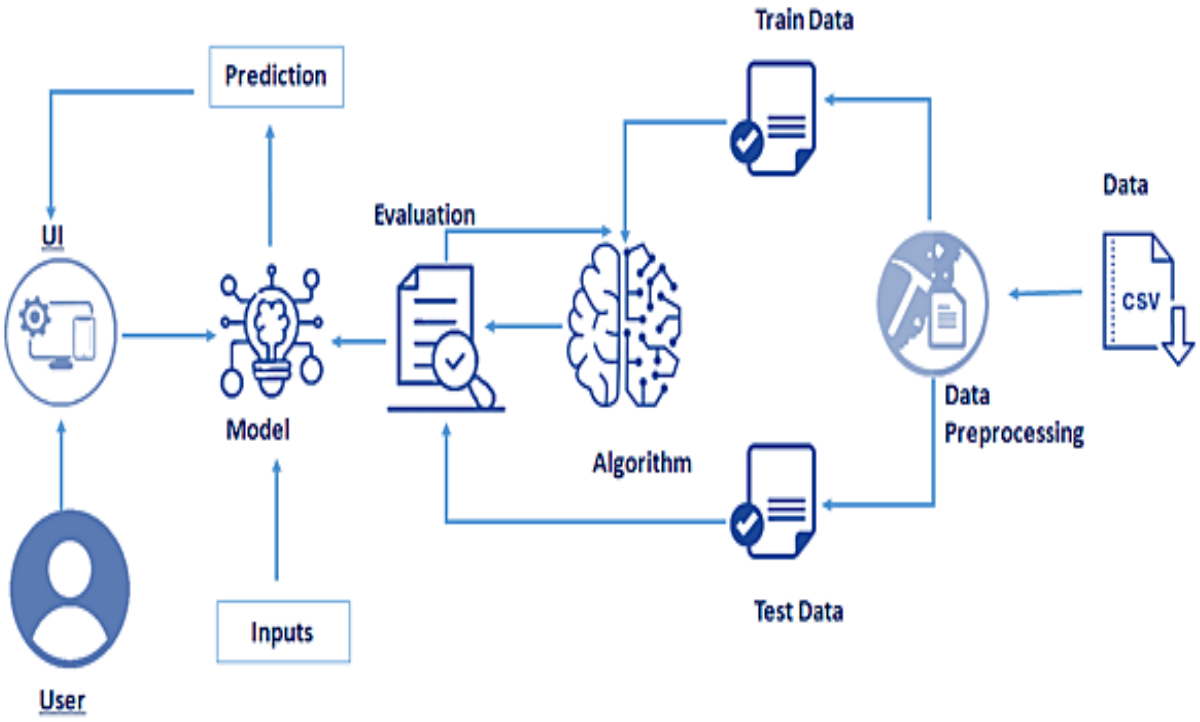
# EMPLOYEE PROMOTION PREDICTION USING IBM WATSON MACHINE LEARNING




Submitted by,

Sithara R

# 1. INTRODUCTION

## 1.1 Overview

Promotion or career advancement is a process through which an employee of a company is given a higher share of duties, a higher pay scale, or both. Promotion is not just beneficial for employees but is also highly crucial for the employer or business owners. It boosts the morale of promoted employees, increases their productivity, and hence improves upon the overall profits earned by the organization. The client is facing a problem in identifying the right people for promotion. The company needs help in identifying the eligible candidates at a particular checkpoint so that they can expedite the entire promotion cycle. This problem can be solved by building a machine learning that automates the process of promoting an employee. we make use of employee datasets to build different classification ML models such as Decision tree, Random forest, KNN, and xgboost. The best model is selected and saved for integration with the flask application. For better training results we make use of IBM to train the model to deploy the model on IBM.

## 1.2 Purpose

The company needs help in identifying the eligible candidates at a particular checkpoint so that they can expedite the entire promotion cycle. This problem can be solved by building a machine learning that automates the process of promoting an employee. we make use of employee datasets to build different classification ML models such as Decision tree, Random forest, KNN, and xgboost. The best model is selected and saved for integration with the flask application. For better training results we make use of IBM to train the model to deploy the model on IBM.

# 2. LITERATURE SURVEY

### 2.1 Existing problem

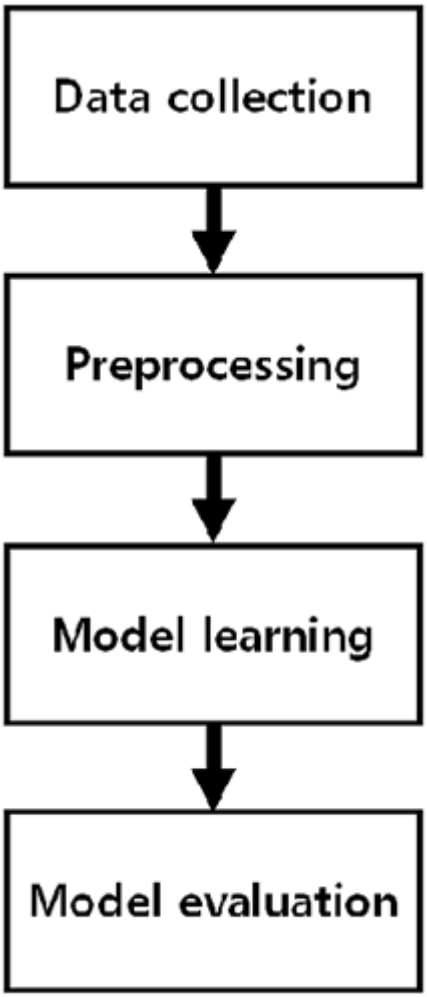
Employee Promotion means the ascension of an employee to higher ranks. It involves an increase in salary, position, responsibilities, status, and benefits. This aspect of the job drives employees the most—the ultimate reward for dedication and loyalty towards an organization. In theory, a promotion requires more work and effort in a job. Based on organizational policies, these promotion-based decisions are taken on different aspects. These can be the **length of service**, experience, seniority, performance, etc.

### 2.2 Proposed solution

The company needs help in identifying the eligible candidates at a particular checkpoint so that they can expedite the entire promotion cycle. This problem can be solved by building a machine learning that automates the process of promoting an employee. we make use of employee datasets to build different classification ML models such as Decision tree, Random forest, KNN, and xgboost.

## 3. THEORITICAL ANALYSIS

3.1 Block diagram




3.2 Hardware / Software requirements

- Processors:Intel® Core™ i5 processor 4300M at 2.60 GHz or 2.59 GHz (1 socket, 2 cores, 2 threads per core), 8 GB of DRAMIntel® Xeon® processor E5-2698 v3 at 2.30 GHz (2 sockets, 16 cores each, 1 thread per core), 64 GB of DRAMIntel® Xeon Phi™ processor 7210 at 1.30 GHz (1 socket, 64 cores, 4 threads per core), 32 GB of DRAM, 16 GB of MCDRAM (flat mode enabled)
- Disk space: 2 to 3 GB
- Operating systems: Windows® 10, macOS*, and Linux*

**Minimum System Requirements**

- Processors: Intel Atom® processor or Intel® Core™ i3 processor
- Disk space: 1 GB
- Operating systems: Windows 8.1
- Python* versions: 3.9

## Software requirements:

### Anaconda navigator:

Anaconda is an open-source distribution for python and R. It iused for data science, machine learning, deep learning, etc. With the availability of more than 300 libraries for data science, it becomes fairly optimal for any programmer to work on anaconda for data science.

### Jupyter notebook:

The *Jupyter Notebook App* is a server-client application that allows editing and running notebook documents via a web browser. The *Jupyter Notebook App* can be executed on a local desktop requiring no internet access (as described in this document) or can be installed on a remote server and accessed through the internet.

# 4 EXPERIMENTAL INVESTIGATIONS

### Decision Tree Model

A function named decisionTree is created and train and test data are passed as the parameters. Inside the function, DecisionTreeClassifier algorithm is initialized and training data is passed to the model with .fit() function. Test data is predicted with .predict() function and saved in new variable. For evaluating the model, confusion matrix and classification report is done.

### Random Forest Model

A function named randomForest is created and train and test data are passed as the parameters. Inside the function, RandomForestClassifier algorithm is initialized and training data is passed to the model with .fit() function. Test data is predicted with .predict() function and saved in new variable. For evaluating the model, confusion matrix and classification report is done.

### KNN Model

A function named KNN is created and train and test data are passed as the parameters. Inside the function, KNeighborsClassifier algorithm is initialized and training data is passed to the model with .fit() function. Test data is predicted with .predict() function and saved in new variable. For evaluating the model, confusion matrix and classification report is done.

### Xgboost Model

A function named xgboost is created and train and test data are passed as the parameters. Inside the function, GradientBoostingClassifier algorithm is initialized and training data is passed to the model with .fit() function. Test data is predicted with .predict() function and saved in new variable. For evaluating the model, confusion matrix and classification report is done.

**Compare The Model**

For comparing the above four models compareModel function is defined.

After calling the function, the results of models are displayed as output. From the four model random forest and decision tree is performing well. From the below image, we can see the accuracy of the models. Both models have 95% and 93% accuracy. Random forest model accuracy is high. And from confusion matrix random forest has higher number of true positive and true negative. So, here random forest is selected and evaluated with cross validation.

**Evaluating Performance Of The Model And Saving The Model**

From sklearn, cross_val_score is used to evaluate the score of the model. On the parameters, we have given rf (model name), x_resample, y_resample, cv (as 5 folds). Our model is performing well. So, we are saving the model by pickle.dump().

# 5. Flowchart

**Project Flow:**

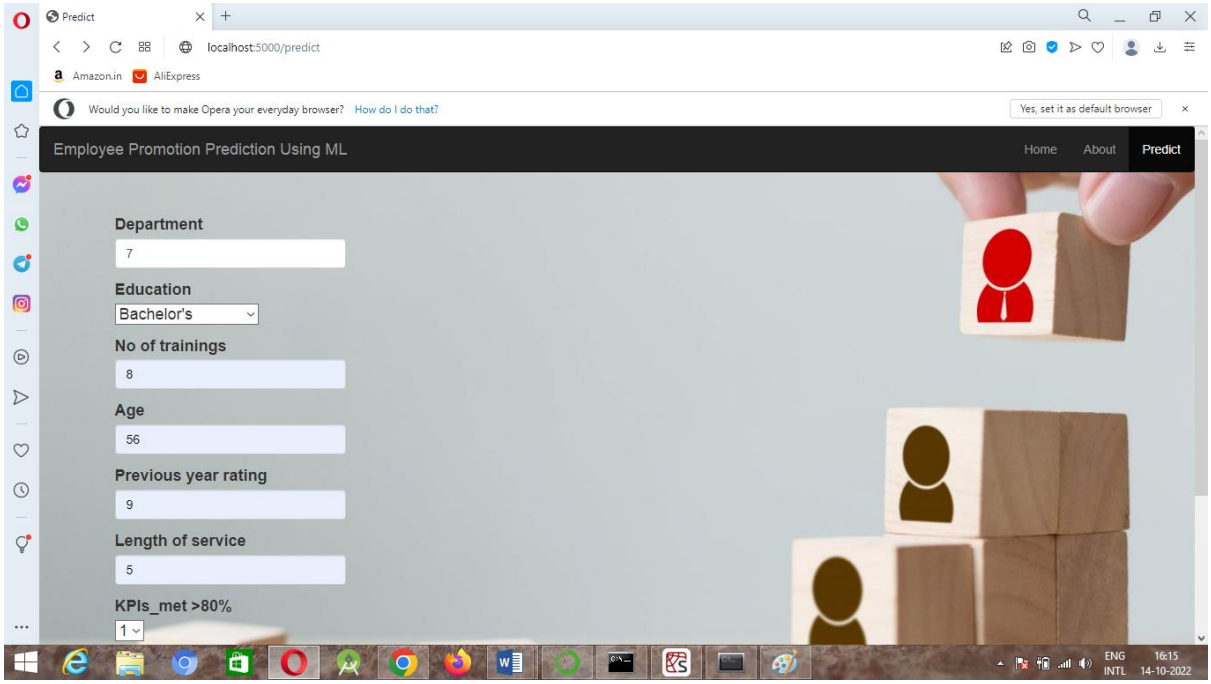
- The user interacts with the UI to enter the input.
- Entered input is analyzed by the model which is integrated.
- Once the model analyses the input the prediction is showcased on the UI

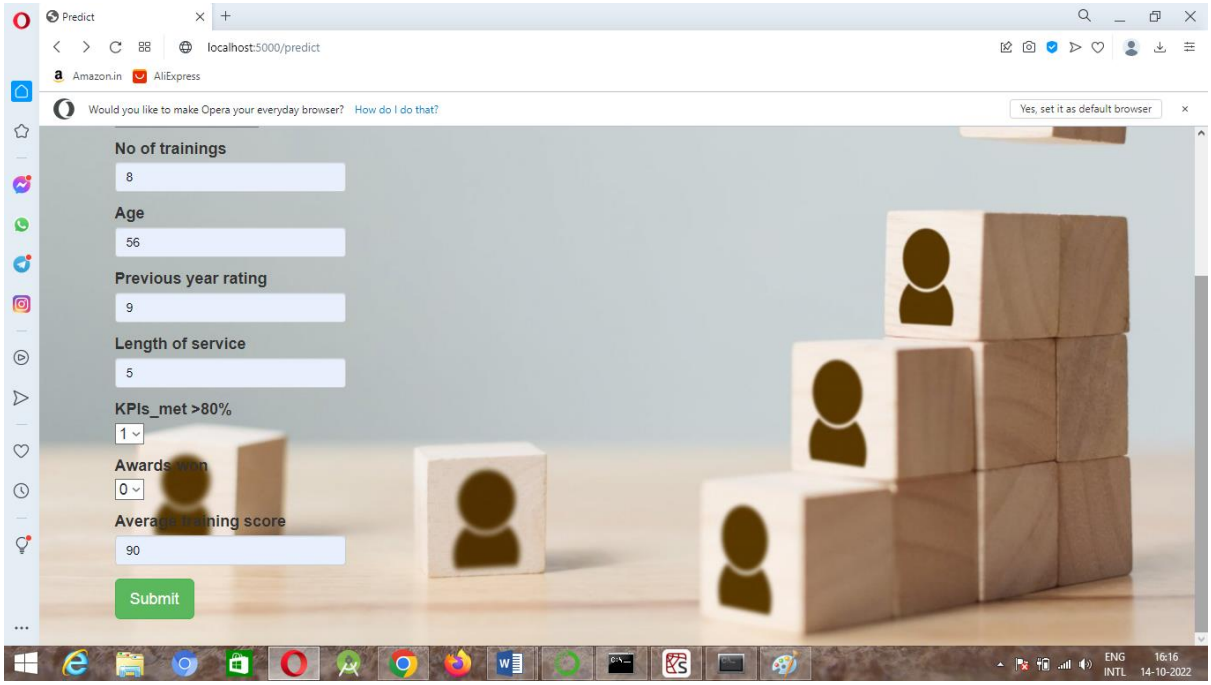
To accomplish this, we have to complete all the activities listed below,

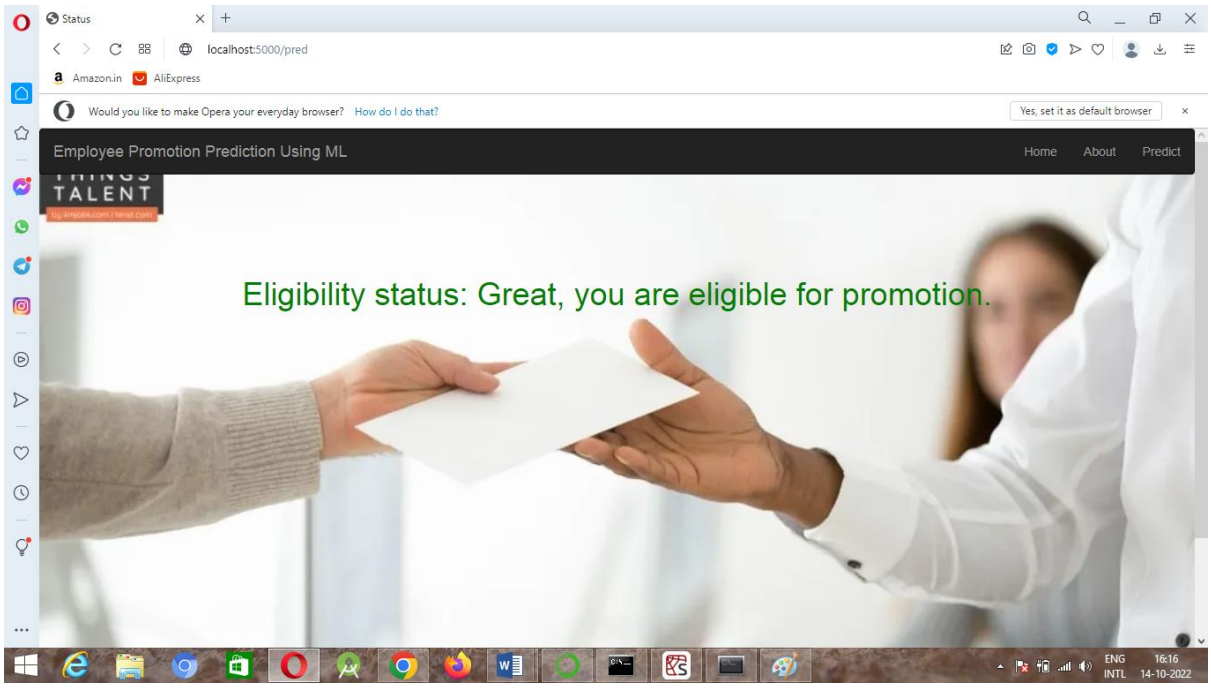
- Data collection
  - Collect the dataset or create the dataset
- Visualizing and analyzing data
  - Univariate analysis
  - Multivariate analysis
  - Descriptive analysis
- Data pre-processing
  - Drop unwanted features
  - Checking for null values
  - Remove negative data
  - Handling outlier
  - Handling categorical data
  - Handling Imbalanced data
  - Splitting data into train and test
- Model building
  - Import the model building libraries
  - Initializing the model
  - Training and testing the model
  - Evaluating performance of the model
  - Save the model
- Application Building
  - Create an HTML file
  - Build python code

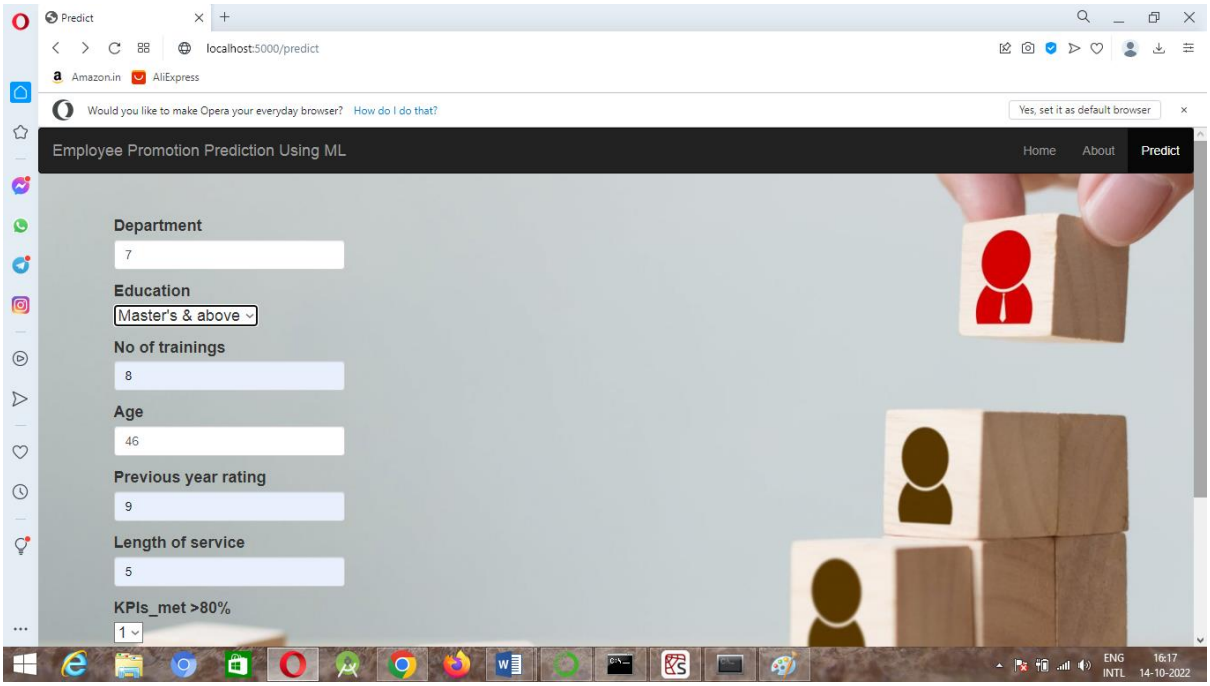
- Run the Application

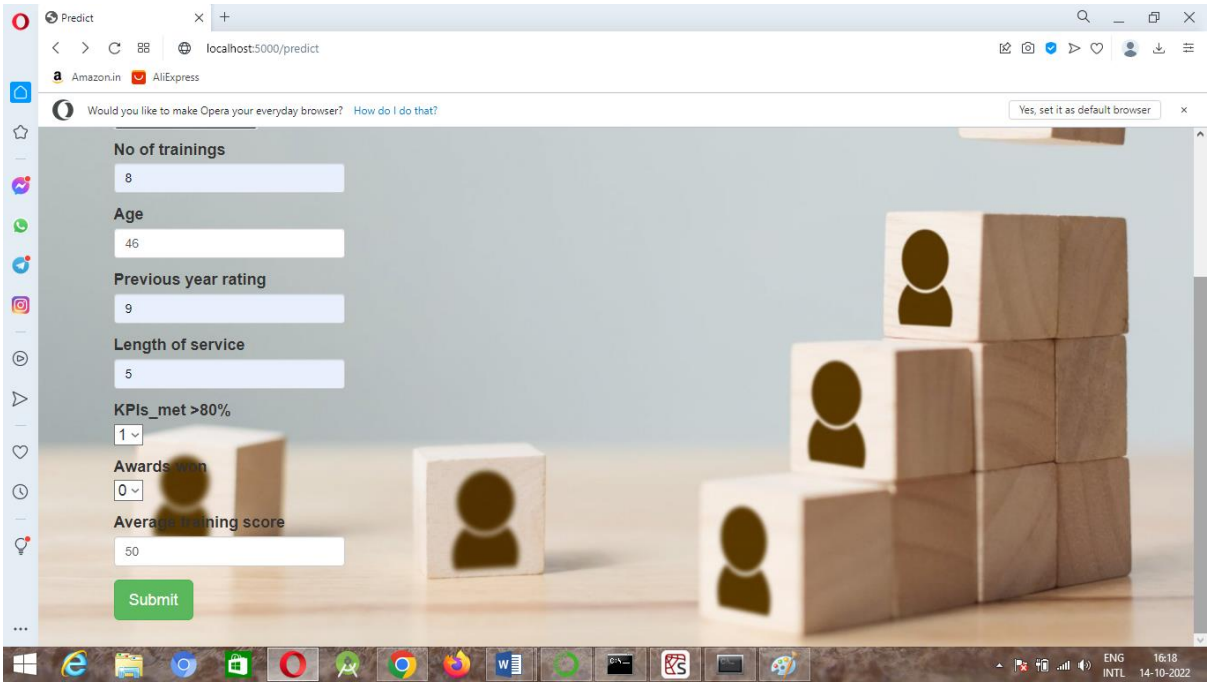
# 6. RESULT

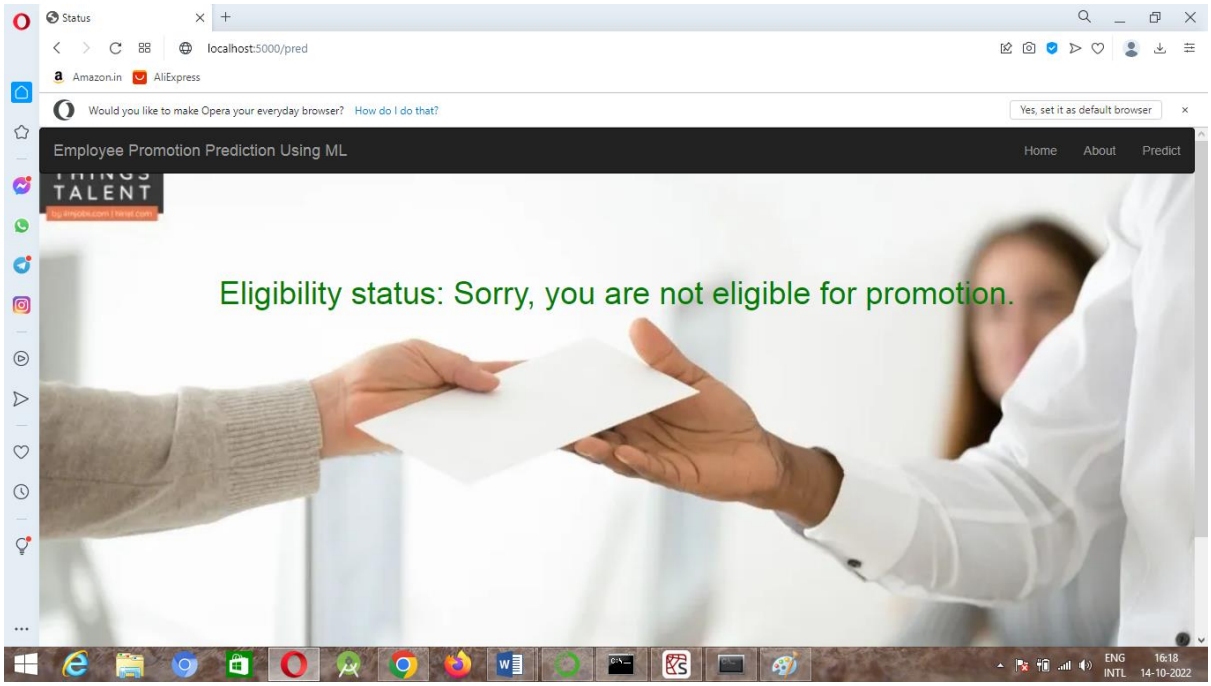


Predict
localhost:5000/predict
Amazon.in
AliExpress
Would you like to make Opera your everyday browser?
How do I do that?
Yes, set it as default browser
Employee Promotion Prediction Using ML
Home
About
Predict
Department
7
Education
Bachelor's
No of trainings
8
Age
56
Previous year rating
9
Length of service
5
KPIs_met >80%
1
ENG
INTL
16:15
14-10-2022


Predict
localhost:5000/predict
Amazon.in
AliExpress
Would you like to make Opera your everyday browser?
How do I do that?
Yes, set it as default browser
No of trainings
8
Age
56
Previous year rating
9
Length of service
5
KPIs_met >80%
1
Awards won
0
Average training score
90
Submit
ENG
INTL
16:16
14-10-2022


Status
localhost:5000/pred
Amazon.in
AliExpress
Would you like to make Opera your everyday browser?
How do I do that?
Yes, set it as default browser
Employee Promotion Prediction Using ML
Home
About
Predict
Eligibility status: Great, you are eligible for promotion.
ENG
INTL
16:16
14-10-2022


Predict
localhost:5000/predict
Amazon.in
AliExpress
Would you like to make Opera your everyday browser?
How do I do that?
Yes, set it as default browser
Employee Promotion Prediction Using ML
Home
About
Predict
Department
7
Education
Master's & above
No of trainings
8
Age
46
Previous year rating
9
Length of service
5
KPIs_met >80%
1
ENG
INTL
16:17
14-10-2022


Predict
localhost:5000/predict
Amazon.in
AliExpress
Would you like to make Opera your everyday browser?
How do I do that?
Yes, set it as default browser
No of trainings
8
Age
46
Previous year rating
9
Length of service
5
KPIs_met >80%
1
Awards won
0
Average training score
50
Submit
ENG
INTL
16:18
14-10-2022


Status
localhost:5000/pred
Amazon.in
AliExpress
Would you like to make Opera your everyday browser?
How do I do that?
Yes, set it as default browser
Employee Promotion Prediction Using ML
Home
About
Predict
Eligibility status: Sorry, you are not eligible for promotion.
ENG
INTL
16:18
14-10-2022

## 7. ADVANTAGES & DISADVANTAGES OF A PROJECT

### ADVANTAGES :

There is an endless number of advantages of ML. We can take a look at the one. which are really helpful.

The advantages of Machine Learning tell us how using ML would benefit us. So, let's have a look at the advantages of Machine Learning

→ **Automation of Everything**

→ **Wide Range of Applications**

→ **Scope of Improvement**

→ **Efficient Handling of Data**

### DISADVANTAGES :

Similar to the advantages of Machine Learning, we should also know the disadvantages of Machine Learning. If you don't know the cons, you won't know the risks of ML. So, let's have a look at these disadvantages:

→ **Possibility of High Error**

→ **Algorithm Selection**

→ **Data Acquisition**

→**Time and Space**

→ **Internet Issues**

## 8. CONCLUSION

From this project we came to a conclusion that the average training score is the key factor that brings the output. When the average training score is 90 or above then the

employee is selected for promotion. And when the training score is below that the employee is not selected for promotion.

## 9. FUTURE SCOPE

This project works very well and in future it will surely help and makes easy for the organization to select the appropriate employee for promotion. It helps to enhance the organization's productivity.

## 10. BIBILOGRAPHY

### 1).Installation of Anaconda Navigator:

**https://www.youtube.com/embed/5mDYijMfSzs**

Anaconda Navigator is a free and open-source distribution of the Python and R programming languages for data science and machine learning-related applications. It can be installed on Windows, Linux, and macOS.Conda is an open-source, cross-platform, package management system. Anaconda comes with great tools like JupyterLab, Jupyter Notebook, QtConsole, Spyder, Glueviz, Orange, Rstudio, Visual Studio Code.

For this project, we will be using a Jupyter notebook and Spyder.

### 2).Python packages

To build Machine learning models you must require the following packages

- Sklearn: Scikit-learn is a library in Python that provides many unsupervised and supervised learning algorithms.
- NumPy: NumPy is a Python package that stands for 'Numerical Python. It is the core library for scientific computing, which contains a powerful n-dimensional array of object
- Pandas: pandas is a fast, powerful, flexible, and easy-to-use open-source data analysis and manipulation tool, built on top of the Python programming language.
- Matplotlib: It provides an object-oriented API for embedding plots into applications using general-purpose GUI toolkits.

### 3). Data collection

There are many popular open sources for collecting the data. Eg: kaggle.com, UCI repository, etc.

In this project, we have used emp_promotion.csv data. This data is downloaded from kaggle.com.

## 4).Flask - Web framework used for building Web applications.

If you are using anaconda navigator, follow the below steps to download the required packages:

Open anaconda prompt as administrator

- Type **"pip install numpy "** and click enter
- Type **"pip install pandas "** and click enter
- Type **"pip install scikit-learn"** and click enter.
- Type **"pip install matplotlib"** and click enter.
- Type **"pip install scipy"** and click enter.
- Type **"pip install pickle-mixin"** and click enter.
- Type **"pip install seaborn"** and click enter.
- Type **"pip install Flask "** and click enter.

## 5).Register For IBM Cloud

## IBM Account:

- Please click here to register for IBM
- Please click here to log in to IBM Account

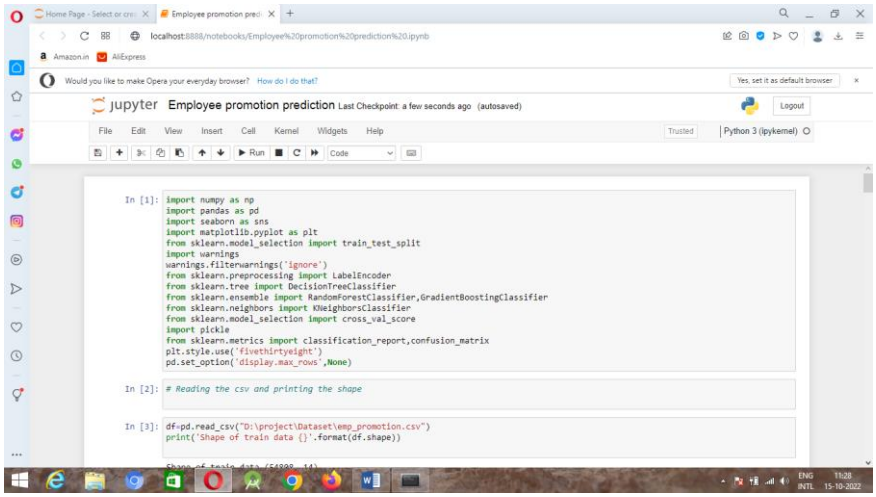
## 6). Train The ML Model On IBM

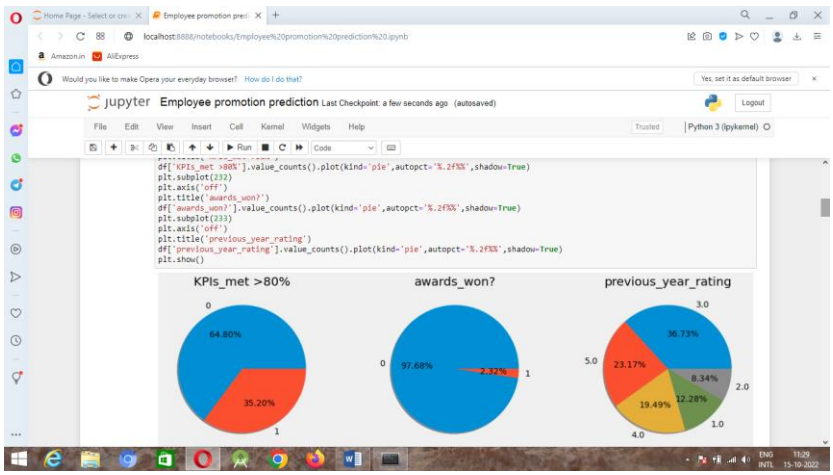
## 7). Integrate Flask With Scoring End Point

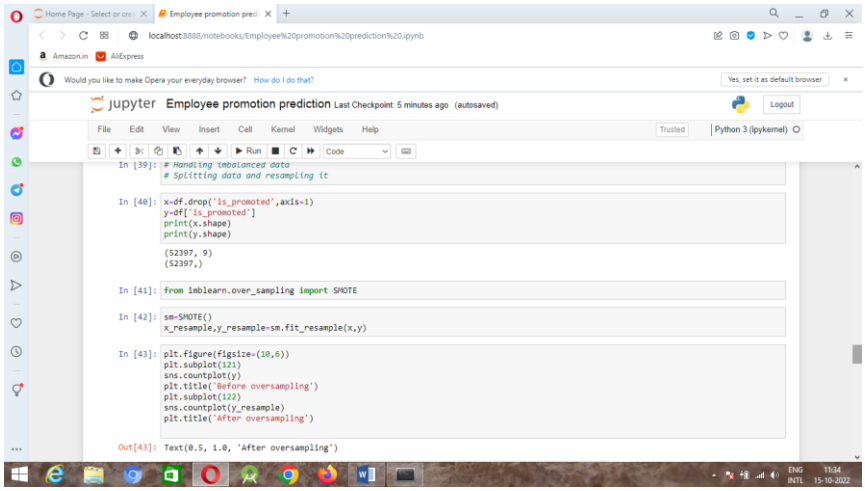
## APPENDIX

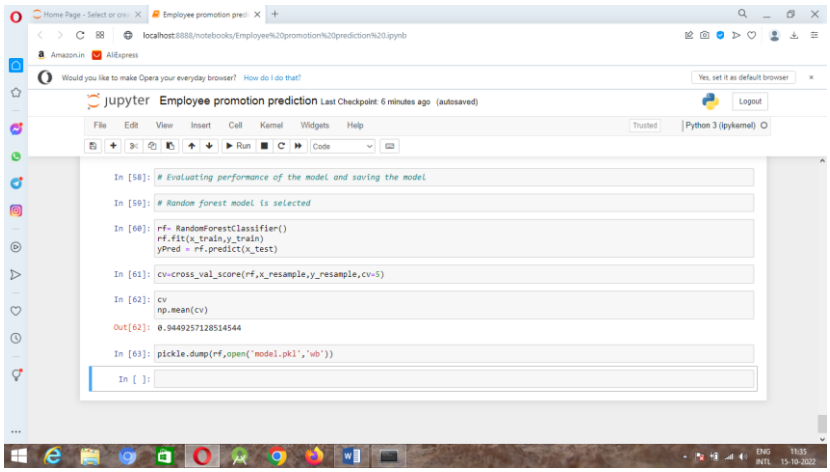
### A. Source code

Employee prediction promotion.pynb

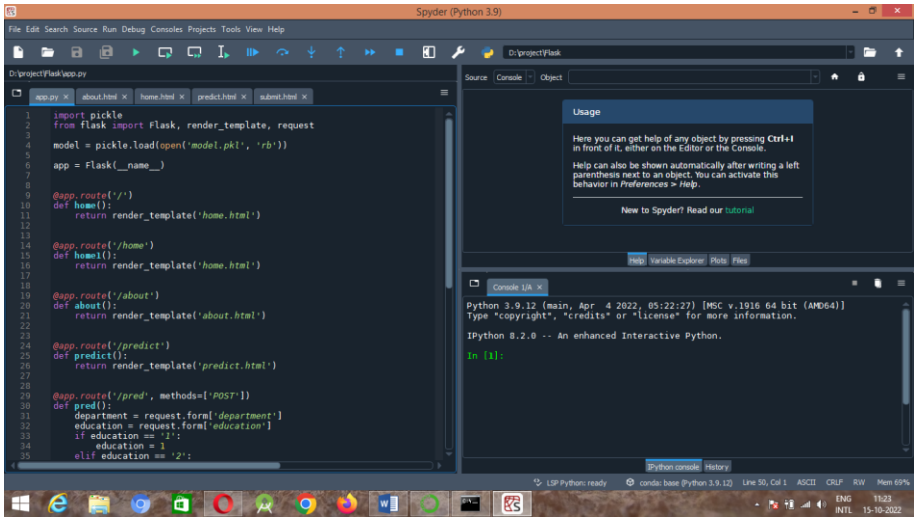

jupyter Employee promotion prediction Last Checkpoint: a few seconds ago (autosaved)
In [1]:
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
import warnings
warnings.filterwarnings('ignore')
from sklearn.preprocessing import LabelEncoder
from sklearn.tree import DecisionTreeClassifier
from sklearn.ensemble import RandomForestClassifier,GradientBoostingClassifier
from sklearn.neighbors import KNeighborsClassifier
from sklearn.model_selection import cross_val_score
import pickle
from sklearn.metrics import classification_report,confusion_matrix
plt.style.use('fivethirtyeight')
pd.set_option('display.max_rows',None)
In [2]:
# Reading the csv and printing the shape
In [3]:
df=pd.read_csv("D:\project\Dataset\emp_promotion.csv")
print('Shape of train data {}'.format(df.shape))


jupyter Employee promotion prediction Last Checkpoint: a few seconds ago (autosaved)
df['KPIs_met >80%'].value_counts().plot(kind='pie',autopct='%.2f%%',shadow=True)
plt.subplot(232)
plt.axis('off')
plt.title('awards_won?')
df['awards_won?'].value_counts().plot(kind='pie',autopct='%.2f%%',shadow=True)
plt.subplot(233)
plt.axis('off')
plt.title('previous_year_rating')
df['previous_year_rating'].value_counts().plot(kind='pie',autopct='%.2f%%',shadow=True)
plt.show()
KPIs_met >80%
0
64.80%
35.20%
1
awards_won?
0
97.68%
2.32%
1
previous_year_rating
3.0
36.73%
5.0
23.17%
8.34%
2.0
12.28%
19.49%
1.0
4.0


jupyter Employee promotion prediction Last Checkpoint: 5 minutes ago (autosaved)
In [39]:
# Handling imbalanced data
# Splitting data and resampling it
In [40]:
x=df.drop('is_promoted',axis=1)
y=df['is_promoted']
print(x.shape)
print(y.shape)
(52397, 9)
(52397,)
In [41]:
from imblearn.over_sampling import SMOTE
In [42]:
sm=SMOTE()
x_resample,y_resample=sm.fit_resample(x,y)
In [43]:
plt.figure(figsize=(10,6))
plt.subplot(121)
sns.countplot(y)
plt.title('Before oversampling')
plt.subplot(122)
sns.countplot(y_resample)
plt.title('After oversampling')
Out[43]:
Text(0.5, 1.0, 'After oversampling')


Employee promotion prediction
In [58]: # Evaluating performance of the model and saving the model
In [59]: # Random forest model is selected
In [60]: rf= RandomForestClassifier()
rf.fit(x_train,y_train)
yPred = rf.predict(x_test)
In [61]: cv=cross_val_score(rf,x_resample,y_resample,cv=5)
In [62]: cv
np.mean(cv)
Out[62]: 0.944925728514544
In [63]: pickle.dump(rf,open('model.pkl','wb'))
In [ ]:

App.py


Spyder (Python 3.9)
import pickle
from flask import Flask, render_template, request
model = pickle.load(open('model.pkl', 'rb'))
app = Flask(__name__)
@app.route('/')
def home():
return render_template('home.html')
@app.route('/home')
def home1():
return render_template('home.html')
@app.route('/about')
def about():
return render_template('about.html')
@app.route('/predict')
def predict():
return render_template('predict.html')
@app.route('/pred', methods=['POST'])
def pred():
department = request.form['department']
education = request.form['education']
if education == '1':
education = 1
elif education == '2':
Usage
Here you can get help of any object by pressing Ctrl+I in front of it, either on the Editor or the Console.
Help can also be shown automatically after writing a left parenthesis next to an object. You can activate this behavior in Preferences > Help.
New to Spyder? Read our tutorial
Python 3.9.12 (main, Apr 4 2022, 05:22:27) [MSC v.1916 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.
IPython 8.2.0 -- An enhanced Interactive Python.
In [1]: