**Data Mining Techniques on Customer Relationships**

John Thorme

Fairfield University

SWEG6518, Data Mining and Business Intelligence

**1. Introduction**

**1.1 Task Description: What are you doing?**

To operate a company to its best ability, one must understand its customers to keep an edge. A way to do this is to divide the customers into groups based on data to see what suits your best customers and where you can improve.In this research article, we explore the use of machine learning algorithms, namely K-nearest neighbors (KNN), decision tree, and logistic regression, to segment customers based on their behavior and predict their likelihood of making a purchase or being satisfied with the company's service. We aim to determine which algorithm provides the best results for customer segmentation and prediction**.**

We begin by describing the dataset used in this study and the preprocessing steps taken to prepare the data for analysis. Then, we present each algorithm's results and discuss their strengths and weaknesses. Finally, we compare the algorithms' performance and draw conclusions on which algorithm is most effective for customer relationship/similarities and prediction.

Overall, the study's findings can help businesses make informed decisions about their customer similarities which can improve strategies to increase customer satisfaction and sales.

**1.2 Data Description:**

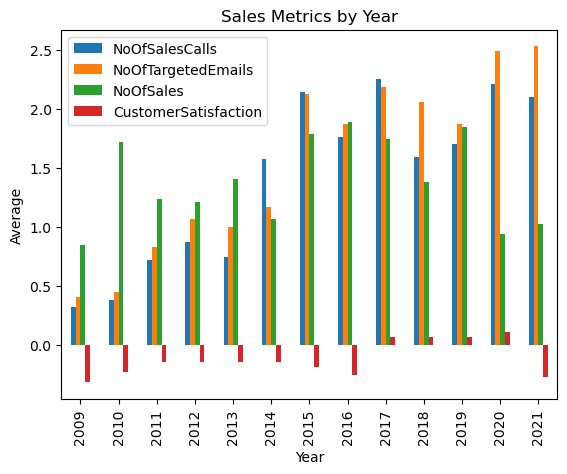
Our dataset does not have that many columns and the columns are, Customer, year, NoOfSalesCalls, NoOfTargetedEmails, NoOfSales, CustomerSatisfaction. Customer column is an id column for each customer ranging 1-47. Year is the year of purchase ranging from 2009-2021. NoOfSalesCalls counts how many sales calls were made to the customer in a given year ranging 0-5. NoOfTargetedEmails counts how many emails were sent to each customer for each year ranging 0-3. NoOfSales counts how many sales were made to the customer each year. Customer Satisfaction is a scale from -1 to 1. Our data was collected on a grain of individual collection.

**2. Data Preparation**

**2.1 Data Exploration**

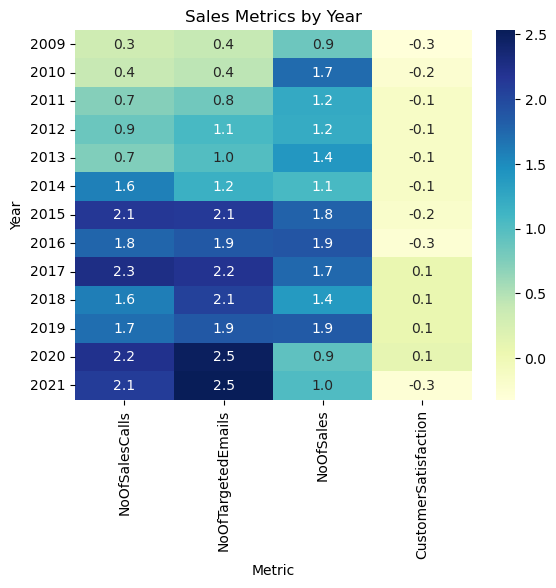
To explore the data, we used several techniques such as statistical analysis and visualization. The statistical analysis helped us understand the distribution of the data and identify any outliers. The visualization techniques, such as bar charts and scatter plots, helped us identify any correlation between the features and the target variable. While exploring the data we weighed the different learning algorithms and which can be best.

**2.2 Data Visualization**

To explore the data a bar chart and heat map were generated on year and customer average for all categories. 

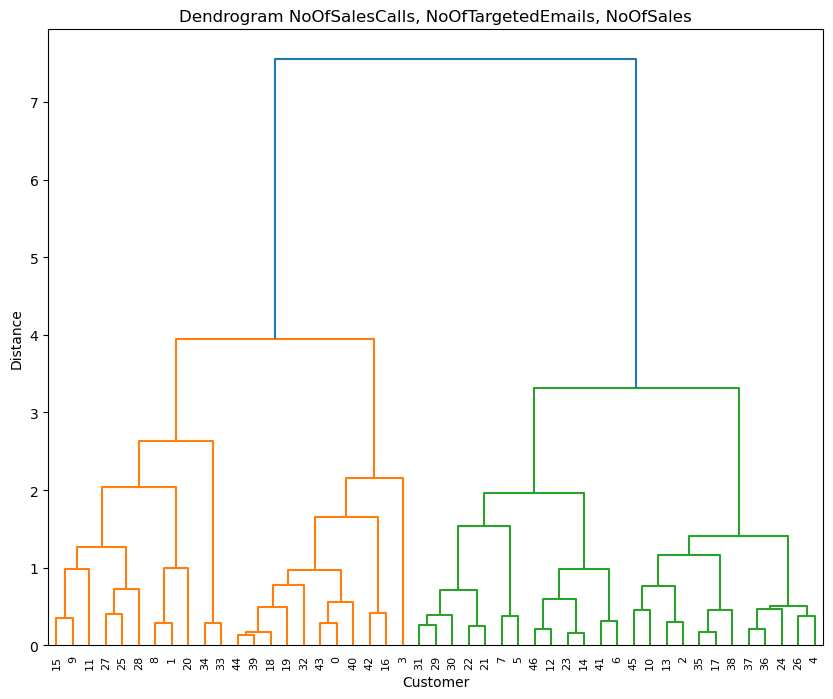
*Fig 1. Bar chart grouping by year and average of all customers in the year.*

This bar chart shows some interesting findings. Year 2017 through year 2020 were the only years in which customer satisfaction was on average positive. It should be noted the scale for CustomerSatisfaction is -1,0,1 so for those years it averaged around 0 or neutral. We also learn that the year 2020 and 2021 had the NoOfTargetedEmails average. Lastly, the year 2010 had one of the highest average NoOfSales while having the second lowest NoOfSalesCalls and NoOfTargetedEmails. The insights were derived from the bar chart above and the heat map below.

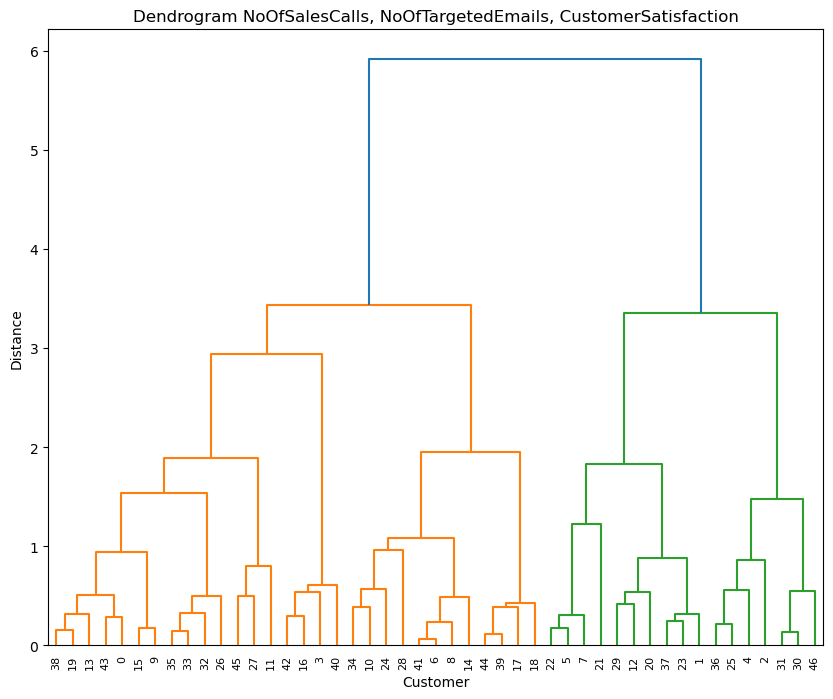


*Fig 2. Heat map of years and averages for customers*

Lastly, hierarchical clustering was used to see if there are natural groups. The idea is to explore grouping customers based on sales and compare to the grouping customers based on customer satisfaction. The distance on the y-axis is the distance between the customers on the X-axis, or how similar two customers are and so on as distance grows. In Fig. 3, customers 35 and 19 have a distance close to 0 meaning they are very similar.



*Fig 3. Dendrogram of natural groupings NoOfSalesCalls, NoOfTargetedEmails, and NoOfSales*

*****Fig 3. Dendrogram of natural groupings NoOfSalesCalls, NoOfTargetedEmails, and CustomerSatisfaction*

**2.3 Preprocessing**

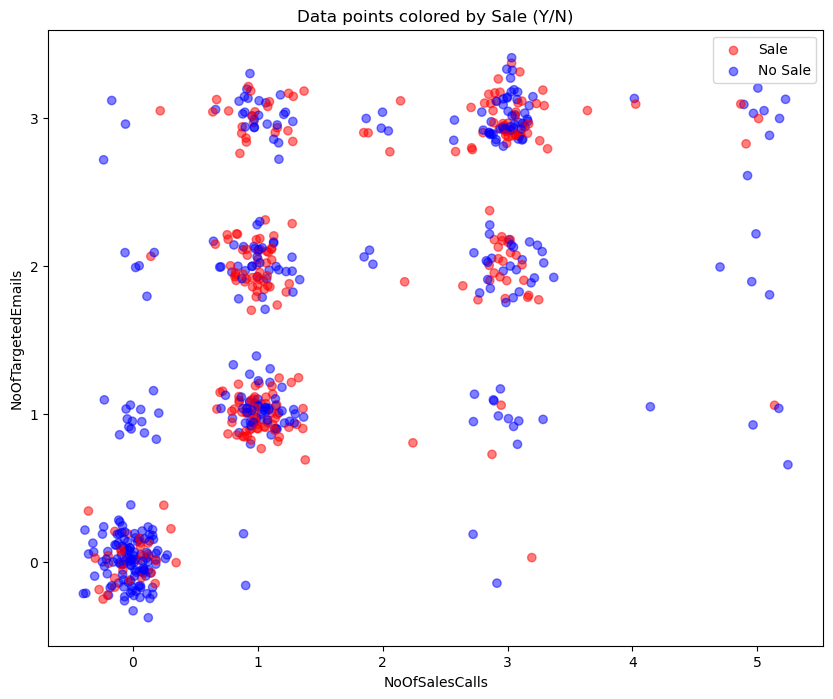
The first step we took when exploring the data was to see what data is being dealt with and if there were any nulls to handle appropriately. The data is clean and had no missing values dealing with and if there factors could be leveraged to see a relationship develop between the customers. KNN is best when the X are numeric values and the y value is a binary yes and no. The two columns CustomerSatisfaction and NoOfSales can be of use to help for a relationship and groupings.

**3. Feature generation and transformation : scaling, dimension reduction etc**

After evaluating the data visualizations it was clear to execute a KNN algorithm in which to cluster the customers by to see relationships and similarities. The customer column repeats customer id numbers every year and to avoid the data being read as only one line accidently a new column Customer\_Id\_Year was created by combining the Customer column along with year to read as such, 1.2019. This way on the scatter chart for KNN, all points can be read on the chart. As to develop the scatter points color to signify additional columns were created. NoOfSales was used to create Sale\_Status in which values greater than 0 became “Yes Sale” and 0 values became “No Sales”, there were no negative values for this column. Customer Satisfaction was used to create CustomerSatisfactionLeve**l** in which the value -1 is “Not Satisfied” and all else is “Satisfied”. A value of 0 was interpreted as neutral and grouped into the “Satisfied” category. The use of these categorical values did not go too well in KNN and in turn led to the creation of binary columns with values of 0 or 1. Sale\_Status became NoOfSales\_YN where 0 is “No Sale” and 1 is “Sale”. CustomerSatisfactionLeve**l** becameCustomerSatisfaction\_YN where 0 is “Not satisfied.

Also, the sklearn import standard scaler was implemented on the data to scale appropriately. Standard scaler is beneficial in KNN to scale the features to a similar range, so that the distances between data points are not dominated by a single feature. KNN creates its clusters and groupings based on distance. If one feature has a much larger range of values than the other features, then that feature will dominate the distance calculation and may lead to incorrect nearest neighbors being identified. This allows KNN to give the appreciated weight to each feature and ensure the mean is 0 and standard deviation of 1.

In our code it is possible that two or more customers can have the same NoOfSalesCalls, NoOfTargetedEmails, NoOfSales, and CustomerSatisfaction rating. To see the customers clearly, np.random.normal(scale=0.1, size=len(data), was applied to increase or decrease the customers data to see the data points more clearly. The KNN algorithm is a supervised learning algorithm that makes predictions based on the distances between the nearest data points. The algorithm does not depend on any randomness from the data, so the use of np.random.normal should not affect the performance of KNN.

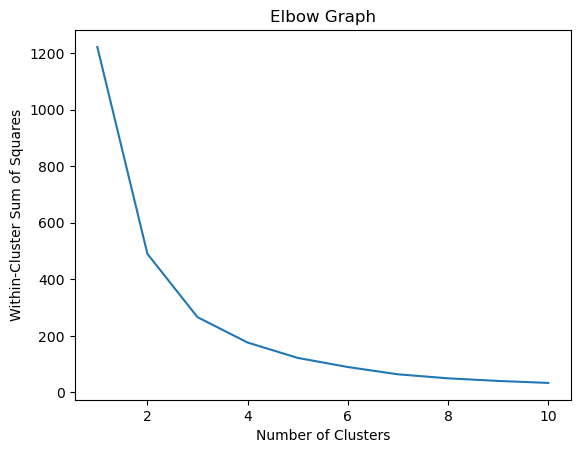


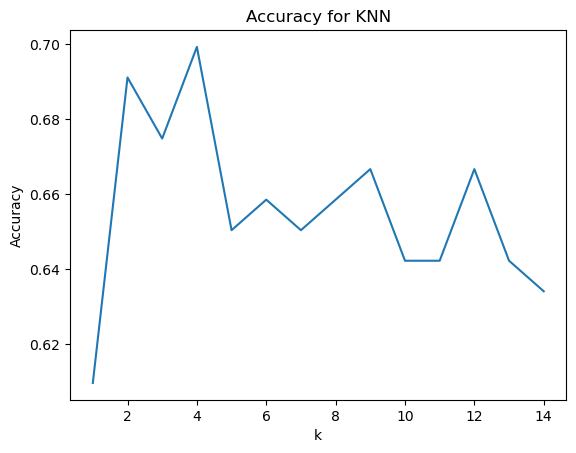
*Fig. 4 Scatter plot of pre analyzed KNN data with noise*

**4. Model development**

Our model was developed using KNN and was made twice to compare which would be the best for identifying similarities between the customers. The first KNN model was developed by looking at the relationship between NoOfSalesCalls and NoOfTargetedEmails where the points on the scatter plot represented NoOfSales\_YN. NoOfSalesCalls and NoOfTargetedEmails were then standardized to be on the same range.

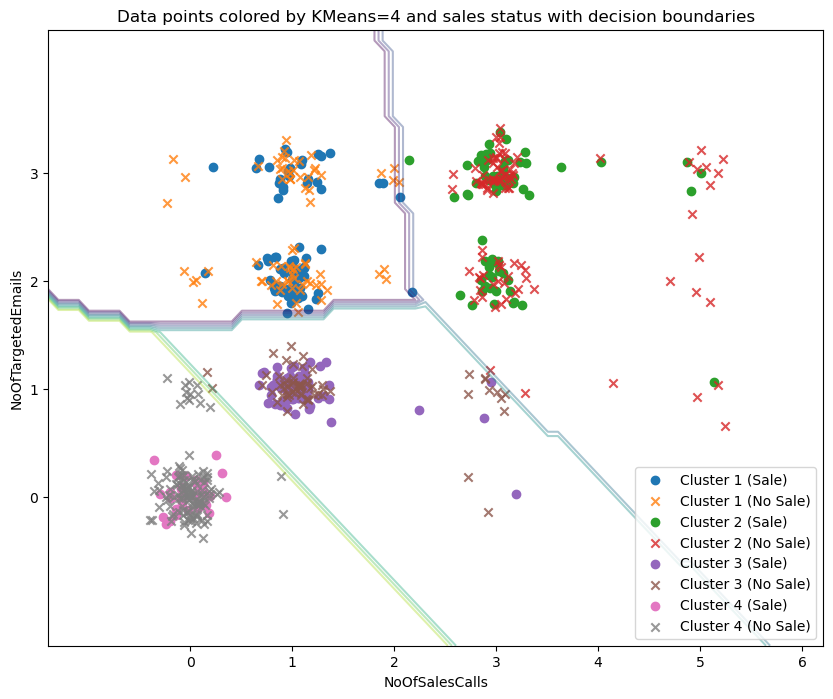
To obtain the optimal K two charts were used, Elbow graph and a chart comparing accuracy on the y and K on the X.





*Fig. 5 Elbow graph and an Accuracy x K graph for optimal K.*

Based on the Accuracy x K chart it appears the most optimal K is K=4. If K is too small (e.g., k=1), the model can be overly sensitive to noise in the data and may overfit. This can result in poor generalization to new data. If K is too large (e.g., k=n, where n is the total number of training samples), the model may become too simple and may underfit the data. This can result in poor performance on the training set and new data.

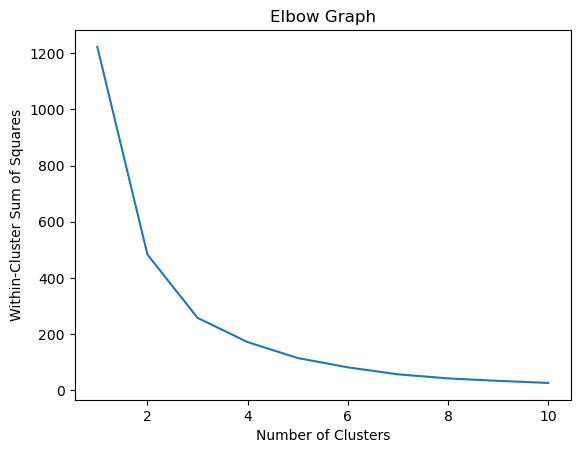


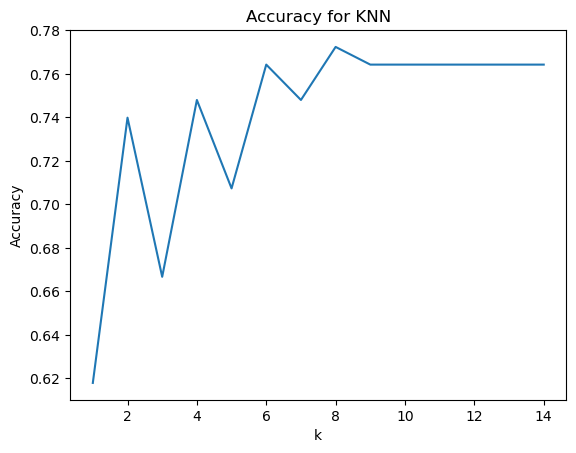
*Fig 6. K=4 cluster for sale status relationships (Sale or No Sale) with decision boundaries*

Following the development of the KNN model, we decided to explore the effectiveness of logistic regression in predicting customer sales. Logistic regression is a commonly used classification technique that can be especially useful in identifying the relationship between predictor variables and a binary outcome, such as whether or not a customer makes a purchase. For this logistic regression our predictors were NoOfSalesCalls and NoOfTargetedEmails and binary outcome NoOfSales\_YN.

Lasty, we developed decision trees in predicting customer sales. Decision trees were developed because they map out all possible outcomes and their probabilities. They can be particularly useful in identifying the most important variables for predicting a binary outcome, such as whether a customer makes a purchase or not, using the same inputs

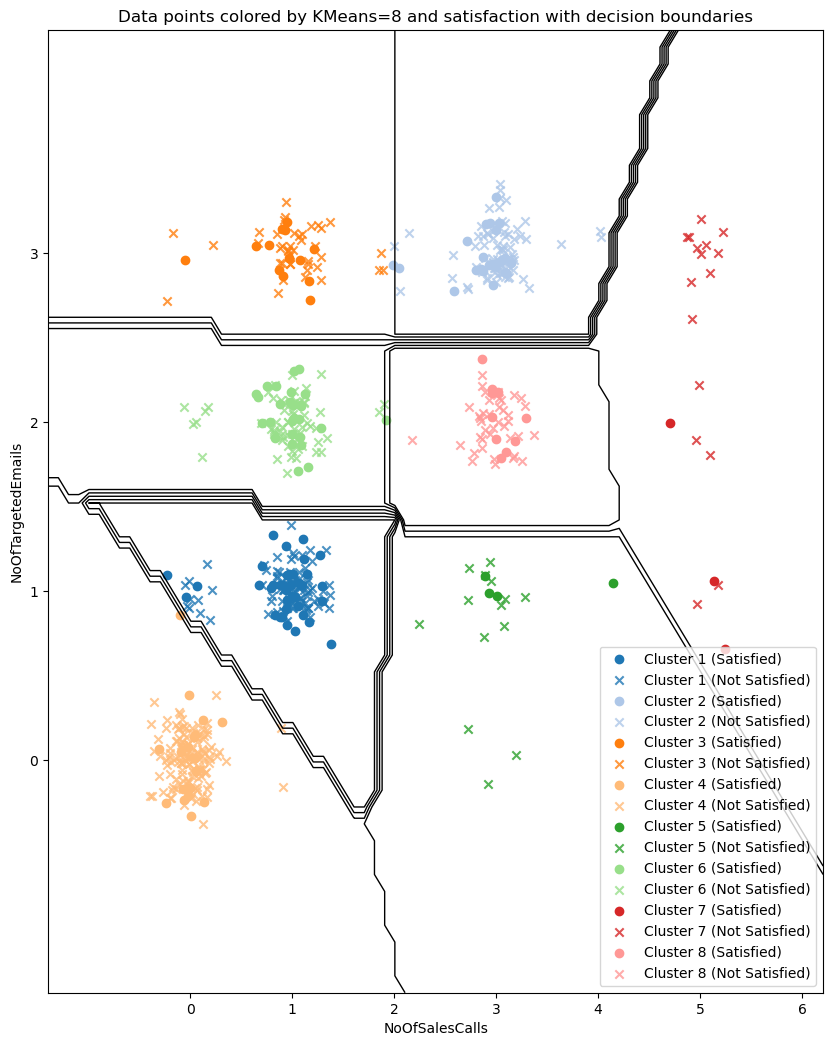
We then developed the same learning algorithms but this time looking at the relationship between NoOfSalesCalls and NoOfTargetedEmails where the points on the scatter plot represented CustomerSatisfaction\_YN. NoOfSalesCalls and NoOfTargetedEmails were then standardized to be on the same range.

To obtain the optimal K two charts were used, Elbow graph and a chart comparing accuracy on the y and K on the X.



*Fig. 7 Elbow graph and an Accuracy x K graph for optimal K.*

Based on the Accuracy x K chart it appears the most optimal K is K=8. This is good for building clusters as it is specific enough to not be considered over fit and avoids under fitting as accuracy for K=10 and on plateaus.



*Fig 8. K=8 cluster for customer satisfaction (Satisfied or Not Satisfied) with decision boundaries*

We then developed logistic regression; our predictors were NoOfSalesCalls and NoOfTargetedEmails and binary outcome CustomerSatisfaction\_YN. Lastly, we developed the decision tree using the same inputs as the logistic regression.

**5. Results and Conclusion**

When analyzing the relationship and similarities of the customers based on Sale or No Sale, the accuracy of K=4 placing new data into its respective cluster is .69 or 69%, rounded to 70%. This means the algorithm is able to successfully place new data based into the correct grouping and predict whether or not it will be a sale based on NoOfSalesCalls and NoOfTargetedEmails close to every 7 out of 10 times. That is a pretty good accuracy and can help the business identify where the new customer will fall and if there will be a sale or not. Some insights were learned about the clustering and customer relationships using K=4. In Cluster 4 of Fig 6 there were 20.75% sales, 79.25% no sales, mean sales calls: -0.01, mean targeted emails: 0.08. This cluster has the lowest amount of sales because the customers were rarely reached out too. On the contrary, in Fig 6, Cluster 3 has 51.91% sales, 48.09% no sales, mean sales calls: 1.22, mean targeted emails: 0.99. This cluster has the highest sales when the customer was reached out to through email and received a sales call only one of each. An important note for the relationship of customers was seen in Cluster 1 and 2, between those two clusters the higher sales came with a target email around 1. Cluster 2 had low sales with a high sales calls average meaning, the customers may get annoyed at the amount of calls or sales people aren't good.

When developing the logistic regression model its accuracy is rounded to 58% and the decision tree produces the same around 58%.

If the business owner wishes to move forward with analyzing relationships between customers based on sales made or not made, influenced by target emails and sales calls, we recommend using the KNN technique along with using K=4. To create better understanding of relationships, the business should look to recommend adding location of customer, product, and spend amount.

KNN Decision tree Logistic regression where not good identifiers of relationships based on sales and do not recommend to use going forward for the business.

|  | KNN | Log | Decision tree |
| --- | --- | --- | --- |
| Accuracy for sales | 70% | 58% | 58% |
| Accuracy for satisfaction | 76% | 77% | 77% |

*Fig 9. Accuracy results for each learning algorithm*

When analyzing the relationship and similarities of the customers based on Satisfaction or No Satisfaction, the accuracy of K=8 placing new data into its respective cluster is 0.76 or 76%, rounded to 80%. This means the algorithm is able to successfully place new data based into the correct grouping and predict whether or not a customer will be satisfied based on NoOfSalesCalls and NoOfTargetedEmails close to every 8 out of 10 times. That is a pretty good accuracy and can help the business identify where the new customer will fall and if there will be a sale or not. Some insights were learned about the clustering and customer relationships using K=8. In the clusters of Fig. 8, a lot of customers are not satisfied, but Clusters 1 and 6 had the highest satisfaction rate with around 1 sales call made. The one of the lowest satisfaction rates is CLuster 5 which had an average of 5 sales calls made. This business does not satisfy a lot of customers and highly recommend working on improving that for future growth.

When developing the logistic regression model its accuracy is rounded to 77% and the decision tree produces the same around 77%.

If the business wishes to see relationships between customers based on satisfaction, influenced by target emails and sales calls, they can choose any of the three as they produce the same accuracy. We recommend the KNN where K=8 as the visual is easy to follow and understand.

In conclusion, this research aimed to explore the effectiveness of different machine learning algorithms in predicting customer behavior in the context of sales and satisfaction. Through the use of KNN, decision tree, and logistic regression models, we were able to accurately predict both sales and satisfaction outcomes for customers based on their historical interactions with the business.

These findings have important implications for businesses looking to improve their understanding of customer behavior and enhance their sales and customer satisfaction outcomes. By leveraging machine learning algorithms, businesses can gain insights into customer patterns and behaviors that can inform targeted marketing and customer engagement strategies.

The code and supporting materials can be reviewed on my [GitHub repository](https://github.com/jthorme/SWEG_Datamining.git).