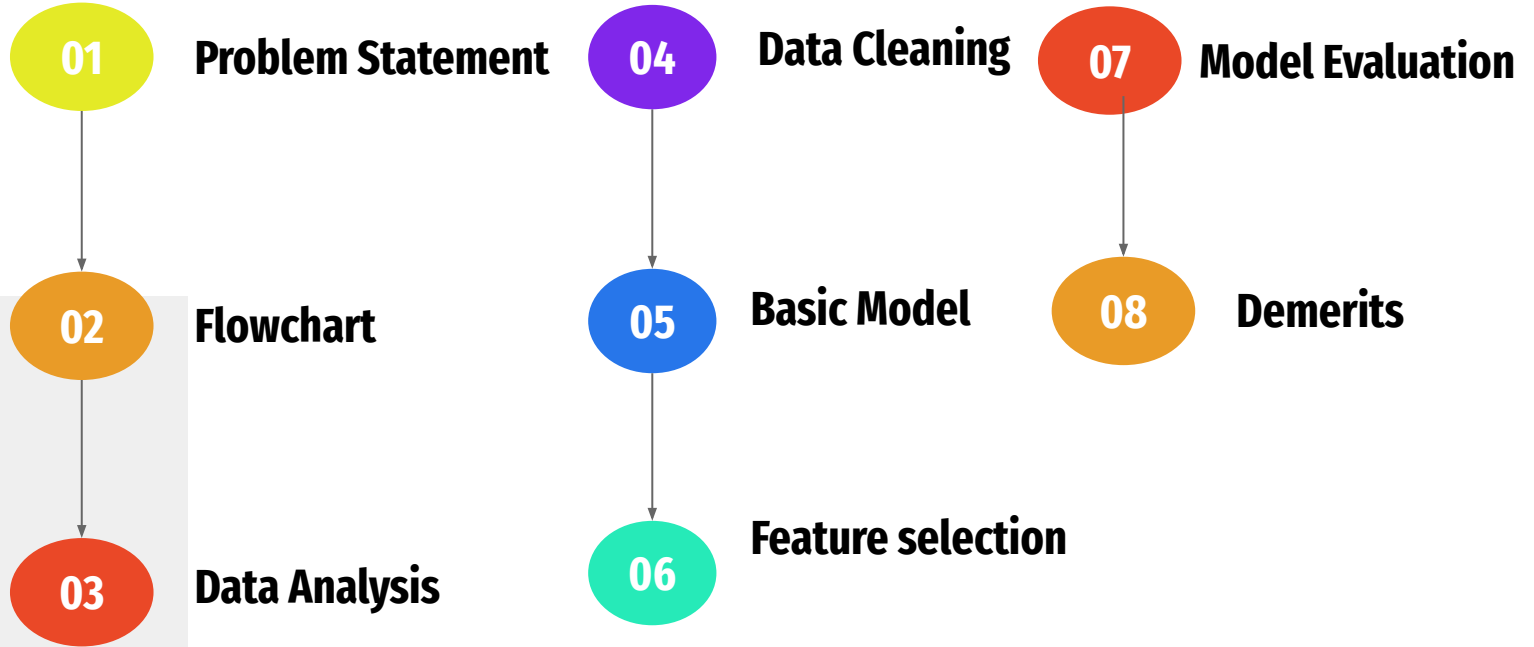


Predictive Analysis for Bank Telemarketing Success

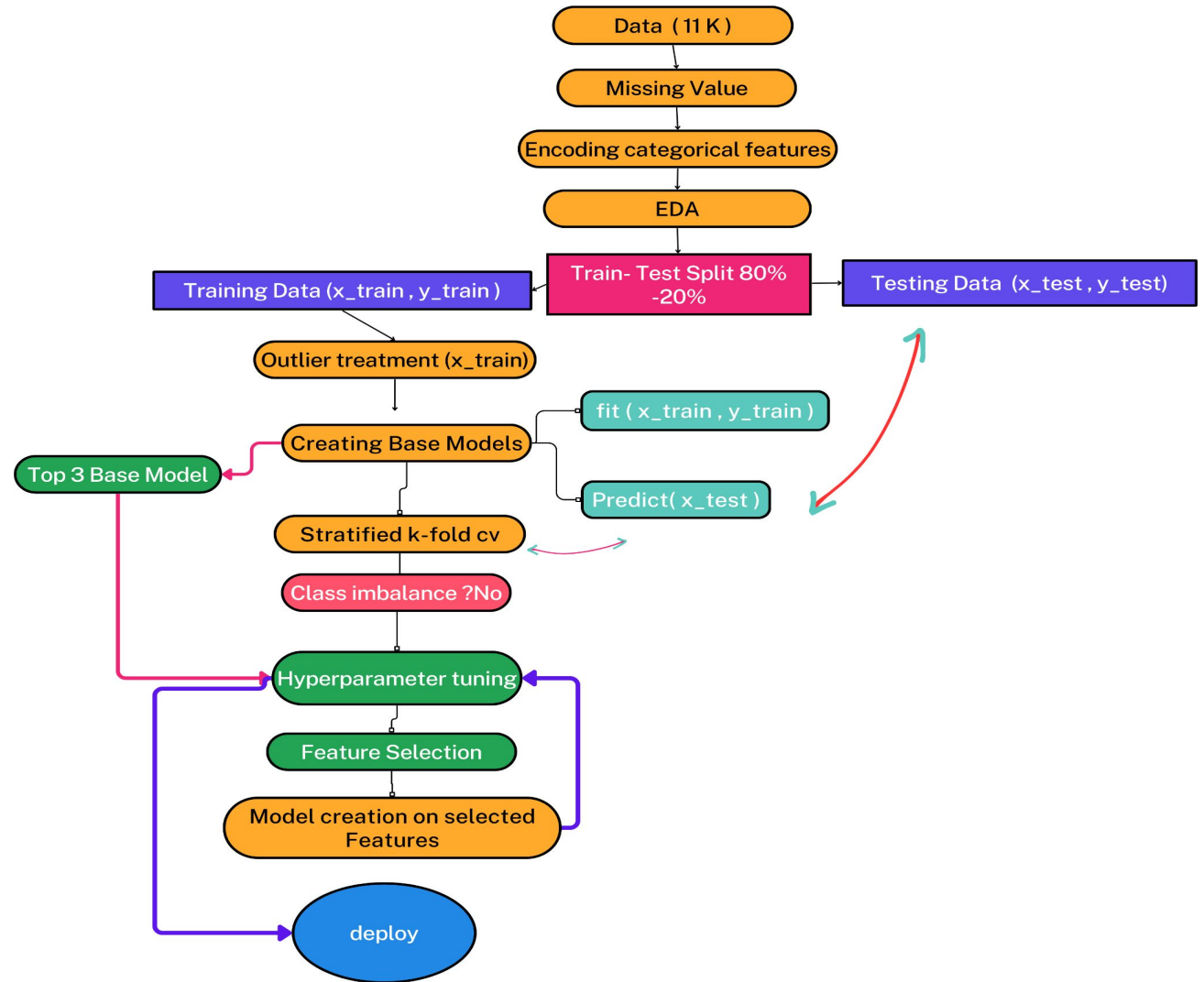
Contents



Problem Statement

Campaign Success Prediction:
**Building a Predictive Model to Determine Customer
Subscription to Term Deposit Policy**

Project Flowchart



Dataset Information

Dataset Shape - (11162, 17)

Dataset format - CSV File

Target Feature - Deposit (yes , no)

Independent Features -

Age,job,marital,education,default,balance,housing,loan,contact,day
,month,duration,campaign,pdays,previous,poutcome

Data Analysis (EDA)

1. No Class Imbalance

2. Positive correlation between call duration and subscription likelihood.

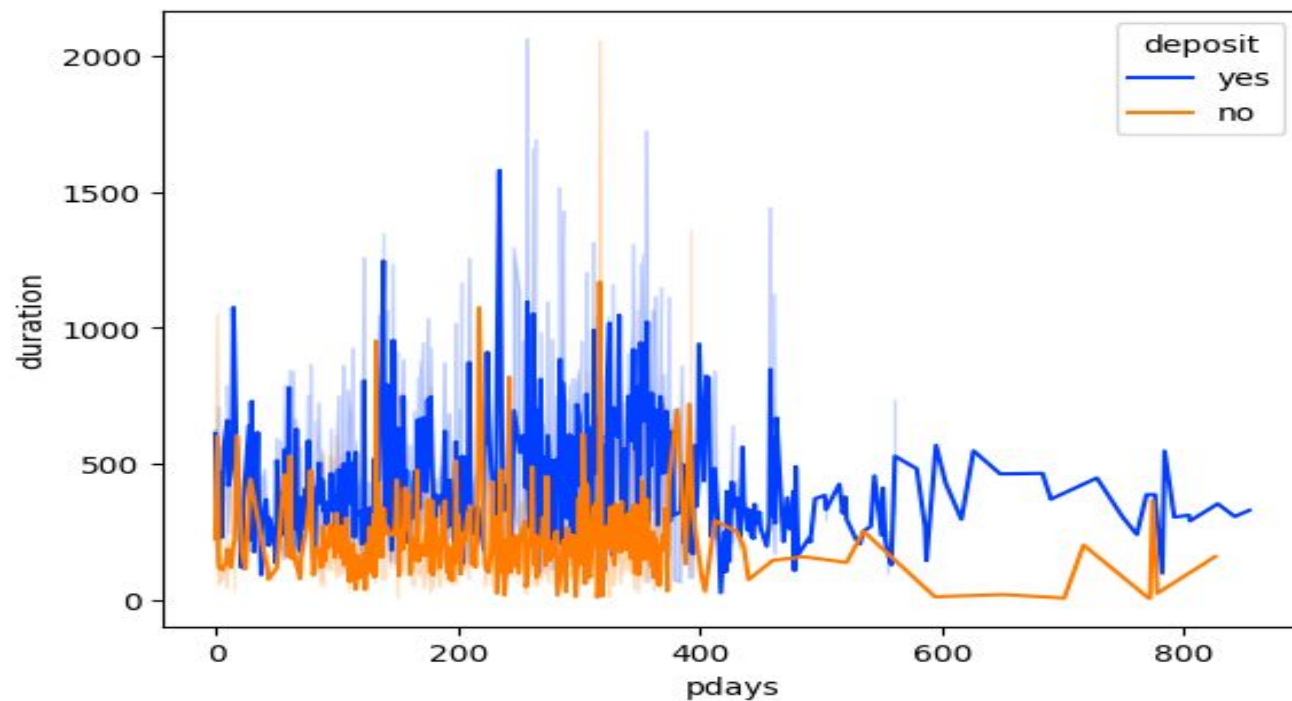
3.Impact of Categorical Columns on the Target Variable:

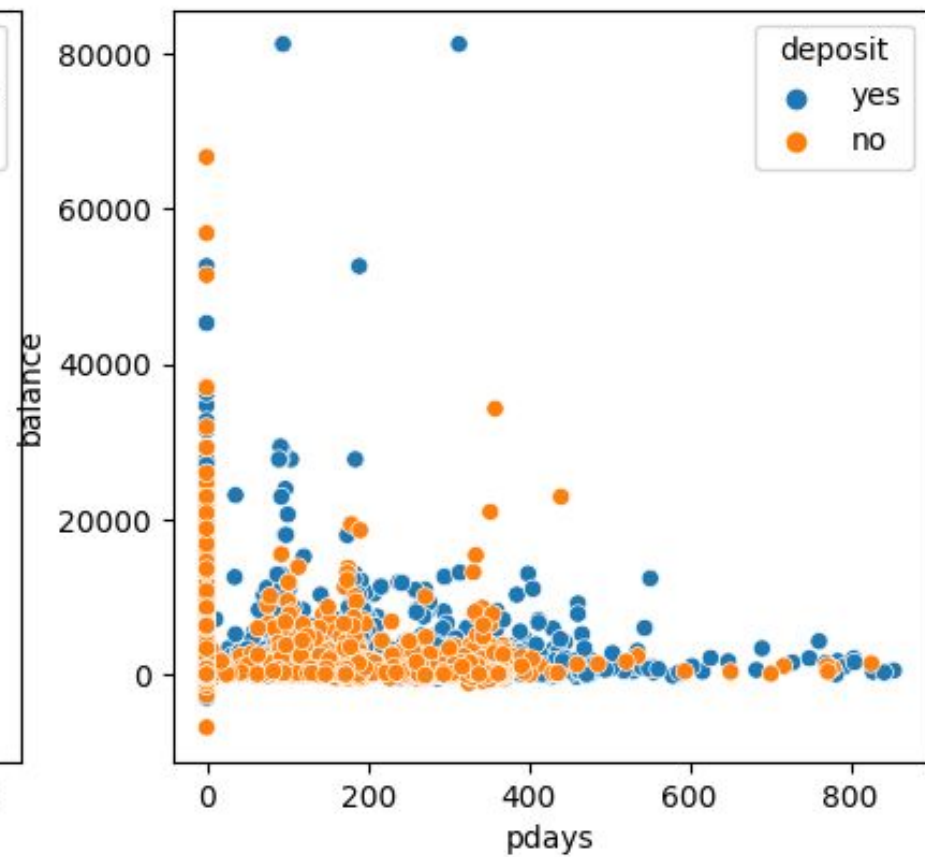
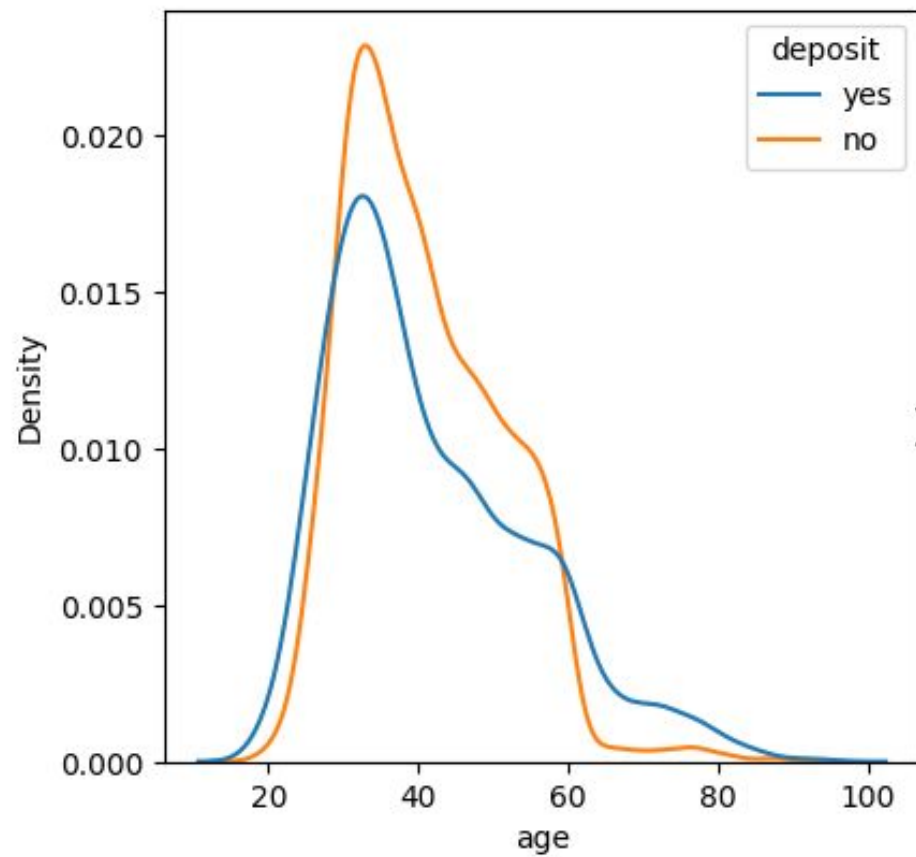
Single Marital Status, tertiary education, and do not have an existing housing loan.
These demographic factors appear to influence the subscription rate positively.

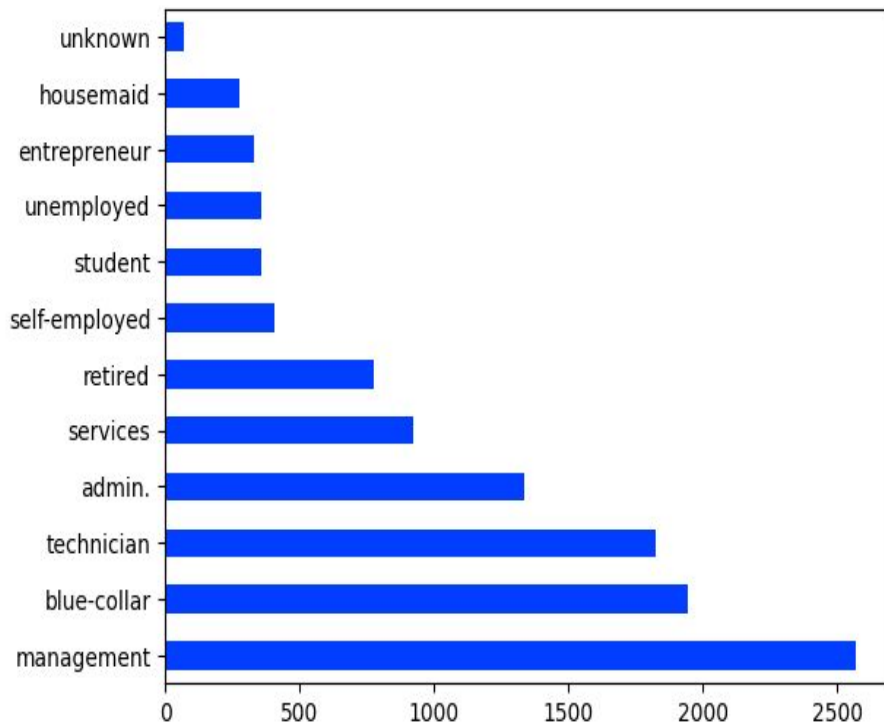
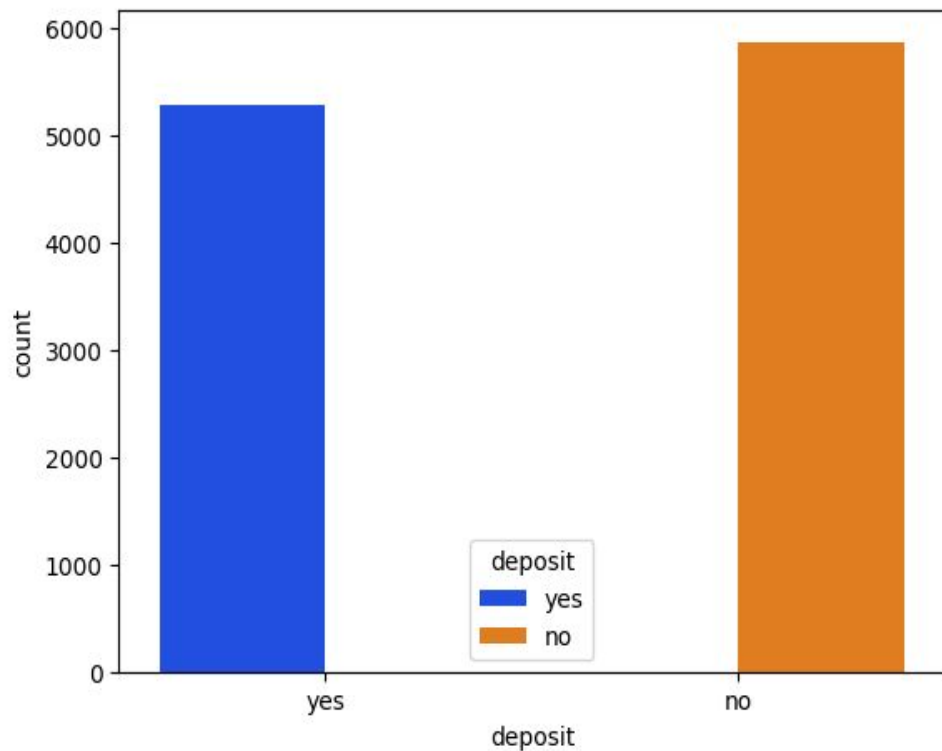
5.Effect of Last Contact Month on Subscription:

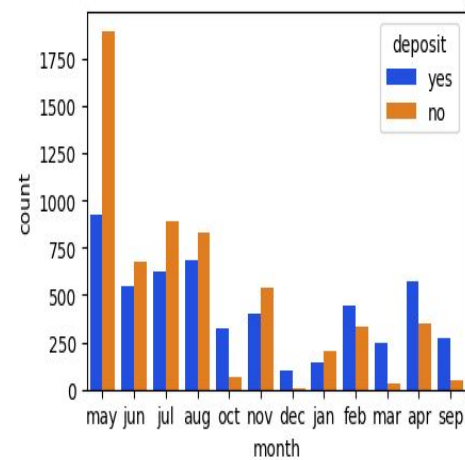
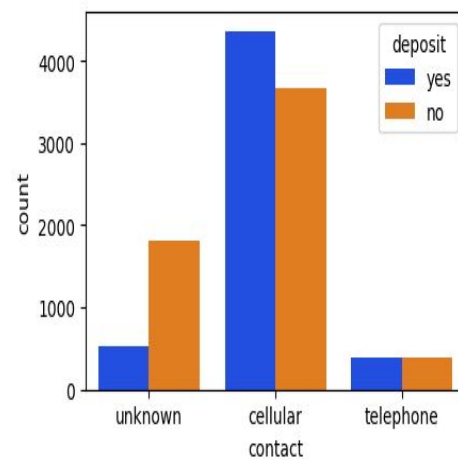
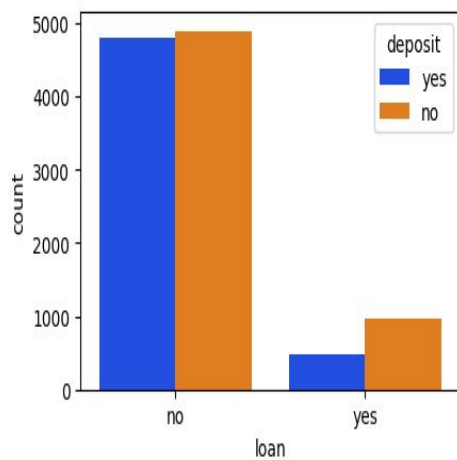
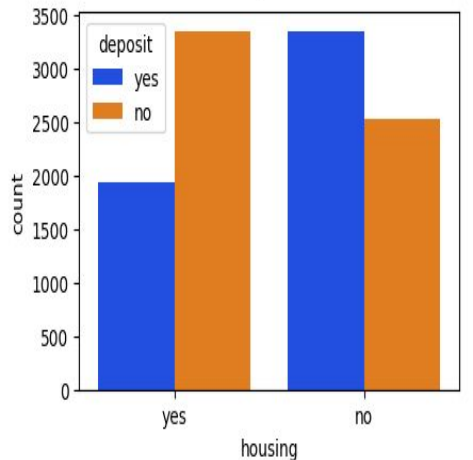
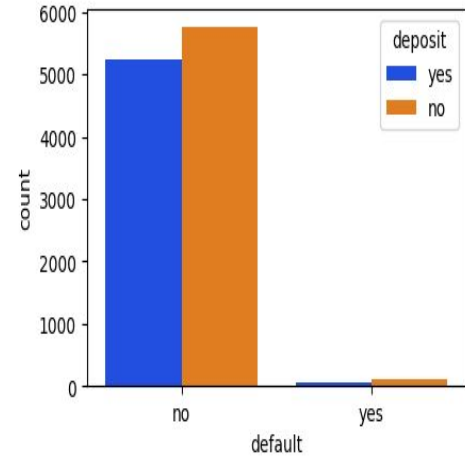
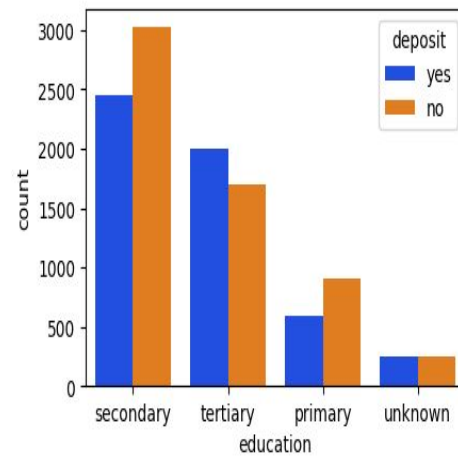
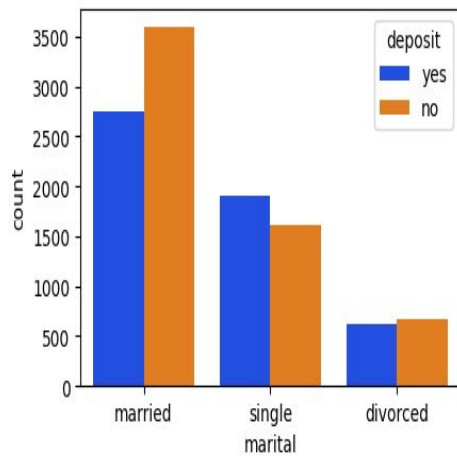
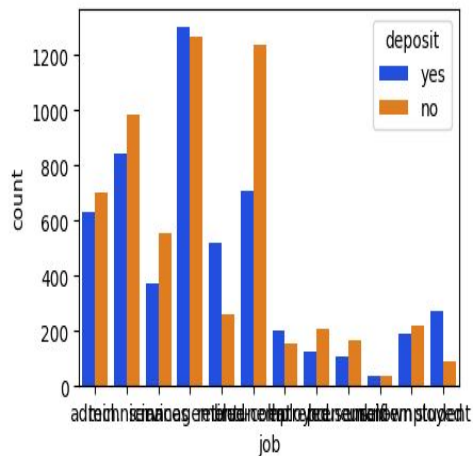
The count of customers subscribing to a term deposit is notably higher when they were last contacted in the months of February, March, April, and September.

	age	job	marital	education	default	balance	housing	loan	contact	day	month	duration	campaign	pdays	previous	poutcome	deposit
7567	37	admin.	married	secondary	no	641	yes	no	unknown	5	jun	42	1	-1	0	unknown	no
6780	30	services	single	secondary	no	-100	yes	yes	cellular	15	may	292	1	-1	0	unknown	no
5220	61	retired	married	tertiary	no	3140	yes	yes	cellular	6	aug	975	4	98	1	unknown	yes









Data Cleaning and Preprocessing

1. Special Codes Handling:

Pdays Adjustment: Replaced -1 with 0 for uniform representation of non-contacted clients.

2. Managing Missing Values:

Filled 'job', 'education', 'contact' having NaNs present as 'unknown' in data

Dropped 'poutcome' Column: More than 50% missing records

3. Missing Value Imputation:

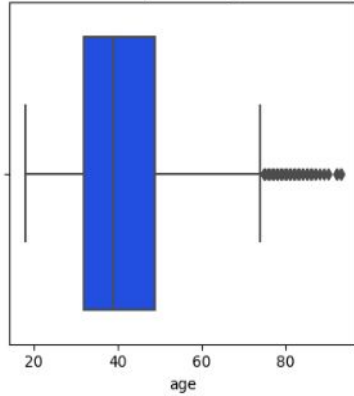
Strategy: Used SimpleImputer replaced missing values with most frequent values.

4. Categorical to Numeric Conversion:

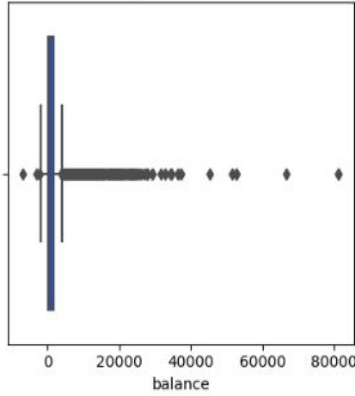
Label Encoding(LabelEncoder): Transformed categorical features into numeric values for analysis.

Outlier Detection

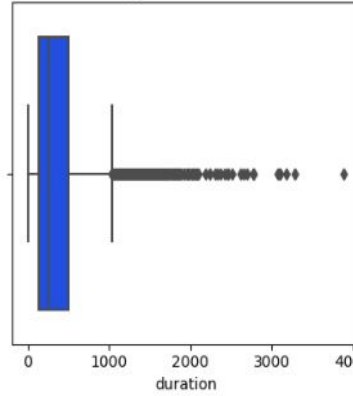
Boxplot for age



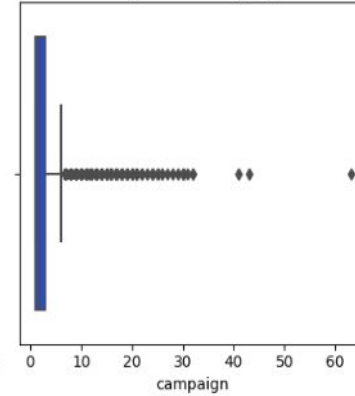
Boxplot for balance



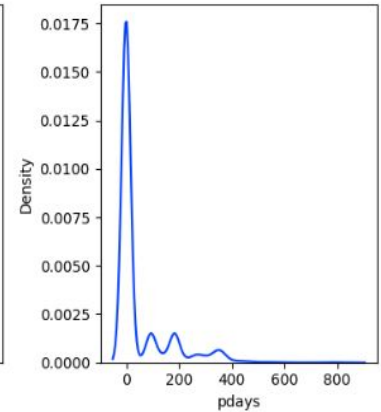
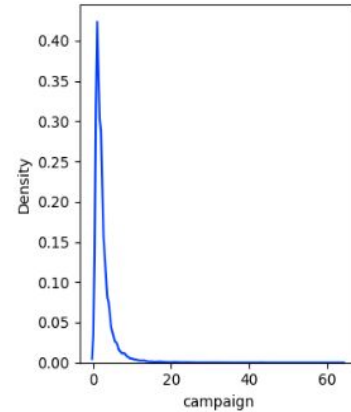
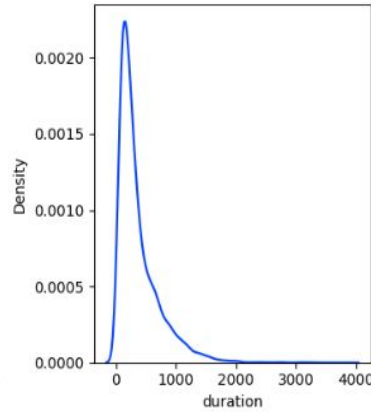
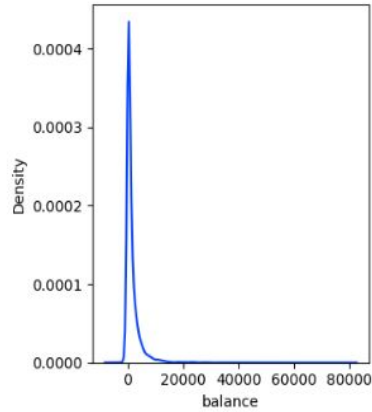
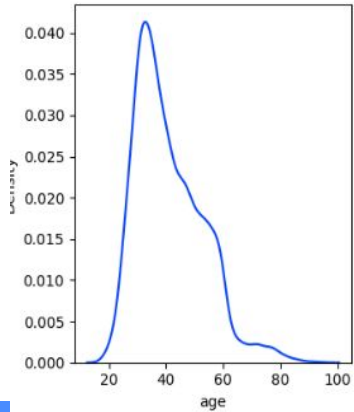
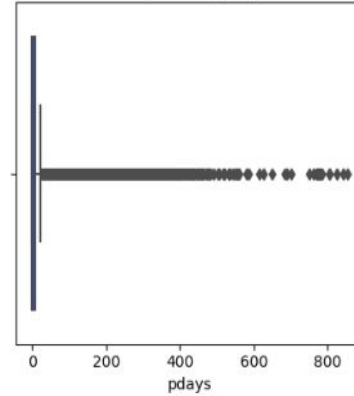
Boxplot for duration



Boxplot for campaign



Boxplot for pdays



Base Models Performance Evaluation

	Model	Accuracy	Precision	Recall	F1 Score	AUC	Specificity	PRC score
0	log	75.279893	0.724138	0.793548	0.757256	0.753917	0.714286	0.674952
1	SVC	69.726825	0.702676	0.653456	0.677173	0.696066	0.738676	0.627551
2	KNN	72.637707	0.714286	0.728111	0.721132	0.726425	0.724739	0.652188
3	Random Forest	82.489924	0.807080	0.840553	0.823476	0.825329	0.810105	0.755867
4	adaboost	79.892521	0.798311	0.784332	0.791260	0.798525	0.812718	0.730933
5	Gradient Boosting	82.400358	0.808929	0.835023	0.821769	0.824306	0.813589	0.755635
6	catBoost	84.773847	0.826468	0.869124	0.847260	0.848325	0.827526	0.781895
7	XGBoost	83.161666	0.814552	0.846083	0.830018	0.832014	0.817944	0.763966
8	Decision Tree	76.175549	0.760603	0.743779	0.752097	0.761262	0.778746	0.690217

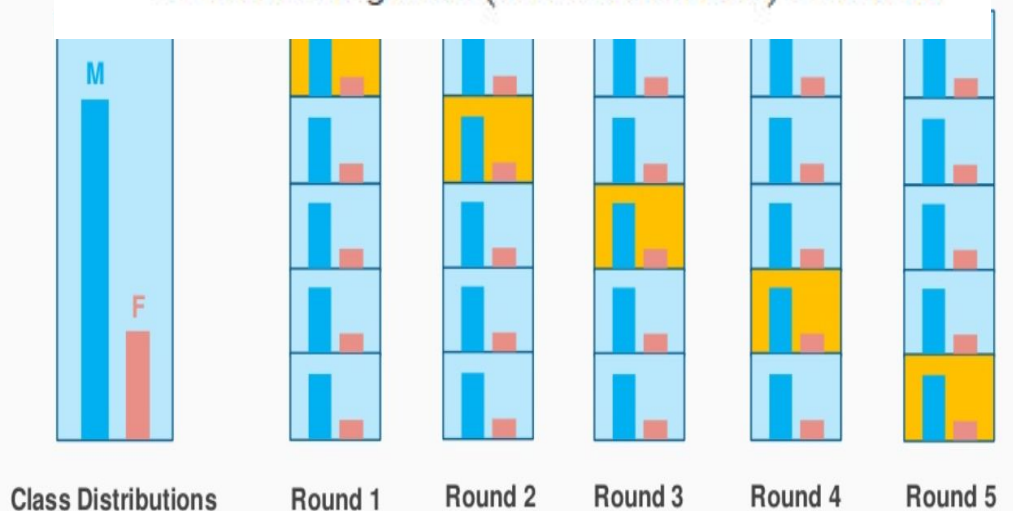
Stratified K- fold Cross Validation Results

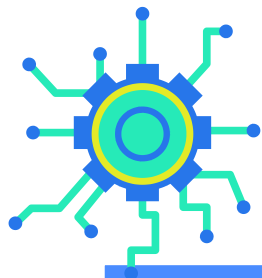
CatBoostClassifier (cross validation results)

- on testing data => 84.77
- on training data (cross validation) => 85.32

XGBClassifier (cross validation results)

- on testing data => 83.16
- on training data (cross validation) => 84.18





Hyperparameter tuning on **Top 3 base models**

RandomSearchCv

Finding Best
Hyperparametrs

Feature
selection

Selecting
Best Model

- Done Hyperparameter tuning for =>
- **CatBoostClassifier**
- **XGBClassifier**
- **RandomForestClassifier**
- Selected features using **feature_importances_** attribute of models
- Generated models on their selected features
- "Evaluated model performance; **CatBoostClassifier** emerged as the top performer based on selected features."

Feature selection - Recursive Feature Elimination

```
from sklearn.feature_selection import RFE
selector = RFE(CatBoostClassifier(verbose=False), n_features_to_select=7, step=1)
selector.fit(x_train,y_train)
```

Selected Features: ['age', 'balance', 'housing', 'day', 'month', 'duration', 'pdays']

Feature_Score_cat	columns_cat	Feature_Score_xgb2	columns_xgb	Feature_Score_rf2	columns_rf
33.414440	duration	0.277740	pdays	0.447330	duration
19.184392	month	0.171153	duration	0.090855	month
10.767175	day	0.147272	housing	0.081055	age
5.716065	age	0.092132	month	0.074622	balance
5.343753	housing	0.057053	loan	0.064487	day
5.073916	pdays	0.038255	previous	0.045662	housing
4.224723	balance	0.034531	day	0.043628	previous
4.158954	job	0.030671	age	0.038503	pdays
2.787719	education	0.027352	education	0.032323	job
2.670489	previous	0.026517	contact	0.030191	campaign
2.292366	campaign	0.026381	campaign	0.017447	education
2.069355	marital	0.026284	balance	0.016666	marital
1.540071	loan	0.023796	marital	0.011808	loan
0.648264	contact	0.020865	job	0.004521	contact
0.108317	default	0.000000	default	0.000901	default

```
x_train1=x_train[['duration','housing', 'age', 'day', 'month','balance', 'pdays']]
x_test1=x_test[['duration','housing', 'age', 'day', 'month','balance', 'pdays']]
```


Procedure on New Train -Test set

01

Training and Testing

Trained the top 3 performing models on new independent features

02

ROC-AUC , PRC Curve

Plot a beautiful ROC-AUC And PRC CURVE. Finding **Optimal threshold value**
=> got 0.46

03

Evaluate model using threshold

Best Performer is CatBoostClassifier

04

Cross Validation

Getting perfect model

05

Deploy

Evaluation on New Train & test

CatBoostClassifier

Accuracy: 85.53515450067174
f1_score : 0.8553515450067174
roc_score : 0.85603293244914
prc_score : 0.790426735771544
Precision: 0.8318815331010453
Recall: 0.880184331797235
specificity : 0.8318815331010
Confusion Matrix:
[[955 193]
[130 955]]

XGBClassifier

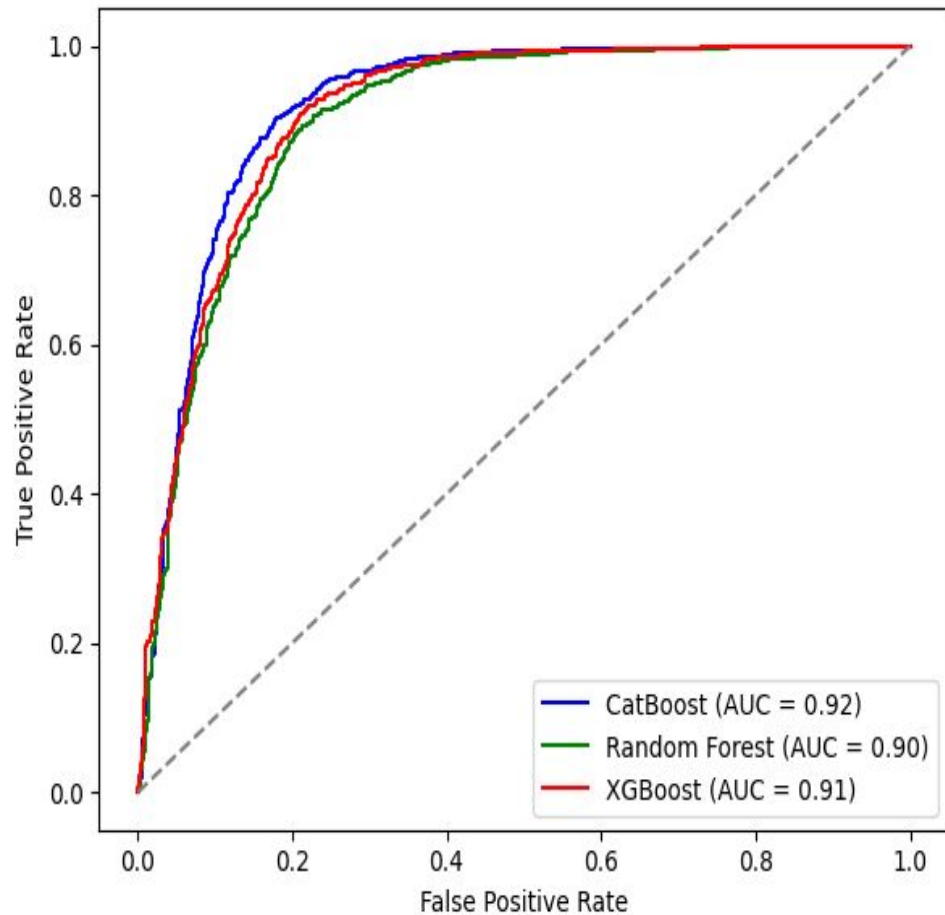
Accuracy: 83.69905956112854
f1_score : 0.8354430379746834
roc_score : 0.8373918174665618
prc_score : 0.7703171037020241
Precision: 0.8198757763975155
Recall: 0.8516129032258064
specificity : 0.82317073170731
Confusion Matrix:
[[945 203]
[161 924]]

RandomForestClassifier

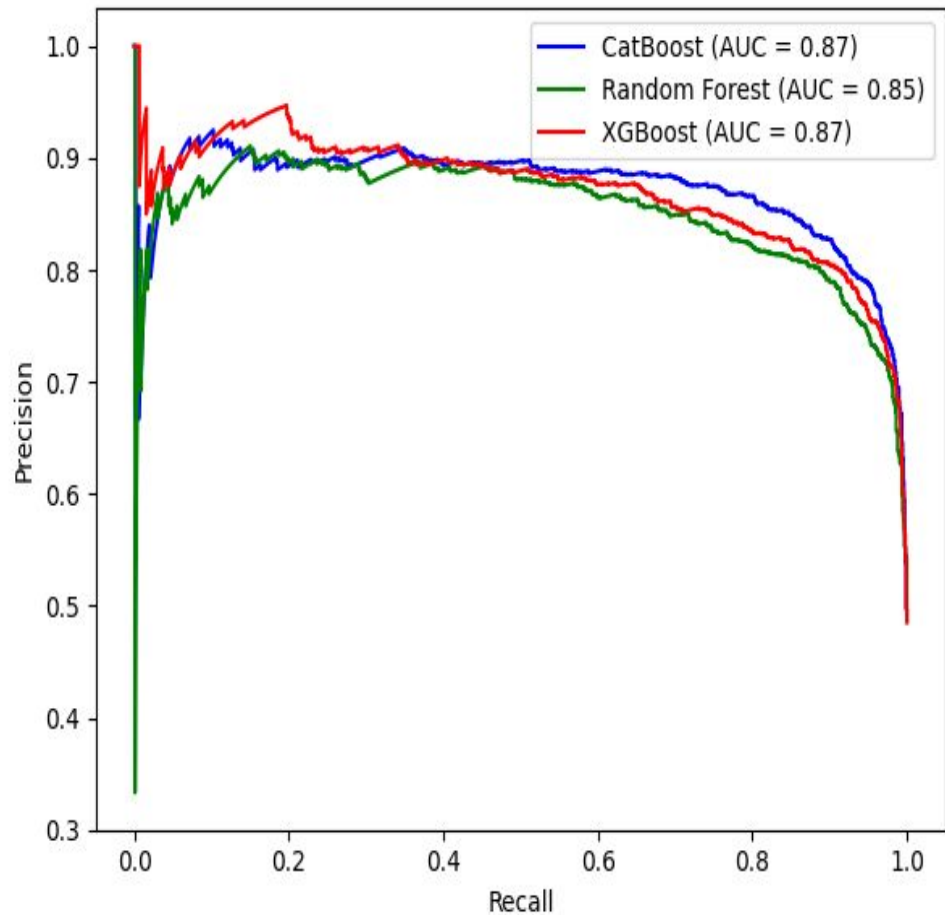
Accuracy: 82.75862068965517
f1_score : 0.8263419034731619
roc_score : 0.828043160615938
prc_score : 0.7588309079852491
Precision: 0.8091872791519434
Recall: 0.8442396313364056
specificity : 0.8118466898954704
Confusion Matrix:
[[932 216]
[169 916]]

	Model	Accuracy	Precision	Recall	F1 Score	AUC	Specificity
0	Random Forest	83.79	0.81	0.87	0.84	0.84	0.80
1	catBoost	86.07	0.83	0.90	0.86	0.86	0.82
2	XGBoost	84.19	0.82	0.87	0.84	0.84	0.82

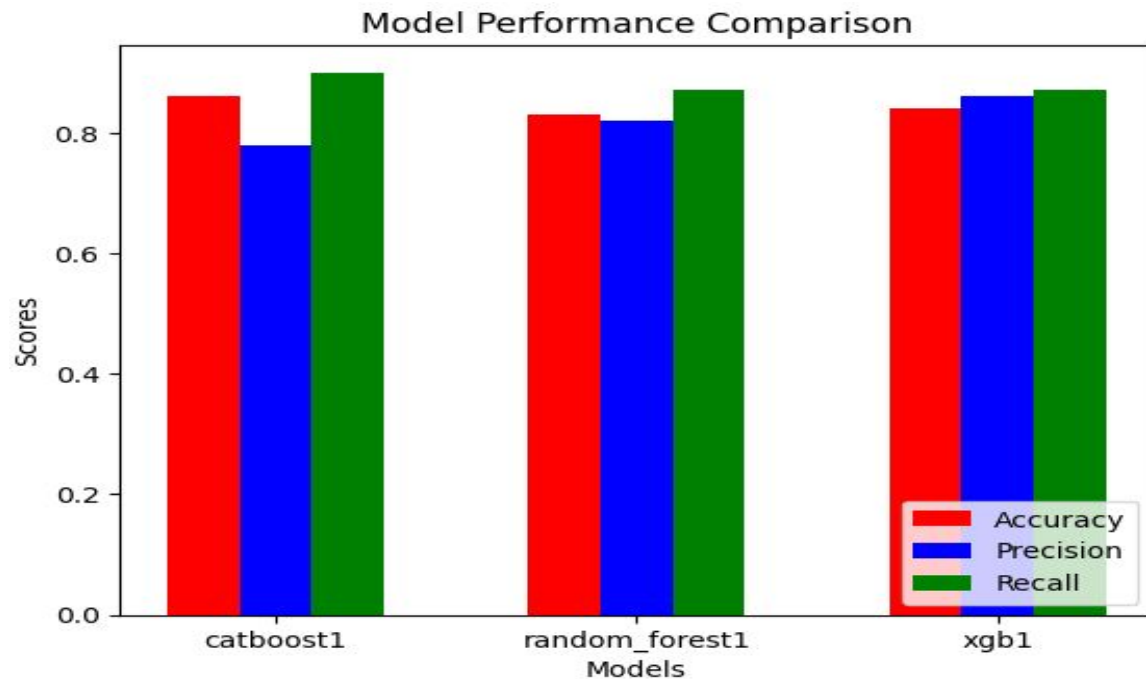
ROC Curves of Classifiers



Precision-Recall Curves of Classifiers



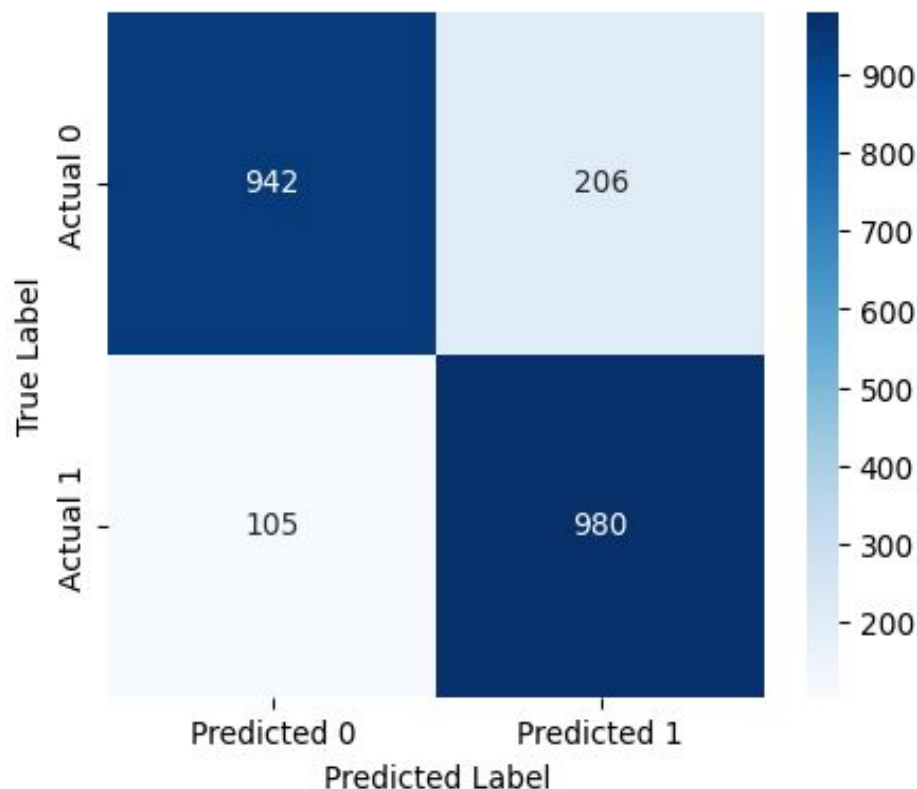
Evaluation on New Train & test



	Model	Accuracy	Precision	Recall	F1 Score	AUC	Specificity
0	Random Forest	83.79	0.81	0.87	0.84	0.84	0.80
1	catBoost	86.07	0.83	0.90	0.86	0.86	0.82
2	XGBoost	84.19	0.82	0.87	0.84	0.84	0.82

Final Result

Confusion Matrix CatBoostClassifier



Accuracy: 86.07254814151366
f1 score : 0.8630559225011009
roc_score : 0.8618916488704058
prc_score : 0.793363672344815
Precision: 0.8263069139966274
Recall: 0.9032258064516129
specificity : 0.8205574912891986
Confusion Matrix:
[[942 206]
 [105 980]]

	precision	recall	f1-score	support
0	0.90	0.82	0.86	1148
1	0.83	0.90	0.86	1085
accuracy			0.86	2233
macro avg	0.86	0.86	0.86	2233
weighted avg	0.86	0.86	0.86	2233

Conclusion and Future Enhancement

1. The following variables seem to be the most relevant inputs in predicting the Success rate of bank direct marketing campaign
 - **Duration - call duration**
 - **Pdays - Number of days since last contact**
 - **Month - month of contact**
 - **Age - customer age**
 - **housing - weather customer having housing loan.**
2. A client is more likely to subscribe term deposit if customer talks for more duration. Campaign is more likely to be successful during March, September, December (end of every trimester).
3. **An essential future enhancement** for our project involves **deploying the trained machine learning model** into real-world applications
4. "Incorporating **advanced ensemble techniques, such as stacking and voting**, will enhance our model's accuracy and robustness, serving as a key future enhancement."

Demerits

1. The absence of **'advanced ensemble techniques'** such as **stacking and voting** in the current implementation suggests a possibility for increased accuracy.

Thanks