Faculty of Information Technology Engineering

Damascus University

Software Engineering Department

2021 - 2022

RESEARCH PAPER FOR CLASSIFYING COMPANY'S CUSTOMERS USING MACHINE LEARNING

- Professor

- Hala Al-Nemeh

- Prepared By

- Joseph Kalash (جوزيف كلش)
- Yazan Mohsen (يزن محسن)
- Cezar Terzian (سيزار ترزيان)

Content

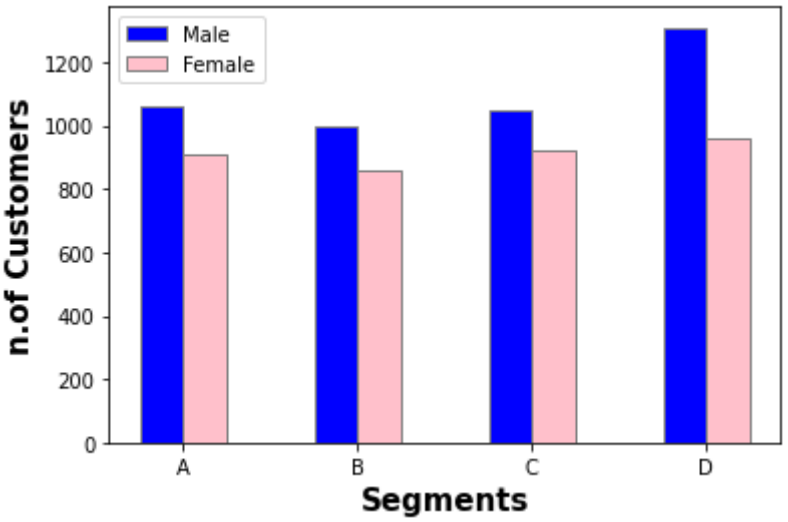
Study Objectives

The main objective of this study is to classify a company's customers into different segments **(A, B, C,** and **D)**, this will be achieved by predicting the segment of the new customer based on exploring and analyzing an existing dataset, then training a model to predict the appropriate segment using machine learning algorithms such as Decision Tree and Random Forest Classifier.

Data Exploration and Analysis

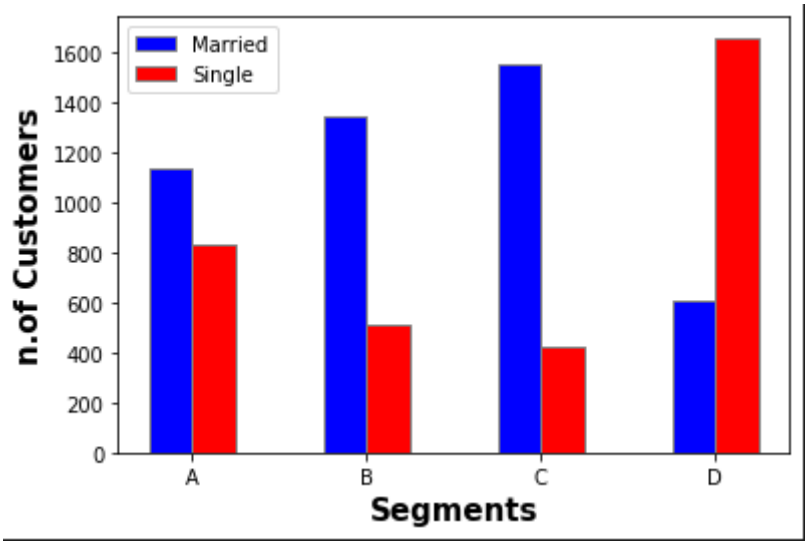
1) Gender Distribution over Segments

As a result of analyzing gender data of the customers, we found that the number of males is bigger than females for all segments, The difference gets bigger on segment **D.**



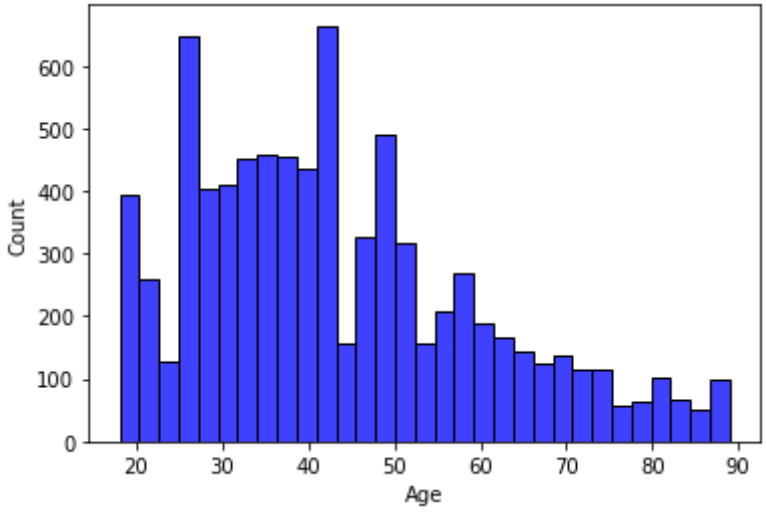
2) Marital Status Distribution over Segments

There is a clear difference between the number of married customers and the single ones. It's more likely for the customer to be married in segments **A, B,** and **C**. In contrast, the number of single customers in segment **D** is much more bigger than the married ones.



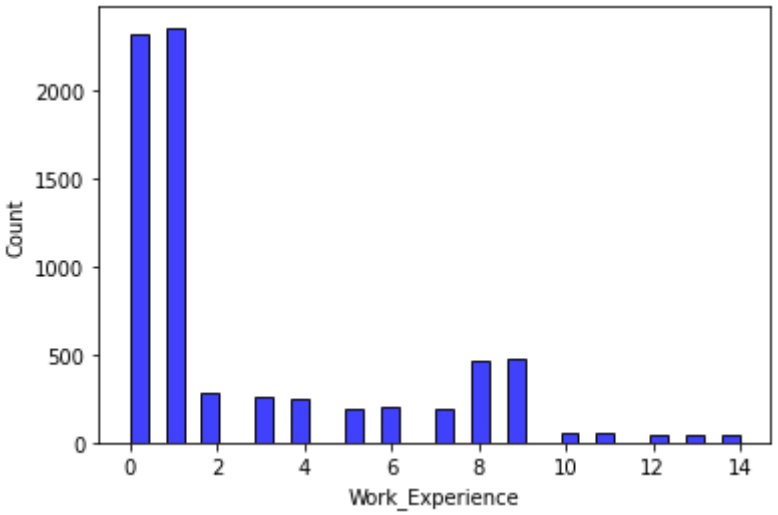
3) Age Histogram

We notice that the majority of customers' ages are between (**20** and **50**), but we have a gap around age ~25 and ~44.



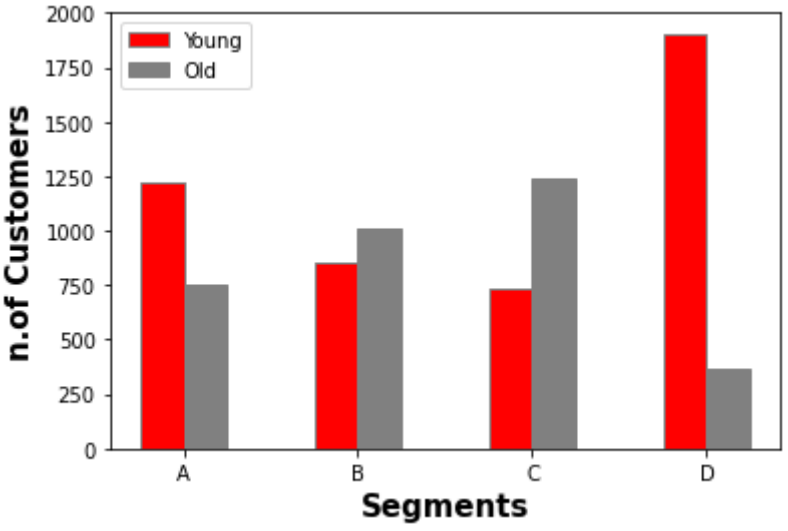
4) Work Experience Histogram

The majority of customers have at most **1** year of work experience.



5) Age Distribution over segments

Let's calculate the average age, then we will assume the following:
if customer age **>** (is greater than) average age ⇒ old customer
else ⇒ It's considered a young customer
The majority of customers are younger in segment **D.**
We have slightly more older customers in segments **B** and **C.**
Segment **A** has more younger customers.



Used Methodology

First, We should apply pre-processing techniques to the dataset, Those techniques are:

1) Taking care of missing data, Replace the NaN (Not a Number) values with meaningful values that is related to the data, there are few replacement strategies such as (mean, median, most_frequent, constant)
   We used the most_frequent strategy to cover the numeric and non-numeric values.
2) Encoding Categorical Data (Label Encoder or One Hot Encoder)
3) Feature Scaling (Data Normalization), This technique is used to prevent the algorithm from being biased toward the feature which has values higher in magnitude.

Then The data will be splitted into two sections (Train and Test data)
The next step is to build a model and train it with the Train dataset.
Two models will be built using two different algorithms (**Decision Tree Classifier** and **Random Forest Classifier**)

After training we should evaluate the results and measure the accuracy of the model using the test dataset.

Experimental Study

First we will import the dataset from an excel (csv) file, Then we will separate the features (as an input) and the segmentation (as an output). The next step is to apply the pre-processing techniques that are mentioned above.

1) Remove empty / NaN values and replace them with meaningful data
2) Encoding Categorical Data
   We will apply the Label Encoding technique on the following features: (**Gender, Ever_Married, Graduated** and **Spending_Score**)
   Apply Label Encoding on the **Segmentation**
   Apply the One Hot Encoding technique on the **Profession** feature
3) Apply Feature Scaling on the following features (**Age** and **Work_Experience**)

We should split the dataset into training set (70% of the dataset) and test set (30% of the dataset)

Now the data is ready to be processed and the model is ready to be trained.
We will train two different models using (**Decision Tree Classifier** and **Random Forest Classifier**)

For each model (algorithm) we will do the following:
When the training is over, we should **Predict the Test set results** then make a confusion matrix, and finally calculate the **Accuracy Score** for the model.

If the accuracy is acceptable we should rely on the model for predicting the segmentation of the new customers.

Importing new customers dataset:
We should do the pre-processing techniques that are mentioned above to the new dataset, Then use the above models to predict the results.

Results comparison and discussion

Decision tree classifier

After we used a decision tree classifier algorithm to classify our dataset and then predict segmentations for customers, we found that the accuracy is 53% and 52% for features_train and features_test respectively.
As we represent (A, B, C, D) as (0, 1, 2, 3) we found that D -or 3 in this case- has the biggest results.

features train classification data report

	0	1	2	3	accuracy	macro avg	weighted avg
precision	0.43	0.44	0.55	0.66	0.53	0.52	0.53
recall	0.51	0.37	0.48	0.72	0.53	0.52	0.53
f1-score	0.47	0.40	0.52	0.69	0.53	0.52	0.52
support	1378.00	1314.00	1363.00	1592.00	0.53	5647.00	5647.00

features train data report

features test classification data report

	0	1	2	3	accuracy	macro avg	weighted avg
precision	0.45	0.37	0.58	0.64	0.52	0.51	0.52
recall	0.53	0.32	0.49	0.70	0.52	0.51	0.52
f1-score	0.49	0.34	0.53	0.67	0.52	0.51	0.52
support	594.00	544.00	607.00	676.00	0.52	2421.00	2421.00

features test data report

Random forest classifier

After we used a random forest classifier algorithm to classify our dataset and then predict segmentations for customers, we found that the accuracy is 57% and 54% for features_train and features_test respectively.
As we represent (A, B, C, D) as (0, 1, 2, 3) we found that D -or 3 in this case- has the biggest results.

features train classification data report

	0	1	2	3	accuracy	macro avg	weighted avg
precision	0.50	0.49	0.57	0.67	0.57	0.56	0.56
recall	0.52	0.39	0.59	0.74	0.57	0.56	0.57
f1-score	0.51	0.44	0.58	0.71	0.57	0.56	0.56
support	1378.00	1314.00	1363.00	1592.00	0.57	5647.00	5647.00

features train data report

features test classification data report

	0	1	2	3	accuracy	macro avg	weighted avg
precision	0.46	0.41	0.58	0.64	0.54	0.52	0.53
recall	0.49	0.31	0.60	0.69	0.54	0.52	0.54
f1-score	0.47	0.35	0.59	0.66	0.54	0.52	0.53
support	594.00	544.00	607.00	676.00	0.54	2421.00	2421.00

features test data report

Comparison between algorithms

	Train (%)	Test (%)
Decision tree classifier	53	52
Random forest classifier	57	54

From the table above we can notice that the random forest classifier algorithm increases the accuracy by 4% for train data and 2% for test data compared to decision tree classifier algorithm, this increase in accuracy because random forest depends on many numbers of decision trees. so that, the result come from all of them.