Classification Models

Employee Promotion

, Abdulmajeed Alnfaie
, Nouf Alshabani
, Ahmad Hakami

INTRODUCTION

There are many companies trying to determine which employees are eligible for a promotion by a certain evaluation, and with thousands of employees, this is delaying the transition to new positions. Hence, the company needs to help identify the qualified candidates at a particular checkpoint so that they can speed up the entire promotion cycle.

INSPIRATION

We try predict whether a potential promote at checkpoint in the test set will be promoted or not after the evaluation process.

Methodology
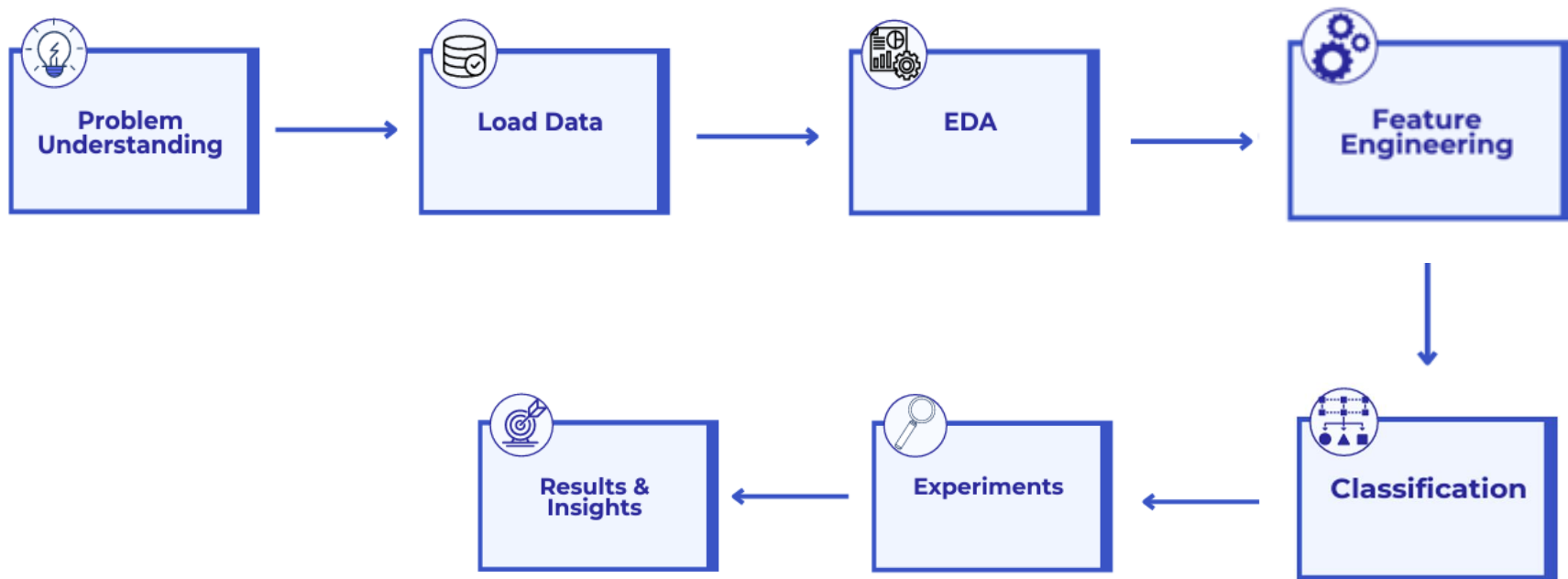
Dataset

Source	**HR Analytics: Employee Promotion Data**
	It was uploaded in Kaggle.com
Records	**54808 Employee Records**
Features	**13 Features**
	id , department, education, gender, lenght_of_service,... etc.
Target	**Predict the eligible candidates for promotion**

Dataset

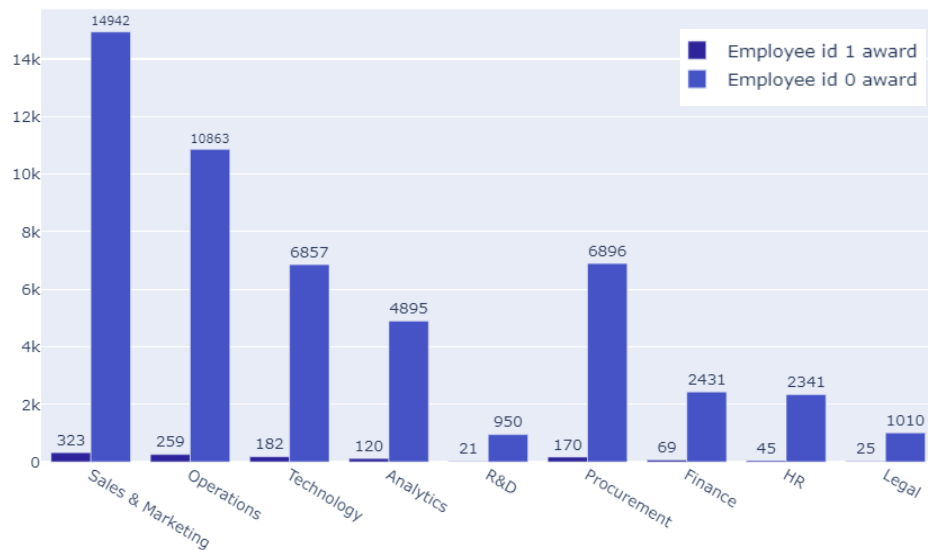| Name of Columns	Describtion	Type
Employee_id	Unique ID for employee	int64
department	Department of employee	object
region	Region of employment (unordered)	object
Education	Education Level	object
Gender	Gender of Employee	object
recruitment_channel	Channel of recruitment for employee	object
no_of_trainings	no of other trainings completed in previous year on soft skills, technical skills etc.	int64
age	Age of Employee	int64
previous_year_rating	Employee Rating for the previous year	float64
length_of_service	Length of service in years	int64
awards_won?	if awards won during previous year then 1 else 0	int64
avg_training_score	Average score in current training evaluations	int64
is_promoted	Recommended for promotion (Target)	int64

Exploratory Data Analysis

Distribution of Employees in Different Departments

The highest number of award by departement
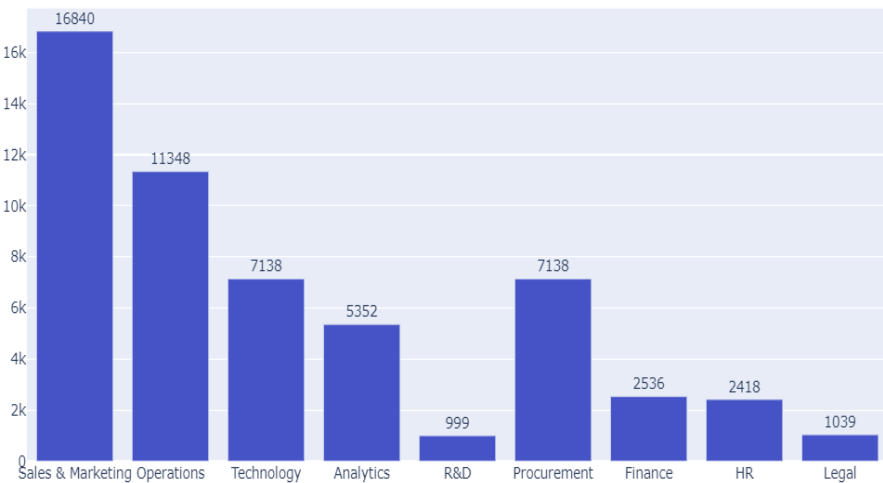
The highest number of department on dataframe is about 16840 in Sales & marketing and the lowest number is about 999 in R&D
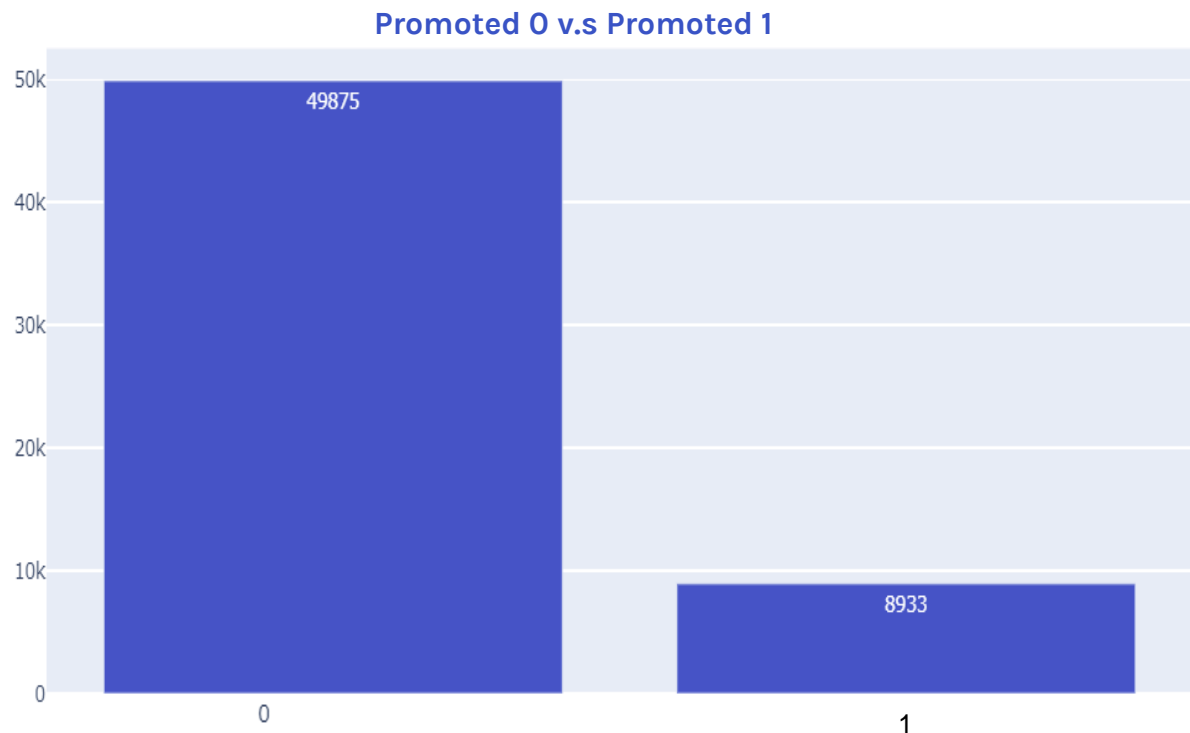
The highest number of award by departement on dataframe is about 323 in Sales & marketing and the lowest number is about 25 in Legal

Features Engineering

Convert Categorical Columns to label Variables

Resulted in decreasing the models' scores.

Data Imbalance

Logistic Regression()

Confusion Matrix
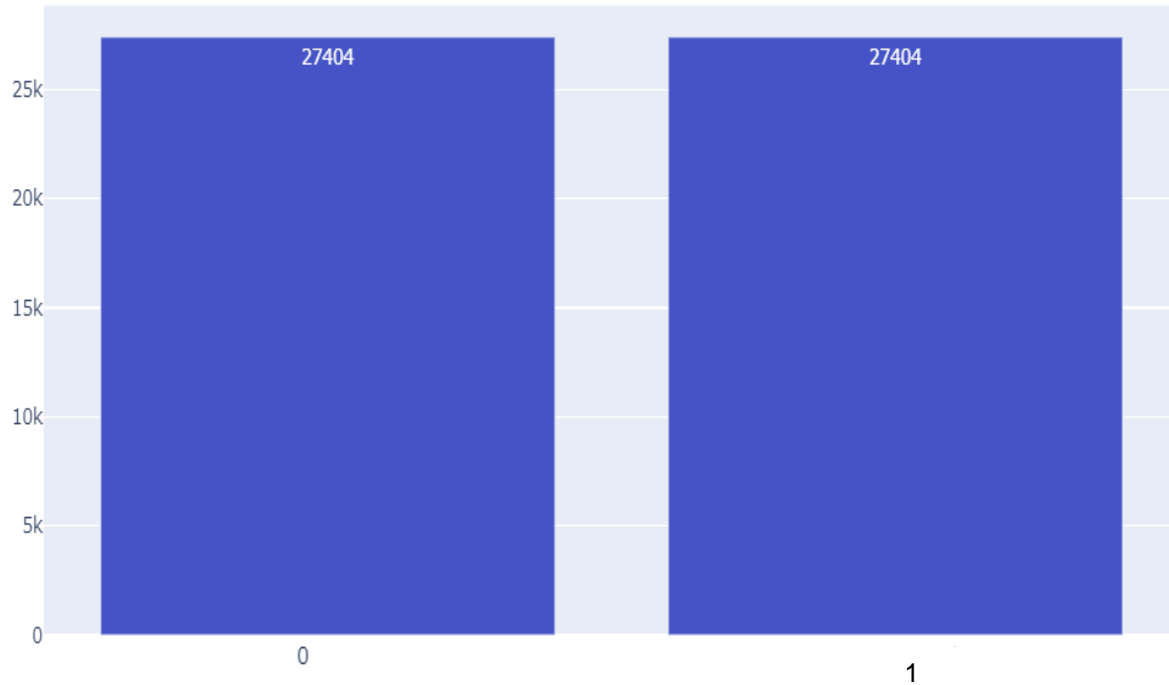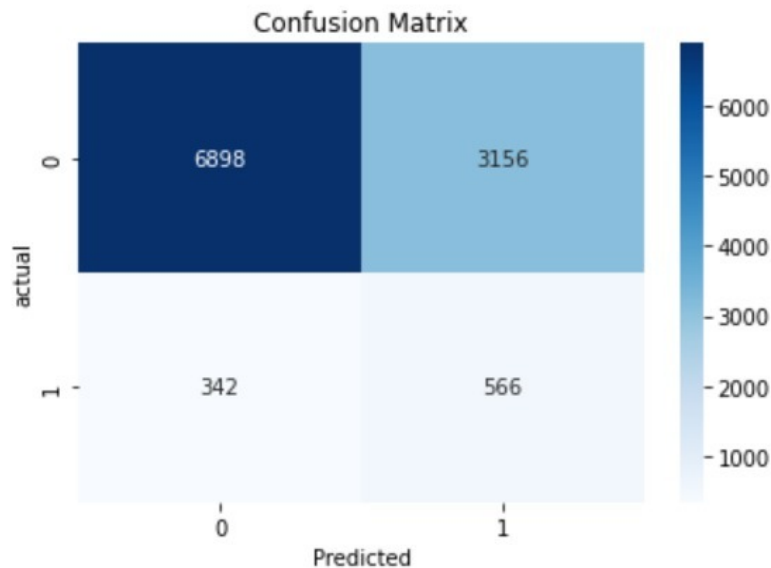
Testing Scores

Accuracy	=	68.08
Precision	=	15.20
✓ Recall	=	62.33
F-1	=	24.44

	precision	recall	f1	support
0	0.95	0.69	0.80	1054
1	0.15	0.62	0.24	908
accuracy			0.68	10962
macro avg	0.55	0.65	0.52	10962
weighted avg	0.89	0.68	0.75	10962

Feature importance

for random forest
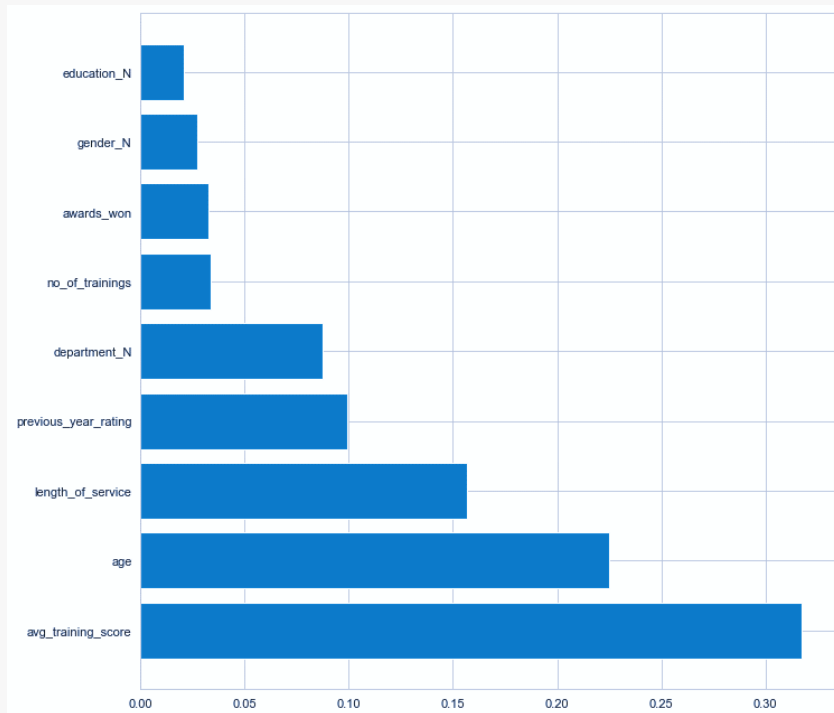
low Feature importance

- Education_N
- Gender_N
- Awards_won

Experiments

Classifier		Accuracy	Precision	Recall	F-1
KNN	Validation	65.31	11.37	46.72	18.29
KNN	Test	64.92	11.48	45.53	18.34
Logistic Regression	Validation	63	12.74	58.89	20.96
Logistic Regression	Test	63.31	13.20	58.10	21.51
Random Forest	Validation	62	12.75	60.39	21
Random Forest	Test	62	13	59.75	21.54
XGBoost	Validation	52.91	11.77	71.87	20.23
XGBoost	Test	53.48	11.96	68.90	20.38
LGBM	Validation	65.96	12.53	51.77	20.18
LGBM	Test	66.33	13.23	52	21
Stacking	Validation	92	85.56	9.52	17.14
Stacking	Test	75	13	32.74	18.53

Split Data

60% train, 20% Validation
20% test

Resampling Technique

Random Over-Sampling

Experiments

| Classifier		Accuracy	Precision	Recall	F-1
KNN	Validation	65.31	11.37	46.72	18.29
	Test	64.92	11.48	45.53	18.34
Logistic Regression	Validation	63	12.74	58.89	20.96
	Test	63.31	13.20	58.10	21.51
Random Forest	Validation	62	12.75	60.39	21
	Test	62	13	59.75	21.54
XGBoost	Validation	52.91	11.77	71.87	20.23
	Test	53.48	11.96	68.90	20.38
LGBM	Validation	65.96	12.53	51.77	20.18
	Test	66.33	13.23	52	21
Stacking	Validation	92	85.56	9.52	17.14
	Test	75	13	32.74	18.53

Split Data

60% train, 20% Validation
20% test

Resampling Technique

Random Over-Sampling

XGBoost (Imbalanced)
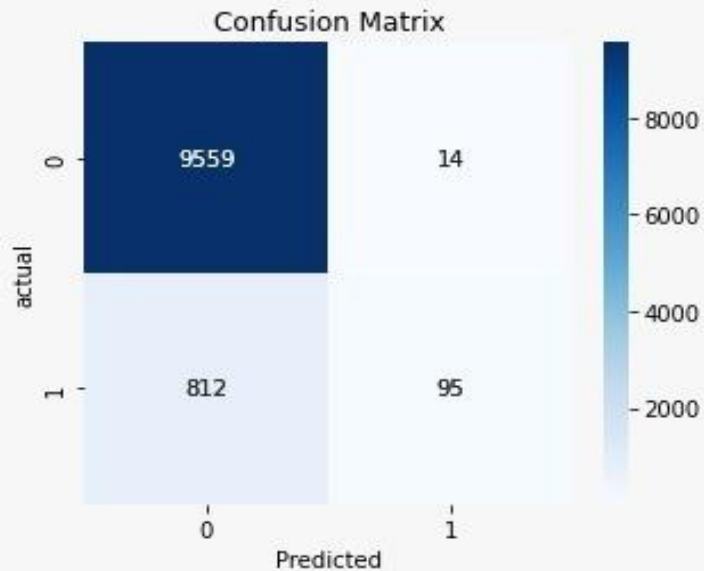
Testing Scores

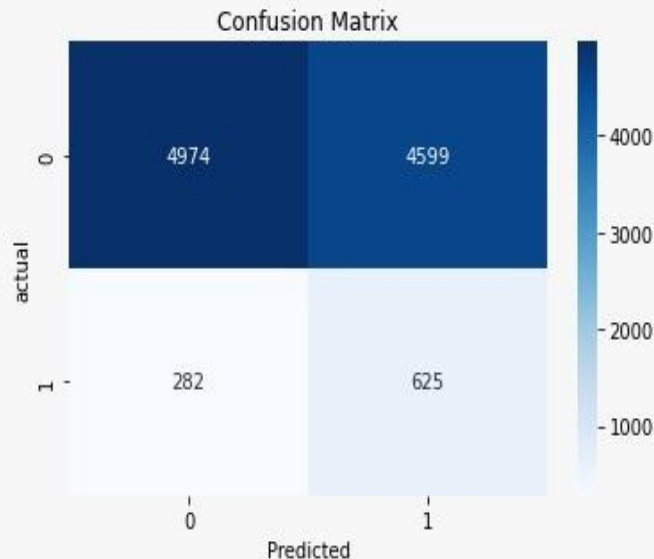|		
Accuracy	=	92.12
Precision	=	87.16
Recall	=	10.5
F-1	=	19

	precision	recall	f1	support
0	0.92	1.00	0.96	9573
1	0.87	0.10	0.19	907
accuracy			0.92	10480
macro avg	0.90	0.55	0.57	10480
weighted avg	0.92	0.92	0.89	10480

XGBoost (balanced)

Confusion Matrix

Testing Scores

Accuracy	=	53.43
Precision	=	12
✓ Recall	=	69
F-1	=	20.4

	precision	recall	f1	support
0	0.95	0.52	0.67	9573
1	0.12	0.69	0.20	907
accuracy			0.53	10480
macro avg	0.53	060	0.44	10480
weighted avg	0.87	0.53	0.63	10480

Conclusion

1. **Classifier Performance Metrics of interest**
   Accuracy, Recall, Precision, F-1

2. **Random over-sampling for Handling Imbalance Data**

3. **XGBoost is the Best Classifier for this dataset**
   With Accuracy = 53.43, precision = 12, recall = 69, F-1 = 20.4

Future work

1. **Correcting errors, if any**

2. **Work on tuning the classifiers more, and try other classifiers**

Tools

Jupyter Notebook

Scikit-learn

Seaborn

Python Programming Language

Plotly

Pandas

Thank You
Dr. Patrick Saoud
For Everything!

Thank You!

Any Question?

| Abdulmajeed | Nouf | Ahmad |

Github @AbdulamjeedAlnefaie

Github @NoufAlshabani

Github @AhmadHakami