Data Glacier

Your Deep Learning Partner

Bank Marketing Campaign Case Study: Modeling

Virtual Internship

Company: ABC Bank

Authors: Ammar Sidhu and Islom Pulatov

Date: 12/30/2022

Team Details

Group Name: AI Boys

Group Member ID	Name	Email	Country	College	Specialization
1	Islom Pulatov	islompulatov115@gmail.com	Poland	Epicode Global	Data Science
2	Ammar Sidhu	ammarsidhu@outlook.com	Canada	University of Toronto	Data Science

Github Repo: https://github.com/Islompulatov/Bank_marketing

AGENDA

Executive Summary

Problem Statement

Approach

EDA

EDA Summary

Correlation Analysis

Modelling

Problem Description and Business Understanding

Problem Description:

- ABC Bank wants to sell its term deposit product to customers and before launching the product they want to develop a model which help them in understanding whether a particular customer will buy their product or not (based on customer's past interaction with bank or other Financial Institution).

Business Understanding:

- The data is related with direct marketing campaigns of a Portuguese banking institution. The marketing campaigns were based on phone calls. Often, more than one contact to the same client was required, 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) a term deposit (variable y).

Why Machine Learning Models?:

- The ABC Bank wants to use machine learning models to shortlist customer whose chances of buying the product is more so that their marketing channel (tele marketing, SMS/email marketing, etc.) can focus only on those customers whose chances of buying the product is more.

Data Cleaning

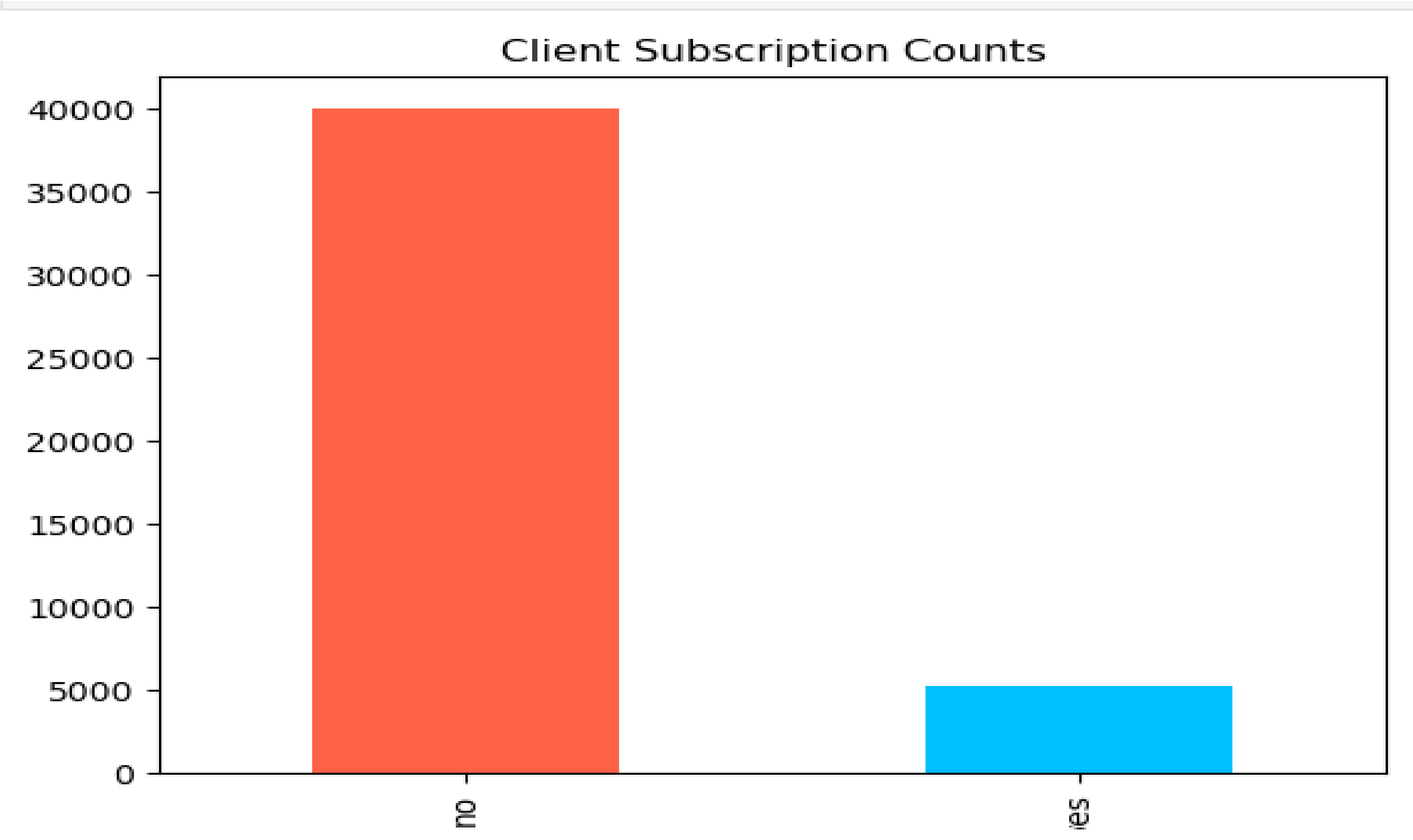
- Found **no missing values** in the dataset

- Found **no duplicate rows** in the dataset

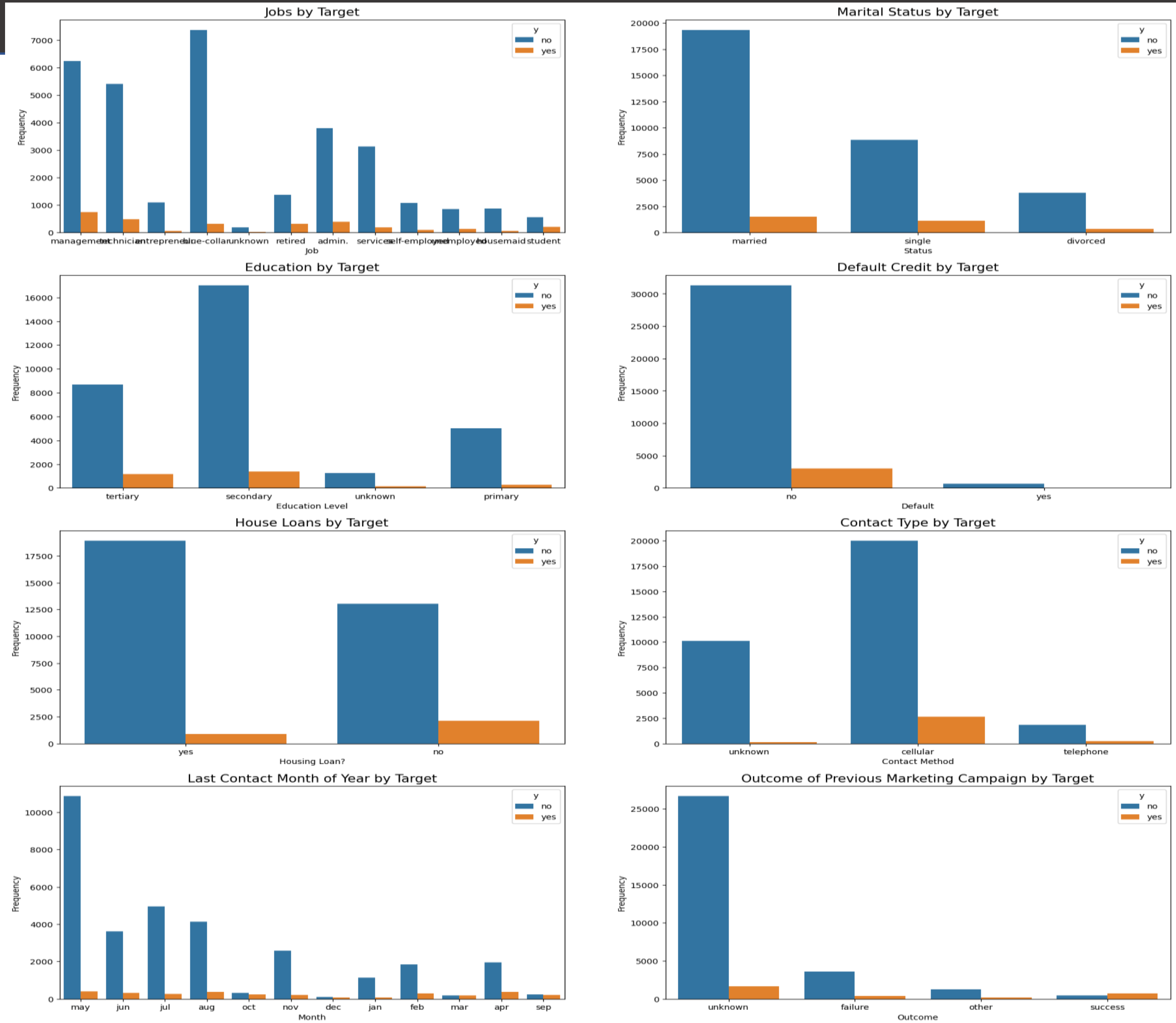
- Handled outliers for the **7 Numerical Features** with two different approaches:
  (1) Do not drop outliers (Islom)
  (2) Drop outliers based on feature and context of the outliers using IQR Method (Ammar)

- 'Unknown' class for categorical variables were handled in two different ways:
  (1) As a unique class so not a missing value (Ammar)
  (2) Treated as a missing value and drop the corresponding row from data frame (Islom)

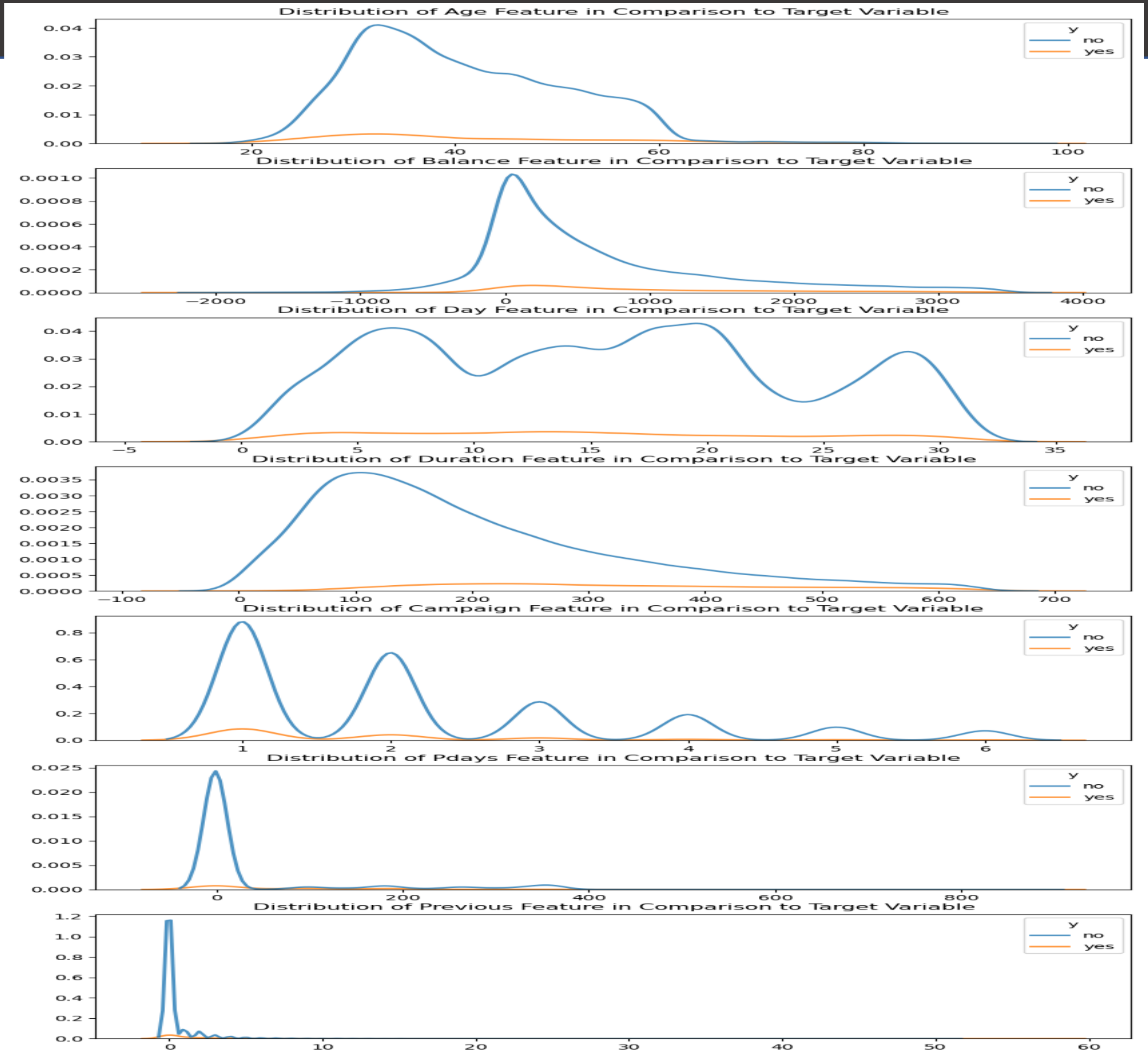
Client Subscriptions Counts – Target (y)

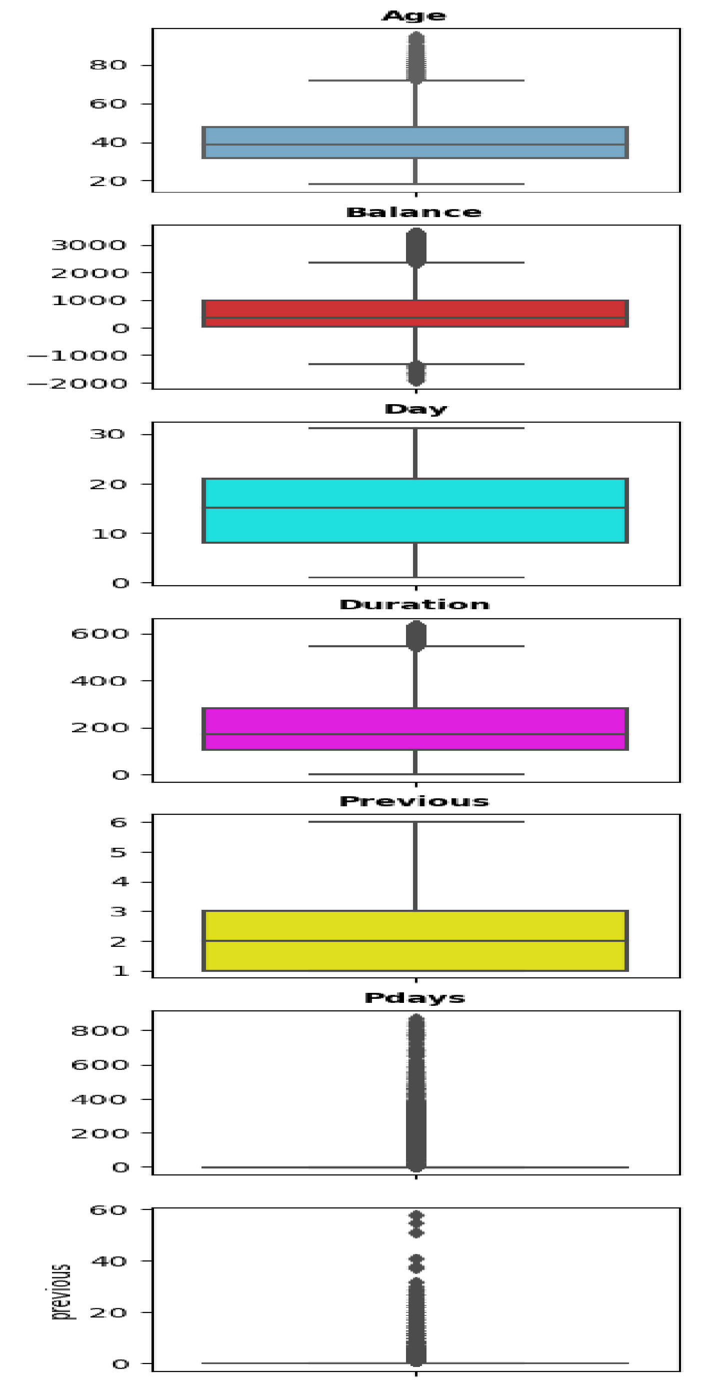


Visualizing Categorical Variables

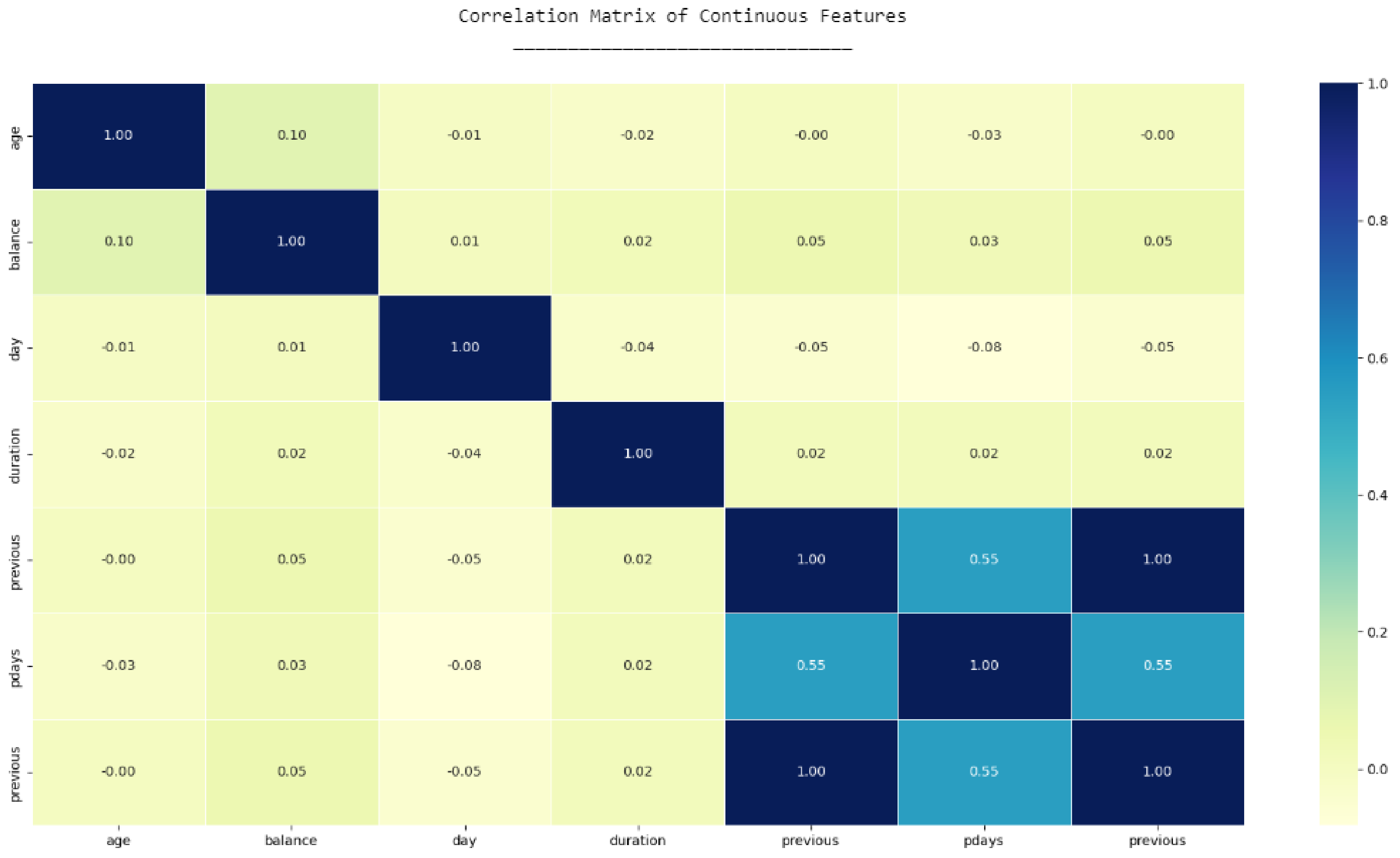


Visualizing Continuous Features





Correlation Analysis



Inference from Correlation Analysis:

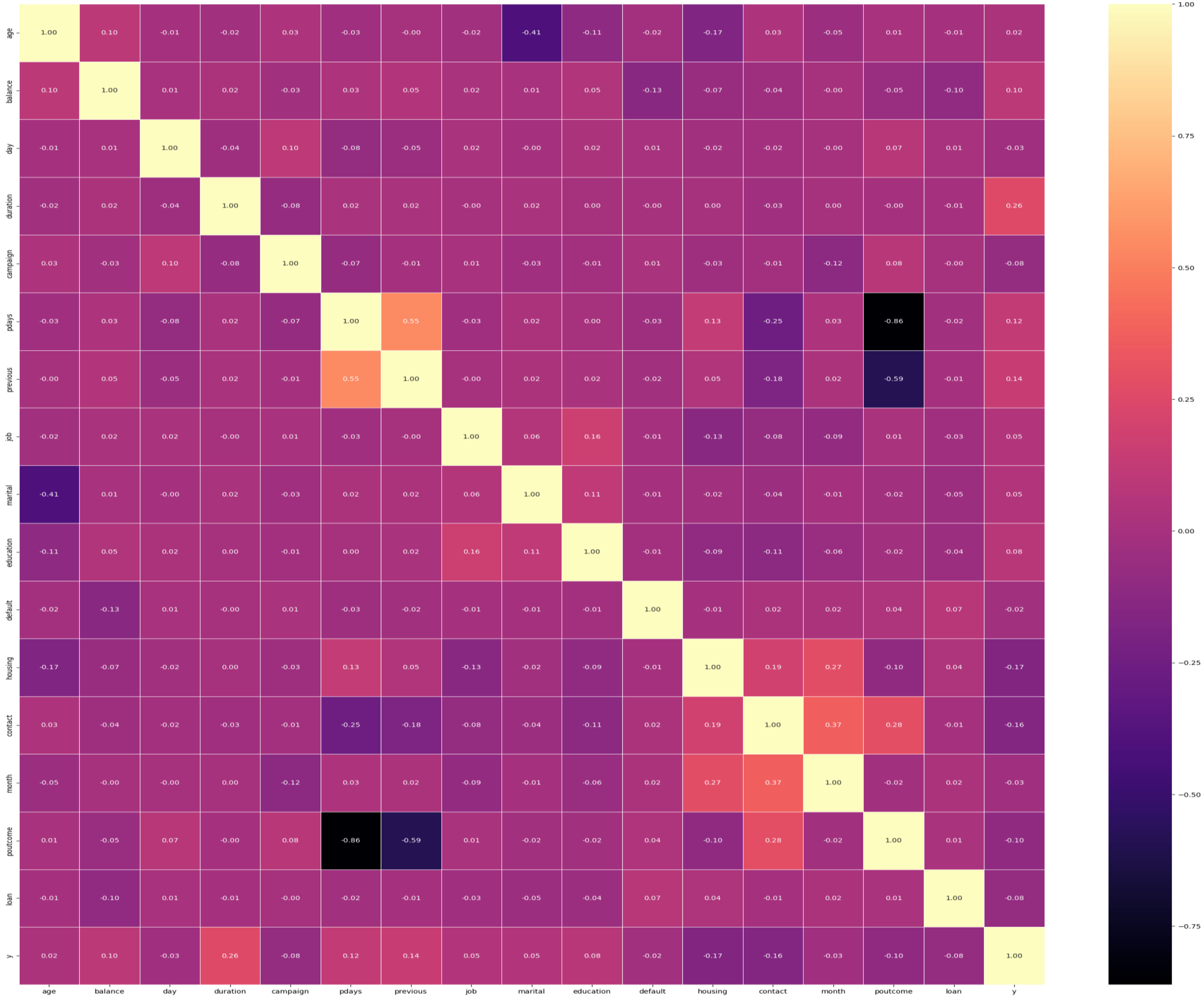
•There is no multicollinearity between the numerical features.

•The only feature with a moderate correlation with the target - y is
the duration feature.

•There is a strong correlation between the
encoded poutcome feature,
and the pdays feature.

•Some features are negatively correlated with each other.

Heatmap of all Features and Including One-Hot Encoded Categorical Features

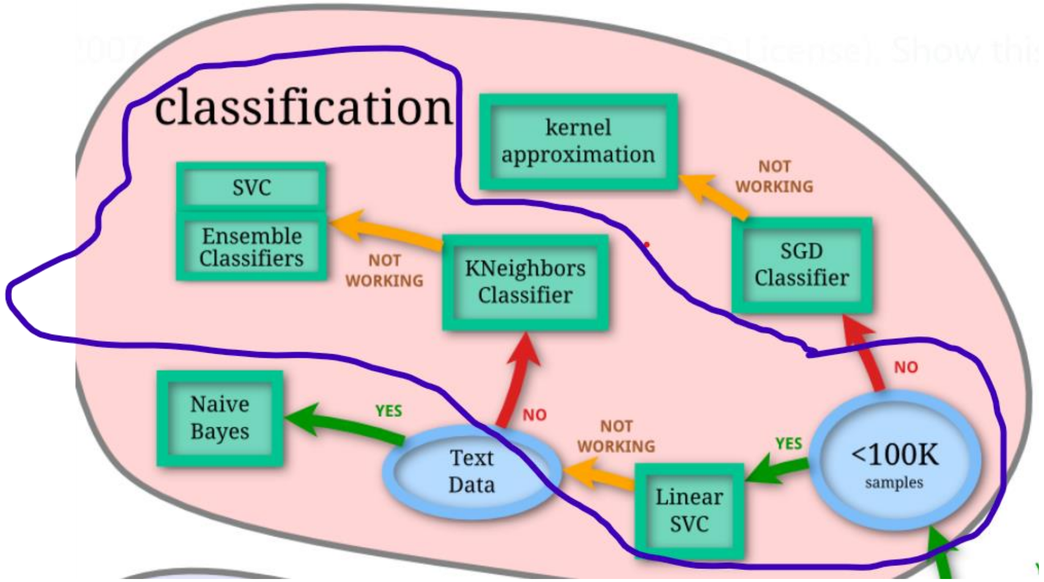


Conclusions from EDA

- There are **no NaN values** and **no duplicate values** in the dataset.
- The **target feature is imbalance**d as there are **more than 8x** the customers who subscribed vs. who dd not.
- Except the duration feature, all the other features have a **low correlation** with the target.
- **Most customers** are married, have loans, and work collar jobs.

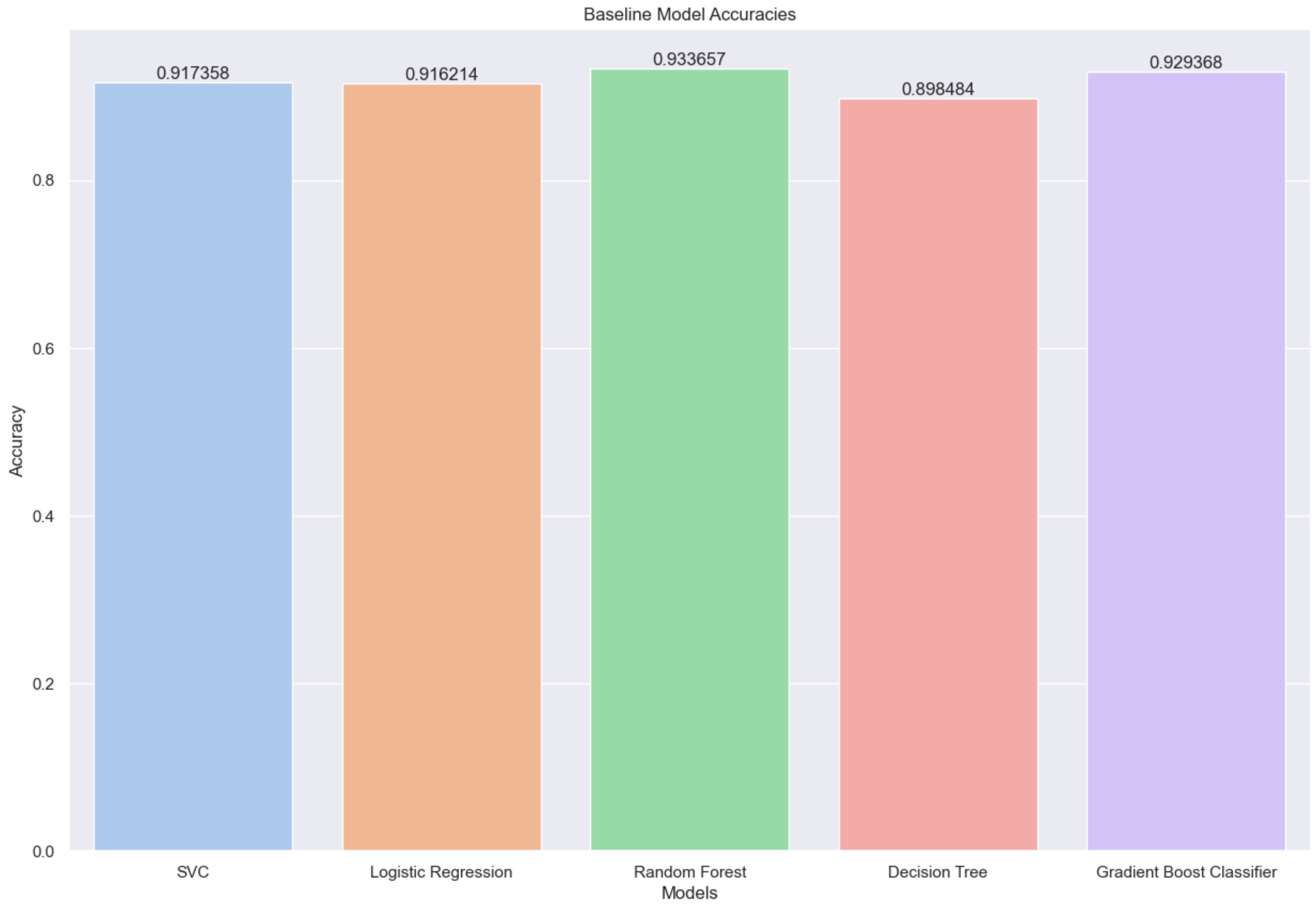
Model Recommendations

- Test and Train Ensemble and Boosting Classification Models with Cost-Sensitive Learning (Ammar).
- Test and Train Ensemble and Boosting Classification Models with SMOTE (Islom).
- Tune Hyperparameters of the Best Performing Models.
- Compare Model Performances and check to see if dropping ‘unknown’ entries had an impact on accuracies.



Source: https://scikit-learn.org/stable/tutorial/machine_learning_map/index.html

Baseline Modeling vs. Cost Sensitive Learning



- 80/20 Training-Test Split
- Achieved 90%+ Accuracy on all models except for Decision Tree
- Random Forest best performing baseline model with approximately 93% accuracy
- Cost-Sensitive Learning yielded a lower accuracy than without on baseline models

Random Forest Model – Confusion Matrix & Classification Report

- The “no” class performed very well
- More data required for the "yes" class given the significantly lower accuracy
- Classification report displays poor performance for “yes” class

	precision	recall	f1-score	support
0	0.94	0.99	0.96	6416
1	0.68	0.35	0.47	578
accuracy			0.93	6994
macro avg	0.81	0.67	0.72	6994
weighted avg	0.92	0.93	0.92	6994

Hyperparameter Tuning and Cross-Validation

```python
# Different RandomForestClassifier() hyperparameters
rf_clf_grid = {'n_estimators': [100, 500, 1000],
         'max_depth': [None, 5, 10],
         'max_features': ['auto', 'sqrt'],
         'min_samples_split': [1, 2],
         'min_samples_leaf': [1, 2]}

# Setup random hyperparameter search for RandomForestClassifier
gs_rf_clf = GridSearchCV(RandomForestClassifier(),
                         param_grid = rf_clf_grid,
                         cv = 5,
                         verbose = True)

# Fit grid hyperparameter search model
gs_rf_clf.fit(X_train, y_train);
```

```
Fitting 5 folds for each of 72 candidates, totalling 360 fits
```

```python
# Check the best parameters
gs_rf_clf.best_params_
```

```
{'max_depth': None,
 'max_features': 'sqrt',
 'min_samples_leaf': 1,
 'min_samples_split': 2,
 'n_estimators': 100}
```

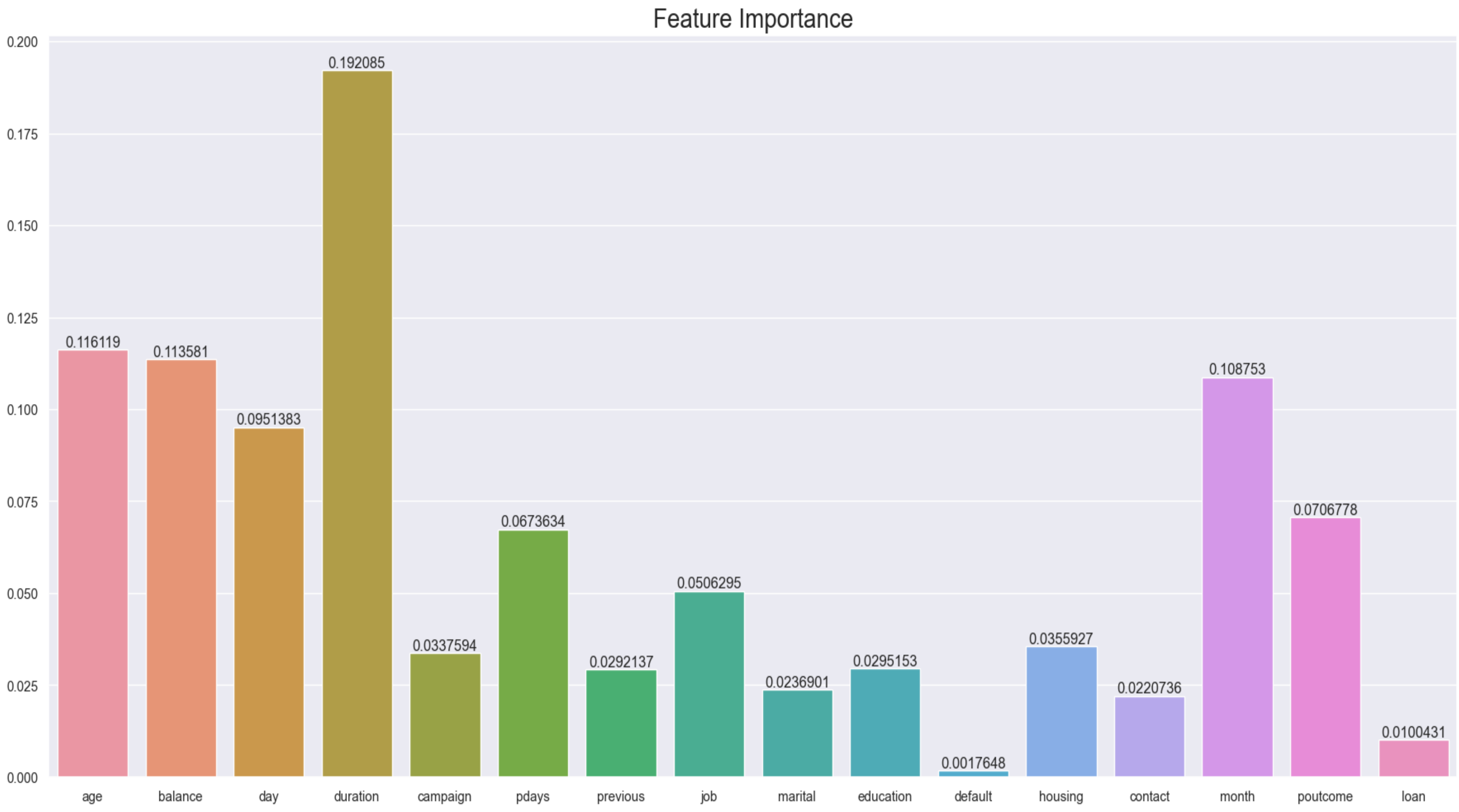
```python
# Evaluate the model
gs_rf_clf.score(X_test, y_test)
```

```
0.9330855018587361
```

The baseline `Random Forest Classifier` outperforms the other baseline models with an accuracy of **93%**, and the `Random Forest Classifiers` with different sets of hyperparameters. Therefore, we will acquire cross-validated evaluation metrics for this model.

- Tuned **Random Forest** Hyperparameters with **5-Fold** Cross-Validation and **360** Total Fits using **GridSearchCV**
- Achieved **93% Accuracy** on **Test Data**

Feature Importance



- Most important feature for classifying subscriptions is **duration** of last class with customer

- **Age**, **Balance**, **Month**, and **Day** are also important features that have a significant influence in classifying a customer's subscription

Conclusion

- Developed a **Random Forest Classifier** that can predict if a client will subscribe to a term deposit with **93% Accuracy**.

- The most important feature for determining if a client will subscribe is the **duration** of the client's last contact.

- Other important features include **Age**, **Balance**, **Month**, and **Day.**

- The ABC Bank should strongly consider leveraging these features for determining if a customer will subscribe to them or not and consider dropping features such as **default**, **contact**, and **previous** should be dropped from the data collection process.

- Model performances for the subscription class can be improved by building models that exclude these features and will ultimately save money and time when collecting future data.

Thank You

Data Glacier
Your Deep Learning Partner