Data Glacier

Your Deep Learning Partner

Bank Marketing (Campaign)

Data Girl

Deadline Date: 30 Dec 2022

LISUM 14 20 Sep – 30 Dec 2022

Fatimah Asiri

Agenda

1- Business Understanding.
2- Data understanding.
3- Exploratory Data Analysis.
4- Data Preparation.
5- Model Building (Logistic Regression, Random Forest, Decision Tree)
6- Model Selection.
7- Converting ML metrics into Business metrics and explaining results to the business.

Data Glacier

Your Deep Learning Partner

1- Business Understanding

Business Understanding

- Bank wants to use the ML model to shortlist customer whose chances of buying the product is more so that their marketing channels marketing SMS/email marketing, etc. can focus only on those customers whose chances of buying the product is more.

- This will save resources and time (which is directly involved in the cost (of resource billing).

- Develop a model with Duration and without duration features and report the performance of the model.

- The duration feature is not recommended as this will be difficult to explain the result to the business and also it will be difficult for businesses to campaign based on duration.

Data Glacier

Your Deep Learning Partner

2- Data understanding.

Data understanding

- Data Set Information:

  - The data is related to direct marketing campaigns of a Portuguese banking institution.

  - The marketing campaigns were based on phone calls. Often, more than one contact with the same client was required, in order to access if the product (bank term deposit) would be ('yes') or not ('no') subscribed.

  - The classification goal is to predict if the client will subscribe (yes/no) to a term deposit (variable y).

Attribute Information:

Input variables:

bank client data:

1 - age (numeric)
2– job: type of job (categorical: ‘admin.’, ‘blue-collar’, ‘entrepreneur’, ‘housemaid’, ’management’, ’retired’, ’self-employed’, 'services’, 'student’, 'technician’, 'unemployed’, 'unknown')
3- marital: marital status (categorical: 'divorced’, 'married’, 'single’, 'unknown'; note: 'divorced' means divorced or widowed)
4- education (categorical: 'basic.4y’, 'basic.6y’, 'basic.9y’, 'high. School’, ‘illiterate’, ‘professional. Course’, ‘university. Degree’, ‘unknown’)
5 – default: has credit in default? (categorical: ‘no’, ‘yes’, ‘unknown’)
6– housing: has a housing loan? (categorical: ‘no’, ‘yes’, ‘unknown’)
7 – loan: has a personal loan? (categorical: 'no’, 'yes’, 'unknown')

Attribute Information: (con…)

8- contact: contact communication type (categorical: 'cellular’, 'telephone')related to the last contact of the current campaign

9 - month: last contact month of the year (categorical: ‘Jan, ‘Feb, 'mar', ..., ‘Nov, ‘Dec')

10- duration: last contact duration, in seconds (numeric). Important note: this attribute highly affects the output target (e.g., if duration=0 then y='no'). Yet, the duration is not known before a call is performed. Also, after the end of the call y is obviously known. Thus, this input should only be included for benchmark purposes and should be discarded if the intention is to have a realistic predictive model.

11 - campaign: number of contacts performed during this campaign and for this client (numeric, includes the last contact)

12- pdays: number of days that passed by after the client was last contacted from a previous campaign (numeric; 999 means the client was not previously contacted)

13 - previous: number of contacts performed before this campaign and for this client (numeric)

14 - poutcome: outcome of the previous marketing campaign (categorical: 'failure’, Unknown’, ‘success’)

15 – y: The classification goal is to predict if the client will subscribe (yes/no) to a term deposit (variable y).

Basic Information about Data:

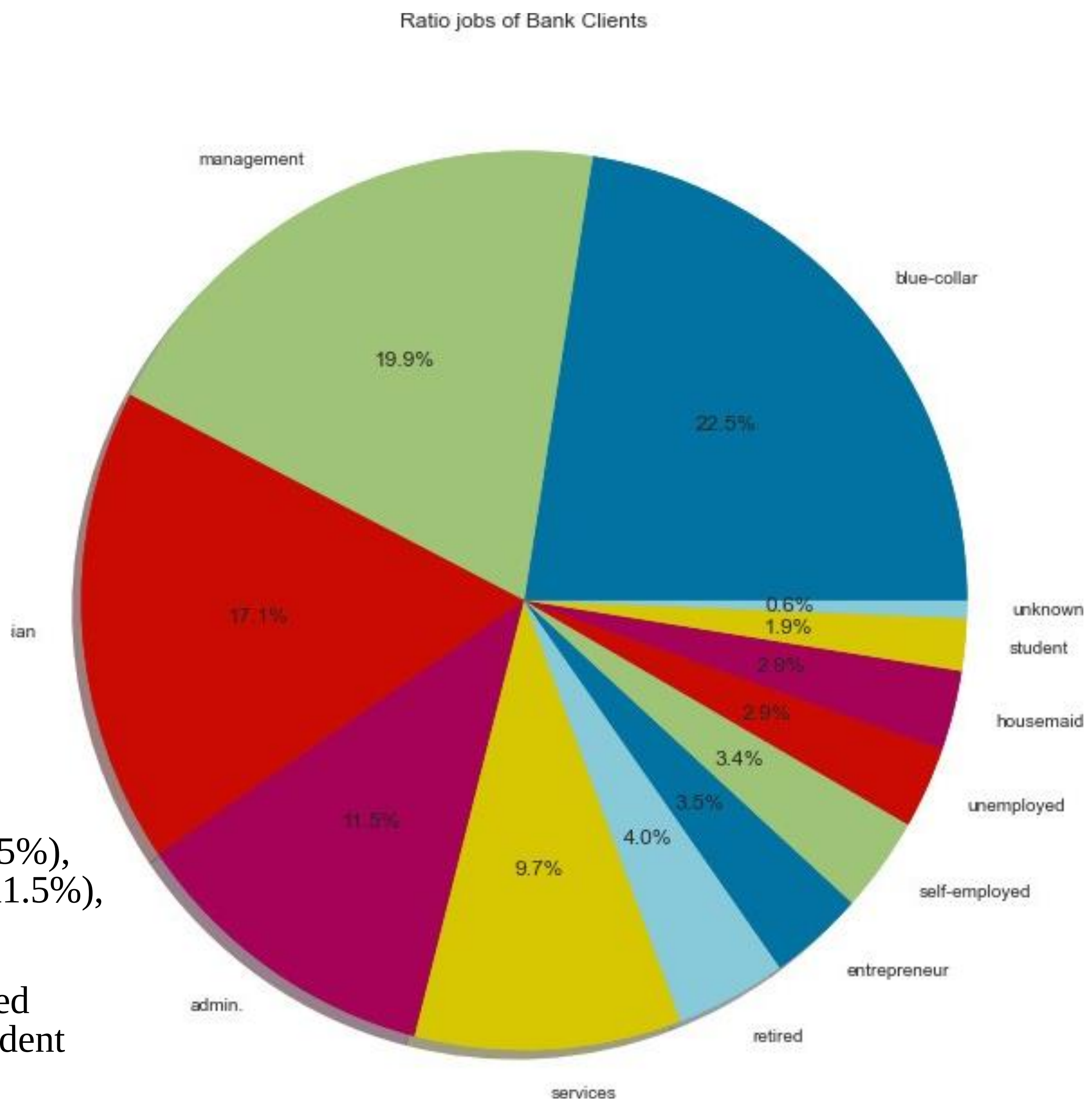
- The Data has 17 columns.
- The Shape of Dataset: (49732, 17).
- The data types: int64(7), object(10)
- Memory usage: 6.5+ MB
- No Missing Data
- The Duplicate rows: (4521 rows)
- Handling with outliers in columns by removing (Age, Balance, Duration, campaign, pdays, previous)

3- Exploratory Data Analysis (EDA)

Data Glacier

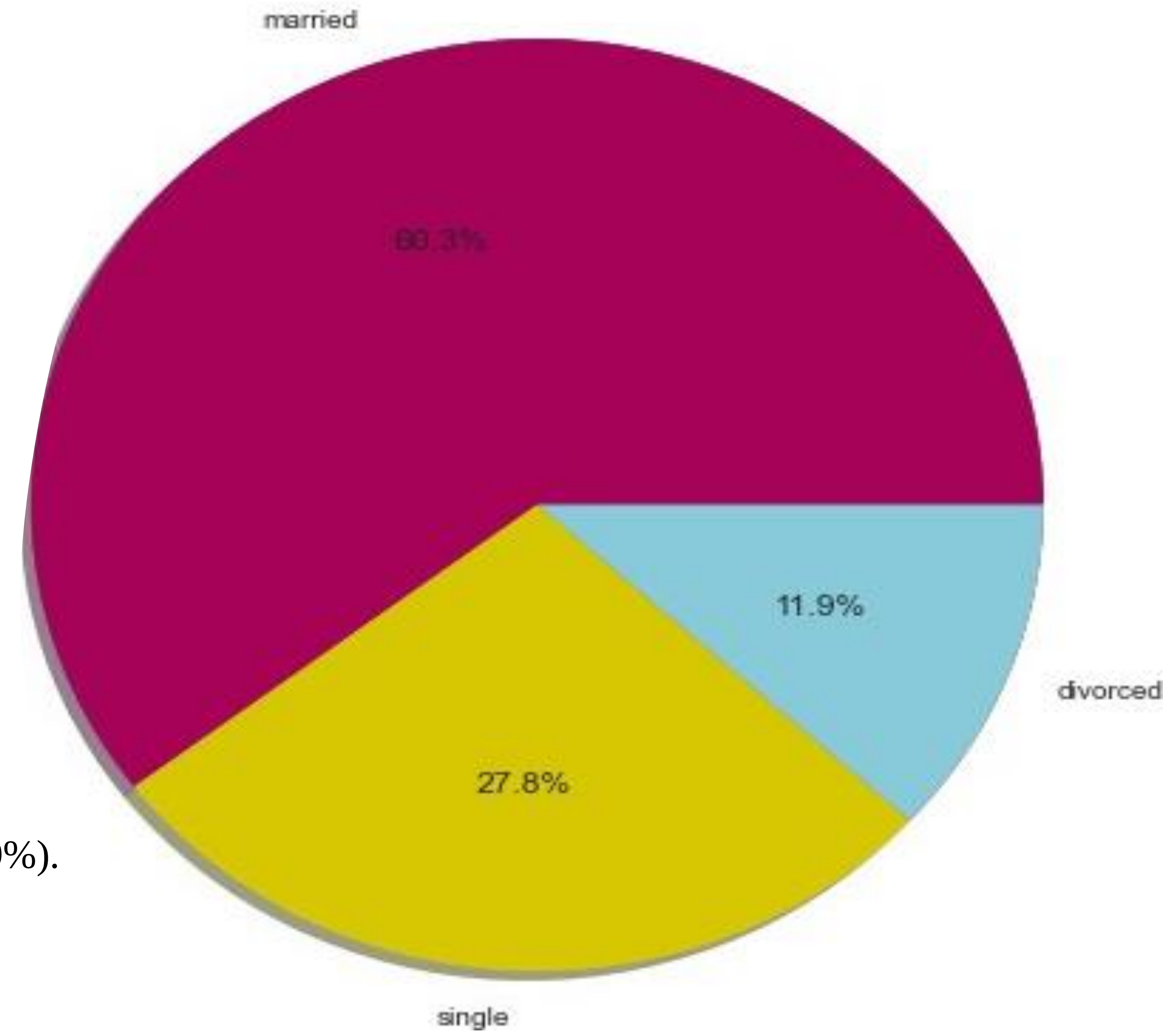
Your Deep Learning Partner

EDA - What's the jobs for bank clients?



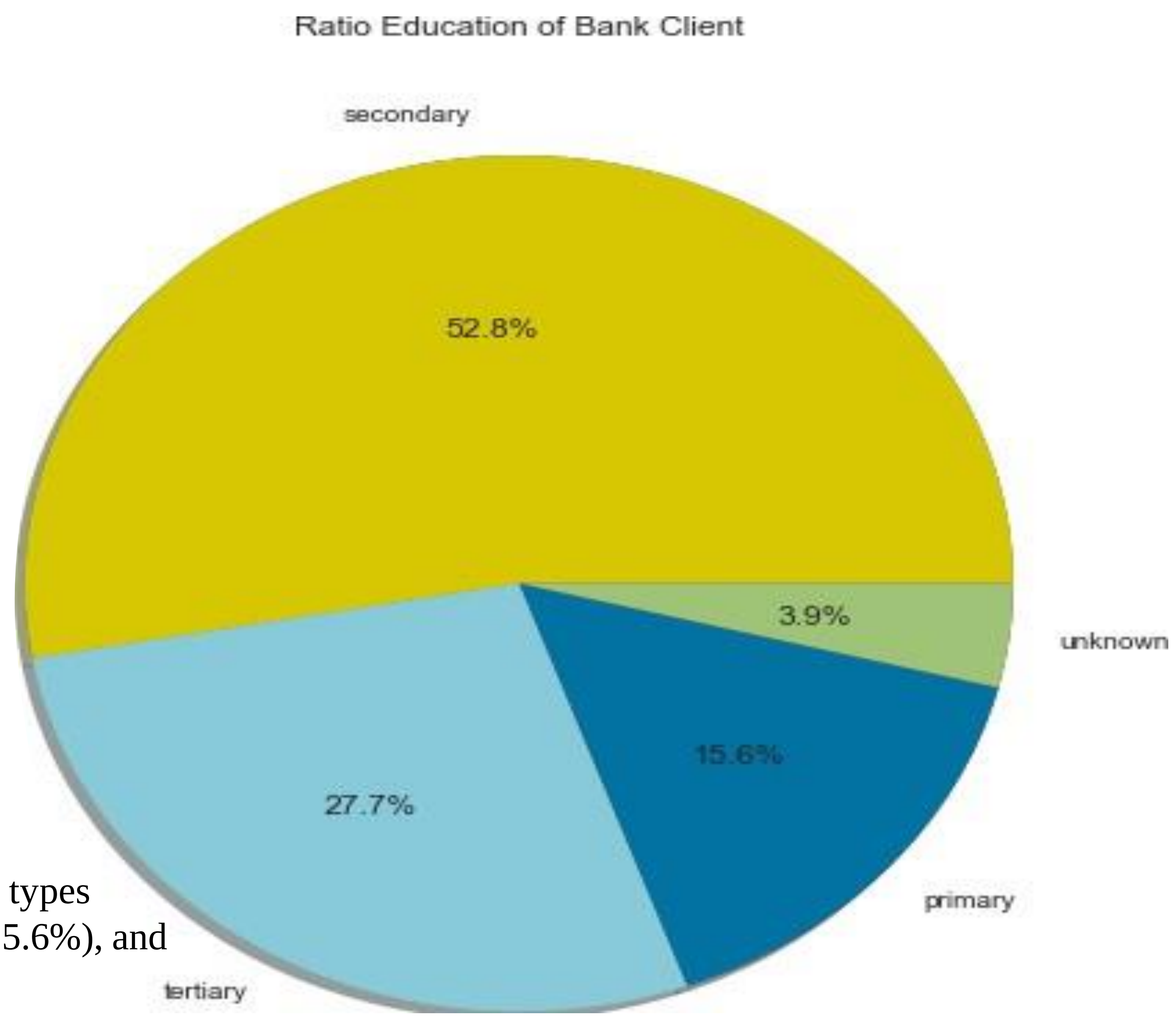
- The top 5 jobs for the client's bank: blue-collar (22.5%), management (19.9%), technician (17.1%), admin (11.5%), and services (9.7%)
- The lowest 5 jobs for the client’s bank: self-employed (3.4%), unemployed (2.9%), housemaid (2.9%), student (1.9%), and unknown (0.6%).

EDA - What is the marital status of bank clients?



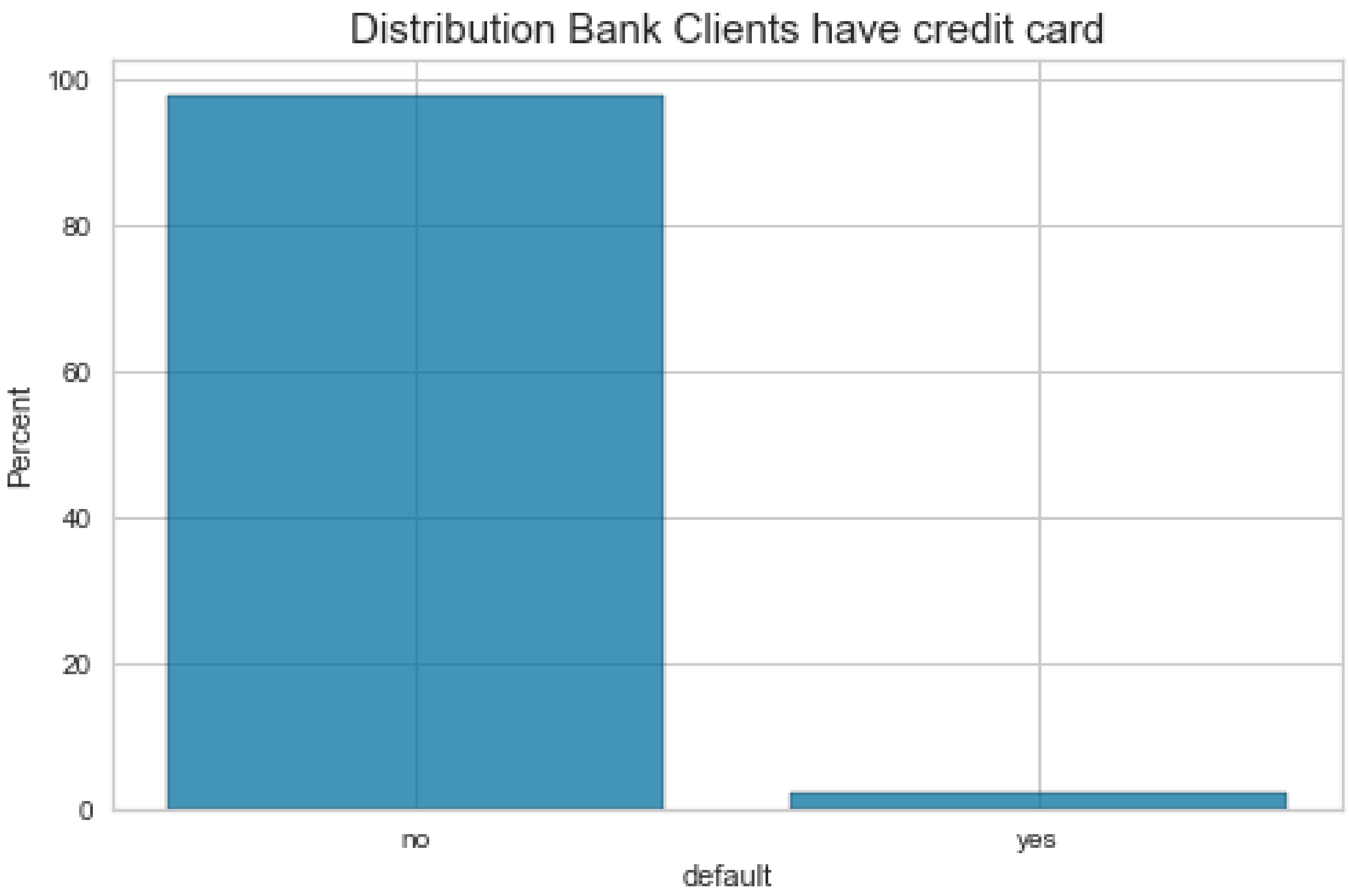
- The ratio marital of bank clients divide to 3 types married(60.3%), single(27.8%), and divorced(11.9%).

EDA – What is the education status of bank clients?



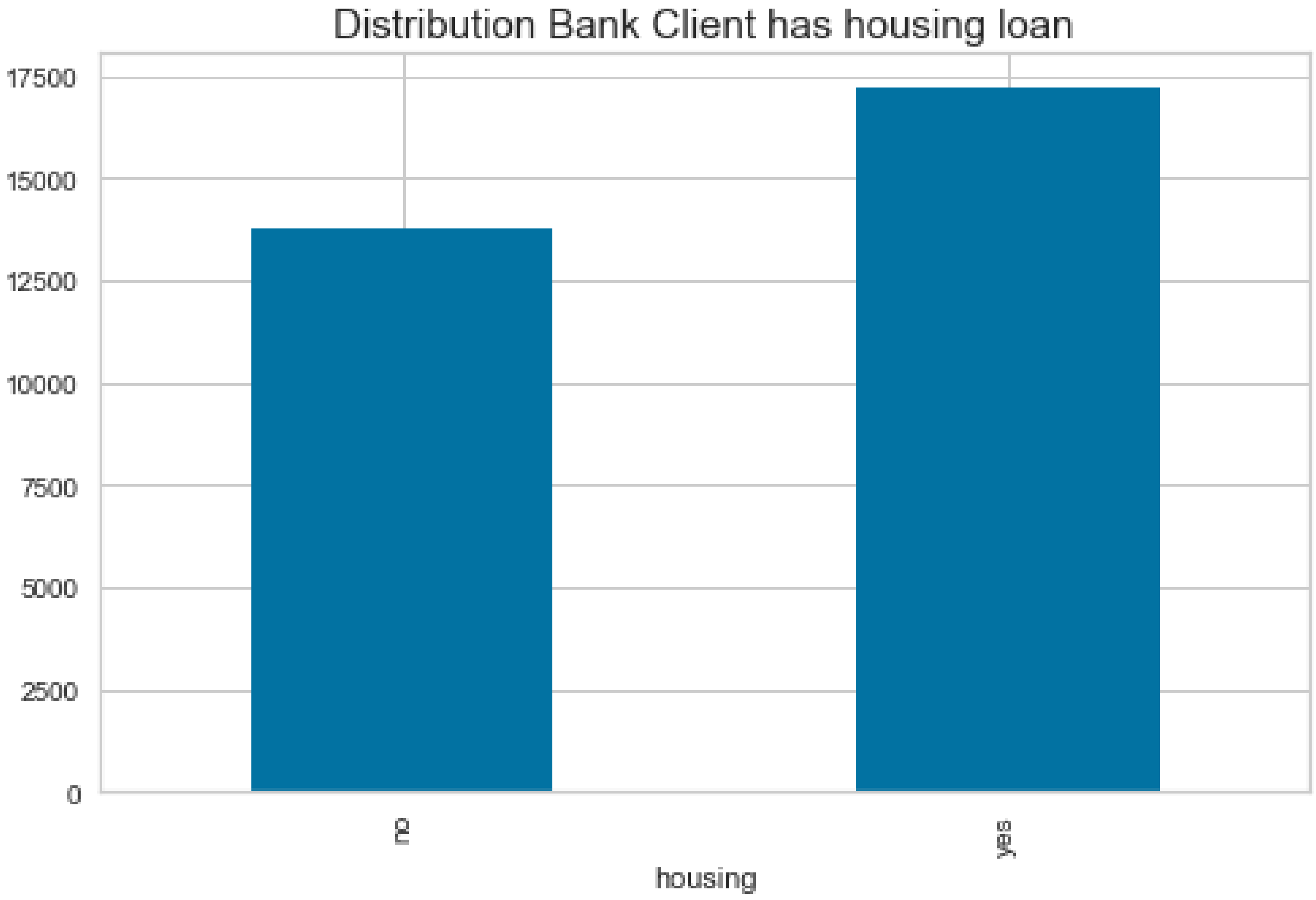
The ratio education of bank clients divide to 4 types Secondary(52.8%), tertiary(27.7%), married(15.6%), and unknown(3.9%).

EDA - Are bank clients have credit cards?



- The ratio of bank clients who have a credit card (98% no) and (2% yes).

EDA - Are bank clients have housing loans?

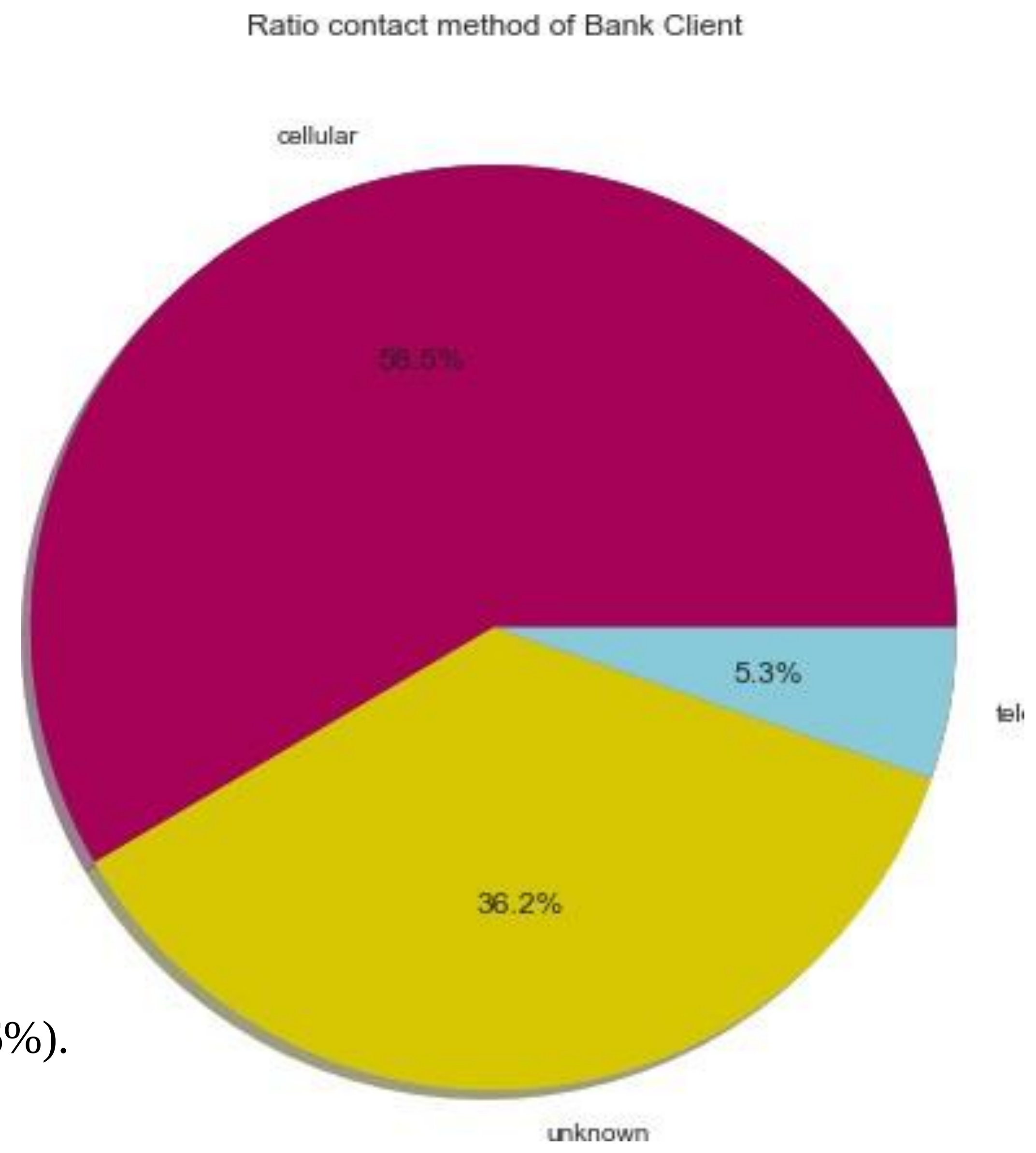


- The number of clients that have housing loans (55.51% yes) and (44.49% no).

EDA - Are bank clients have personal loans?

- The number of clients that have personal loans (17.50 % yes) and (82.50% no).

EDA - what's the method contact with bank clients?



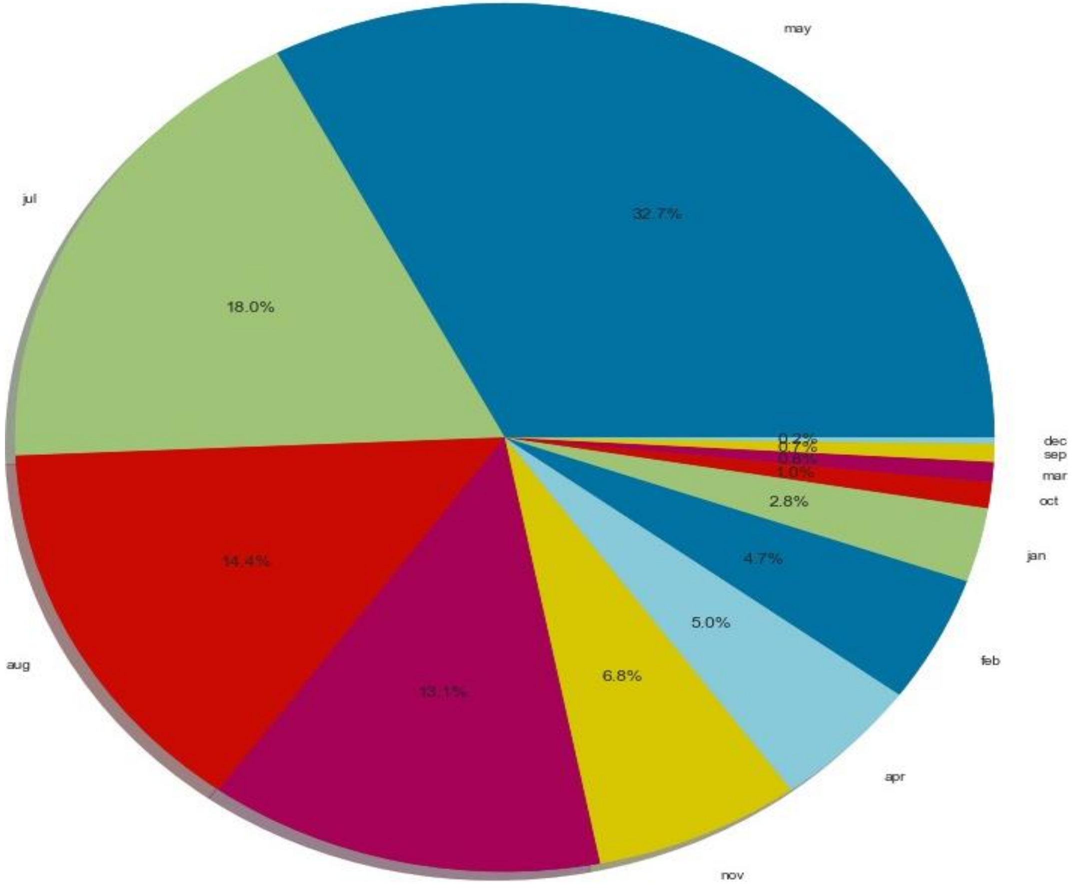
- The ratio contact method of Bank Client 3 types cellular(58.5%), unknown(36.2%), and telephone(15.6%).

EDA - what's the last contact month of the year of bank clients?

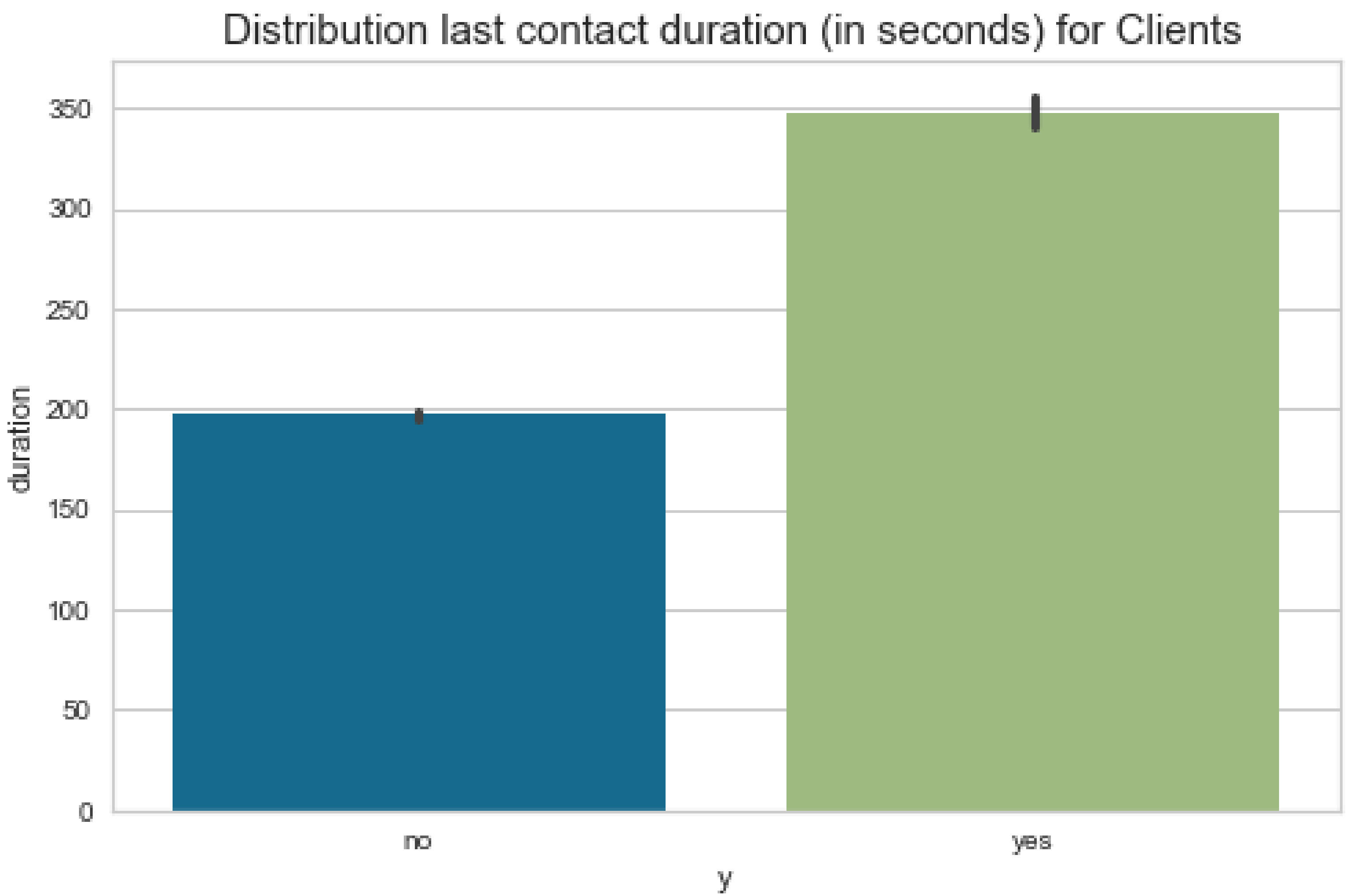
- The ratio last contact month of the year of Bank Clients.

- The top 5 months active Ratio last contact month of the year of Bank Clients: may (32.7%), Jul (18.0%), Aug
- (14.4%), Jun (13.1%), and Nov (6.8%)

- The lowest 5 months active Ratio last contact month of the year of Bank Clients: Jan (2.8%), Oct (1.0%), Mar (0.8%), Sep (0.7%), and Dec (0.2%).

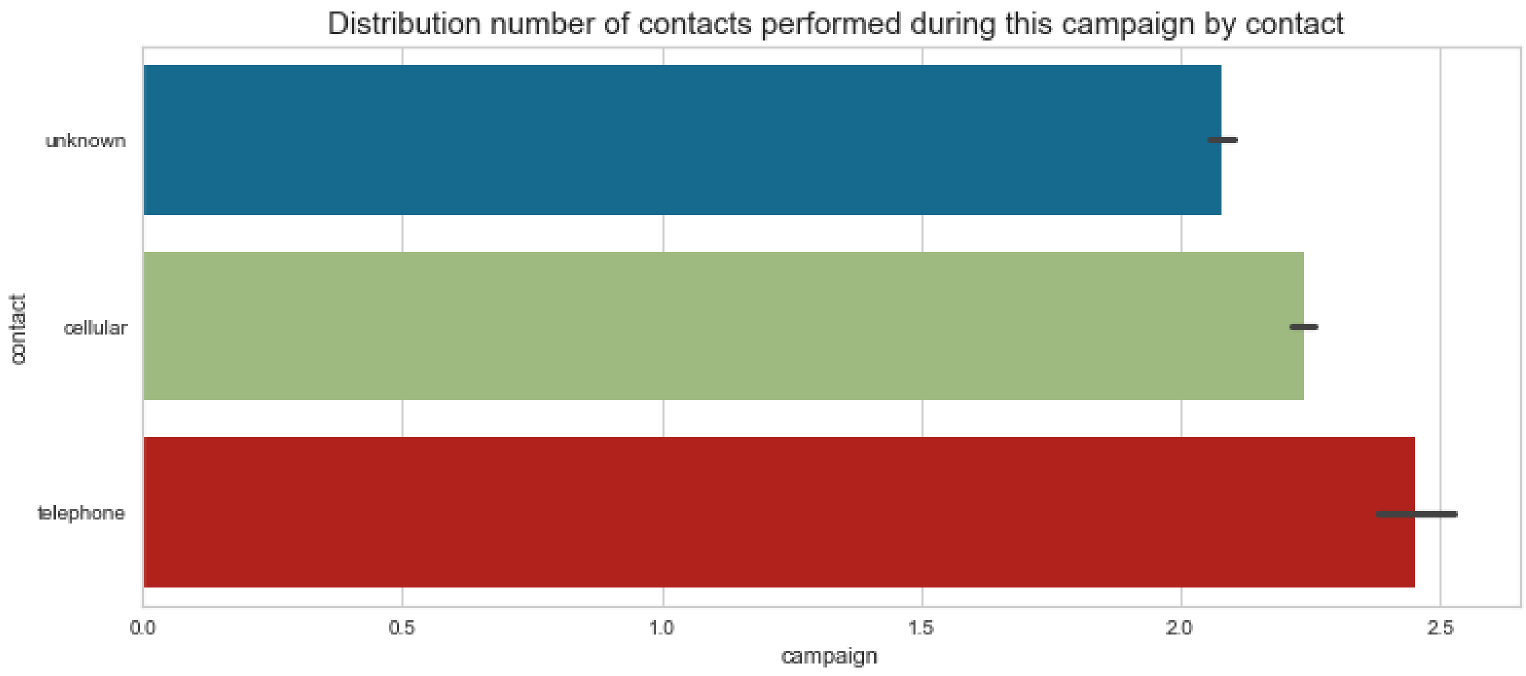


EDA - Do clients subscribe to a term deposit based on the last contact duration (in seconds)?



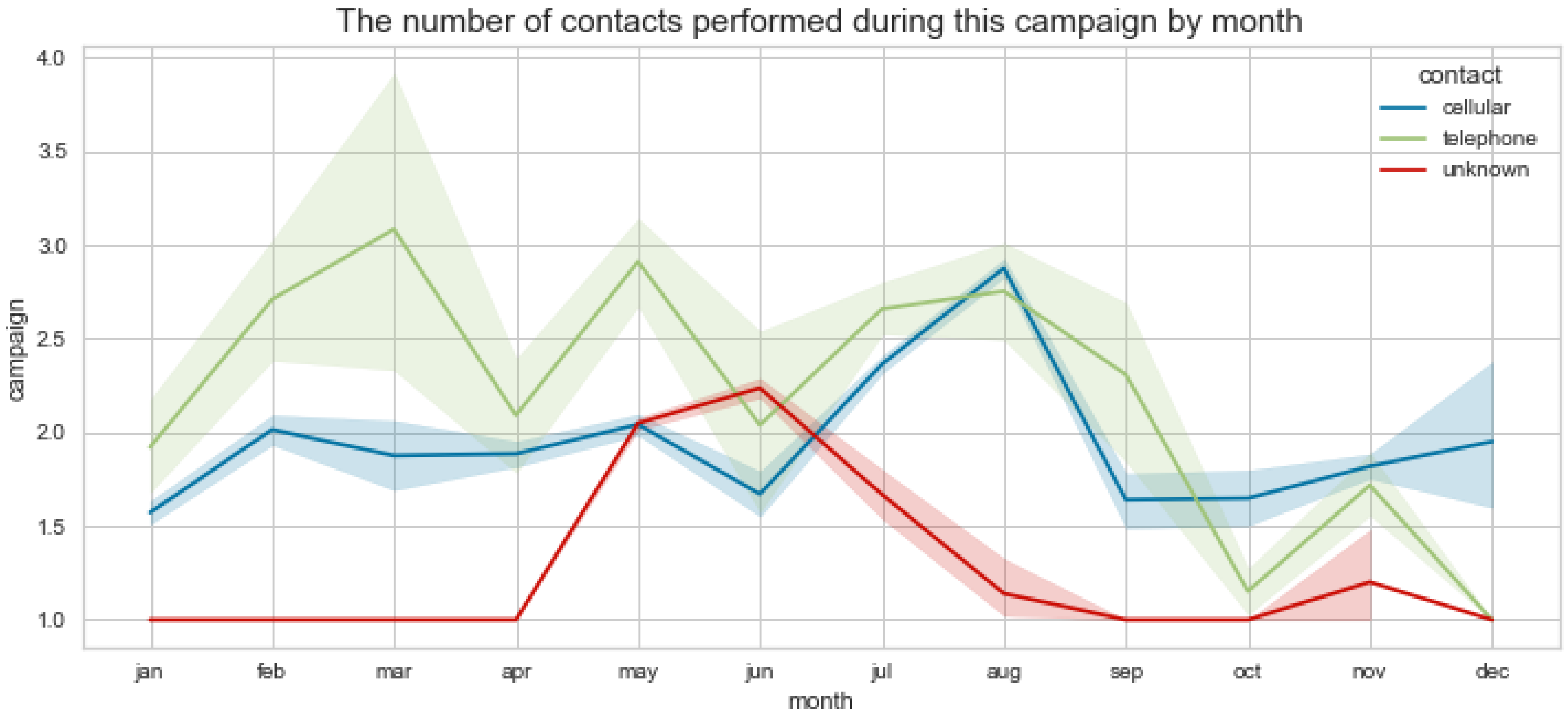
- The number of Clients who accept the campaign in the last contact duration (in seconds) is higher than the rejected.

EDA - what's the method of contact performed during this campaign by contacting?



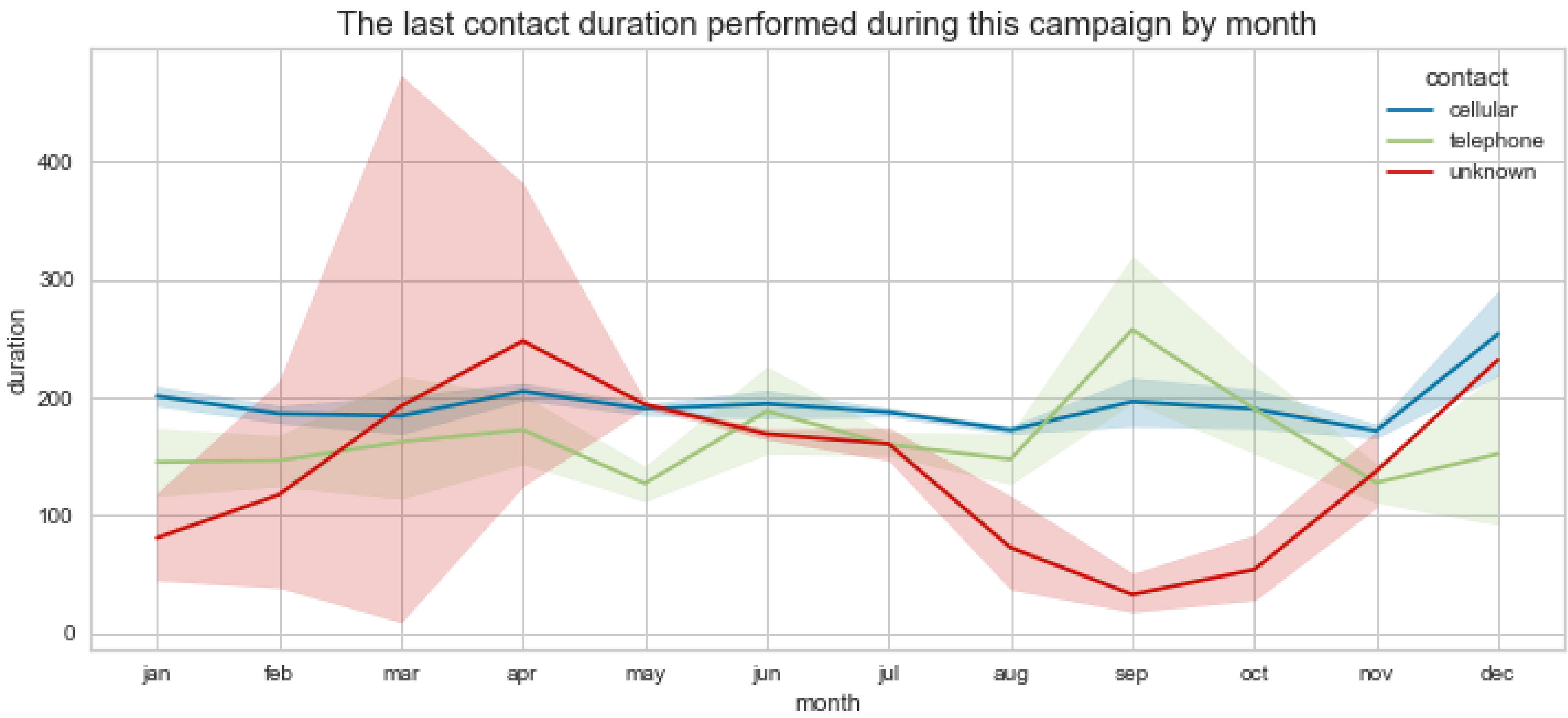
- The number of contacts performed during this campaign was sorted from top to down telephone, cellular and unknown.

EDA - what's the highest and lowest month number of a contact in the campaign?



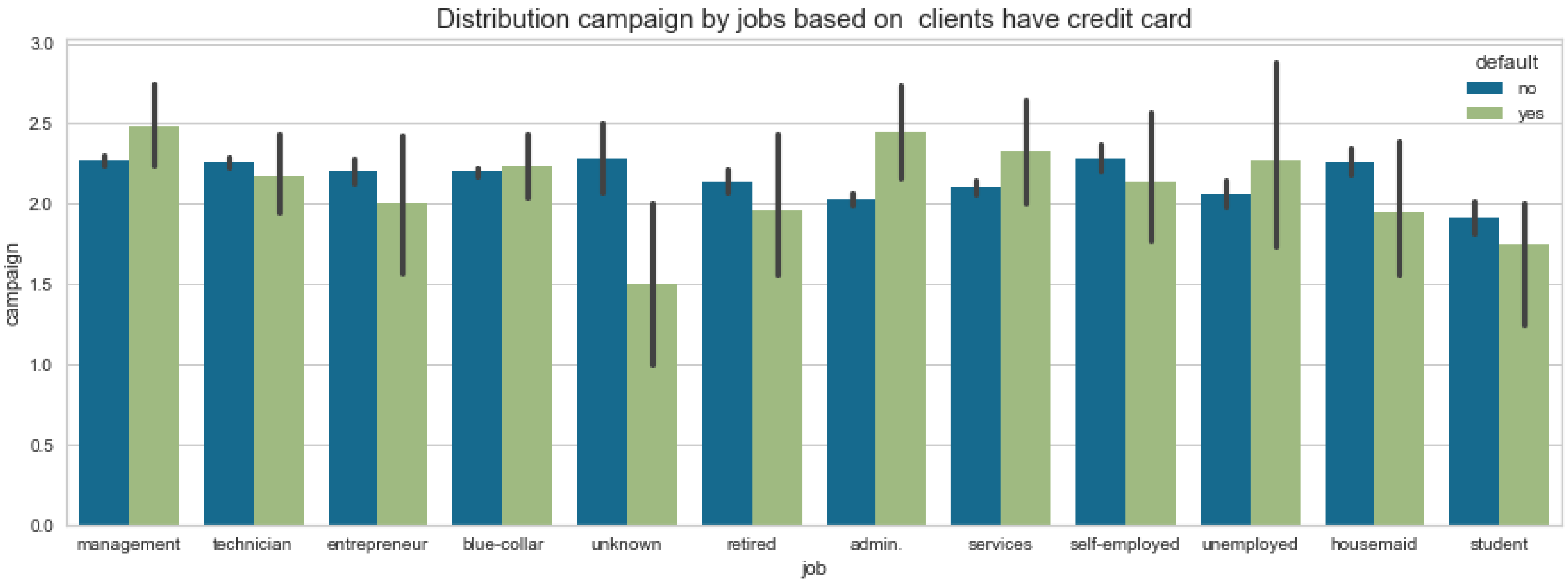
- The number of contacts performed during this campaign by month sorted from top to down telephone, cellular and unknown.

EDA - what's the last contact duration performed during this campaign by month?



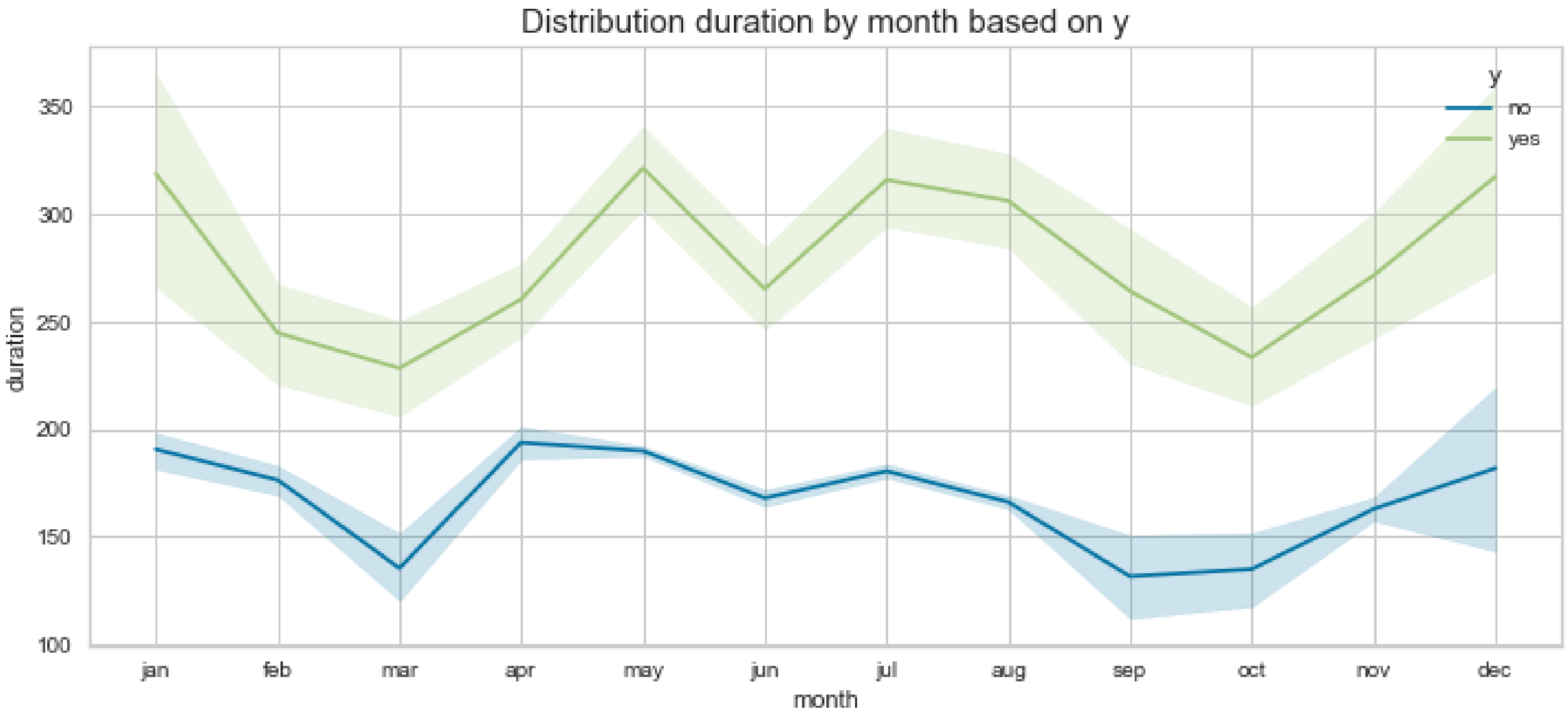
- The clients that have credit cards more than they not based on jobs.

EDA - Distribution campaign by jobs based on clients have credit card



- The clients that have credit cards more than they not based on jobs.

EDA - Distribution duration by month based on y



- The distribution duration by month based on y who accept is higher than rejected.

4- Data Preparation.

Data Preparation

1- Check on missing Data.

2- Label encoding for columns (default, housing, loan, month, poutcome, y).

3- One hot encoding for columns (job, marital, education, contact).

4- PCA - Principal component analysis.

5- Model Building (Logistic Regression, Random Forest, Decision Tree, and Naive Bayes)

Split Data into X1 and y1

Split Data into X1 and y1

```
# split data
X1 = df.drop(['day','month','campaign','pdays', 'previous' , 'duration', 'poutcome', 'result'],
             axis=1)
y1= df['result']

X_train, X_test, y_train, y_test = train_test_split(X1, y1, test_size=0.20, random_state=42)
```

Logistic Regression

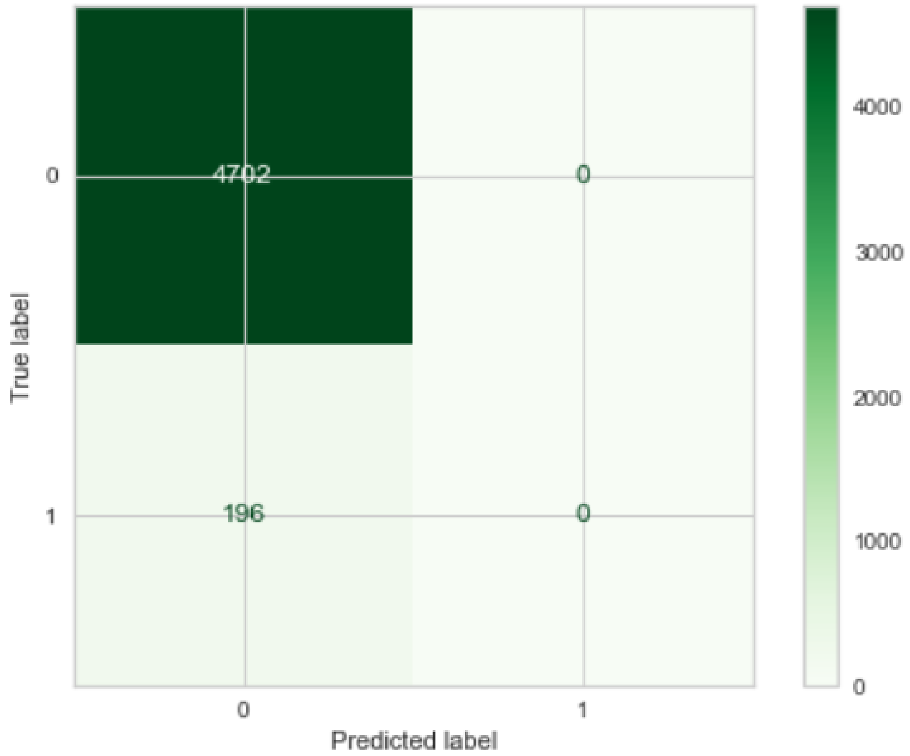
```
log_regression = LogisticRegression(random_state = 42)

log_regression.fit(X_train, y_train)
```

Count values that Actual_Result == Predict_Result

```
(test_df['Actual_Result'] == test_df['Predict_Result']).value_counts()
```

```
True     4702
False     196
dtype: int64
```

Logistic Regression



Calculate accuracy, precision, and recall

```
accuracy = accuracy_score(y_test, y_pred)
print('Accuracy: %f' % accuracy)
# precision tp / (tp + fp)
precision = 644/(405+644)
print('Precision: %f' % precision)
# recall: tp / (tp + fn)
recall = 644 /( 1338 + 644 )
print('Recall: %f' % recall)
```

```
Accuracy: 0.959984
Precision: 0.613918
Recall: 0.324924
```

Random Forest Classifier

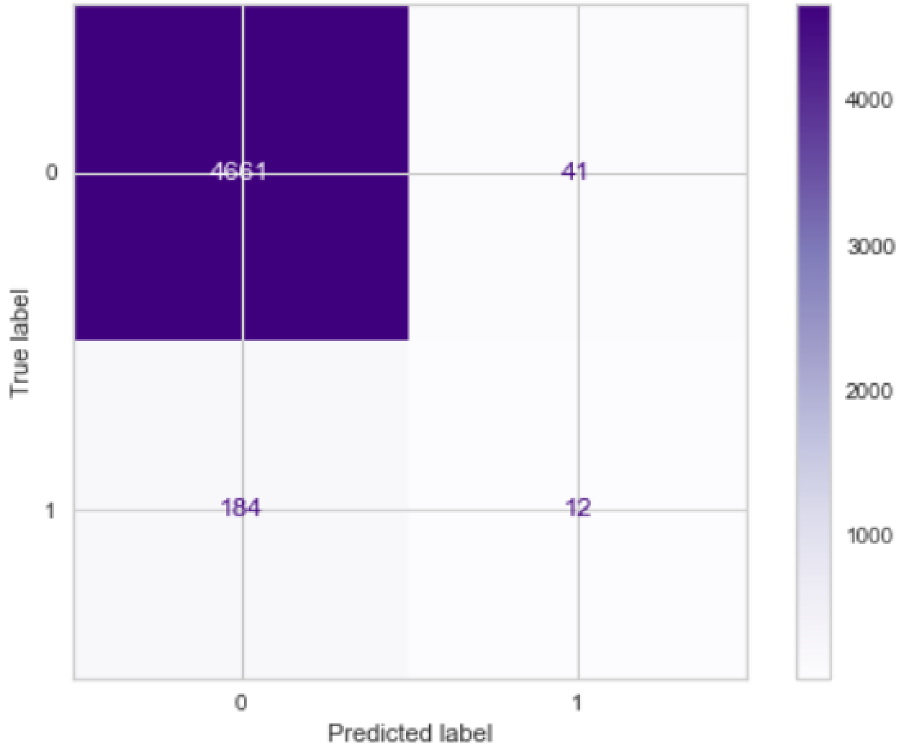
```
class_forest = RandomForestClassifier(criterion='gini',
                                      n_estimators=5,
                                      random_state=1,
                                      n_jobs=2)
```

Count values that Actual_Result == Predict_Result

```
(test_dfr['Actual_Result'] == test_dfr['Predict_Result']).value_counts()
```

```
True     4678
False     220
dtype: int64
```

Random Forest Classifier



Calculate accuracy, precision, and recall

```
accuracy = accuracy_score(y_test, y_predforest)
print('Accuracy: %f' % accuracy)
# precision tp / (tp + fp)
precision = 644/(405+644)
print('Precision: %f' % precision)
# recall: tp / (tp + fn)
recall = 644 /( 1338 + 644 )
print('Recall: %f' % recall)
```

```
Accuracy: 0.955084
Precision: 0.613918
Recall: 0.324924
```

Decision Tree Classifier

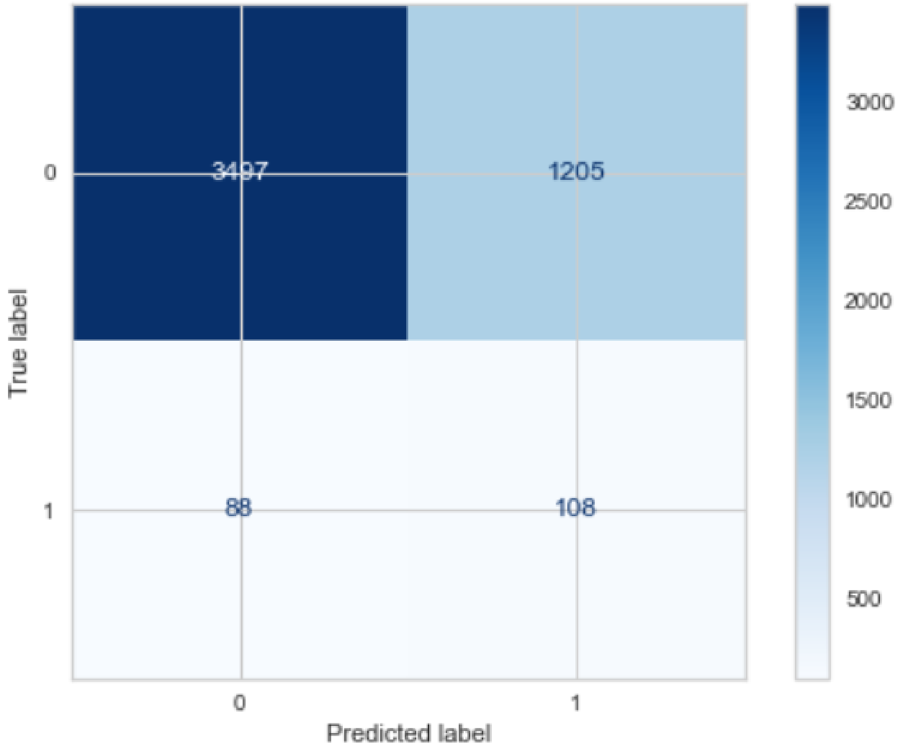
```
# Create Decision Tree classifer object
Dec_Tree = DecisionTreeClassifier()
```

Count values that Actual_Result == Predict_Result

```
(test_dftree['Actual_Result'] == test_dftree['Predict_Result']).value_counts()
```

```
True     4600
False     298
dtype: int64
```

Decision Tree Classifier



Calculate accuracy, precision, and recall

```
accuracy = accuracy_score(y_test, y_predtree)
print('Accuracy: %f' % accuracy)
# precision tp / (tp + fp)
precision = 644/(405+644)
print('Precision: %f' % precision)
# recall: tp / (tp + fn)
recall = 644 /( 1338 + 644 )
print('Recall: %f' % recall)
```

```
Accuracy: 0.939159
Precision: 0.613918
Recall: 0.324924
```

Naive Bayes

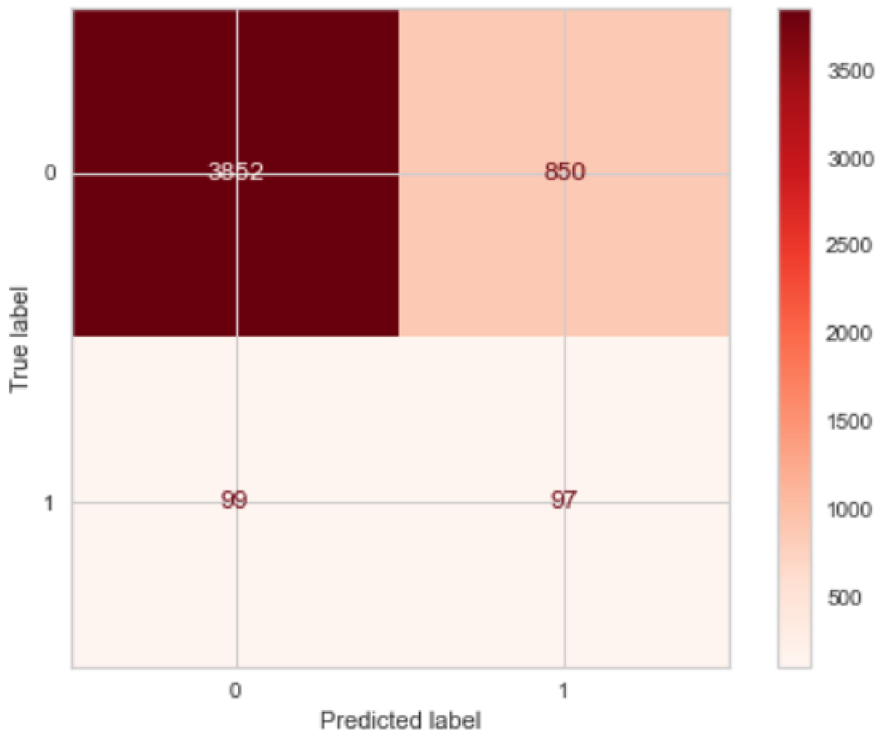
```
# Create Naive Bayes object

model = BernoulliNB().fit(X_train, y_train)
```

Count values that Actual_Result == Predict_Result

```
(test_dftree['Actual_Result'] == test_dftree['Predict_Result']).value_counts()
```

```
True     4619
False     279
dtype: int64
```

Naive Bayes



Calculate accuracy, precision, and recall

```
accuracy = accuracy_score(y_test, predicted_signal)
print('Accuracy: %f' % accuracy)
# precision tp / (tp + fp)
precision = 644/(405+644)
print('Precision: %f' % precision)
# recall: tp / (tp + fn)
recall = 644 /( 1338 + 644 )
print('Recall: %f' % recall)
```

```
Accuracy: 0.943038
Precision: 0.613918
Recall: 0.324924
```

Data Glacier

Your Deep Learning Partner

6- Model Selection.

Comparing Models by Accuracy, Precision, Recall

Tabel of Comparing between Models

Score	Logistic Regression	Random Forest Classifier	Decision Tree	Naive Bayes
Accuracy	0.959	0.955	0.938	0.943
Precision	0.61	0.61	0.61	0.61
Recall	0.33	0.33	0.33	0.33

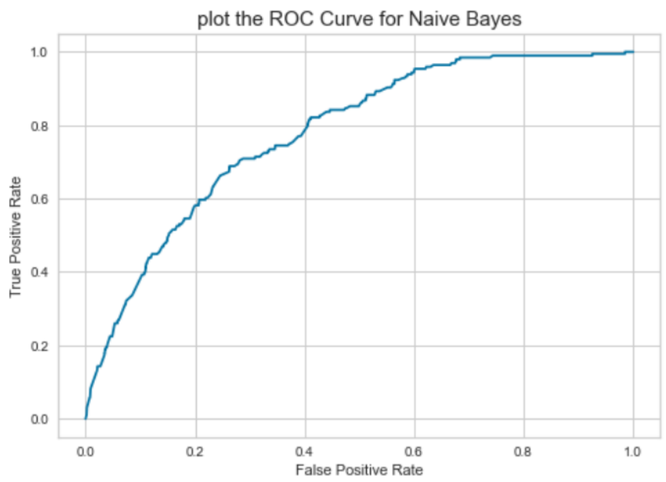
So, The best model is Logistic Regression based on Accuracy, Precision, Recall

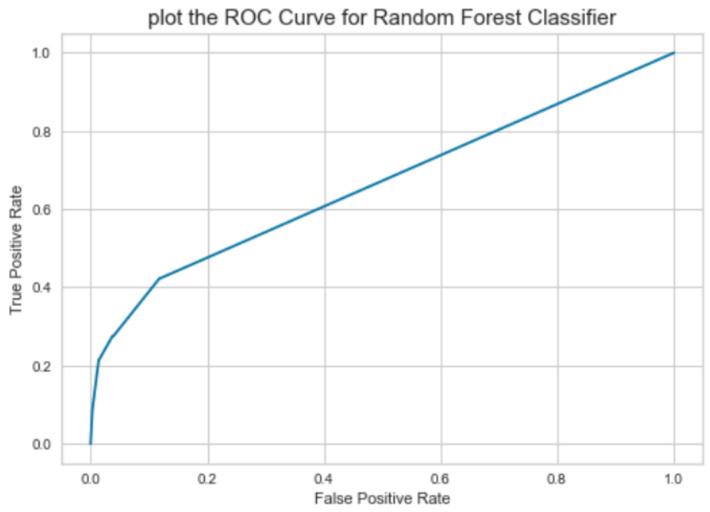
Data Glacier

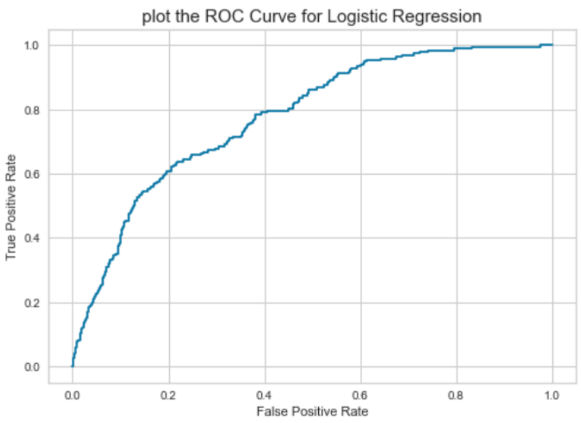
Your Deep Learning Partner

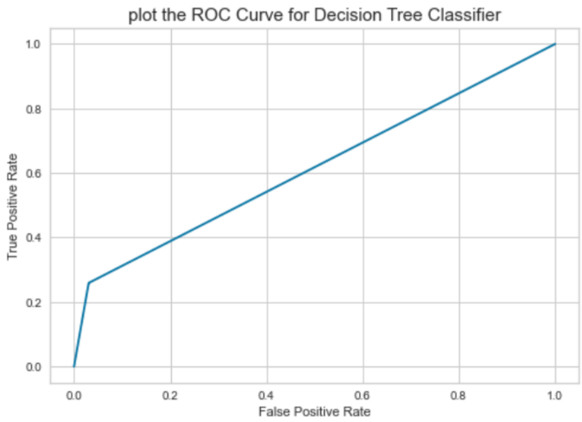
7- Report ROC-AUC

Plots the ROC Curve

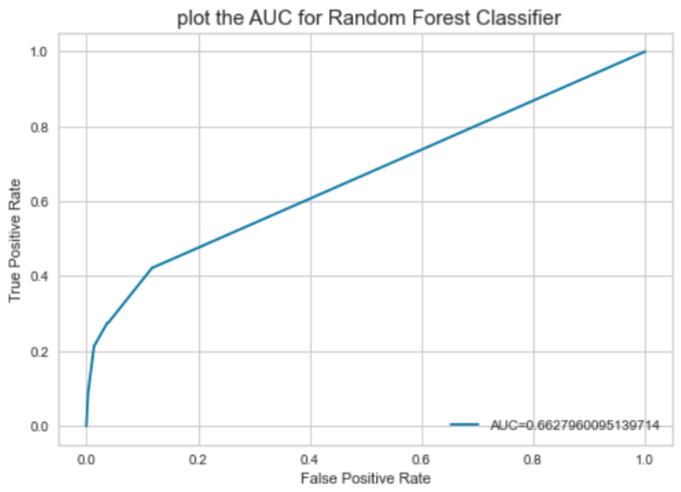


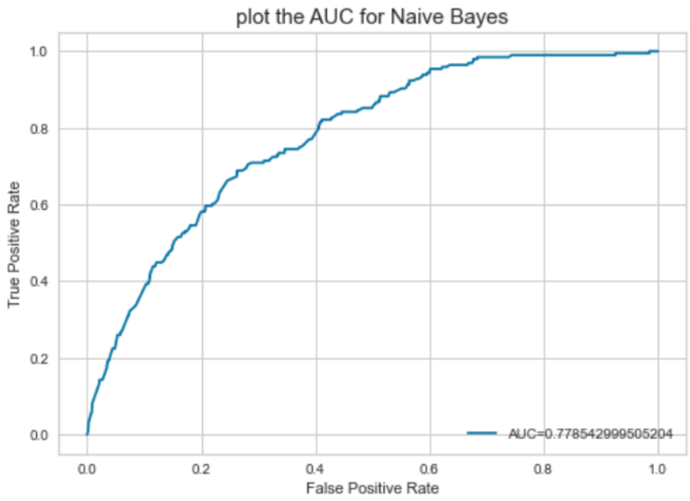


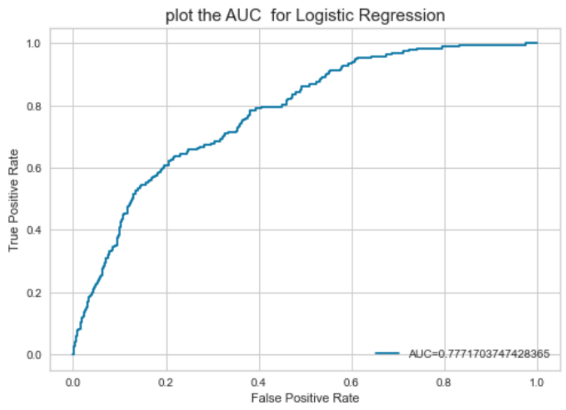


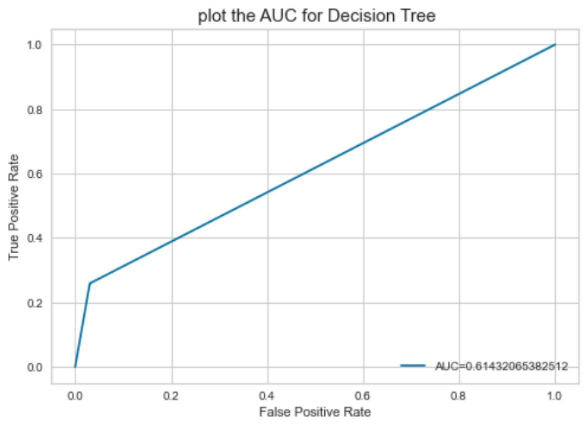


Plots the AUC Curve









Data Glacier

Your Deep Learning Partner

8- Converting ML metrics into Business metrics and explaining results to the business.

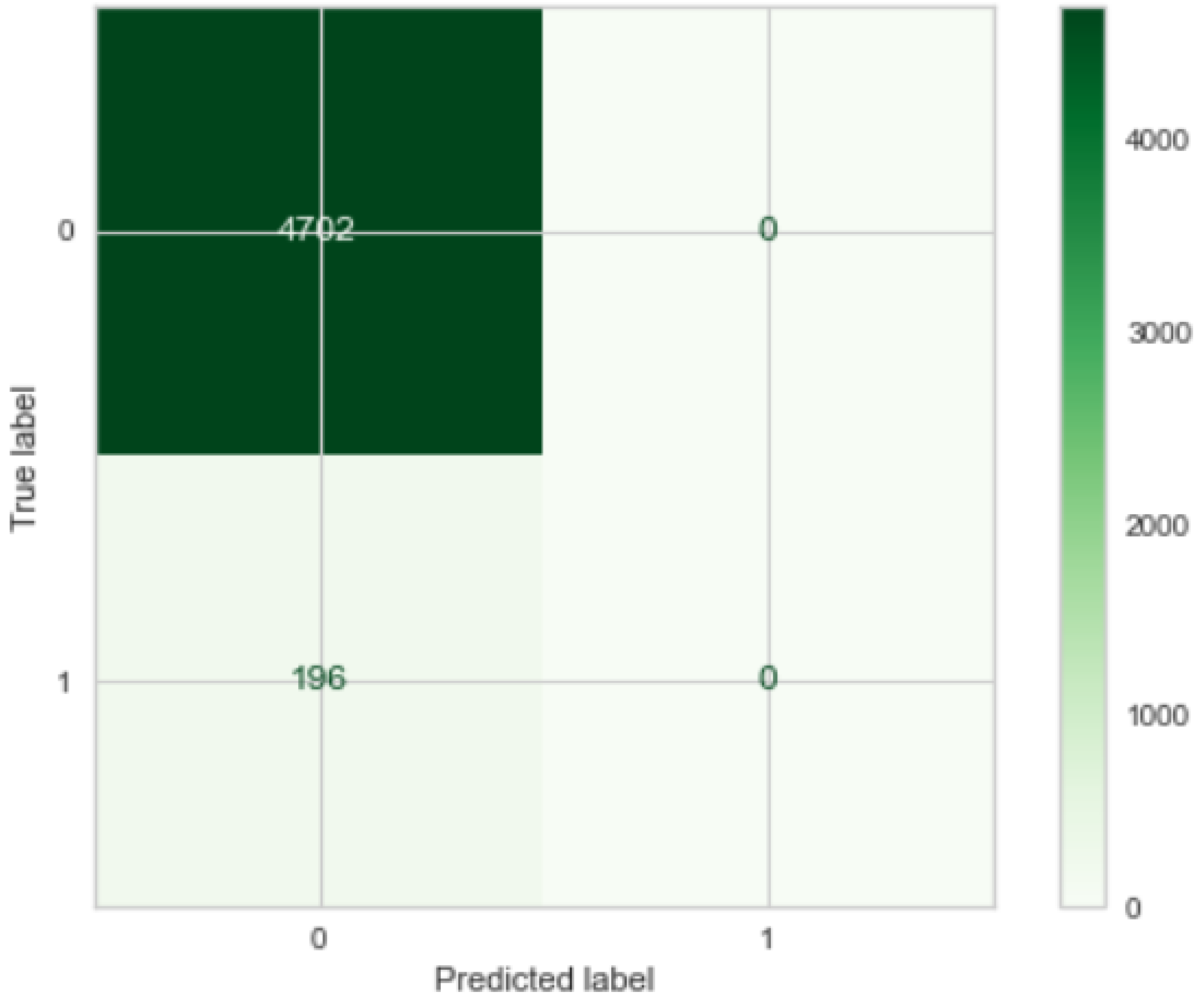
Converting ML metrics into Business metrics and explaining results to the business.

1.Confusion Matrix

2.F1 Score

3.Gain and Lift charts

Confusion Matrix



F1 Score

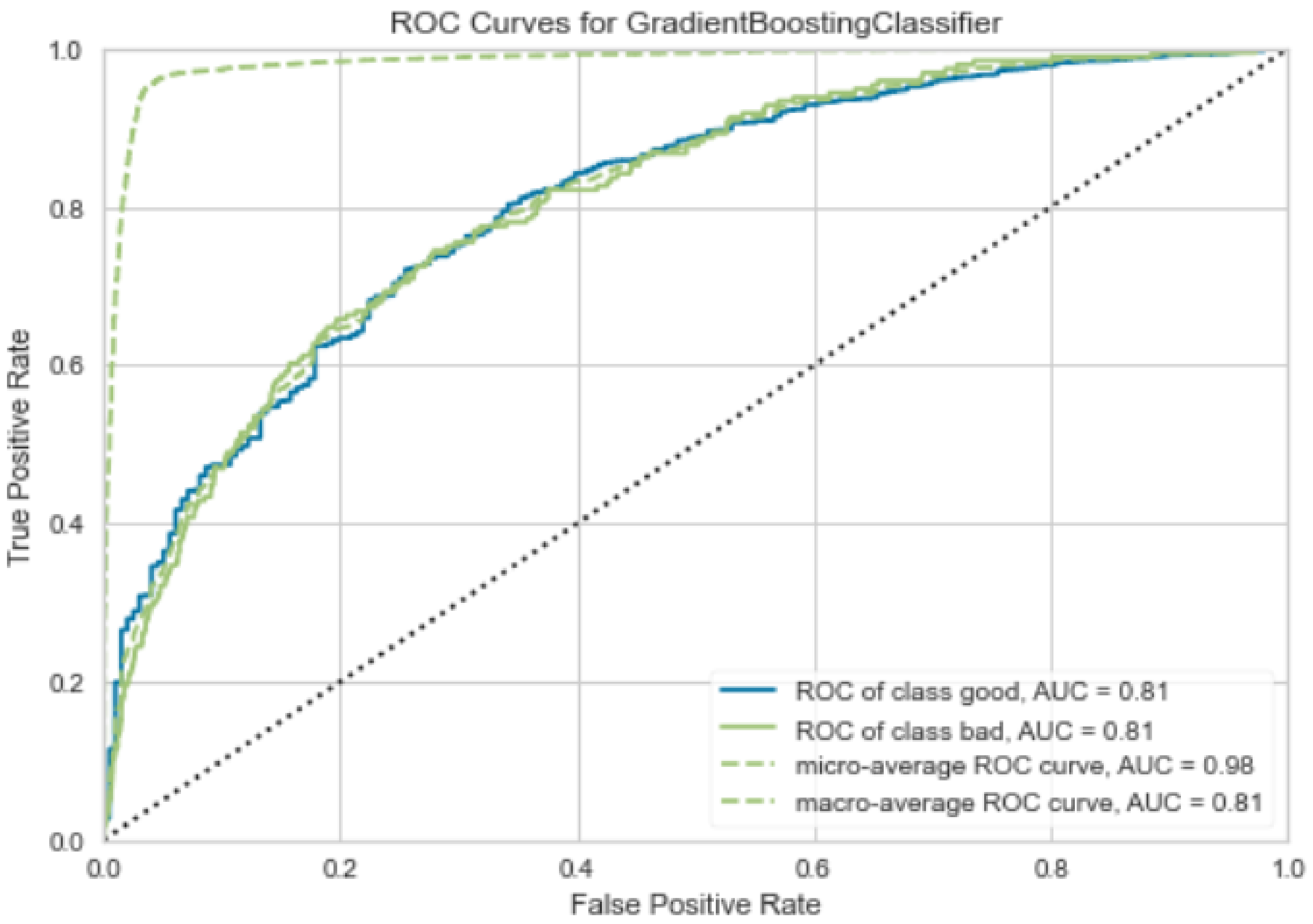
2. F1 Score

```
# F1 Score = 2 * (Precision * Recall) / (Precision + Recall)
# where:
# Precision: Correct positive predictions relative to total positive predictions
# Recall: Correct positive predictions relative to total actual positives

F1_Score = 2 * (0.613918 * 0.324924) / (0.613918 + 0.324924)
F1_Score
```

```
0.4249419864726972
```

Gain and Lift charts

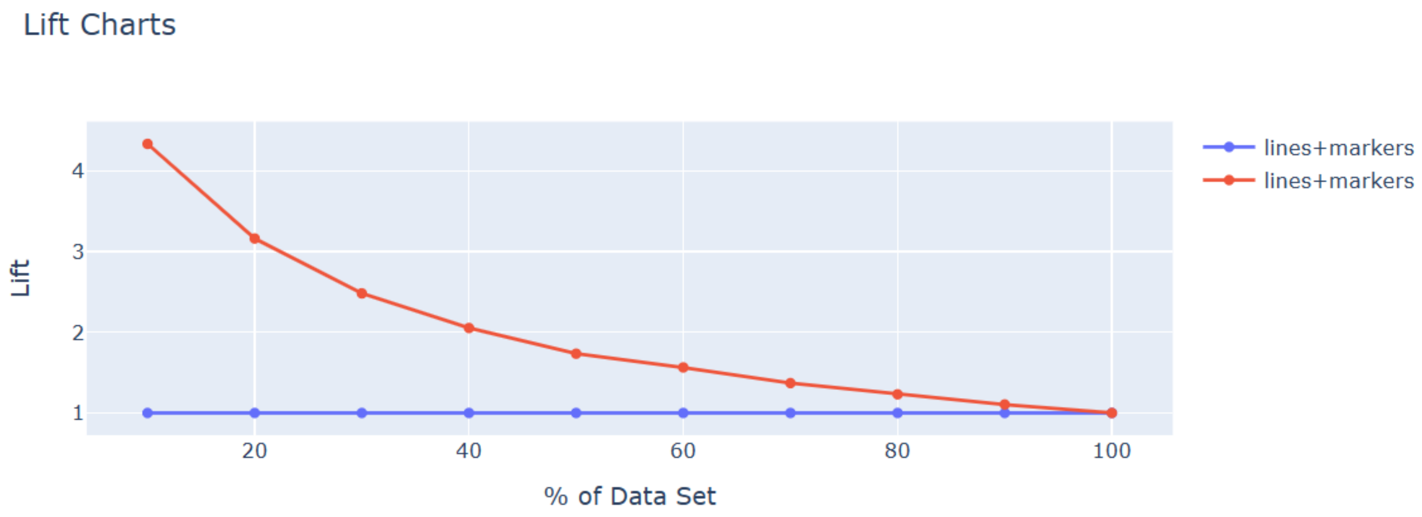
Fit a Gradient Boosting Machine and examine the ROC AUC graph.



Gain chart

Gain Charts
100
50
0
0
20
40
60
80
100
% of Gain
% of Data Set
lines+markers
lines+markers

Lift chart



Lift Charts
4
3
2
1
Lift
20
40
60
80
100
% of Data Set
lines+markers
lines+markers

Bank Marketing (Campaign)

Data Girl

Deadline Date:

30 Dec 2022

LISUM 14

30 Sep – 30 Dec 2022

Fatimah Asiri

Data Glacier

Your Deep Learning Partner

Thank You