DATA MINING FINAL PROJECT

Predicting Term Deposit Subscriptions using Bank Marketing Dataset

INTRODUCTION

MARKETING CAMPAIGNS

- Used by companies to help promote products to current and future clients.

- Helps companies that lose sales due to major negative press often use marketing campaigns to rehabilitate their image.[1]

TERM DEPOSIT

- Type of account held at a financial institution where money is locked up for some set period of time, that pays the depositor compensation in the form of interest on the account balance [2]

DATA SOURCE DETAILS

DATA SOURCE DETAILS

- Extracted from Kaggle under the name **Bank Marketing Dataset.**
- **17** features
- **11,162** observations

EXPLORATORY DATA ANALYSIS (EDA)

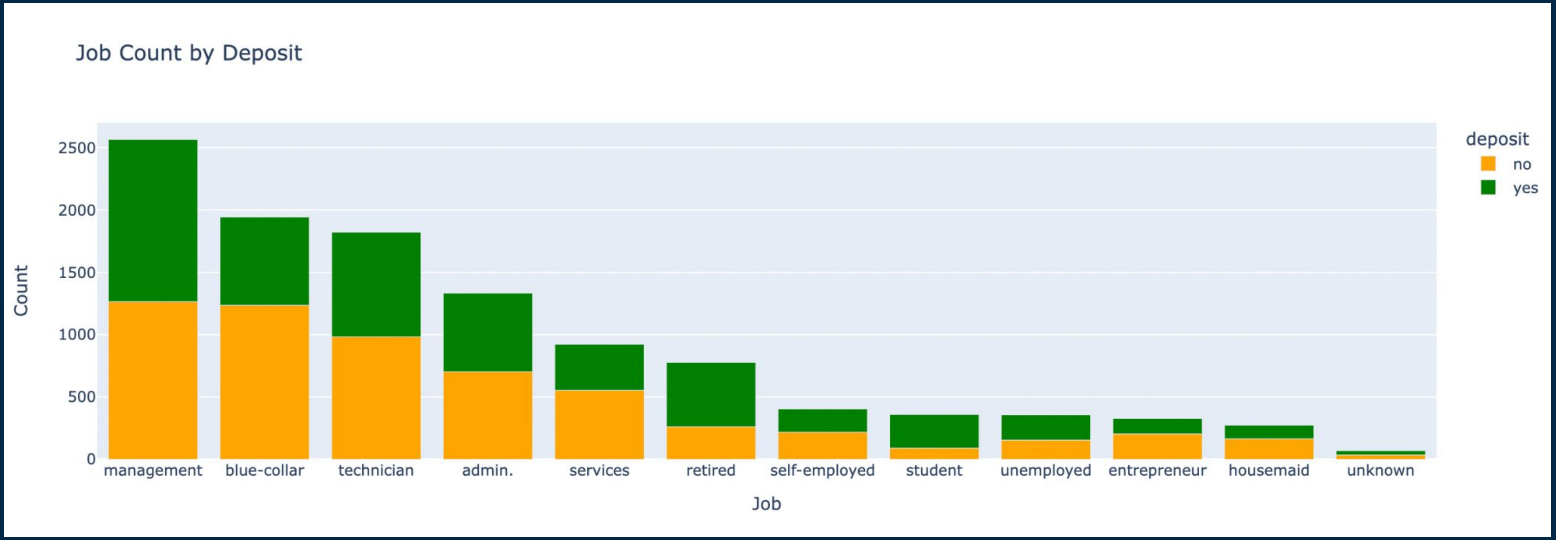
EDA

1. Check for nulls
2. Check for the unique values (categorical features)
3. Visualized how the objective variable behaves
4. Plotted pairwise relationships in a dataset of numerical features
5. Checked the correlation between numerical features
6. Check if dataset is balanced.

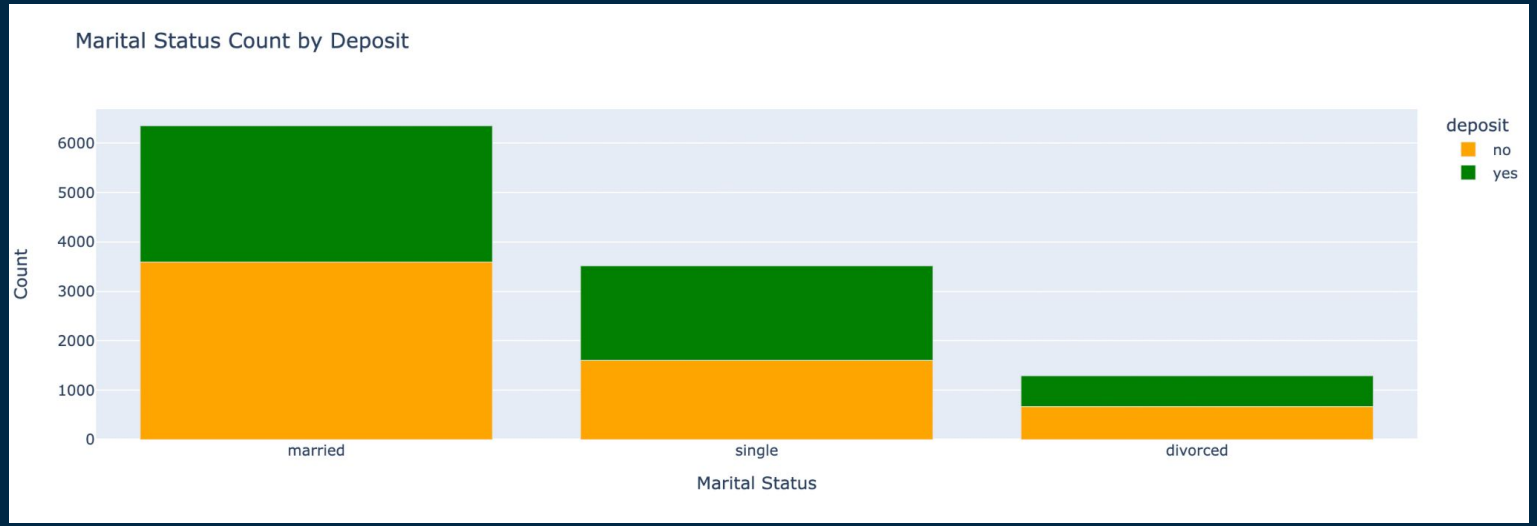
0

NULL VALUES

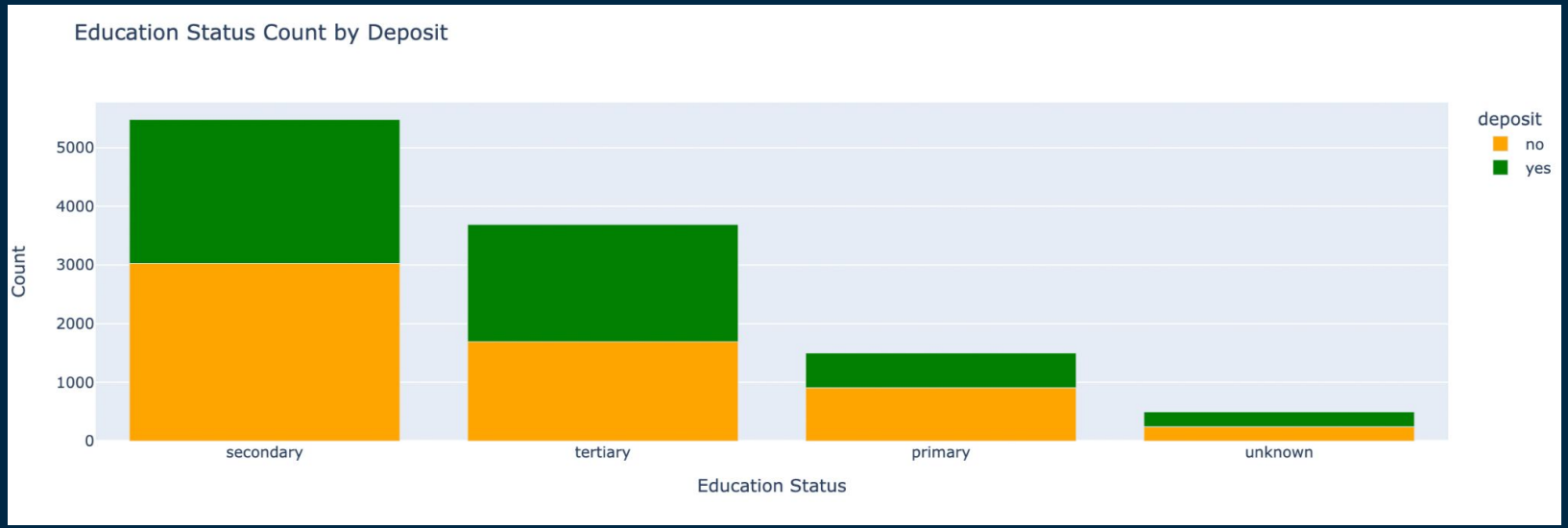
UNIQUE VALUES



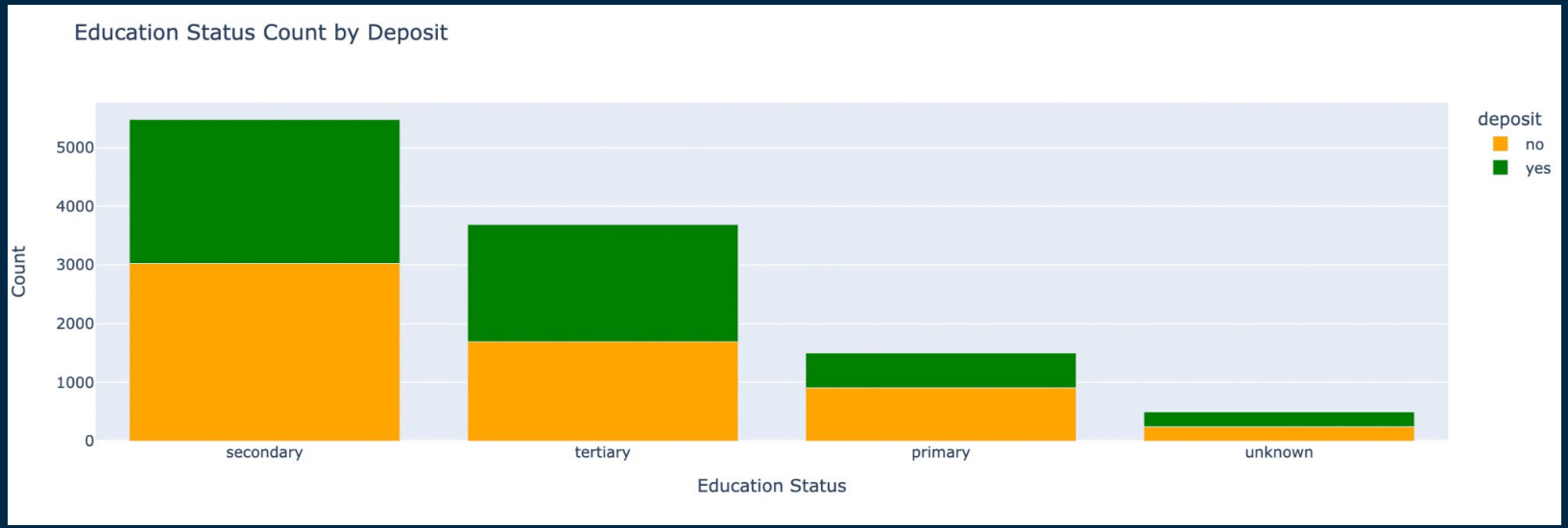
UNIQUE VALUES



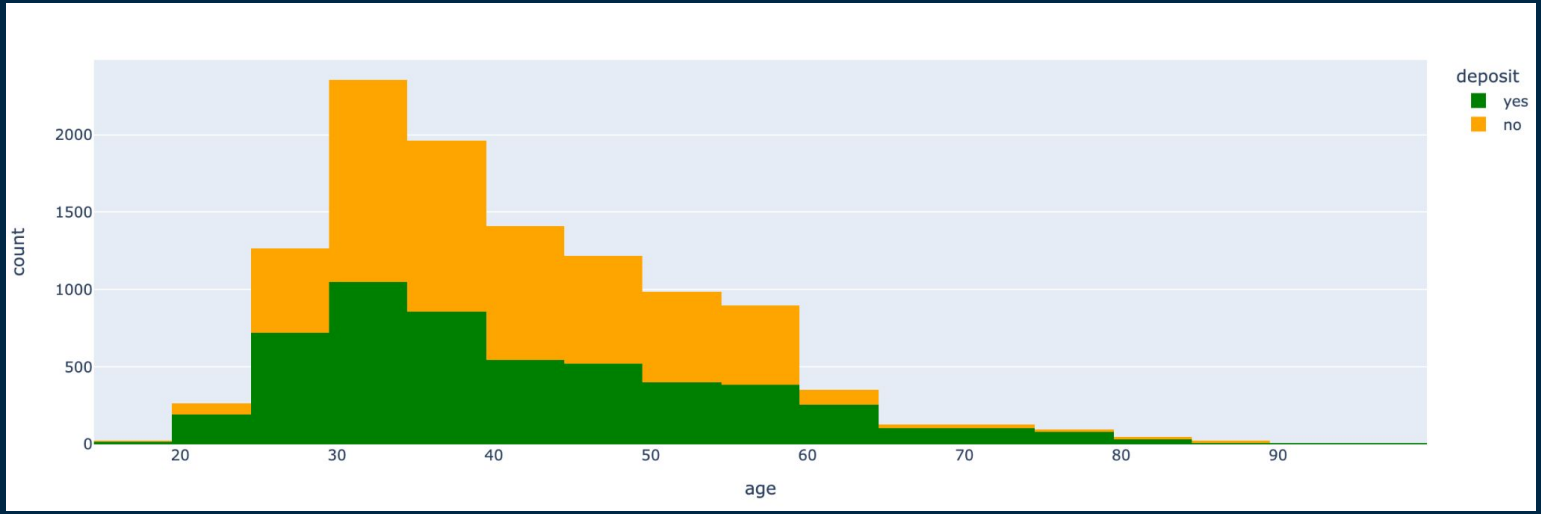
UNIQUE VALUES



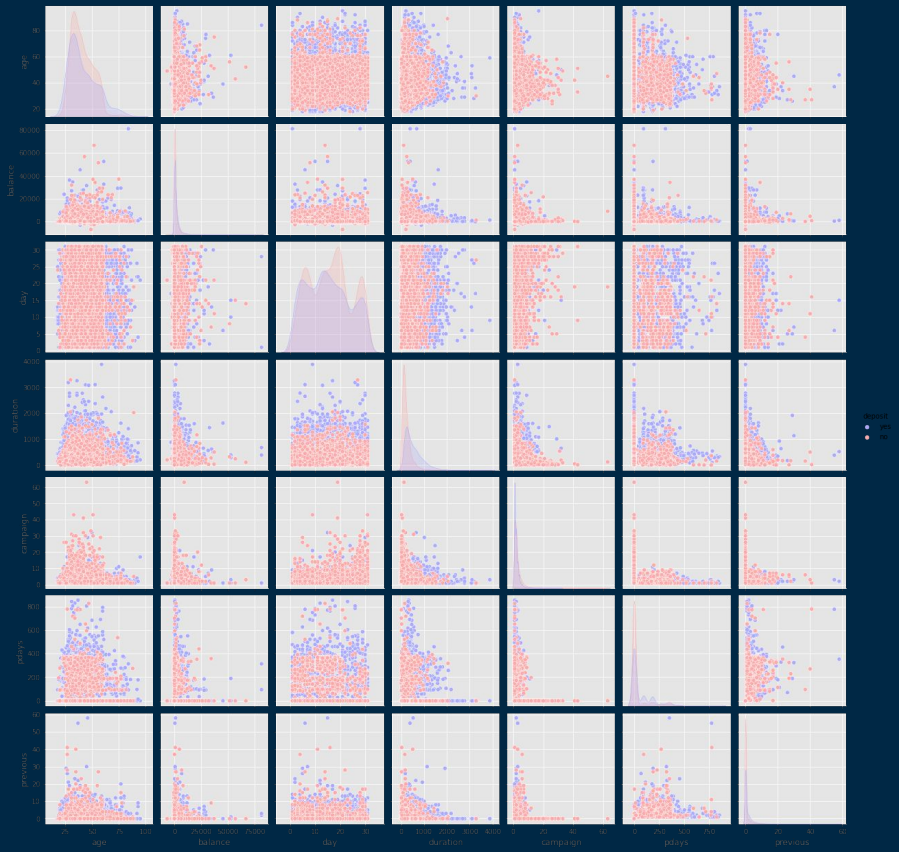
UNIQUE VALUES



UNIQUE VALUES



BEHAVIOUR OF Y vs. FEATURES



CORRELATION OF NUMERICAL FEATURES

BALANCE IN DATASET

```
no      0.52616
yes     0.47384
Name: deposit, dtype: float64
```

52.6% vs 47.4%

BALANCE IN DATASET

NO vs YES

DATA MODELING

DATA MODELING

1. Split the dataset into two.
2. Scaled continuous features.
3. Imputed values
4. Encoded categorical features.
5. Selected a model based on the results of step 3 and 4.
6. Balanced the data
7. Hyperparameter tuning
8. Feature selection / relevant features by models.

ALGORITHMS

1. K-nearest neighbors (k-NN)
2. Logistic Regression*
3. Naive Bayes
4. Decision Tree Classifier
5. Random Forest
6. Ensemble Model applying a hard-voting (mode of all predicted results), using the 4 models previously mentioned.

SPLIT DATA INTO TWO

80% – 20%

TRAIN VS. TEST

SPLIT DATA INTO TWO

80%

TRAIN DATA

VS

20%

TEST DATA

SCALING VALUES

SCALING VALUES

We used Min-Max approach to scale the continues values

IMPUTING MISSING VALUES

EXPERIMENT TIME

IMPUTING MISSING VALUES

- Using Mode
- Using k-NN
- Using Decision Trees
- Leaving as it is, with *"unknowns"* values.

ENCODING CATEGORICAL FEATURES

ENCODING CATEGORICAL FEATURES

- Used ONE HOT ENCODING.
- Method of converting data to prepare it for an algorithm and get a better prediction.
- Each categorical value is a new categorical column and assign a binary value of 1 or 0 to those columns

SELECT BEST FEATURE TREATMENT SETUP

BEST FEATURE TREATMENT SETUP

	F1	Accuracy	Precision	Recall
Decision Tree: Most Frequent	0.761255	0.775786	0.764355	0.758939
Ensemble (Voting): Most Frequent	0.800575	0.818457	0.830341	0.773493
KNN: Most Frequent	0.678978	0.719565	0.737578	0.629770
Linear Regressor: Most Frequent	0.793840	0.814088	0.831460	0.759832
Naive Bayes: Most Frequent	0.633771	0.714974	0.803452	0.523609
Random Forest: Most Frequent	0.842141	0.846230	0.816125	0.870197

BEST FEATURE TREATMENT SETUP

	F1	Accuracy	Precision	Recall
Decision Tree: KNN Imputer	0.774207	0.787996	0.777173	0.771694
Ensemble (Voting): KNN Imputer	0.815218	0.829767	0.834532	0.797159
KNN: KNN Imputer	0.687717	0.726396	0.744246	0.639743
Linear Regressor: KNN Imputer	0.807612	0.824617	0.835827	0.781622
Naive Bayes: KNN Imputer	0.639412	0.715309	0.793030	0.535944
Random Forest: KNN Imputer	0.848982	0.852615	0.821129	0.879115

BEST FEATURE TREATMENT SETUP

	F1	Accuracy	Precision	Recall
Decision Tree: Imputed with "unknown": DTC Imputer	0.772827	0.786764	0.776383	0.769809
Ensemble (Voting): Imputed with "unknown": DTC Imputer	0.818402	0.832904	0.838647	0.799415
KNN: Imputed with "unknown": DTC Imputer	0.689664	0.727853	0.745656	0.642109
Linear Regressor: Imputed with "unknown": DTC Imputer	0.808728	0.825737	0.837674	0.782139
Naive Bayes: Imputed with "unknown": DTC Imputer	0.639770	0.715869	0.794878	0.535713
Random Forest: Imputed with "unknown": DTC Imputer	0.850859	0.854518	0.823509	0.880620

BEST FEATURE TREATMENT SETUP

	F1	Accuracy	Precision	Recall
Decision Tree: Imputed with "unknown"	0.773440	0.787100	0.775858	0.771735
Ensemble (Voting): Imputed with "unknown"	0.819004	0.832792	0.835704	0.803237
KNN: Imputed with "unknown"	0.698800	0.736252	0.756935	0.649497
Linear Regressor: Imputed with "unknown"	0.808444	0.825288	0.836523	0.782618
Naive Bayes: Imputed with "unknown"	0.643700	0.716429	0.789339	0.543808
Random Forest: Imputed with "unknown"	0.851258	0.854967	0.823960	0.880846

SUMMARY

	Imputed with "unkown"	Imputed with Most Frequent	KNN Imputer	DTC Imputer
F1	0.765774	0.751760	0.762191	0.763375
Accuracy	0.792138	0.781517	0.789450	0.790607
Precision	0.803053	0.797218	0.800990	0.802791
Recall	0.738624	0.719307	0.734213	0.734967

BALANCING DATA

BALANCING DATA

- Under-sampling
- Over-sampling
- Synthetic Minority Over-sampling Technique (SMOTE)

RESULTS WITH UNDERSAMPLING

	F1	Accuracy	Precision	Recall
Decision Tree	0.780050	0.791465	0.775836	0.784886
Ensemble (Voting)	0.812453	0.827865	0.834792	0.791626
KNN	0.708958	0.738156	0.744794	0.676902
Linear Regressor	0.814966	0.827753	0.825739	0.804720
Naive Bayes	0.646255	0.717661	0.789275	0.547570
Random Forest	0.850874	0.852615	0.813728	0.892026

RESULTS WITH OVERSAMPLING

	F1	Accuracy	Precision	Recall
Decision Tree	0.772003	0.787434	0.780218	0.764311
Ensemble (Voting)	0.810891	0.827640	0.840223	0.783769
KNN	0.705123	0.735356	0.742655	0.671744
Linear Regressor	0.818038	0.830328	0.827281	0.809225
Naive Bayes	0.648033	0.718556	0.789359	0.550014
Random Forest	0.848854	0.851943	0.818085	0.882515

RESULTS WITH SMOTE

	F1	Accuracy	Precision	Recall
Decision Tree	0.777913	0.790796	0.778508	0.777521
Ensemble (Voting)	0.815231	0.831000	0.840970	0.791502
KNN	0.705974	0.731996	0.731019	0.683092
Linear Regressor	0.817395	0.829880	0.827458	0.807834
Naive Bayes	0.652638	0.720684	0.788639	0.557099
Random Forest	0.852412	0.855415	0.821878	0.885632

FINAL RESULTS

	Base Model	Undersampling	Oversampling	SMOTE
F1	0.762191	0.768926	0.767157	0.770261
Accuracy	0.789450	0.792586	0.791876	0.793295
Precision	0.800990	0.797361	0.799637	0.798079
Recall	0.734213	0.749622	0.743596	0.750447

HYPERPARAMETER TUNING

HYPERPARAMETER TUNING

- Hyperparameters are the variables that govern the training process itself.
- These are tuned by running your whole training job, looking at the aggregate metrics, and adjusting.

HYPERPARAMETER TUNING

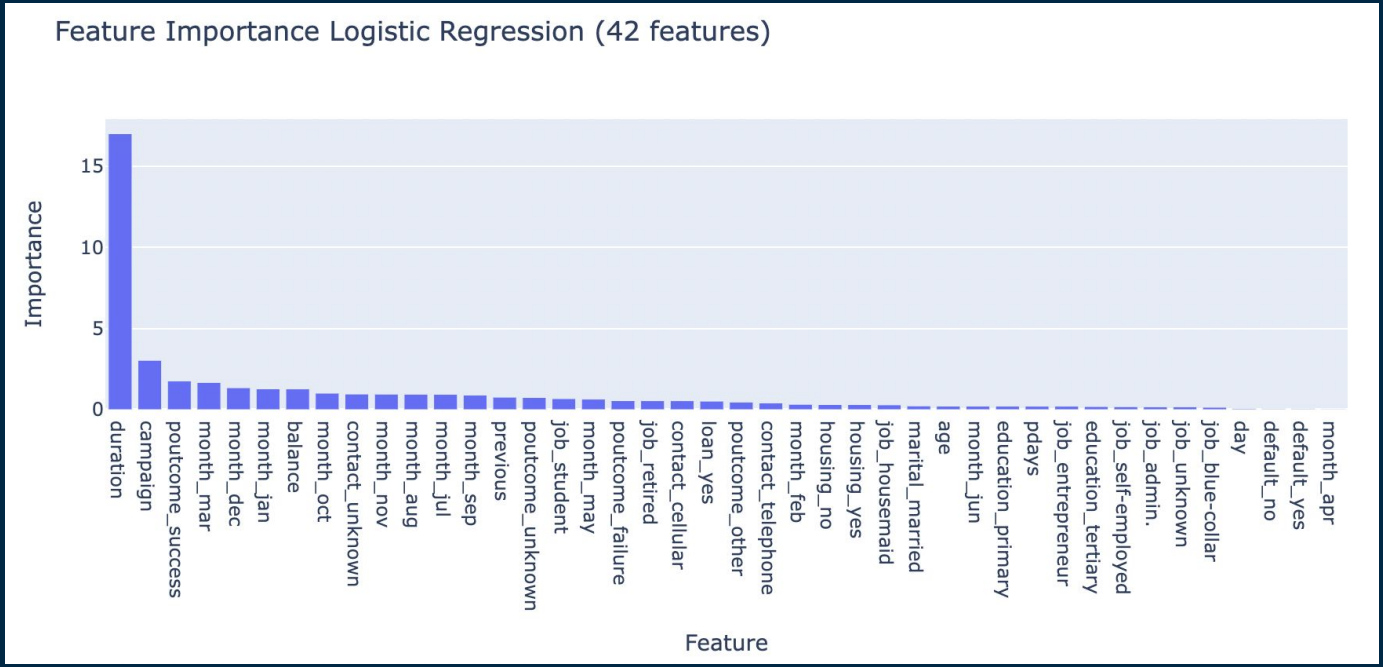
```
Parameters KNN:
n_neighbors='9',

Parameters NB:
var_smoothing='1e-05',

Parameters LR:
penalty='none',

Parameters DTC:
criterion='entropy',
max_depth='10',
min_samples_leaf='2',
min_samples_split='5',
splitter='random',

Parameters RF:
criterion='gini',
n_estimators='800',
```

FEATURE TUNING

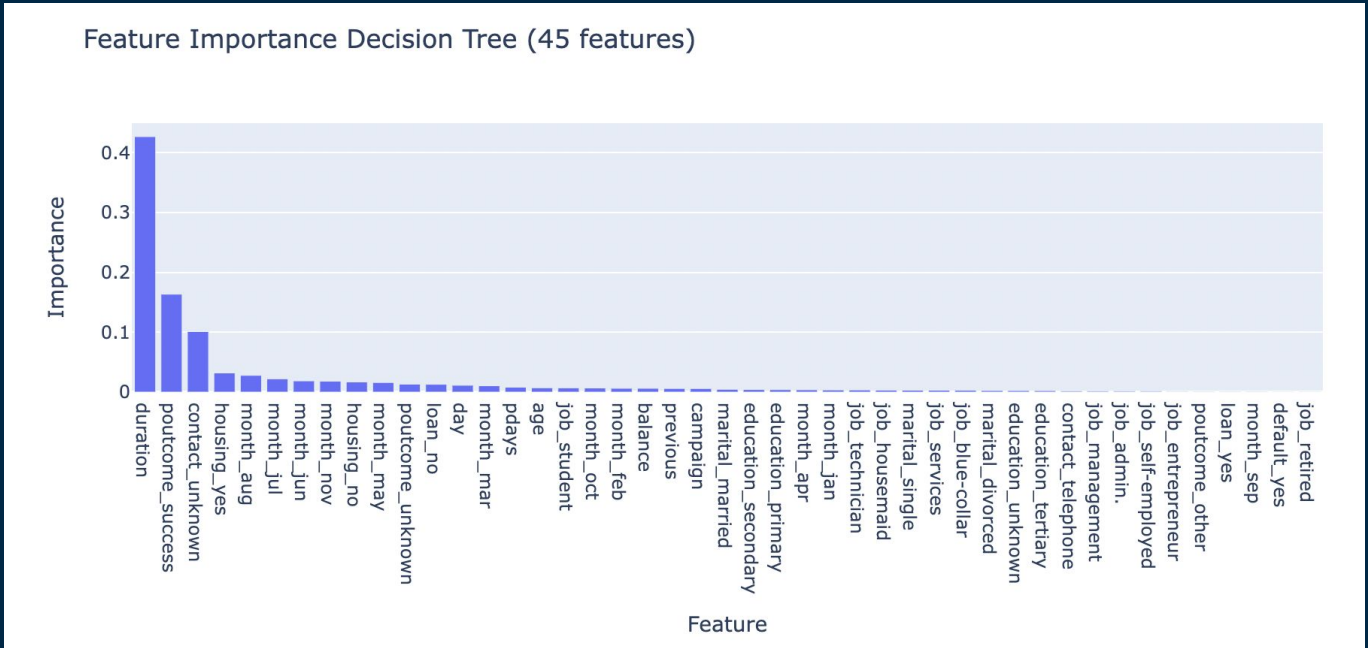
FEATURE TUNING

- Sequential Forward Search
  - k-NN and for Naives Bayes
- Filter Method (Recursively)
  - Logistic Regression
  - Decision Tree
  - Random Forest

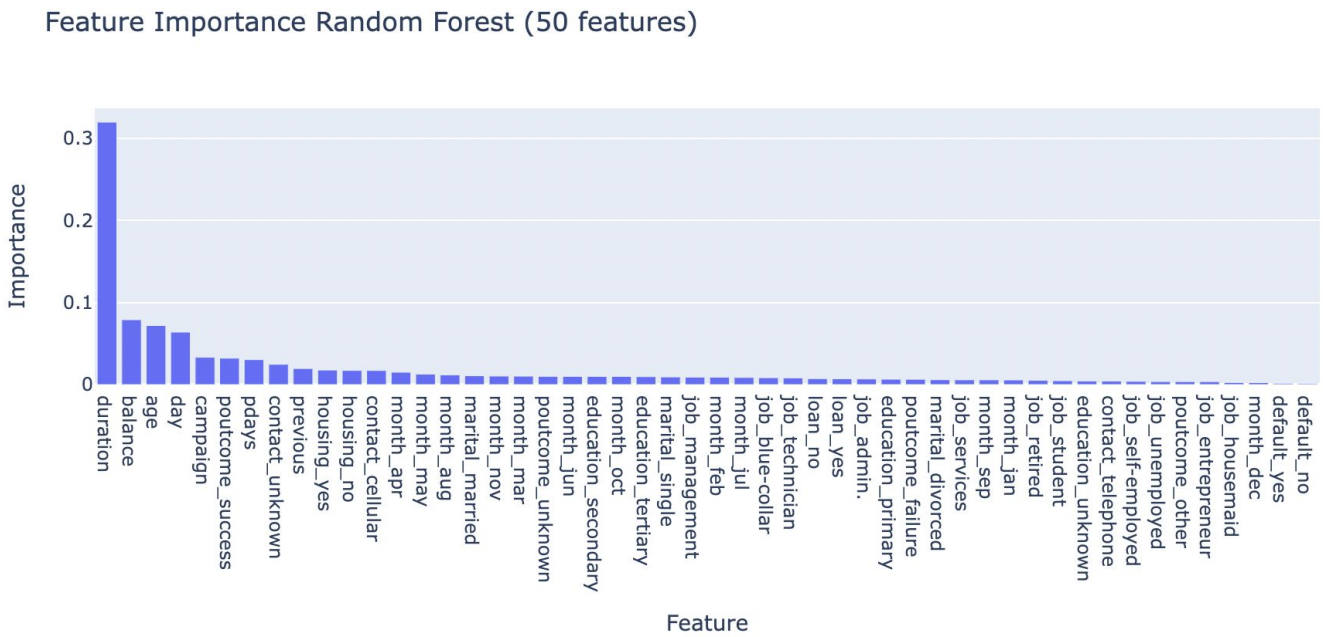
FEATURE TUNING



FEATURE TUNING



FEATURE TUNING



METRICS ON TEST

METRICS

	F1	Accuracy	Precision	Recall
Random Forest	0.855739	0.855352	0.816709	0.898687
Random Forest_HyperParameter_Features_Selected	0.853571	0.853112	0.814310	0.896811
Random Forest_HyperParameter	0.852166	0.851769	0.813299	0.894934
KNN_HyperParameter_Features_Selected	0.852022	0.855799	0.835135	0.869606
Ensemble (Voting)_HyperParameter	0.831084	0.837438	0.824561	0.837711
Ensemble (Voting)	0.827032	0.836095	0.833333	0.820826
Ensemble (Voting)_HyperParameter_Features_Selected	0.824684	0.832064	0.821994	0.827392
Linear Regressor_HyperParameter_Features_Selected	0.820804	0.830273	0.827455	0.814259
Decision Tree_HyperParameter	0.820767	0.822212	0.791123	0.852720
Linear Regressor_HyperParameter	0.819563	0.829825	0.829808	0.809568
Linear Regressor	0.819563	0.829825	0.829808	0.809568
Decision Tree_HyperParameter_Features_Selected	0.817633	0.824004	0.808999	0.826454
Decision Tree	0.784314	0.793103	0.780669	0.787992
Naive Bayes_HyperParameter_Features_Selected	0.776055	0.767129	0.717357	0.845216
KNN	0.703922	0.729512	0.737166	0.673546
KNN_HyperParameter	0.703187	0.733094	0.749469	0.662289
Naive Bayes	0.642417	0.713838	0.796117	0.538462
Naive Bayes_HyperParameter	0.642417	0.713838	0.796117	0.538462

RECOMMENDATIONS & FURTHER STEPS

ANY QUESTIONS?

THANK YOU!!!!