded. By January 2020 about 10% of the money we made from sales was refunded. The plot above titled *Monthly Refunds Per Sale* demonstrates the decrease in refunds.



BONUS QUESTION

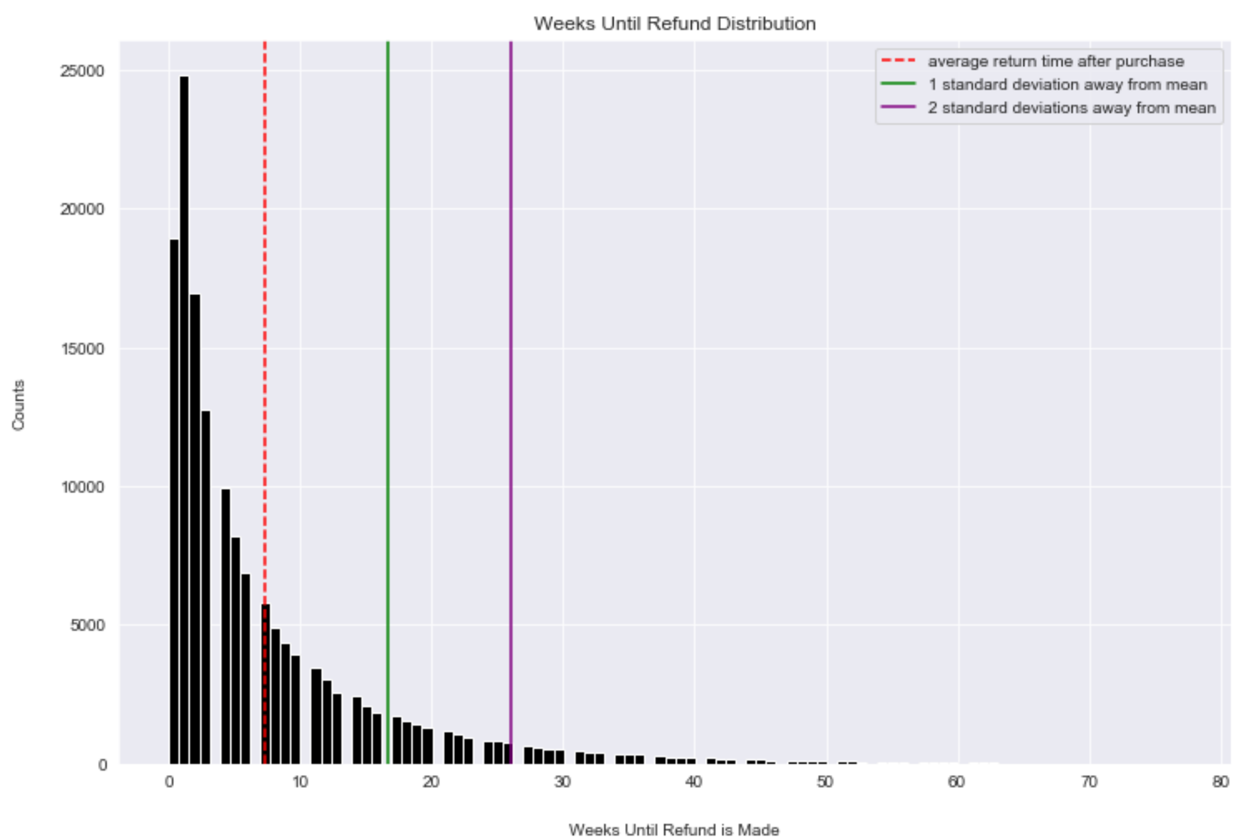
The customer to win is customer_id **889031**. I believe the timezone difference between how the data was stored, and the time the store actually operates was the root of contention. The *sales_at* and *refund_at* data is logged in the database in UTC time zone. However the store operates on an EST time zone. So in our database, the first customer you see making a purchase on 2020-01-01 is actually making the purchase on 2019-12-31 in the '*America/New_York*' EST time zone. The first person who makes a purchase in the stores time zone is **889031**.

Appendix

Amount of Time it Takes for a Refund

Although the percentage of refunds are going way down in relation to sales over time, we need to make sure that we take in account how much time on average a customer takes to make a refund. This way we can make sure our findings for the last couple of months are more reliable.

My analysis has shown that on average it takes roughly 51 days after a purchase for a customer to make a return. That is roughly equivalent to 7 weeks, or 2 months. 95% of our customers are making refunds within 27 weeks after the date of purchase. That is equivalent to around 7 months.



If we take a look at the percentage of refunds seven months prior to our final month, we see the percentage of refunds was only 12%. That is still almost a 50% decrease in the percentage of refunds from October 2018. This still fits in line with our analysis that the percentage of refunds are decreasing.

The plot above titled *Weeks Until Refund Distribution* shows how many refunds we had for each week after a purchase. The red dotted line displays the average number of weeks until a refund is submitted after the purchase. The green and purple lines display the standard deviations away from the mean. The purple line displays the second standard deviation away from the mean, and it shows that 95% of refunds occur 27 weeks after the purchase date.

Python Code

Here is the python code I wrote to explore the data in addition to SQL queries I made.

```
# loading monthly refund data
monthly_refunds = pd.read_csv('Data/A2_Monthly_Grouped_Data.csv')

# loading weekly refund data
weekly_refunds = pd.read_csv('Data/A2_Weekly_Grouped_Data.csv')

# plotting monthly refund percentage
plt.figure(figsize=(20, 10))
g=sns.lineplot(x='month_start_date', y='refund_ratio', data=monthly_refunds)
plt.xticks(rotation=30)
plt.xlabel('Month', labelpad=20)
new_labels = [float(item.get_text()) * 100 for item in g.get_yticklabels()[1:]]
g.set_yticklabels([str(int(label)) + '%' for label in new_labels])
plt.ylabel('Refund Percentage per Monthly Sales', labelpad=20)
plt.title('Monthly Refunds Per Sale')
plt.show()


# plotting weekly refund percentage
plt.figure(figsize=(20, 10))
l=sns.lineplot(x='week_start_date', y='refund_ratio', data=weekly_refunds, c='red')
plt.xticks(rotation=70)
plt.xlabel('Week', labelpad=20)
new_labels = [float(item.get_text()) * 100 for item in l.get_yticklabels()[1:]]
l.set_yticklabels([str(int(label)) + '%' for label in new_labels])
plt.ylabel('Refund Percentage per Weekly Sales', labelpad=20)
plt.title('Weekly Refunds Per Sale ratio')
plt.show()
```