CSI 4142 - Project Phase 4: Data Mining

uOttawa

Elli Traboulsi #300175764

Tamara Micic #300163312

Teodora Vukojevic #300199584

Group #4

Prof. Yazan Otoum

TA Lansu Dai

April 9th, 2024

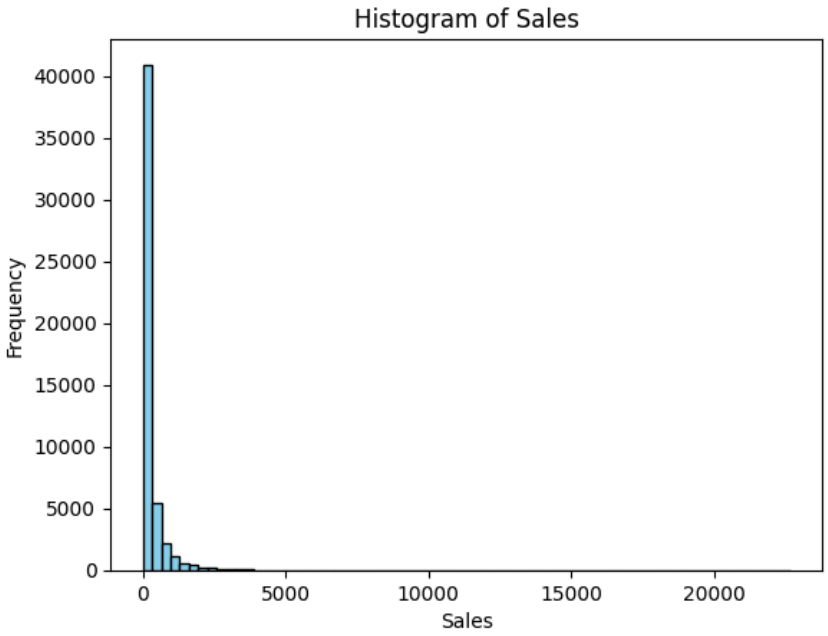
Table of Contents

Part A. Data summarization, data preprocessing and feature selections

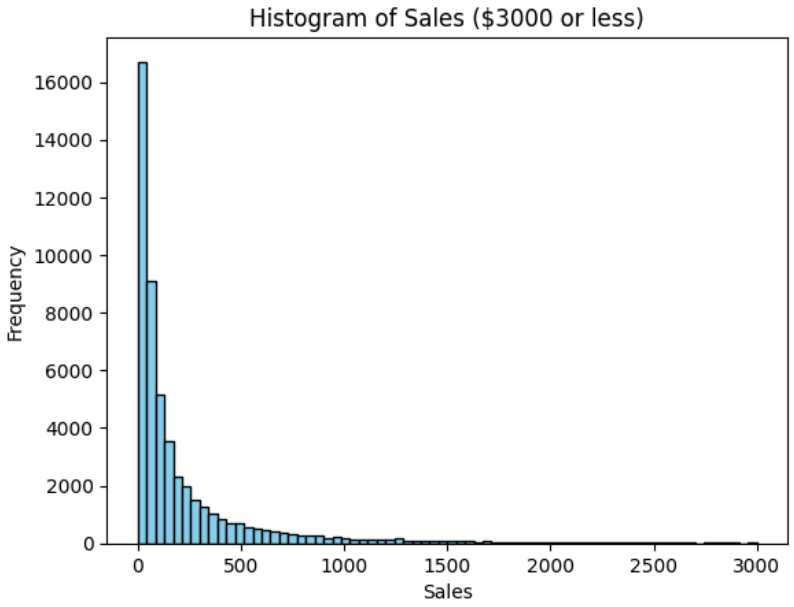
1. Data Exploration and Summarization

Histograms

First, we created histograms for all the continuous variables. That is, we have the continuous variable on the x-axis and the frequency on the y-axis.

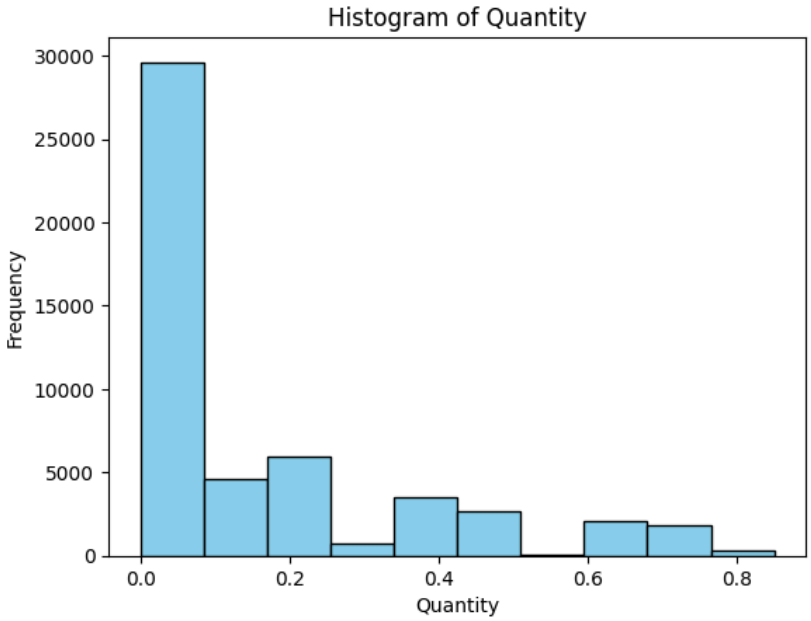


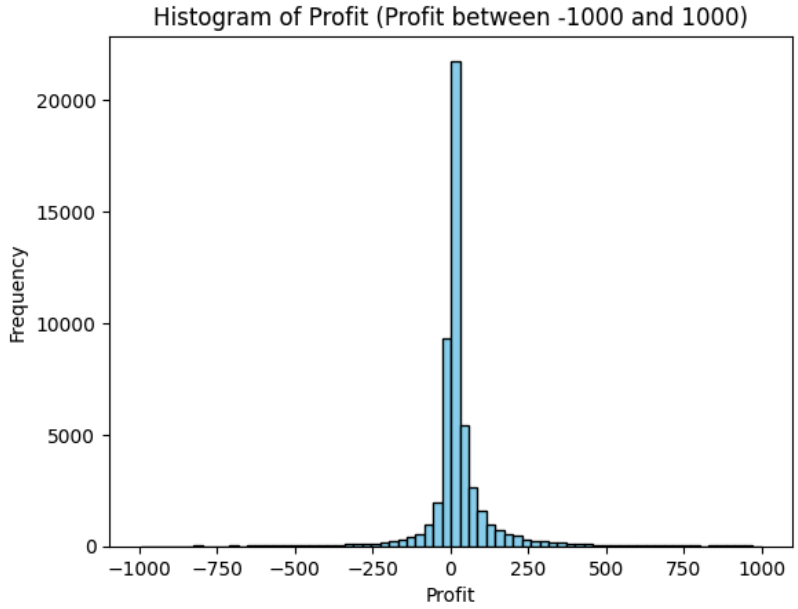
For example, in this histogram, we see that most of the sales are of low value. We can zoom in to get a better picture:

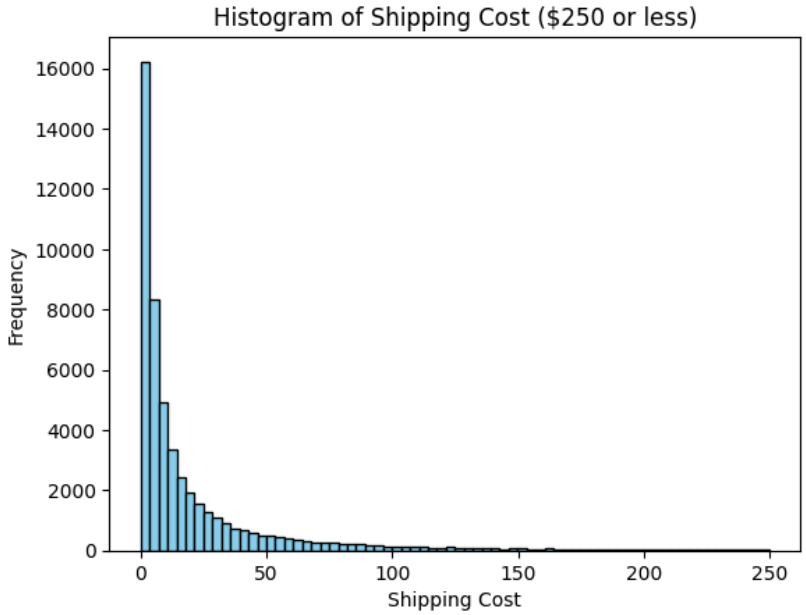


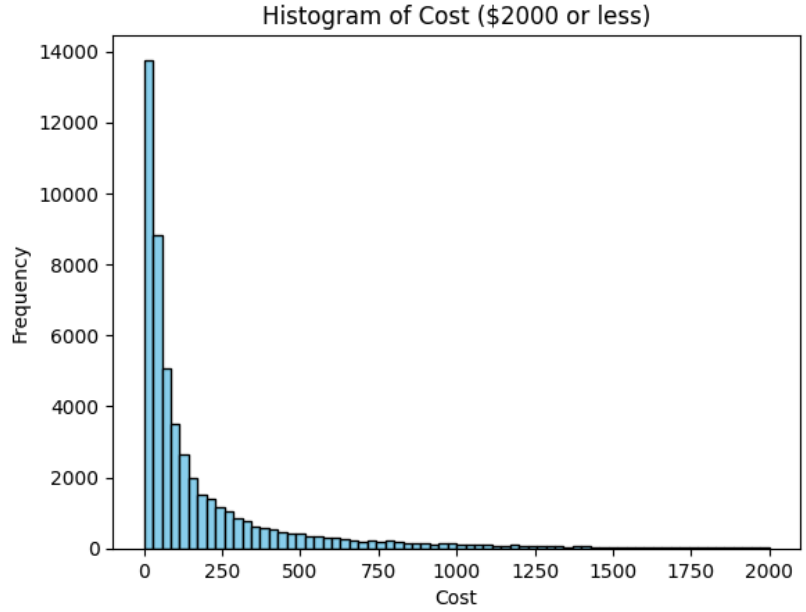
As the sale value increases, the number of sales decreases.

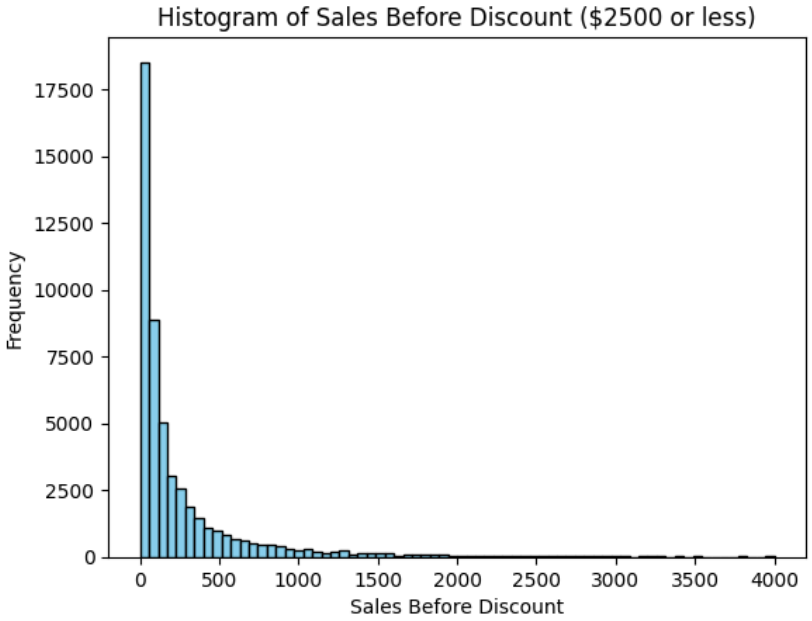
Here are the histograms for the other continuous variables:

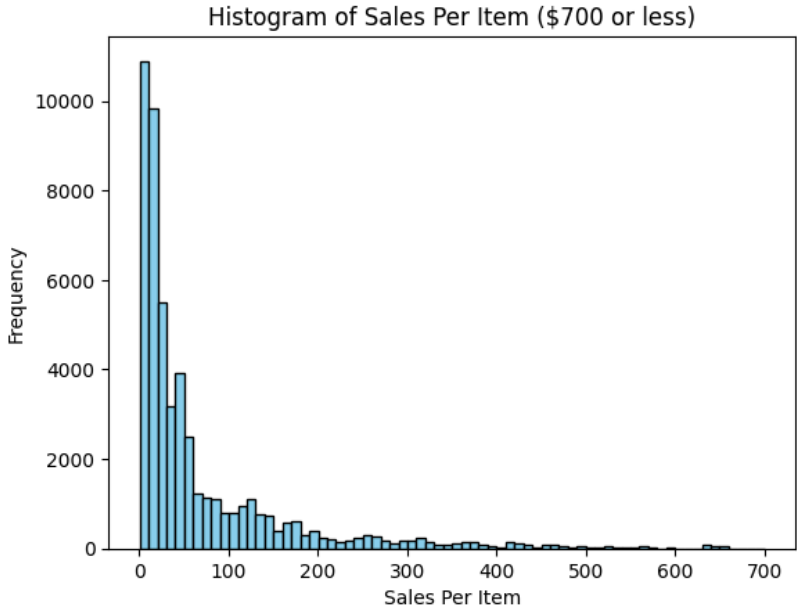

Histogram of Quantity
Frequency
Quantity
30000
25000
20000
15000
10000
5000
0
0.0
0.2
0.4
0.6
0.8

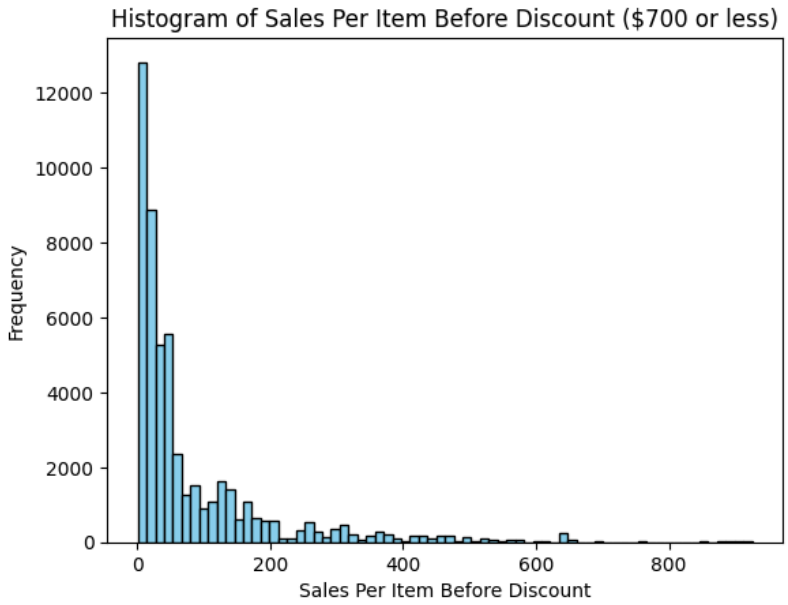

Histogram of Profit (Profit between -1000 and 1000)
Frequency
Profit
20000
15000
10000
5000
0
−1000
−750
−500
−250
0
250
500
750
1000


Histogram of Shipping Cost ($250 or less)
Frequency
Shipping Cost
0
2000
4000
6000
8000
10000
12000
14000
16000
0
50
100
150
200
250


Histogram of Cost ($2000 or less)
Frequency
Cost
0
2000
4000
6000
8000
10000
12000
14000
0
250
500
750
1000
1250
1500
1750
2000

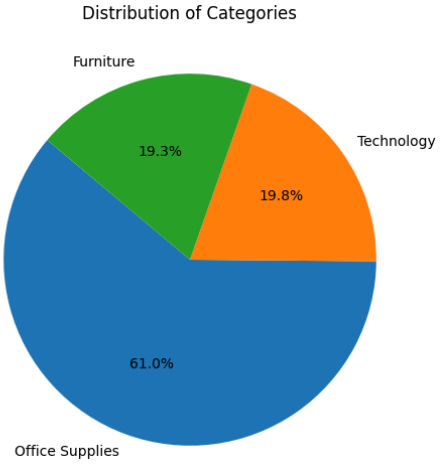

Histogram of Sales Before Discount ($2500 or less)
Frequency
Sales Before Discount
0
2500
5000
7500
10000
12500
15000
17500
0
500
1000
1500
2000
2500
3000
3500
4000


Histogram of Sales Per Item ($700 or less)
Frequency
Sales Per Item
0
2000
4000
6000
8000
10000
0
100
200
300
400
500
600
700

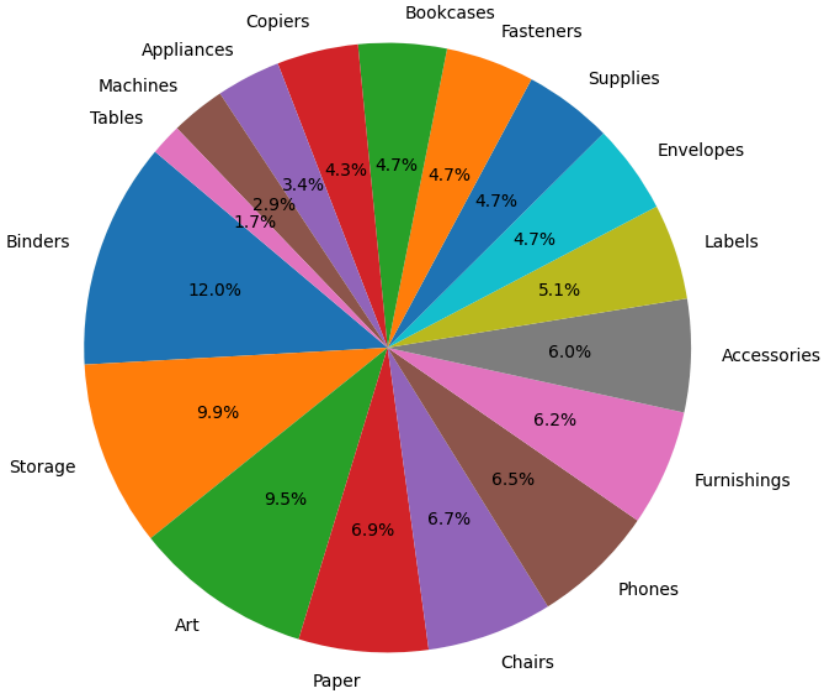


Pie-Charts

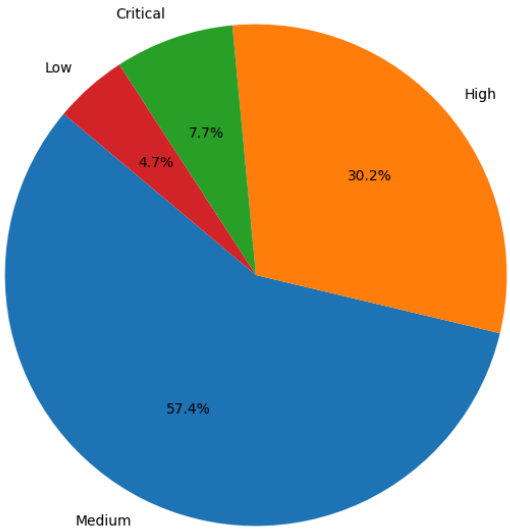
We created pie-charts to show the distribution of categories and sub-categories:

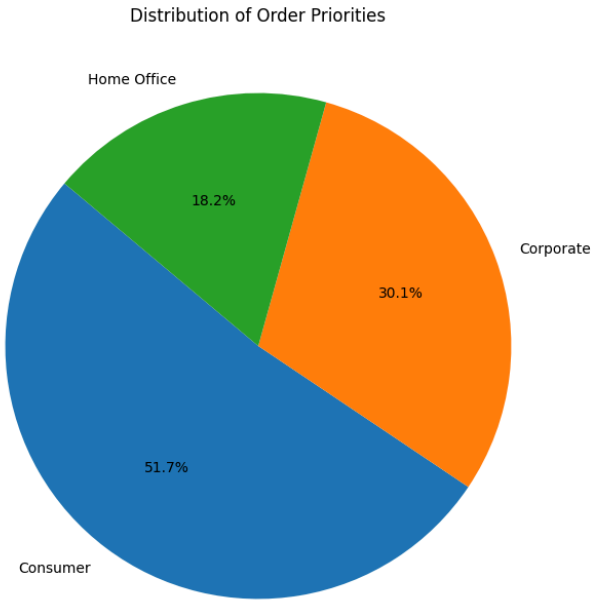


Distribution of Sub-Categories



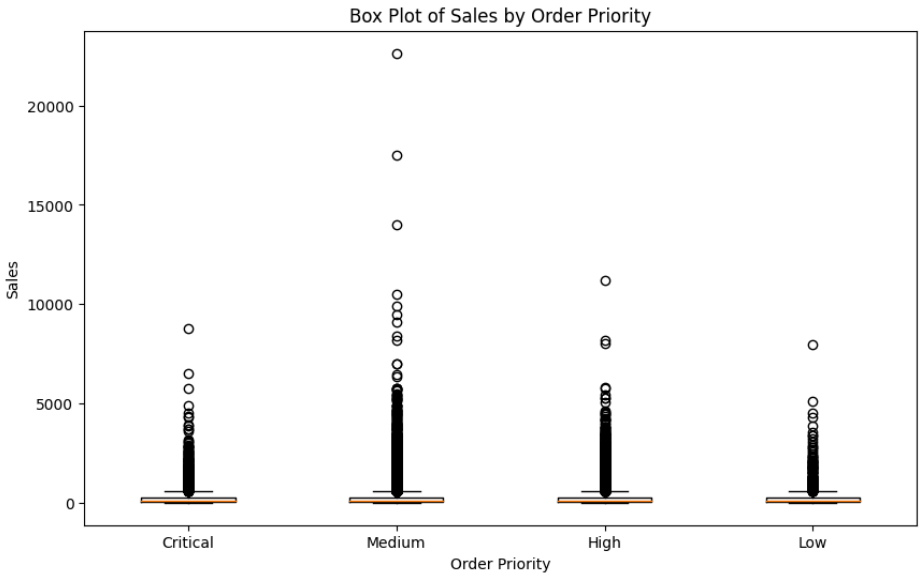
Distribution of Order Priorities





Box-Plot

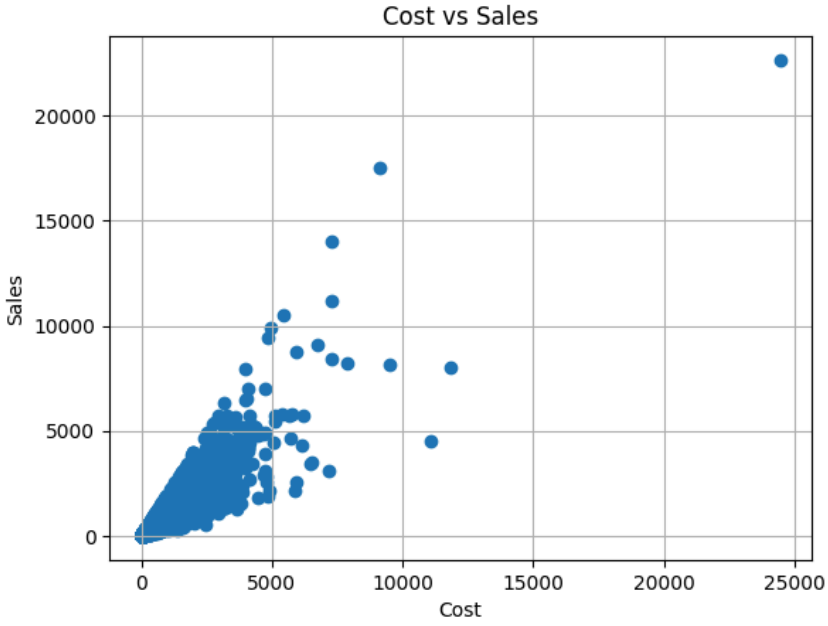
We created the following box plot that shows the distribution of sales per order priority:

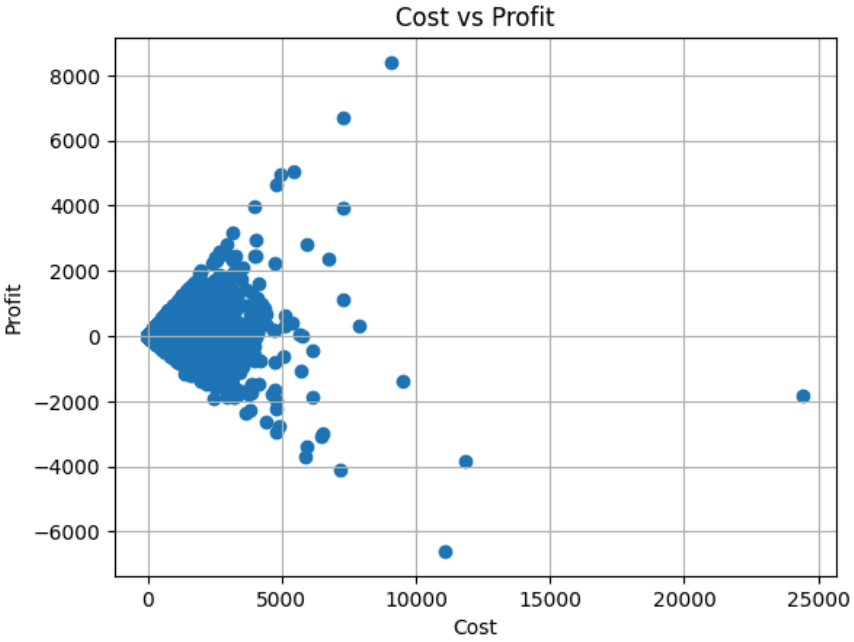


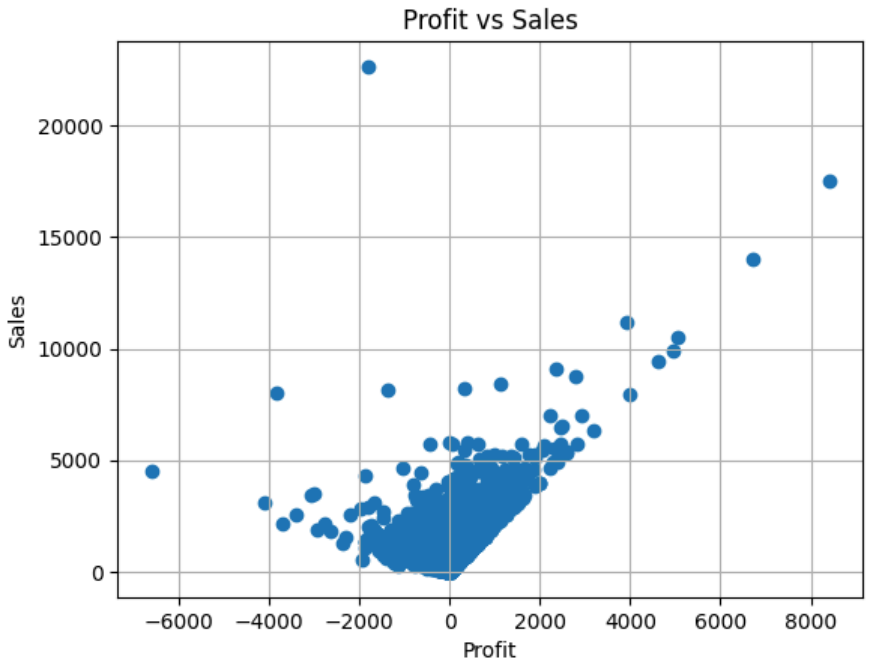
However, since there are so many data points, and many of the sales are small in value, the interquartile range is small and there are many data points that are outliers. So boxplots do not make a lot of sense given the nature of our data.

Scatter-Plots

We did scatter plots for Cost, Sales, and Profit, in different combinations:

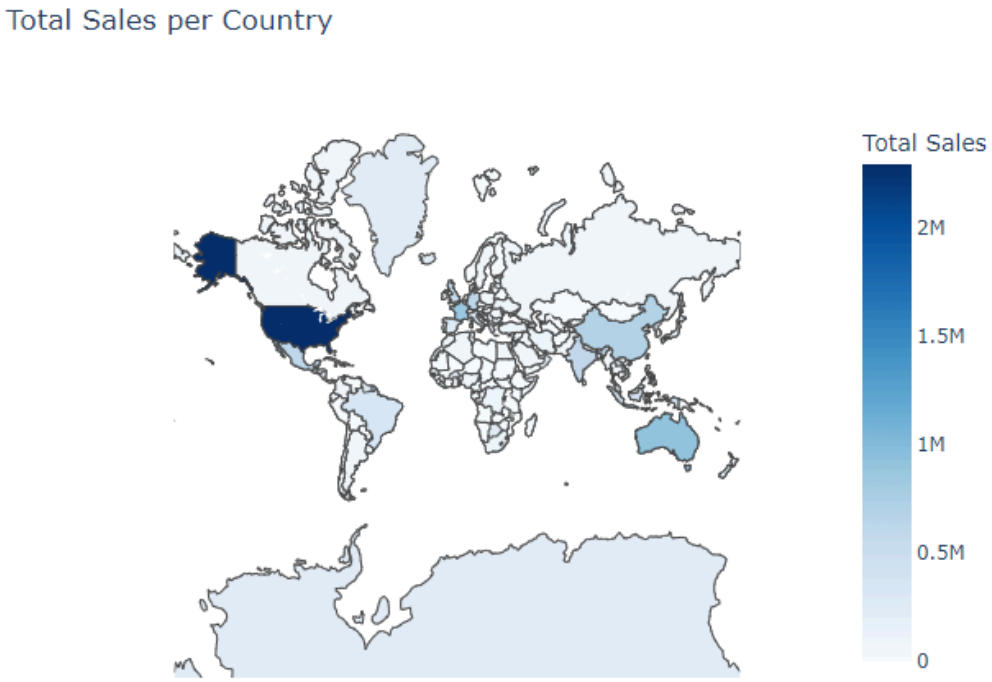






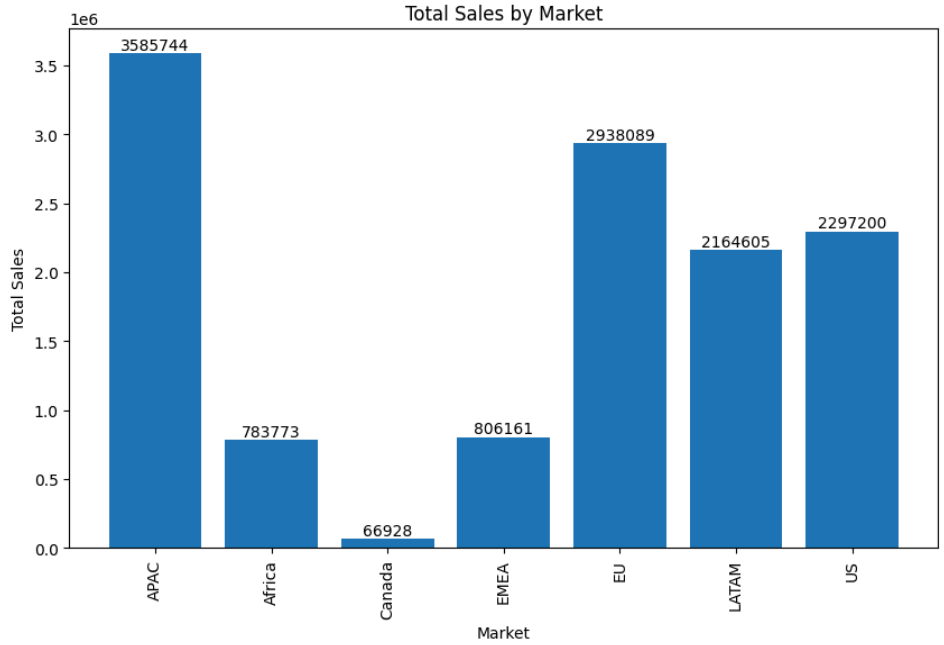
Map

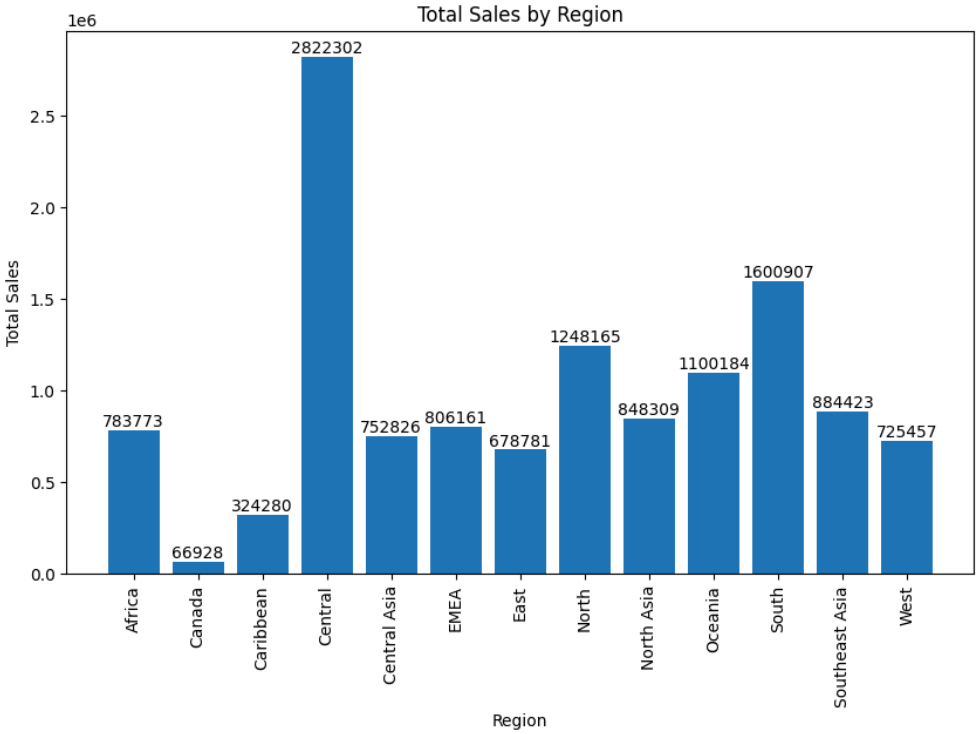
We did a map that shows the distribution of sales by country. The darker colored countries have more sales. The most sales are in the United States.



Bar Charts

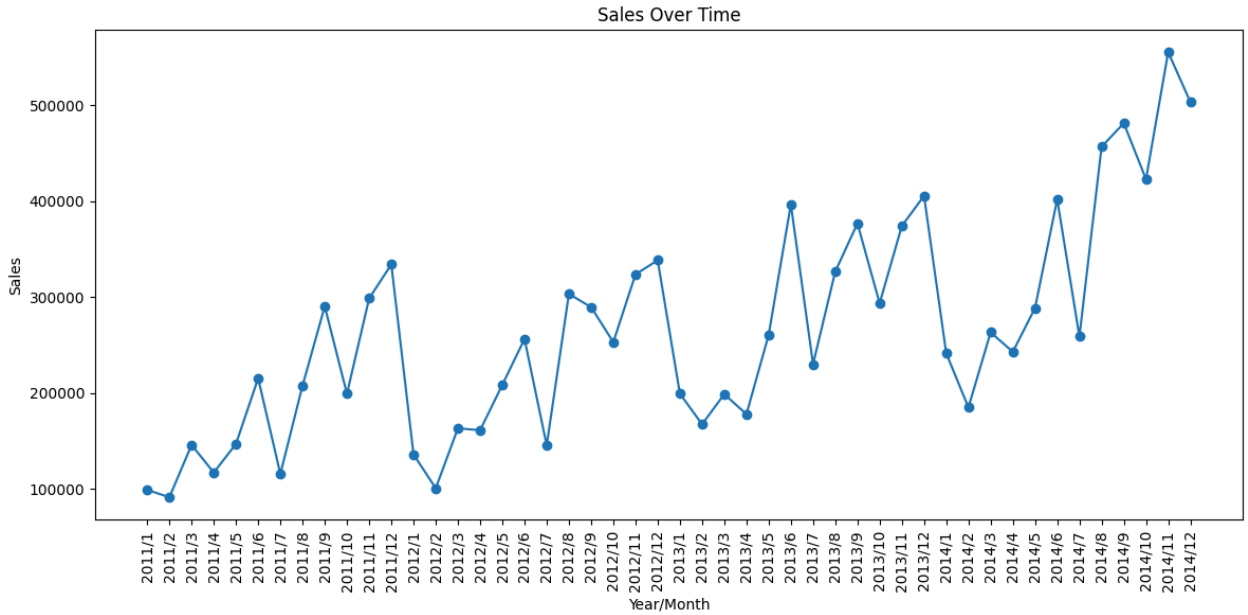
We did bar charts that show sales by Market and by Region. APAC (Asia-Pacific) is the highest of the markets, and Central (America) is the highest for Region.



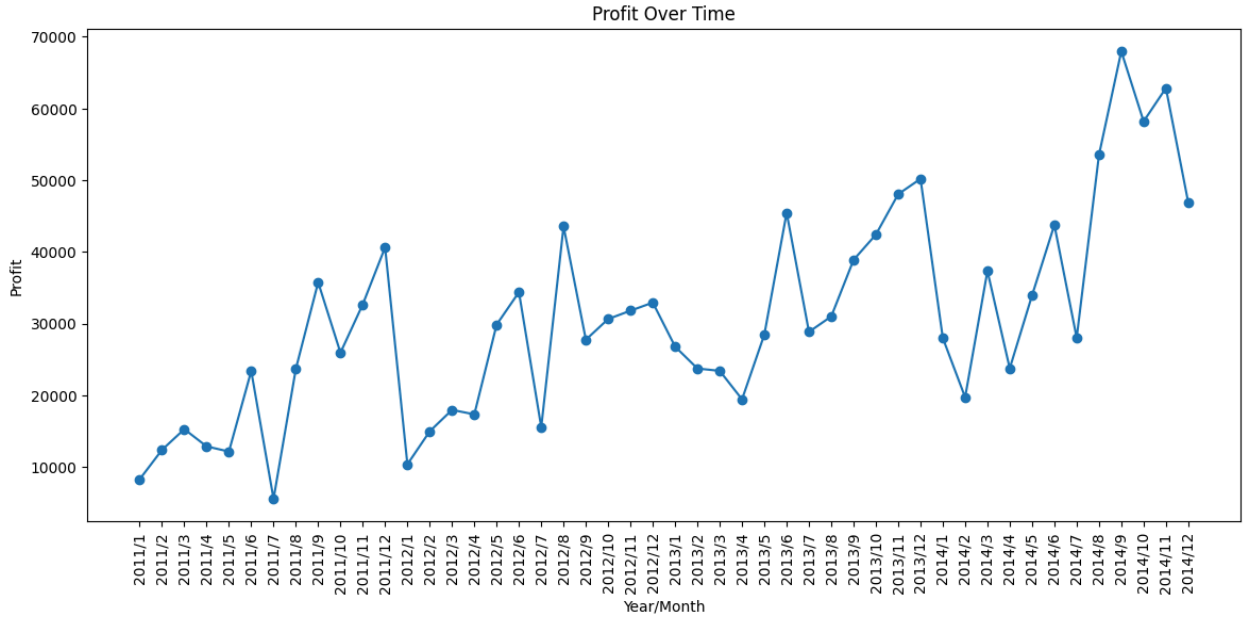


Line Graphs

We created two line graphs that show sales over time and profit over time.



It is interesting to see a repeated annual trend. Sales drop in July (2011/7, 2012/7, 2013/7, 2014/7) and are at a high in December (2011/12, 2012/12, 2013/12, 2014/12). There is also an overall increase in sales.



Similar patterns are seen for profit.

2. Data Preprocessing and Feature Selections

One page summary of Data Preprocessing (Deliverable Part A):

To do classification, first we needed to do some preprocessing of the data. First, we dropped certain columns. We dropped ‘Order Priority’, since we also have ‘Order Priority Numerical’ that is more useful, since it is a numerical ordering of order priority (4, 3, 2, 1, instead of ‘Critical’, ‘High’, ‘Medium’, ‘Low’).

Next, we attempted to do one hot encoding for all categorical variables, but there were so many that this used all available RAM and the session crashed. So we decided to drop Order ID, Order Date, Ship Date, Customer ID, Customer Name, and Product ID, since those categorical variables have a lot of unique values (so they require a lot of space for encoding) and we believe that they are not significant for sub-category prediction. So we ended up applying one hot encoding to Ship Mode, Segment, City, State, Country, Market, Region, Category, and Product Name.

Product Name_Zebra ZM400 Thermal Label Printer	Product Name_Zebra Zazzle Fluorescent Highlighters	Product Name_Zipper Ring Binder Pockets	Product Name_i.Sound Portable Power - 8000 mAh	Product Name_iHome FM Clock Radio with Lightning Dock	Product Name_iKross Bluetooth Portable Keyboard + Cell Phone Stand Holder + Brush for Apple iPhone 5S 5C 5, 4S 4	Product Name_iOttie HLCRIO102 Car Mount	Product Name_iOttie XL Car Mount	Product Name_invisibleSHIELD by ZAGG Smudge-Free Screen Protector	Product Name_netTALK DUO VoIP Telephone Service
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
...	...	...	...	...	...	...	...	...	...
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

Now we see that many of the values are 0s, and there are 1s as well (not shown in the screenshot), and the dataframe now has 8712 columns.

Then, we normalized the numerical attributes: Sales, Quantity, Discount, Profit, Shipping Cost, Order Priority Numerical, Cost, Sales Before Discount, Sales Per Item, Sales Per Item Before Discount, Order Year, Order Month, Order Day, Ship Year, Ship Month, and Ship Day. After normalization, all numerical values are between 0 and 1:

```
global_superstore_preprocessed_phase4['Profit']
```

```
0        0.490812
1        0.420749
2        0.501331
3        0.433564
4        0.460768
           ...
51285    0.440300
51286    0.439926
51287    0.440749
51288    0.440160
51289    0.440120
Name: Profit, Length: 51290, dtype: float64
```

Finally, we took a look at what the 10 best features for predicting Sub-Category are. They were found to be Sales, Shipping Cost, Cost, Sales Before Discount, Sales Per Item, Sales Per Item Before Discount, Market_US, Category_Furniture, Category_Office Supplies, and Category_Technology. These are the features that will be used in the next classification step.

Part B. Classification (Supervised Learning)

Summary Table

	Decision Tree	Boosting	Random Forest
Accuracy	0.846753753168259	0.6657243127315267	0.7361084031975044
Precision	0.8476075082569683	0.6808726451577332	0.7350313930404999
Recall	0.846753753168259	0.6657243127315267	0.7361084031975044
Runtime	2 seconds	3 minutes	15 seconds

200 Word Summary on Quality of Algorithms:

Interestingly, decision tree performed the best out of the three algorithms, with accuracy, precision, and recall all around 85%. Random Forest was the second best algorithm, with accuracy, precision, and recall all around 74%, and boosting had the worst accuracy, precision, and recall, all between 66-69%. We expected different results, in particular we expected random forest to perform better than decision tree, since random forest is made up of multiple decision trees. We also expected boosting to perform better since this is an ensemble method of multiple machine learning methods that focuses on correcting errors of previous models. Perhaps this is due to the specific characteristics of the dataset and how each algorithm interacts with those characteristics. It is possible that the dataset doesn't have a high degree of variance that random forest could effectively capture through its approach. Similarly, boosting may not have had enough diversity in weak learners to effectively minimize errors. Decision trees might have been able to capture the underlying patterns in the data more accurately due to their simplicity and flexibility.

Overall, we would rank decision tree as the best model, then random forest, then boosting, since that is the order of decreasing accuracy, precision, and recall, and it is the order of increasing runtime.

200-300 Word Summary of Actionable Knowledge Nuggets

While investigating the models, we learned the importance of splitting the data into train and test sets. This is crucial to avoid overfitting. Therefore, we split the data into 80% training and 20% test. We also learned the importance of all the preprocessing steps that were done in the previous part. It is necessary to encode the categorical variables, as the models need numerical values (0s and 1s, in this case) as input. The normalization step is also very important, so that all the features are brought to a similar scale, preventing certain features from dominating the model training process due to differences in their magnitude. This ensures that the model learns from each feature appropriately and avoids biases that may arise from variations in feature scales. Finally, using only the 10 best features was a good idea, as to not clutter the models with too many features. It also helps to reduce the dimensionality of the dataset, which can lead to simpler and more interpretable models, as well as potentially improving model generalization and performance.

We also looked at the precision and recall values of each specific sub-category. For example, here are the results using decision tree:

	precision	recall	f1-score	support
Accessories	0.90	0.92	0.91	595
Appliances	0.88	0.90	0.89	337
Art	0.86	0.86	0.86	953
Binders	0.82	0.80	0.81	1246
Bookcases	0.91	0.94	0.93	470
Chairs	0.91	0.89	0.90	691
Copiers	0.95	0.95	0.95	461
Envelopes	0.76	0.70	0.73	484
Fasteners	0.61	0.65	0.63	469
Furnishings	0.96	0.94	0.95	635
Labels	0.71	0.71	0.71	505
Machines	0.91	0.90	0.90	309
Paper	0.76	0.82	0.79	689
Phones	0.92	0.92	0.92	698
Storage	0.89	0.88	0.89	1051
Supplies	0.80	0.76	0.78	501
Tables	0.79	0.85	0.82	164
accuracy			0.85	10258
macro avg	0.84	0.85	0.84	10258
weighted avg	0.85	0.85	0.85	10258

This shows us which sub-categories are more (higher score) or less (lower score) predictable. For example, ‘Copier’ has a precision/recall of 0.95, meaning it is very predictable based on features. A useful application that we thought of for this is, for the sub-categories with high precision/recall (e.g. copiers), the company could advertise these items to customers with similar features. We could look at the features that make a person likely to buy a copier, and then we would advertise copiers to other people with similar features (e.g. Market US), since they would be more likely to make a purchase.

Part C. Detecting Outliers

200-300 Word Summary on Outliers

To find outliers, we used the one-class SVM (Support Vector Machine) algorithm, which is an unsupervised learning technique that uses the idea of minimizing the hypersphere of a single class of examples in training data and considers all the other samples outside the hypersphere to be outliers or out of the training data distribution. SVM takes numerical attributes as input, so we only extracted the numerical attributes from our dataset for outlier detection. We then scaled the data, so that different attributes would be treated equally. Then we trained the model, with kernel = rbf, which stands for radial basis function. It is the default setting and it is able to capture non-linear relationships, making it more flexible. We also set nu = 0.001. nu is the upper bound on the fraction of margin errors and a lower bound of the fraction of support vectors. A lower value of nu makes the model more sensitive to outliers and results in a larger fraction of data points being classified as outliers. A higher value of nu makes the decision boundary smoother and classifies fewer data points as outliers. We chose nu = 0.001 – this way we got 256 outliers, which we thought was a good amount. When looking at the outliers, we notice that a lot of values are quite extreme. For example, most of the Sales values are more than $1000, whereas the mean Sales value is $246.49. There are also extreme Profit values, particularly negative Profit values, which is odd, since we would expect positive Profit. Furthermore, some Shipping Cost values are over $900, which is quite big. Overall, the outlier data the one-class SVM algorithm selected seems to make sense. These transactions should be further investigated for suspicious activities, such as fraud or identity theft.