PREDICTING FUTURE EMPLOYEE ATTRITION: A MACHINE LEARNING APPROACH

A Major Project Report submitted to the Department of Computer Applications, Bharathiar University in the partial fullfillment of the requirements for the award of degree of

MASTER OF SCIENCE IN DATA ANALYTICS

Submitted by

KIRTHIKA V
(22CSEG16)

Under the guidance of

Dr. J. Komalalakshmi, B.Sc., MCA., M.Phil., B.Ed., M.Ed., Ph.D.,

GUEST FACULTY

DEPARTMENT OF COMPUTER APPLICATIONS

BHARATHIAR UNIVERSITY

COIMBATORE - 641 046

APRIL - 2024

CERTIFICATE

This is to certify that the Major-Project report titled “**PREDICTING FUTURE EMPLOYEE ATTRITION: A MACHINE LEARNING APPROACH”** submitted to the Department of Computer Applications, Bharathiar University in partial fulfilment of the requirement for the award of the degree of the Master of Science in Data Analytics is record of the original work done by **KIRTHIKA V (22CSEG16)** under my supervision and guidance and this project work has not formed the basis for the award of any Degree/Diploma/Associate ship/ Fellowship or similar title to any candidate of any University.

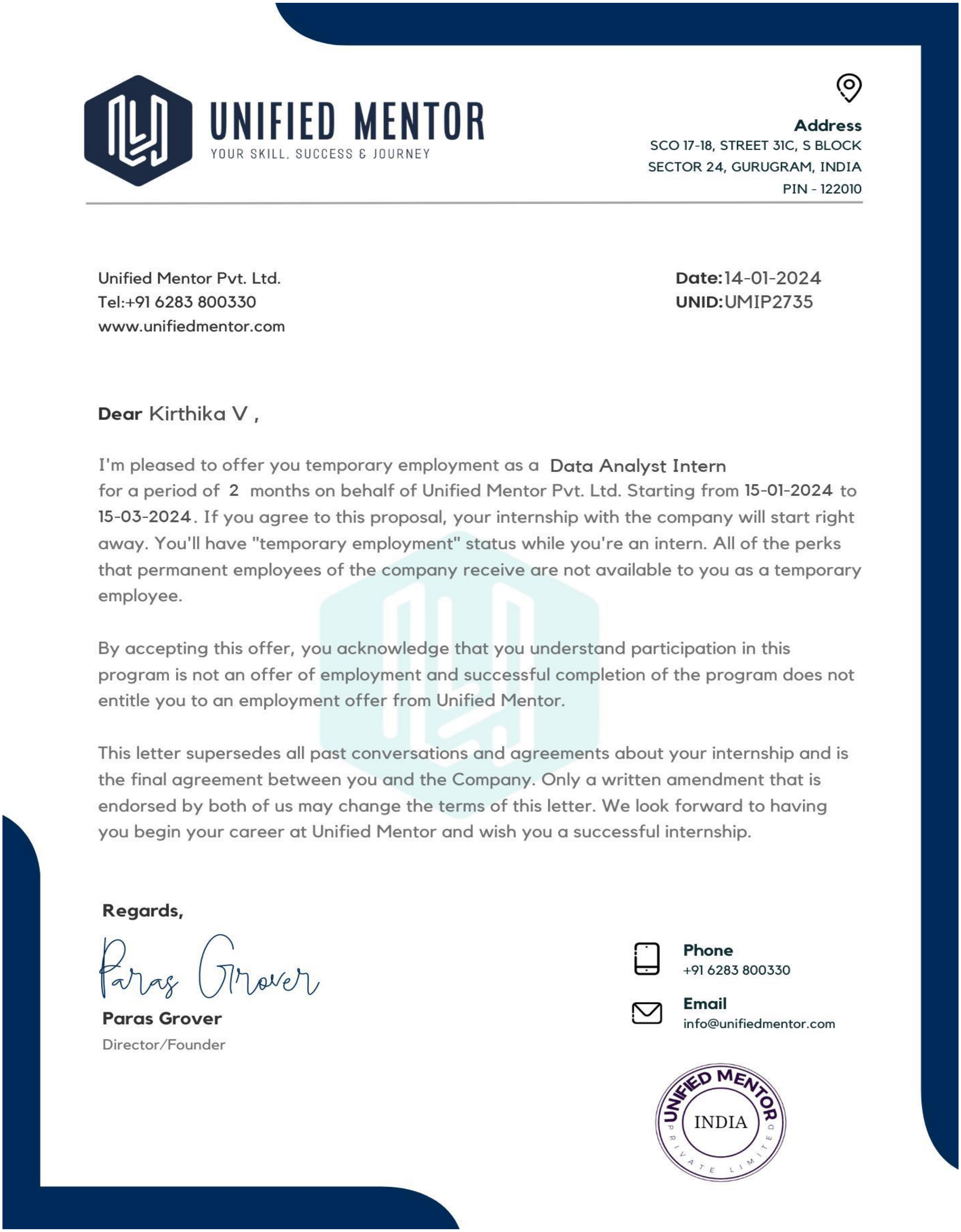
Place: Coimbatore

Date:

Project Guide Head of the Department

Submitted for the University Viva-Voice Examination held on ________________

Internal Examiner External Examiner



UNIFIED MENTOR
YOUR SKILL, SUCCESS & JOURNEY

Address
SCO 17-18, STREET 31C, S BLOCK
SECTOR 24, GURUGRAM, INDIA
PIN - 122010

Unified Mentor Pvt. Ltd.
Tel:+91 6283 800330
www.unifiedmentor.com

Date:14-01-2024
UNID:UMIP2735

Dear Kirthika V ,

I'm pleased to offer you temporary employment as a Data Analyst Intern for a period of 2 months on behalf of Unified Mentor Pvt. Ltd. Starting from 15-01-2024 to 15-03-2024. If you agree to this proposal, your internship with the company will start right away. You'll have "temporary employment" status while you're an intern. All of the perks that permanent employees of the company receive are not available to you as a temporary employee.

By accepting this offer, you acknowledge that you understand participation in this program is not an offer of employment and successful completion of the program does not entitle you to an employment offer from Unified Mentor.

This letter supersedes all past conversations and agreements about your internship and is the final agreement between you and the Company. Only a written amendment that is endorsed by both of us may change the terms of this letter. We look forward to having you begin your career at Unified Mentor and wish you a successful internship.

Regards,

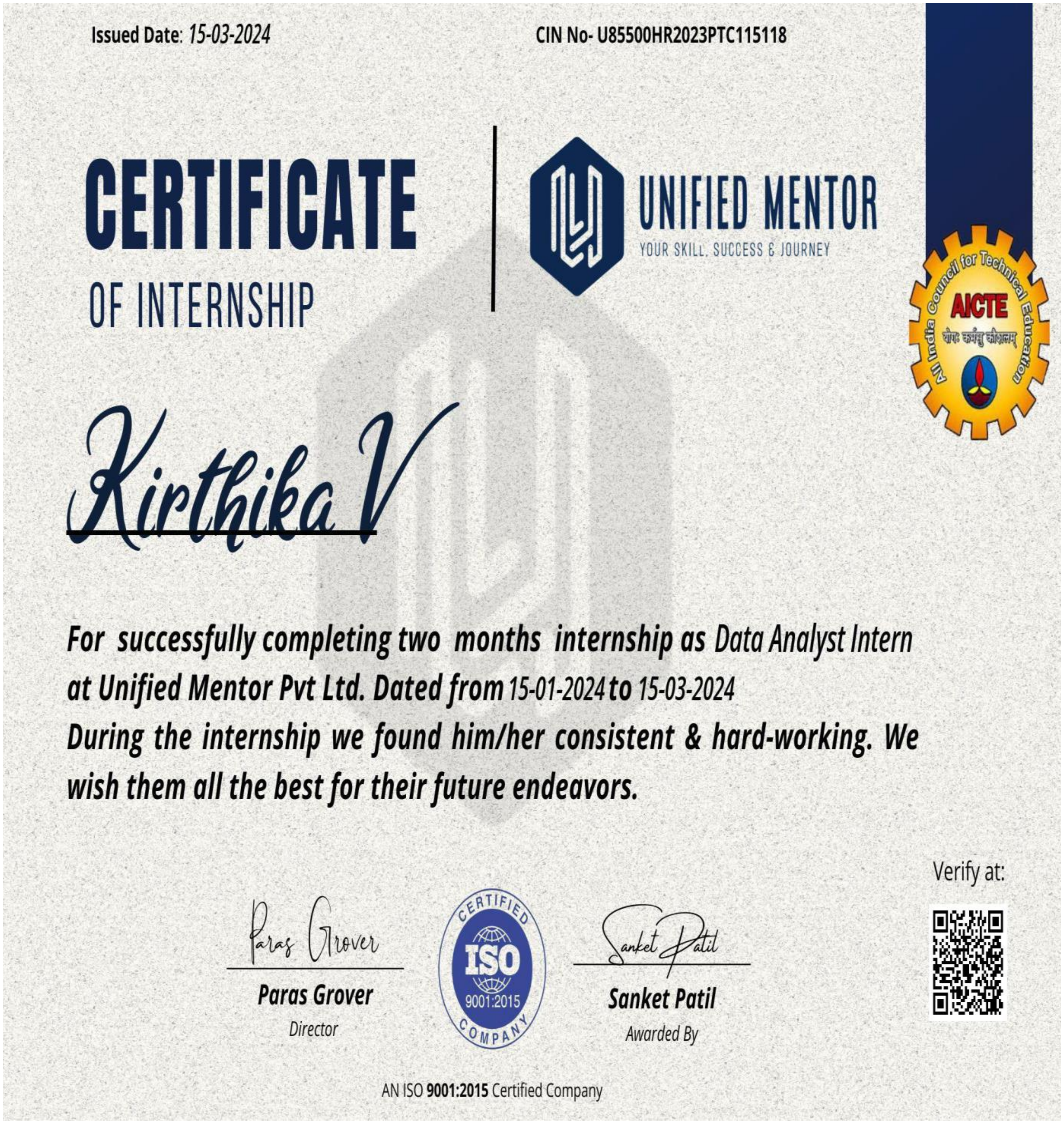
Paras Grover

Paras Grover
Director/Founder

Phone
+91 6283 800330

Email
info@unifiedmentor.com

UNIFIED MENTOR
INDIA
PRIVATE LIMITED



Issued Date: *15-03-2024*

CIN No- U85500HR2023PTC115118

CERTIFICATE

OF INTERNSHIP

UNIFIED MENTOR

YOUR SKILL, SUCCESS & JOURNEY

All India Council for Technical Education

AICTE

योगः कर्मसु कौशलम्

Kirthika V

For successfully completing two months internship as Data Analyst Intern at Unified Mentor Pvt Ltd. Dated from 15-01-2024 to 15-03-2024

During the internship we found him/her consistent & hard-working. We wish them all the best for their future endeavors.

Verify at:

Paras Grover

Paras Grover

Director

CERTIFIED

ISO

9001:2015

COMPANY

Sanket Patil

Sanket Patil

Awarded By

AN ISO **9001:2015** Certified Company

DECLARATION

I hereby declare that this Major-project report titled **"PREDICTING FUTURE EMPLOYEE ATTRITION: A MACHINE LEARNING APPROACH"** submitted to the Department of Computer Applications, Bharathiar University is a record of original work done by **KIRTHIKA V (22CSEG16)** under the guidance of **Dr. J. Komalalakshmi, B.Sc., MCA., M.Phil., B.Ed., M.Ed., Ph.D.,** Guest Faculty, Department of Computer Applications, Bharathiar University and this project work has not formed the basis for the award of any Degree/ Diploma/ Associate ship/ Fellowship or similar title to any candidate of any University.

Place: Coimbatore — Signature of the Candidate

Date: — (KIRTHIKA V)

Countersigned by,

Guide

ACKNOWLEDGEMENT

I express my respectful thanks to our Professor & Head of the Department **Dr. M. Punithavalli, B.Sc., M.Sc., M.Phil., Ph.D,** Department of Computer Applications, Bharathiar University, for permitting me to carry out my major project report work in **"PREDICTING FUTURE EMPLOYEE ATTRITION: A MACHINE LEARNING APPROACH"**

I really deem it a special privilege to convey my prodigious and everlasting thanks to my guide **Dr. J. Komalalakshmi, B.Sc., MCA., M.Phil., B.Ed., M.Ed., Ph.D., Guest Faculty,** Department of Computer Applications, Bharathiar University, for her valuable guidance and personal interest in this major project report work.

Last but not least, I also acknowledge the help done by my parents and acknowledge the encouraging support of my friends who were involved in this major project, in one way or the other. I thank Almighty for showering the divine grace on me and offer prayers to the lord for everything that was given to me.

KIRTHIKA V (22CSEG16)

ABSTRACT

Predicting future Employee attrition project aims to develop predictive models to forecast the likelihood of employee turnover within the next two years for a company's HR department. The dataset comprises various attributes related to employees, and the primary objective is to construct accurate models that aid in identifying potential turnover risks.

Initial steps involve importing the dataset, assessing missing values, and conducting exploratory data analysis (EDA) to compare attributes and glean insights. Following EDA, machine learning models, namely K-Nearest Neighbors (KNN), Random Forest, and Multilayer Perceptron (MLP) Classifier, are employed for predictive analysis. These models are trained and evaluated based on their ability to predict employee attrition accurately. The project delves into a detailed comparison of the performance metrics, such as accuracy, precision, recall, and F1-score, for each model. The findings provide valuable insights into the effectiveness of different machine learning algorithms in predicting employee turnover, thereby assisting HR departments in proactively managing workforce attrition.

❖ **System Development:**

➢ **Backend:**

In this project, the Scikit-Learn metrics library is used to develop a Employee attrition model. The dataset was provided by Unified Mentor Company and was utilized to develop the predictive models. It comprises 4653 instances and encompasses 9 attributes.

These datasets were downloaded and stored as CSV files in local directory for model development and evaluation. This document summarizes the key aspects of the approach and dataset for the Employee attrition project**.**

➢ **Frontend:**

For the development of this Employee attrition project, the Python programming language is employed as the primary tool. Within Python, harnessed the power of machine learning algorithms, with a particular focus on utilizing the Classification algorithm which are Multilayer Perceptron classifier, Random Forest Classification and K-Nearest Neighbors classification are the cornerstone of this project. This choice of algorithm played a crucial role in building an effective Future prediction of Employee attrition model.

➢ **Connectivity:**

The connection between Python and the Structured Employee attrition dataset has been established, and subsequently, the dataset was imported into the Python environment.

(i) pd.read_csv(“train.csv”)
(ii) pd.read_csv(“test.csv”)

This was accomplished by utilizing the Pandas library, employing the pd.read_csv function to read and load the dataset from the CSV file into the Python environment for further analysis and model development.

- **Work Flow:**

The Employee attrition dataset includes nine features which are

(i) Education
(ii) Joining Year
(iii) City
(iv) Payment Tier
(v) Age
(vi) Gender
(vii) Ever Benched
(viii) Experience in Current Field
(ix) Leave or Not

In this project, the dataset provided by Unified Mentor Company is first prepared by importing it into Python and addressing any missing values. Subsequently, an exploratory data analysis is conducted to gain insights into the attributes through descriptive statistics and visualization techniques. Following this, machine learning models, namely the Multilayer Perceptron classifier, Random Forest Classification, and K-Nearest Neighbors classification, are implemented to predict employee attrition. The performance of each model is then evaluated using metrics such as accuracy, precision, recall, and F1-score. The results are compared, and their implications for predicting future employee attrition are discussed. Finally, a conclusion is drawn summarizing the findings and suggesting avenues for further research.

TABLE OF CONTENTS

I. INTRODUCTION

In this project, the focus is on the development of a predictive model for anticipating employee attrition within the next two years, utilizing machine learning techniques. The dataset, provided by Unified Mentor Company, forms the basis for the analysis. Initial steps involve data preparation, including the handling of missing values, followed by an in-depth exploratory data analysis to extract insights from the various attributes. Subsequently, machine learning algorithms such as the Multilayer Perceptron classifier, Random Forest Classification, and K-Nearest Neighbors classification are applied to build predictive models. The performance of each model is evaluated through a comprehensive assessment of metrics including accuracy, precision, recall, and F1-score. This project aims to contribute to the enhancement of workforce management strategies by providing HR departments with reliable tools for proactively addressing employee turnover challenges.

1.1 Background

The background of this project centers around the critical need for organizations to anticipate and address employee attrition effectively. With turnover posing significant challenges to workforce stability and productivity, predictive models offer valuable tools for proactive management. Leveraging data provided by Unified Mentor Company, this project seeks to develop machine learning-based solutions capable of forecasting employee turnover within a two-year timeframe. By understanding the underlying patterns and factors contributing to attrition, organizations can implement targeted strategies to retain talent and optimize workforce management practices.

1.2 Objective

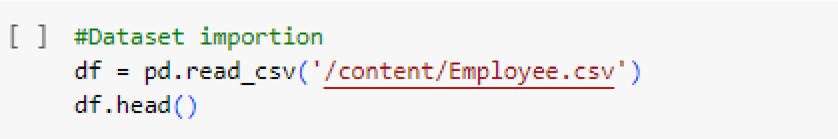
The objective of this project is to develop predictive models using machine learning techniques to forecast employee attrition within the next two years. By analyzing the dataset provided by Unified Mentor Company, the aim is to identify patterns and factors associated with employee turnover. These predictive models will enable organizations, particularly HR departments, to anticipate potential attrition risks and take proactive measures to retain valuable talent. Ultimately, the goal is to empower organizations with actionable insights to optimize workforce management strategies and enhance employee retention efforts.

II. DATA COLLECTION AND PREPROCESSING

The data for this project was collected from Unified Mentor Company, which provided a dataset containing information relevant to employee attrition prediction. The dataset includes 4,653 instances and encompasses 9 attributes. Details regarding the specific attributes and their meanings were provided along with the dataset. The data collection process ensured the anonymization of sensitive information to maintain confidentiality and privacy. This dataset serves as the foundation for the analysis and development of predictive models aimed at forecasting employee turnover within a two-year timeframe.

2.1 Importing Dataset

The dataset importation process involved utilizing the pandas library within the Python programming language. Pandas provides powerful tools for data manipulation and analysis, making it a popular choice for working with tabular data like the dataset provided for this project. By using pandas, the dataset was imported into the Python environment, allowing for seamless access to its contents.

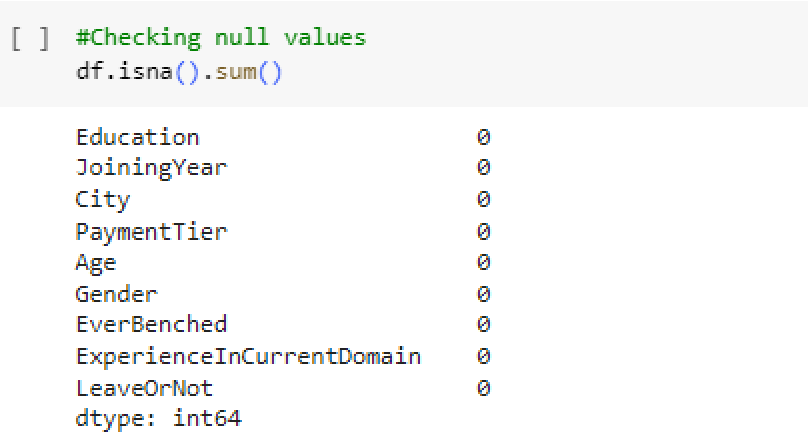


```
[ ]  #Dataset importion
     df = pd.read_csv('/content/Employee.csv')
     df.head()
```

Fig 2.1 Dataset importing

2.2 Handling Missing Values

The dataset was checked for missing values using the isnull().sum() function, which revealed that there are no missing values present in the dataset.



```
[ ]  #Checking null values
     df.isna().sum()

     Education                    0
     JoiningYear                  0
     City                         0
     PaymentTier                  0
     Age                          0
     Gender                       0
     EverBenched                  0
     ExperienceInCurrentDomain    0
     LeaveOrNot                   0
     dtype: int64
```

Fig 2.2 Checking null values

III. EXPLORATORY DATA ANALYSIS(EDA)

Exploratory Data Analysis (EDA) involves the process of examining and visualizing the dataset to gain insights into its structure, distribution, and relationships between variables. In this project, EDA serves as a crucial step towards understanding the dataset provided by Unified Mentor Company and identifying patterns or trends relevant to employee attrition prediction.

3.1 Data Visualization

Data visualization techniques, such as histograms, box plots, and scatter plots, help visualize the distribution of individual variables and explore potential relationships between them.

- **Countplot for Education distributions:**

A countplot is utilized to visualize the distribution of education levels, which include the categories of bachelors, masters, and PhD.

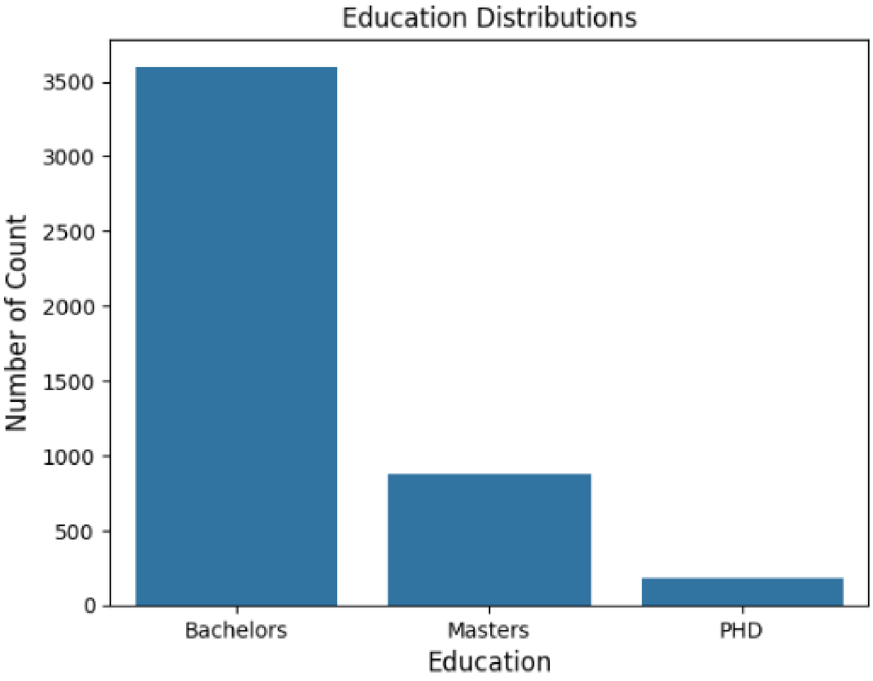


Fig 3.1 Countplot for Education distributions

- **Insight:**

The education level categories consist of bachelor, master, and PhD, with the counts as follows: bachelors, comprising over 3500 instances; masters, containing approximately 1000 instances; and PhD, comprising approximately 250 instances. This distribution indicates a higher prevalence of individuals with bachelor's degrees compared to those with master's or PhD degrees. The reason for this distribution may be attributed to factors such as the entry-level requirements for positions within the company, educational attainment trends within the workforce, and the organization's hiring practices.

- **Piechart for City distribution:**

A Piechart is utilized to visualize the distribution of City, which include the categories of Bangalore, Pune and New delhi.

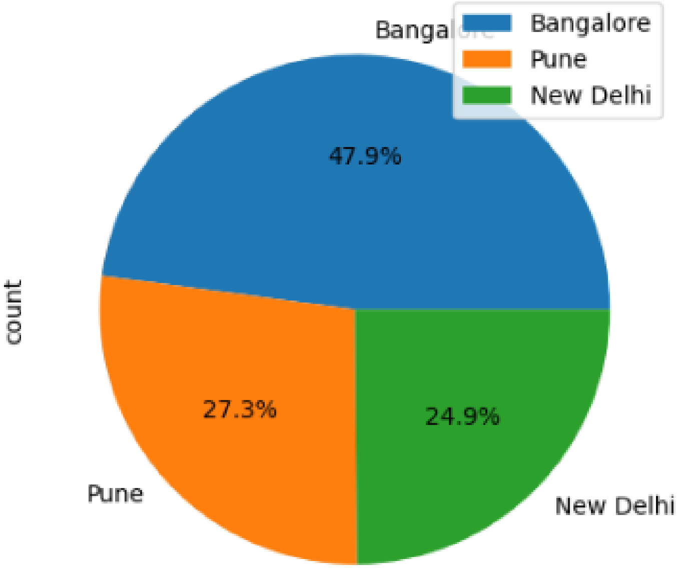


Fig 3.2 Piechart for city distribution

- **Insight:**

Using a pie chart, the distribution of the city attribute is visualized, revealing that Bangalore comprises the highest proportion at 47.9%, followed by Pune at 27.3%, and New Delhi at 24.9%. This distribution suggests that Bangalore has the largest representation among the cities in the dataset, followed by Pune and then New Delhi. The reason for this distribution may be attributed to factors such as the geographic location of the company's offices or branches, population density in these cities, job market opportunities, or specific industry presence in each city.

- **Countplot for city distribution:**

A Countplot is utilized to visualize the distribution of City, which include the categories of Bangalore, Pune and New delhi.

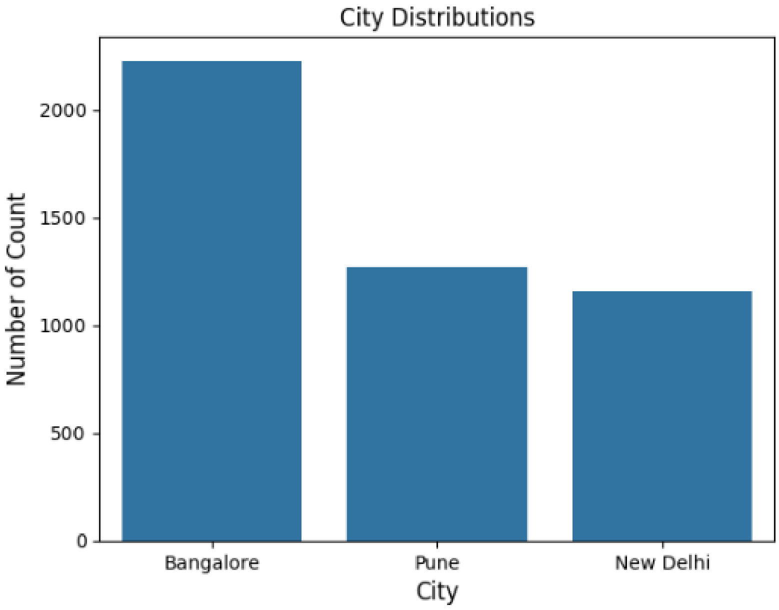


Fig 3.3 Countplot for city distribution

- **Insight:**

The countplot is employed to visualize the distribution of cities, including Pune, Bangalore, and New Delhi. Bangalore exhibits the highest count, surpassing 2000 instances, followed by Pune with over 1200 instances, and New Delhi with over 1000 instances. This distribution suggests that Bangalore has the highest representation among the cities in the dataset, followed by Pune and then New Delhi. The reason for this distribution may be attributed to factors such as the company's headquarters or major operations being located in Bangalore, followed by significant business presence in Pune and New Delhi. Additionally, factors such as population density, economic opportunities, and regional industry development may contribute to the varying counts of instances across these cities.

- **Countplot for Gender distribution:**

A Countplot is utilized to visualize the Gender distributions, which include the categories of Male and Female.

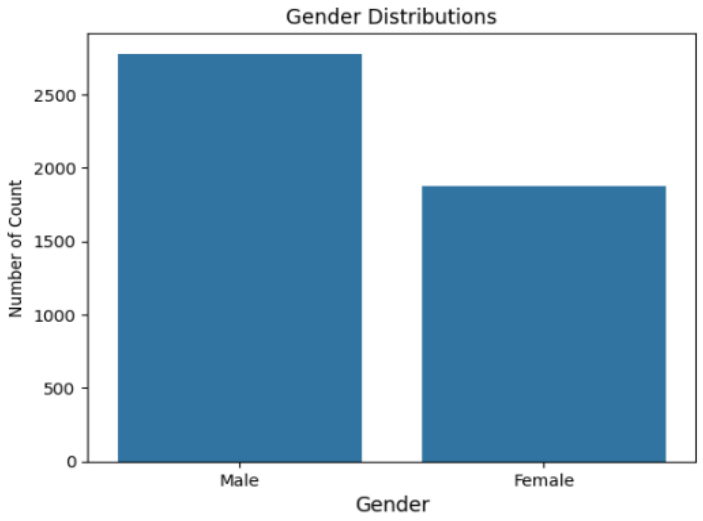


Fig 3.4 Countplot for Gender distribution

- **Insight:**

The countplot is utilized to visualize the distribution of gender, comprising male and female categories. The count of males exceeds 2500 instances, while the count of females exceeds 1500 instances. This distribution indicates a higher representation of males compared to females in the dataset. The reason for this gender distribution may be attributed to various factors, including workforce demographics, industry-specific gender ratios, hiring practices, and societal norms influencing gender participation in the workforce.

- **Piechart for gender distribution:**

A Piechart is utilized to visualize the Gender distributions, which include the categories of Male and Female.

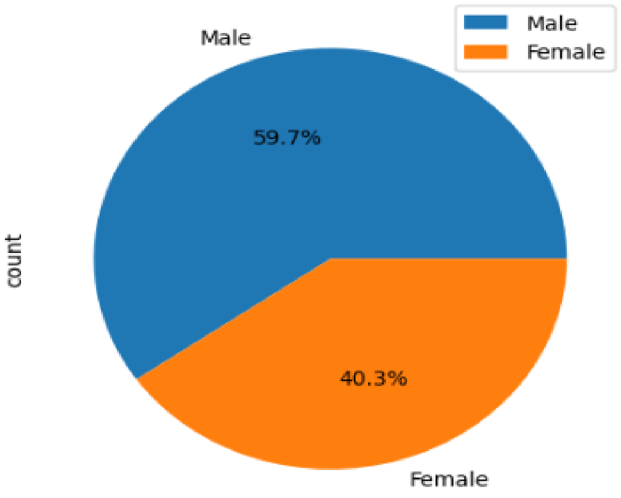


Fig 3.5 Piechart for gender distribution

- **Insight:**

A pie chart is employed to visualize the distribution of gender, with categories including male and female. The male category comprises 59.7%, while the female category comprises 40.3% of the total. This distribution indicates a higher proportion of males compared to females in the dataset. The reason for this gender distribution may be influenced by various factors, including historical gender disparities in certain industries, differential rates of participation in the workforce among genders, and societal norms impacting gender representation in specific occupations or roles.

- **Piechart for Payment tier distributions:**

A Piechart is utilized to visualize the Payment Tier distributions, which include the categories: 1-Highest tier, 2-Middle level and 3-Lowest tier.

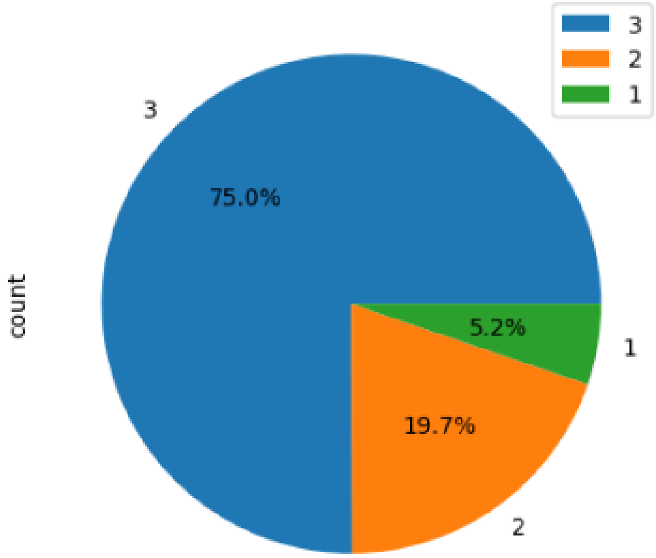


Fig 3.6 Piechart for Payment tier distribution

- **Insight:**

The payment tier is analyzed, revealing three categories: 1 representing the highest tier, 2 representing the middle level, and 3 representing the lowest tier. The pie chart illustrates that the highest tier (category 3) constitutes the largest proportion, accounting for 75.0%, followed by the middle tier (category 2) with 19.7%, and the lowest tier (category 1) with 5.2%. This distribution suggests a predominant presence of employees in the lowest payment tier, followed by the middle and highest tiers. The reason for this distribution may be attributed to factors such as hierarchical salary structures within the organization, varying levels of job roles and responsibilities associated with each tier, and the organization's compensation policies aimed at maintaining a balanced distribution of salaries across different tiers.

- **Countplot for Payment tier distribution:**

A Countplot is utilized to visualize the Payment Tier distributions, which include the categories: 1-Highest tier, 2-Middle level and 3-Lowest tier.

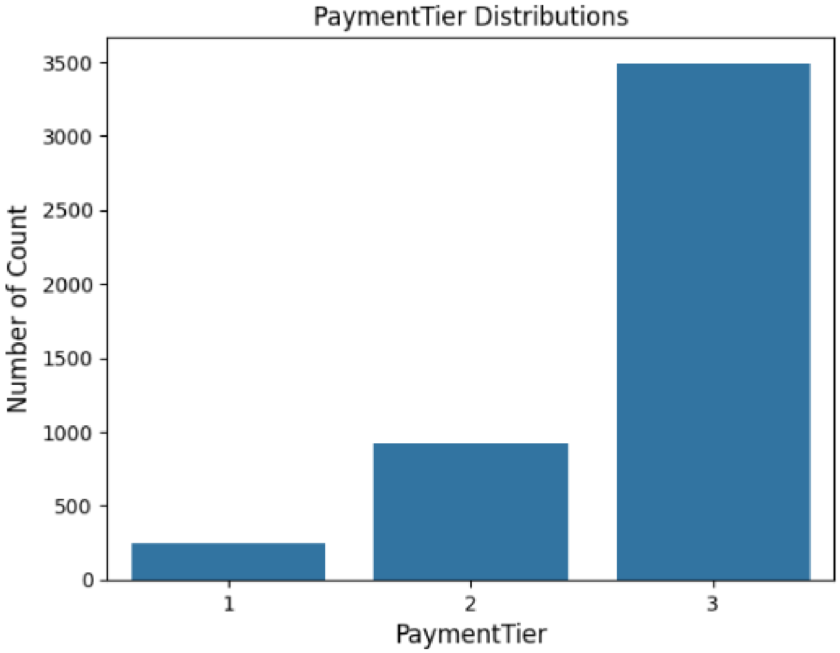


Fig 3.7 Payment tier distribution

- **Insight:**

The payment tier is analyzed using a countplot, revealing three categories: 1 representing the highest tier, 2 representing the middle level, and 3 representing the lowest tier. The countplot indicates that category 3 has a count exceeding 3000 instances, followed by category 2 with approximately 1000 instances, and category 1 with approximately 250 instances. This distribution suggests a higher prevalence of employees in the lowest payment tier, followed by the middle and highest tiers. The reason for this distribution may be influenced by factors such as organizational salary structures, job roles and responsibilities associated with each tier, and the distribution of employees across different departments or job functions within the company.

- **Kdeplot:**

The Kdeplot is utilized to visualize the Age distributions.

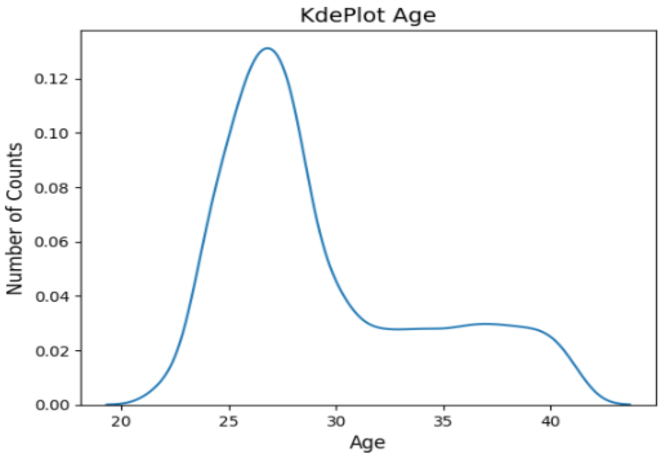


Fig 3.8 Kdeplot

- **Insight:**

The age distributions are visualized using a kdeplot, revealing that the highest density occurs within the age range of 25 to 30, with a subsequent decrease observed for ages above 30. This distribution suggests a concentration of individuals within the younger age bracket, gradually tapering off as age increases beyond 30. The reason for this trend may be attributed to factors such as the demographic composition of the workforce, recruitment practices favoring younger candidates, and career progression trajectories within the organization.

- **Piechart for Everbenched:**

The Piechart is used to visualize the Everbenched attribute and it has the categories yes and no.

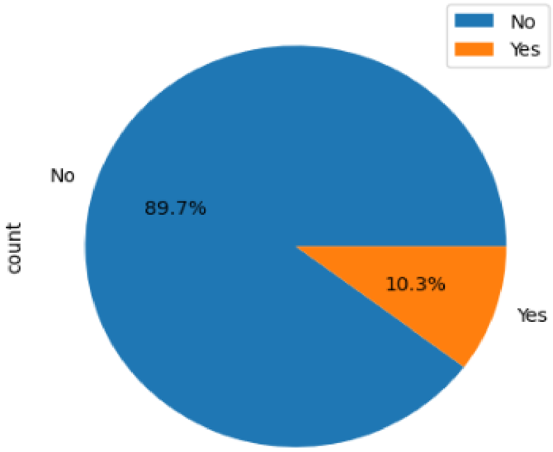


Fig 3.9 Piechart for Everbenched

- **Insight:**

A pie chart is utilized to visualize the distribution of the everbenched attribute, which consists of two categories: yes and no. The category "yes" comprises 10.3% of the total instances, while the category "no" accounts for 89.7%. This distribution indicates a predominant occurrence of instances where employees have not been benched. The reason for this distribution may be influenced by factors such as efficient resource allocation, effective project management practices, and workforce optimization strategies aimed at minimizing employee bench time and maximizing productivity.

- **Countplot for Everbenched:**

The Countplot is used to visualize the Everbenched attribute and it has the categories yes and no.

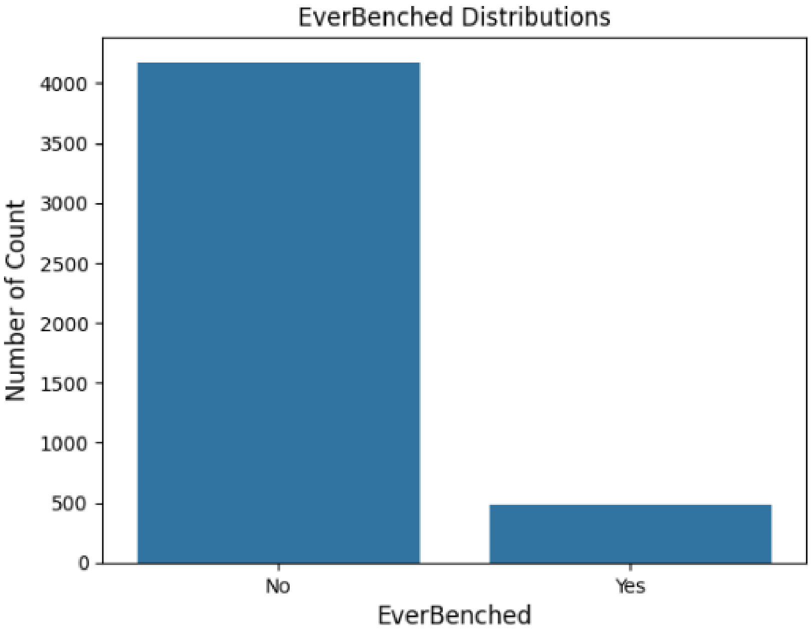


Fig 3.10 Countplot for Everbenched distributions

- **Insight:**

The everbenched attribute is visualized using a countplot, revealing that the category "no" has a count exceeding 4000 instances, while the category "yes" has approximately 500 instances. This distribution indicates a significantly higher prevalence of instances where employees have not been benched compared to those who have. The reason for this distribution may be influenced by factors such as effective project management practices, efficient utilization of workforce resources, and proactive measures taken by the organization to minimize instances of employee benching.

- **Countplot for Experience in current domains:**

The countplot is used to visualize the attribute Experience in current domain.

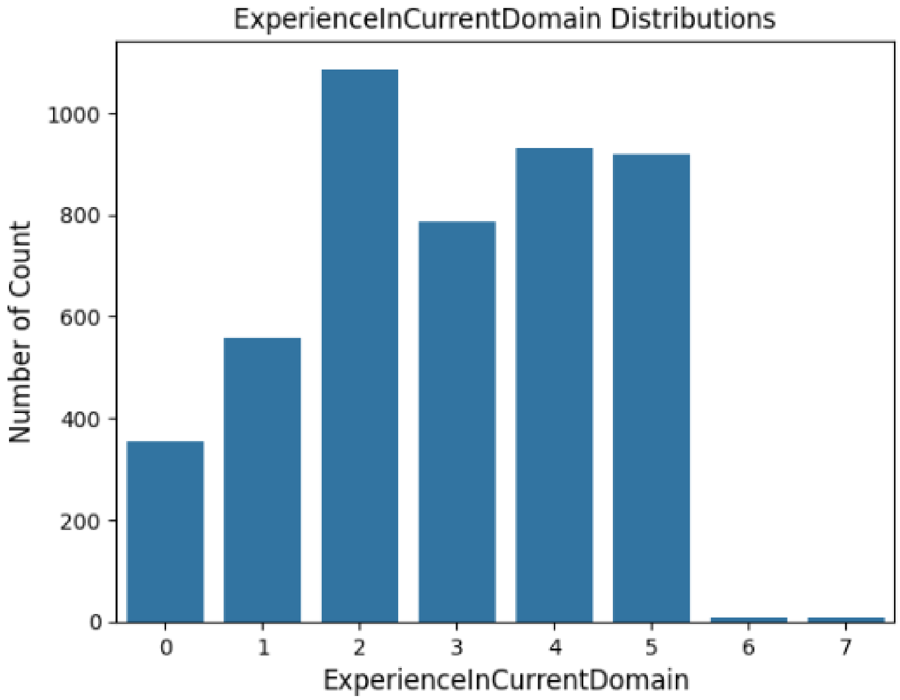


Fig 3.11 Experience in current domain

- **Insight:**

Experience in the current domain is visualized using a countplot, revealing that the category "2" has the highest count, exceeding 1000 instances. Subsequently, categories "4" and "5" exhibit similar counts, both exceeding 800 instances. In contrast, categories "6" and "7" have substantially lower counts, nearing zero instances. This distribution suggests a concentration of employees with experience levels categorized as "2," followed by "4" and "5," while experience levels "6" and "7" are less prevalent. The reason for this distribution may be attributed to factors such as the hiring patterns of the organization, the tenure of employees within the current domain, and the distribution of experience levels across different job roles or departments.

- **Piechart:**

The Piechart is used to visualize the attribute Leave or Not. It contains 0-Leave and 1-Not.

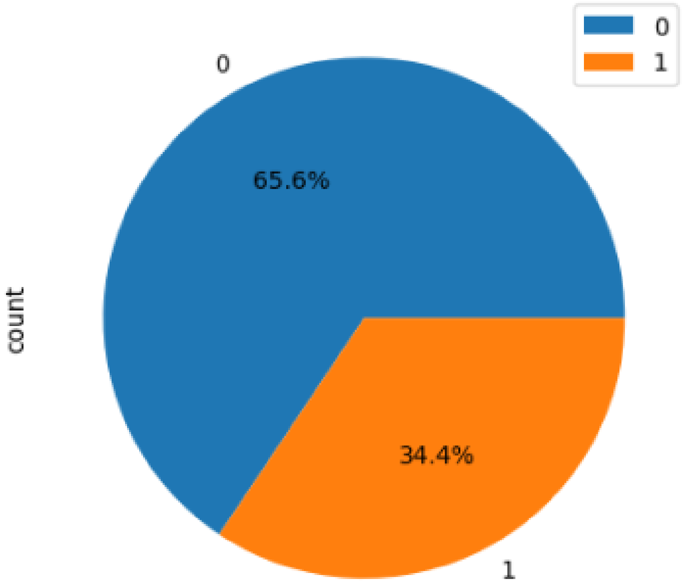


Fig 3.12 Piechart

- **Insight:**

A pie chart is employed to visualize whether employees have taken leave or not, comprising two categories: 0 indicating leave taken and 1 indicating no leave taken. The category "0" represents 65.6% of instances, while the category "1" represents 34.4%. This distribution suggests that a majority of employees have taken leave, while a smaller proportion have not. The reason for this distribution may be influenced by factors such as company policies regarding leave entitlements, employee workload and stress levels, personal circumstances, and the availability of vacation time or other types of leave.

- **Countplot for Leave or Not:**

The Countplot is used to visualize the attribute Leave or not. It contains the categories 0 has leave and 1 has not leave.

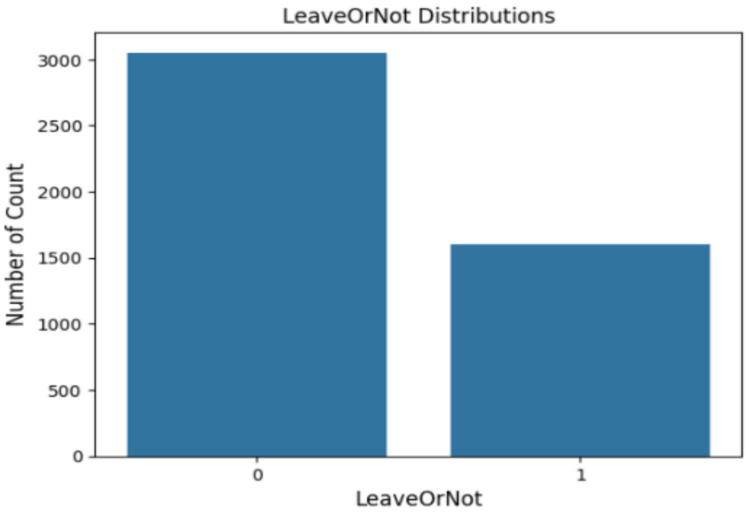


Fig 3.13 Countplot for Leave or Not distribution

- **Insight:**

The analysis of employee leave or not is conducted using a countplot, revealing that the category "0" has a count of 3000 instances, while the category "1" has a count of 1500 instances. This distribution indicates a higher prevalence of instances where employees have taken leave compared to instances where they have not. The reason for this distribution may be influenced by factors such as company policies regarding leave entitlements, employee workload, personal circumstances, and the availability of vacation time or other types of leave, all of which collectively impact employees' decisions to take leave or not.

- **Countplot for joining year distribution:**

The Joining year distribution is visualized using countplot.

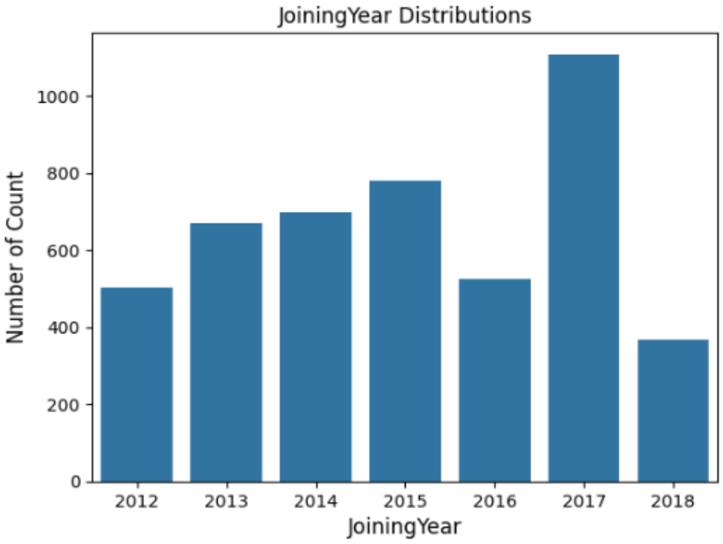


Fig 3.14 Countplot for joining year distribution

- **Insight:**

The visualization of joining year distribution is accomplished using a countplot, revealing that the year 2017 exhibits the highest count, followed by 2015 as the second highest count. Conversely, the year 2018 displays the least count. This distribution suggests a higher influx of employees joining in 2017, followed by those joining in 2015, while 2018 saw comparatively fewer new hires. The reason for this distribution may be attributed to factors such as company growth trends, hiring cycles, recruitment initiatives, and economic conditions impacting hiring activities during those years.

- **Piechart for joining year distribution:**

The Joining year distribution is visualized using Piechart.

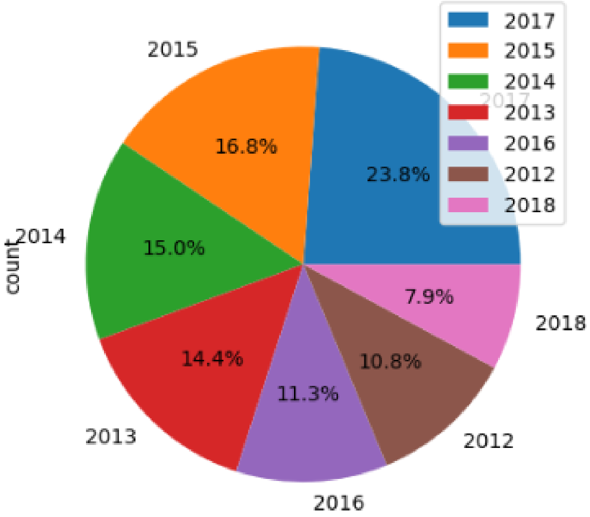


Fig 3.15 Piechart for joining year distribution

- **Insight:**

Joining year distribution is visualized using a pie chart, where the year 2017 comprises the highest proportion at 23.8%, followed by 2015 with 16.8%, and 2018 with the least proportion at 7.9%. This distribution indicates a higher representation of employees joining in 2017, followed by those joining in 2015, while 2018 had the lowest representation. The reason for this distribution may be attributed to factors such as hiring trends influenced by organizational expansion, recruitment drives, job market conditions, and specific initiatives or projects requiring workforce augmentation during those respective years.

- **Kdeplot Overall analysis:**

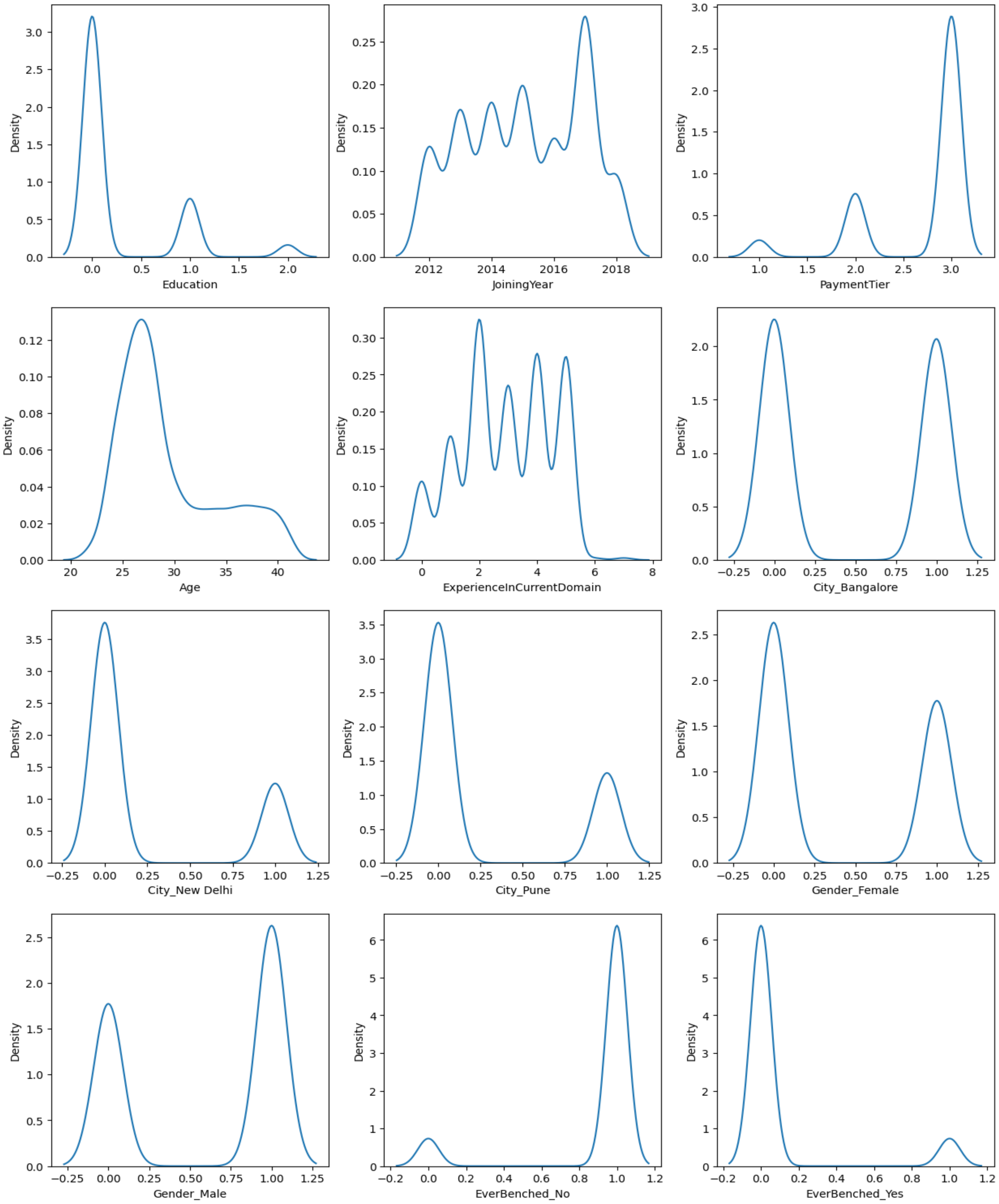
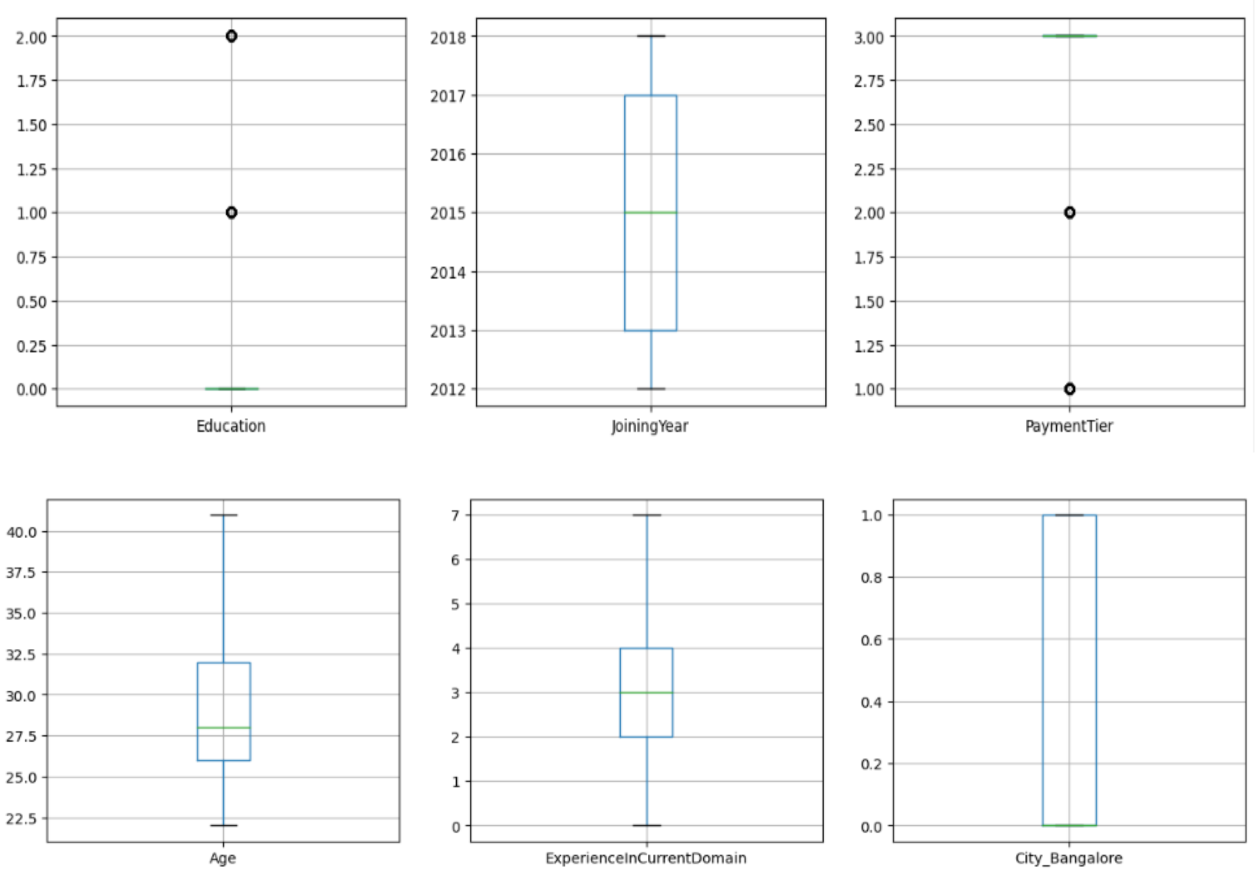


Fig 3.16 Kdeplot

- **Insight:**

The kdeplot analysis across overall attributes reveals several notable findings. Age demonstrates the highest density within the bin ranging from 25 to 30, indicating a concentration of individuals within this age range. Similarly, experience in the current domain exhibits its highest bin at level 2, suggesting a prevalent level of experience among employees. Among cities, Bangalore, Pune, and New Delhi are analyzed, with Bangalore showing the highest representation, followed by Pune and then New Delhi. Additionally, gender distribution is examined, revealing insights into the proportion of males and females within the dataset. Furthermore, the everbenched attribute is analyzed, providing insights into the occurrence of employee benching. These observations collectively contribute to a comprehensive understanding of the dataset's characteristics and trends. The reasons for these findings may be influenced by various factors, including demographic trends, hiring practices, regional preferences, and organizational policies impacting workforce dynamics and employee experiences across different attributes.

- **Boxplot for Outlier detection:**



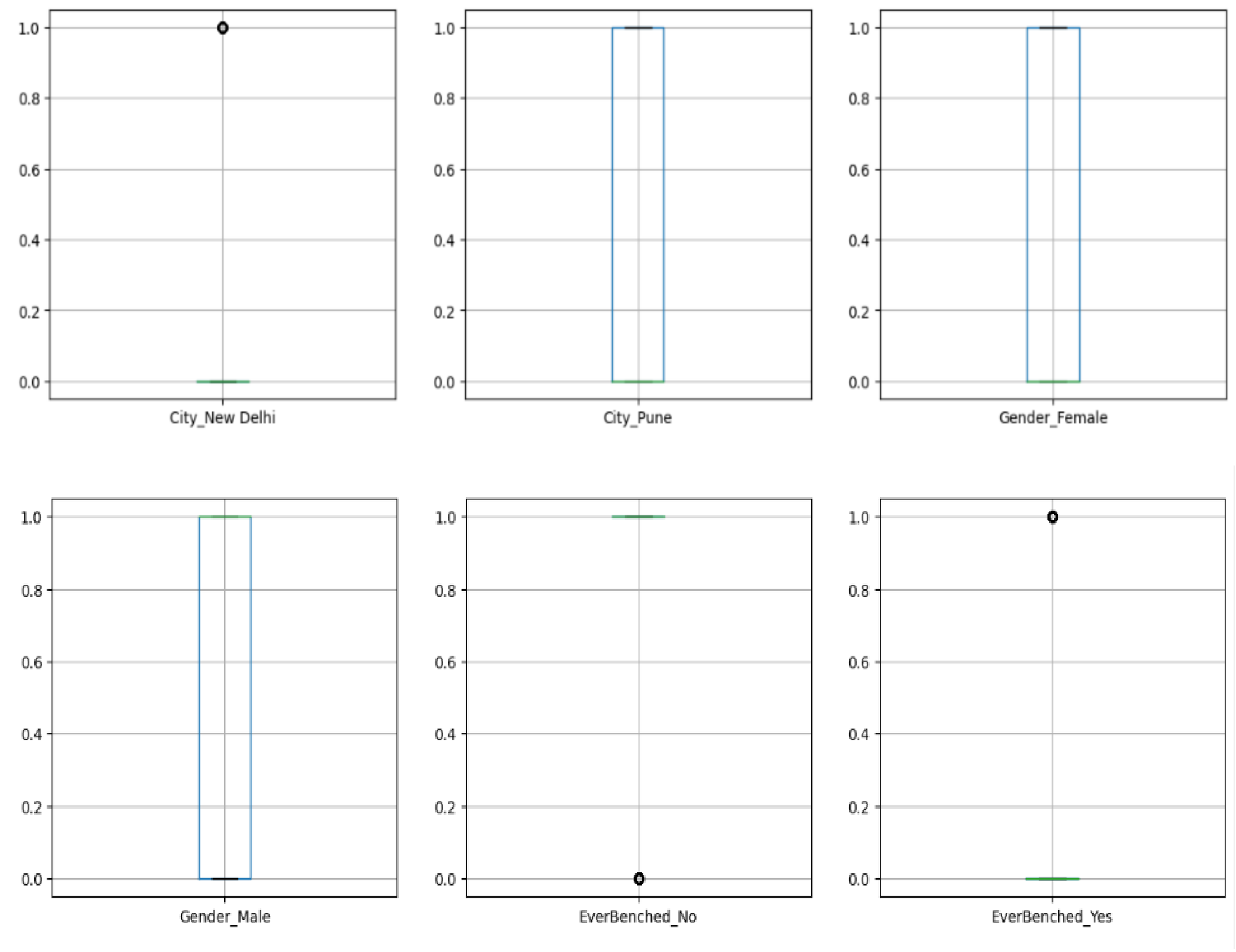


Fig 3.17 Boxplot

- **Insight:**

In the visualization of boxplots for outlier detection, it is observed that the attributes "everbenched" and "city" exhibit outliers, whereas other attributes do not display outliers. The presence of outliers in "everbenched" and "city" suggests the occurrence of unusual or exceptional values beyond the typical range observed in the dataset. This phenomenon may be influenced by factors such as irregularities in the data collection process, data entry errors, or genuine anomalies within the dataset. In contrast, the absence of outliers in other attributes indicates a relatively consistent distribution of values within their respective ranges, implying a more standardized pattern across those attributes.

- **Correlation analysis:**

The heatmap is employed for conducting correlation analysis among the attributes in the dataset.

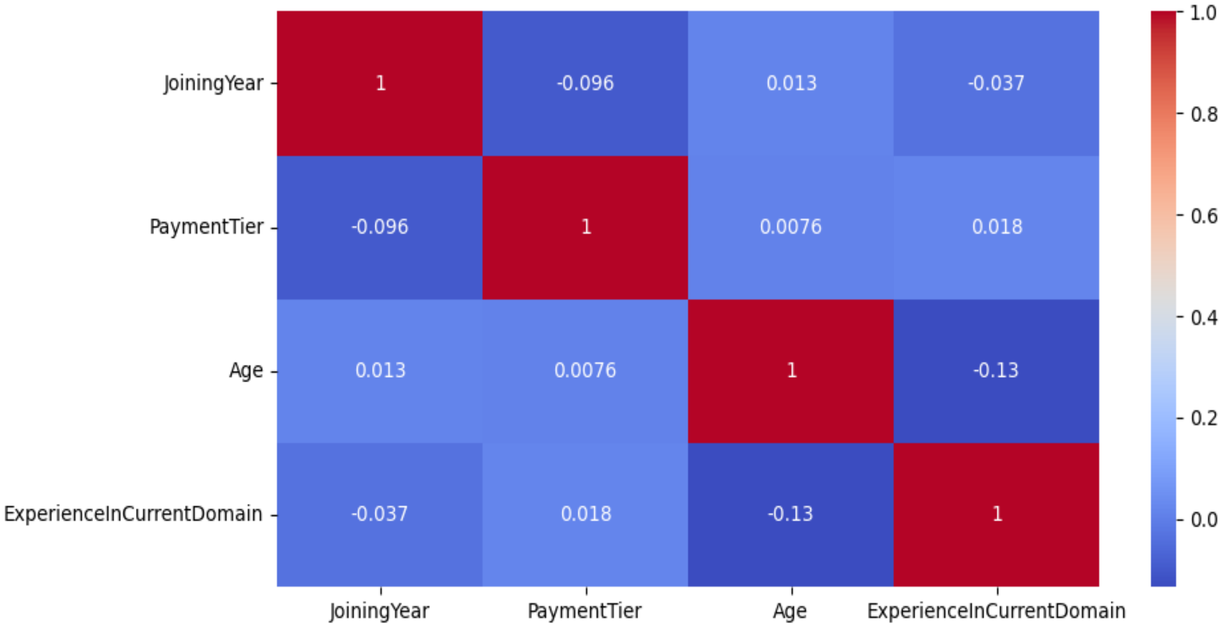


Fig 3.18 Heatmap

- **Insight:**

In the heatmap analysis, it is observed that no attributes exhibit a high level of correlation. Additionally, negative correlations are identified between age and experience in the current domain, as well as between payment tier and joining year. This suggests that as age increases, experience in the current domain may decrease, and as payment tier increases, the joining year may decrease. These findings provide insights into the relationships between different attributes within the dataset, highlighting potential patterns or trends that may inform further analysis or model development.

3.2 Attribute Comparison

Attribute comparison involves examining the relationships and differences between various attributes within the dataset. This process helps in understanding how different attributes interact with each other and their potential impact on the target variable or outcome of interest. In the context of this project, attribute comparison may include:

- ✓ Comparing demographic attributes such as age, gender, and city to identify any patterns or trends.

- ✓ Analyzing the distribution of categorical attributes such as education level or everbenched status across different groups.
- ✓ Exploring the relationship between continuous attributes such as age, experience in the current domain, and payment tier through correlation analysis.
- ✓ Investigating the distribution of target variables, such as employee attrition or leave status, across different levels of categorical attributes.

Overall, attribute comparison allows for a deeper understanding of the dataset and helps in identifying potential predictors or factors influencing the outcome variable, which is crucial for building accurate predictive models.

IV. MACHINE LEARNING MODELS

In the project, machine learning models are utilized to predict employee attrition based on various attributes within the dataset. Several classification algorithms, including K-Nearest Neighbors (KNN), Random Forest Classification, and Multilayer Perceptron (MLP) classifier, are employed for this purpose.

These models are trained on the dataset to learn patterns and relationships between:

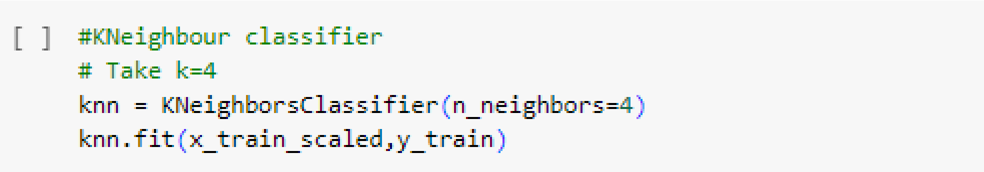
(i) The predictor variables (age, experience, education level)
(ii) The target variable (employee leave or not).

The chosen machine learning models play a pivotal role in developing an accurate and reliable predictive model for identifying potential employee turnover risks.

4.1 Overview of Model

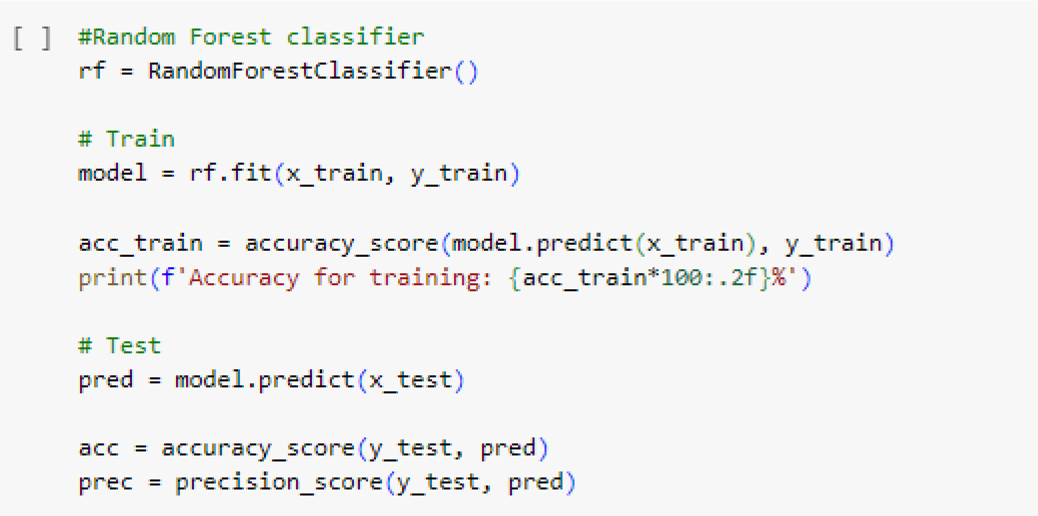
An overview of the machine learning models used for the project is provided as follows:

K-Nearest Neighbors (KNN): The KNN algorithm operates by identifying the K nearest data points to a given instance based on a similarity measure and assigning the most common class label among its neighbors.



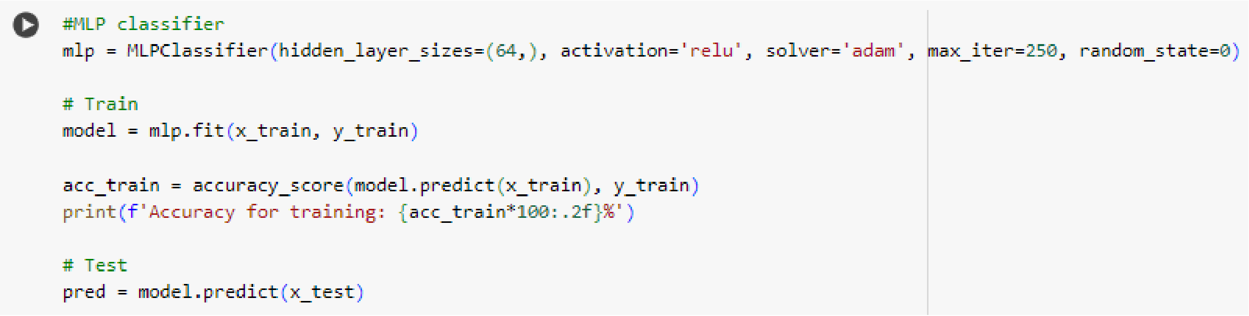
```
[ ]  #KNeighbour classifier
     # Take k=4
     knn = KNeighborsClassifier(n_neighbors=4)
     knn.fit(x_train_scaled,y_train)
```

Random Forest Classification: Random Forest is an ensemble learning technique that builds multiple decision trees during training and aggregates their predictions to make a final classification.



```
[ ]  #Random Forest classifier
     rf = RandomForestClassifier()

     # Train
     model = rf.fit(x_train, y_train)

     acc_train = accuracy_score(model.predict(x_train), y_train)
     print(f'Accuracy for training: {acc_train*100:.2f}%')

     # Test
     pred = model.predict(x_test)

     acc = accuracy_score(y_test, pred)
     prec = precision_score(y_test, pred)
```

Multilayer Perceptron (MLP) Classifier: The MLP Classifier is a type of artificial neural network that consists of multiple layers of interconnected nodes (neurons) with nonlinear activation functions. It learns complex patterns in the data through forward and backward propagation of errors during training.



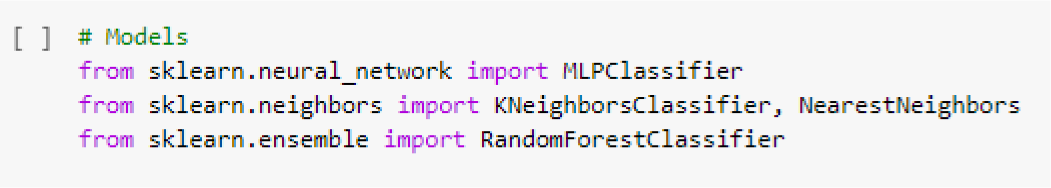
```
#MLP classifier
mlp = MLPClassifier(hidden_layer_sizes=(64,), activation='relu', solver='adam', max_iter=250, random_state=0)

# Train
model = mlp.fit(x_train, y_train)

acc_train = accuracy_score(model.predict(x_train), y_train)
print(f'Accuracy for training: {acc_train*100:.2f}%')

# Test
pred = model.predict(x_test)
```

These machine learning models are trained and evaluated using appropriate techniques to develop an effective predictive model for identifying prospects of future employee attrition.

4.2 Model Implementation

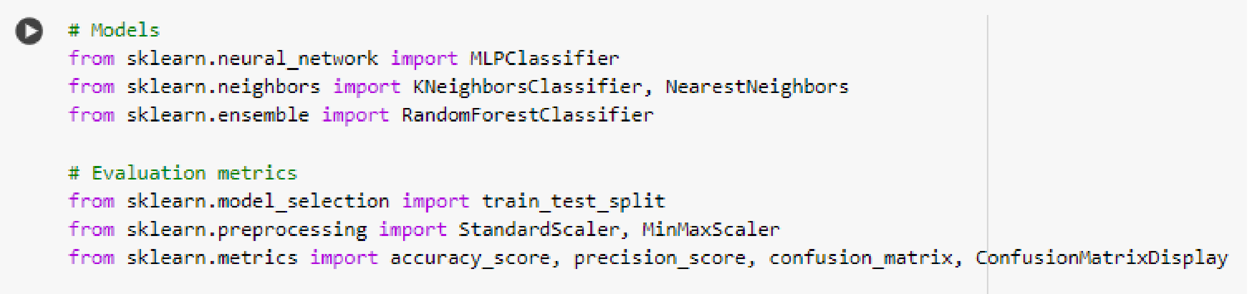
Model implementation for the project is described as follows:

K-Nearest Neighbors (KNN): KNN is implemented using Python libraries such as scikit-learn, with parameters optimized through cross-validation.



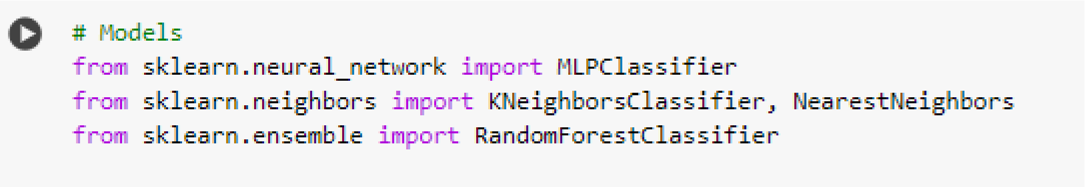
```
# Models
from sklearn.neural_network import MLPClassifier
from sklearn.neighbors import KNeighborsClassifier, NearestNeighbors
from sklearn.ensemble import RandomForestClassifier
```

Random Forest Classification: Random Forest Classification is implemented using scikit-learn, with hyperparameters tuned using techniques such as grid search or random search.



```
# Models
from sklearn.neural_network import MLPClassifier
from sklearn.neighbors import KNeighborsClassifier, NearestNeighbors
from sklearn.ensemble import RandomForestClassifier

# Evaluation metrics
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler, MinMaxScaler
from sklearn.metrics import accuracy_score, precision_score, confusion_matrix, ConfusionMatrixDisplay
```

Multilayer Perceptron (MLP) Classifier: MLP Classifier is implemented using libraries such as scikit-learn, with hyperparameters tuned to optimize performance.



```
# Models
from sklearn.neural_network import MLPClassifier
from sklearn.neighbors import KNeighborsClassifier, NearestNeighbors
from sklearn.ensemble import RandomForestClassifier
```

4.3 Model Evaluation Metrics

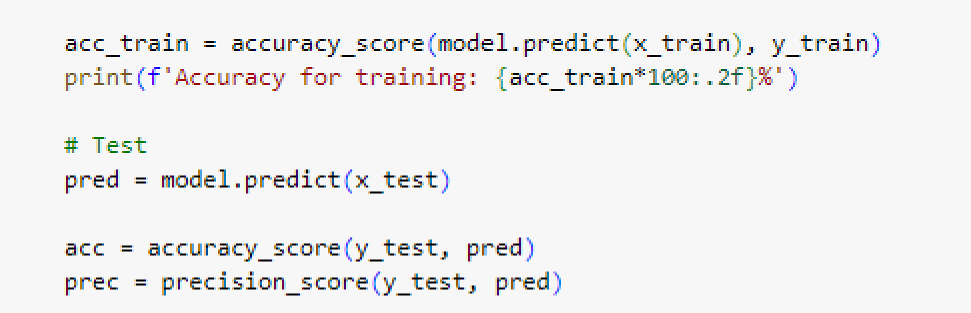
Model evaluation metrics for the project are described as follows:

Accuracy: Accuracy measures the proportion of correctly predicted instances out of the total instances in the dataset.

- **Formula:**

$$\text{Accuracy} = \frac{\text{Number of correctly predicted instances}}{\text{Total number of instances}}$$

It is calculated as the ratio of the number of correctly predicted instances to the total number of instances.

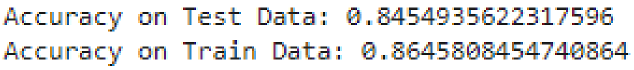


```
acc_train = accuracy_score(model.predict(x_train), y_train)
print(f'Accuracy for training: {acc_train*100:.2f}%')

# Test
pred = model.predict(x_test)

acc = accuracy_score(y_test, pred)
prec = precision_score(y_test, pred)
```

Accuracy for training and testing:

(i) An accuracy score of approximately 84.55% on the test data suggests that the model predicts employee attrition with a high level of accuracy on unseen data.

(ii) An accuracy score of approximately 86.46% on the train data indicates a high level of accuracy in predicting employee attrition on the data used for training the model.



```
Accuracy on Test Data: 0.8454935622317596
Accuracy on Train Data: 0.8645808454740864
```

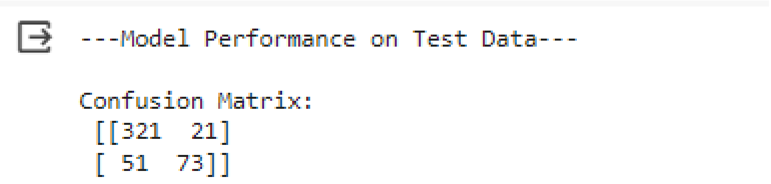
Confusion Matrix: The confusion matrix provides a detailed summary of the performance of a classification model on test data, illustrating the counts of true positive (TP), true negative (TN), false positive (FP), and false negative (FN) predictions.

K Nearest Neighbor (KNN): The confusion matrix for KNN reveals that:

(i) 321 instances are correctly classified as negative (no attrition)

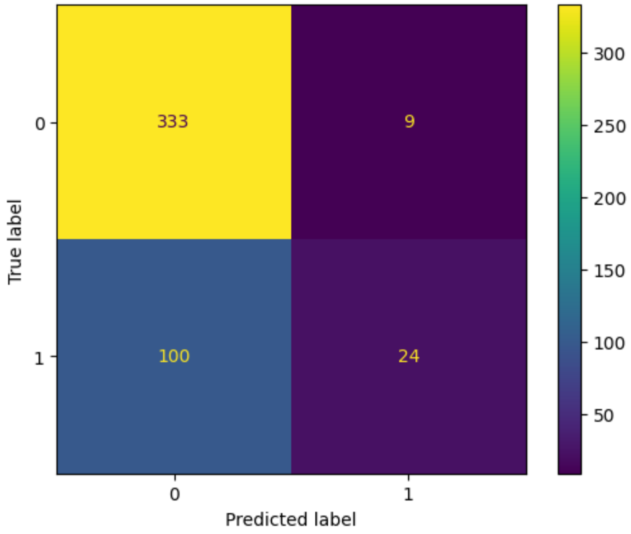
(ii) 21 instances are incorrectly classified as positive (attrition).

(iii) Additionally, 73 instances are correctly classified as positive, and 51 instances are incorrectly classified as negative.

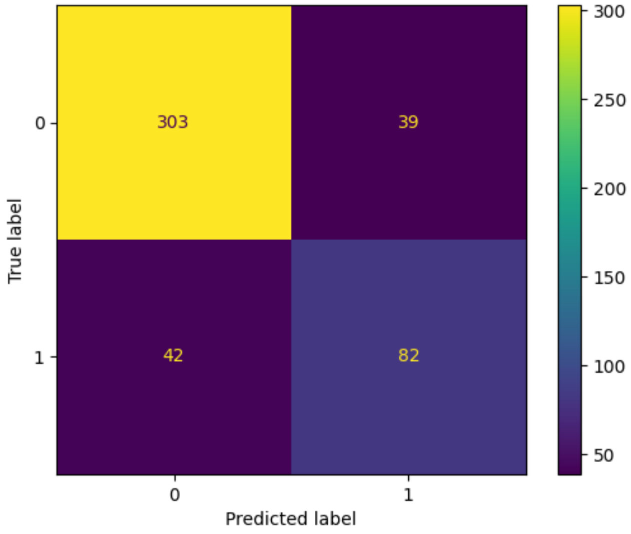


```
---Model Performance on Test Data---

Confusion Matrix:
 [[321  21]
 [ 51  73]]
```

Multilayer Perceptron (MLP): The confusion matrix for MLP shows that 333 instances are correctly classified as negative, while 9 instances are incorrectly classified as positive. Moreover, 24 instances are correctly classified as positive, and 100 instances are incorrectly classified as negative.



Random Forest: The confusion matrix for Random Forest indicates that 303 instances are correctly classified as negative, while 39 instances are incorrectly classified as positive. Furthermore, 82 instances are correctly classified as positive, and 42 instances are incorrectly classified as negative.

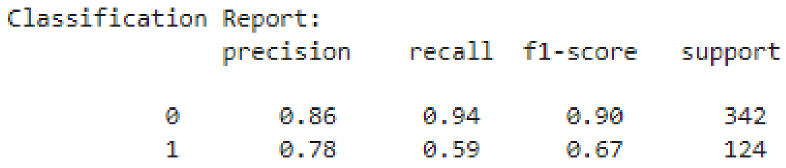


Precision: Precision measures the proportion of true positive predictions (correctly predicted attritions) out of all instances predicted as positive (both true positives and false positives).

- **Formula:**

$$\text{Precision} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive}}$$

It is calculated as the ratio of true positives to the sum of true positives and false positives.



```
Classification Report:
              precision    recall  f1-score   support

           0       0.86      0.94      0.90       342
           1       0.78      0.59      0.67       124
```

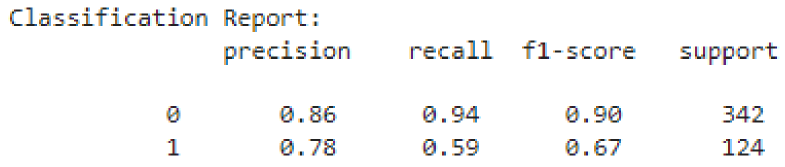
(i) For class 0, the precision is 0.86, indicating that 86% of the instances predicted as class 0 were correctly classified.

(ii) For class 1, the precision is 0.78, indicating that 78% of the instances predicted as class 1 were correctly classified.

Recall: Recall measures the proportion of true positive predictions out of all actual positive instances (true positives and false negatives). It is calculated as the ratio of true positives to the sum of true positives and false negatives.

- **Formula**

$$\text{Recall} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Negative}}$$



```
Classification Report:
              precision    recall  f1-score   support

           0       0.86      0.94      0.90       342
           1       0.78      0.59      0.67       124
```

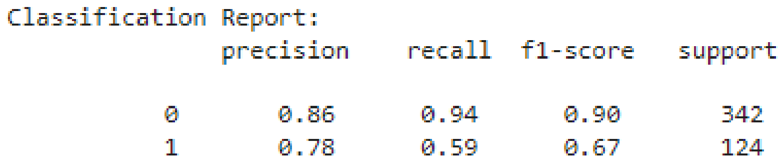
(i) For class 0, the recall is 0.94, indicating that 94% of the actual instances of class 0 were correctly classified.

(ii) For class 1, the recall is 0.59, indicating that 59% of the actual instances of class 1 were correctly classified.

F1 score: F1-score is the harmonic mean of precision and recall. It provides a balance between precision and recall.

- **Formula:**

$$\boxed{\text{F1 score} = 2 * \frac{\text{Precision * Recall}}{\text{Precision + Recall}}}$$



```
Classification Report:
              precision    recall  f1-score   support

           0       0.86      0.94      0.90       342
           1       0.78      0.59      0.67       124
```

(i) For class 0, the F1-score is 0.90, indicating overall performance in classifying instances of class 0.

(ii) For class 1, the F1-score is 0.67, indicating overall performance in classifying instances of class 1.

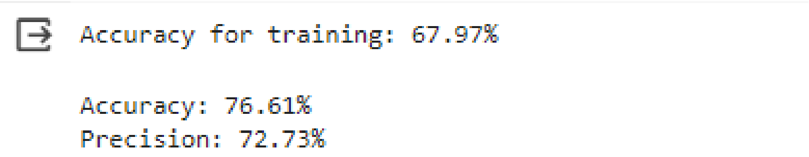
These metrics provide insights into the performance of the classification model for each class, with higher values indicating better performance.

V. RESULTS AND DISCUSSION

5.1 Performance comparison model

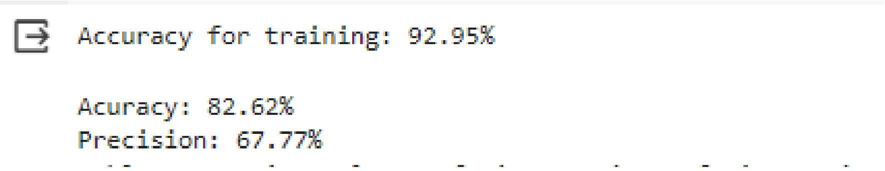
The performance of three machine learning models is compared based on various evaluation metrics.

Multilayer Perceptron (MLP) classifier: For the Multilayer Perceptron (MLP) classifier, the accuracy on the test data is reported as 76.61%, with a precision of 72.73%. The accuracy on the training data is noted as 92.95%.



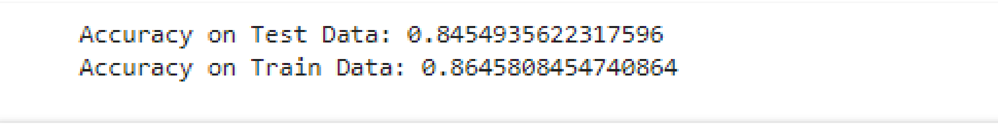
```
Accuracy for training: 67.97%

Accuracy: 76.61%
Precision: 72.73%
```

Random Forest Classification: The Random Forest Classification model achieves an accuracy of 82.62% on the test data, with a precision of 67.77%. The accuracy on the training data for Random Forest Classification is 67.97%.



```
Accuracy for training: 92.95%

Acuracy: 82.62%
Precision: 67.77%
```

K Nearest Neighbor (KNN): The K Nearest Neighbor (KNN) model yields an accuracy of approximately 84.55% on the test data and 86.46% on the training data.



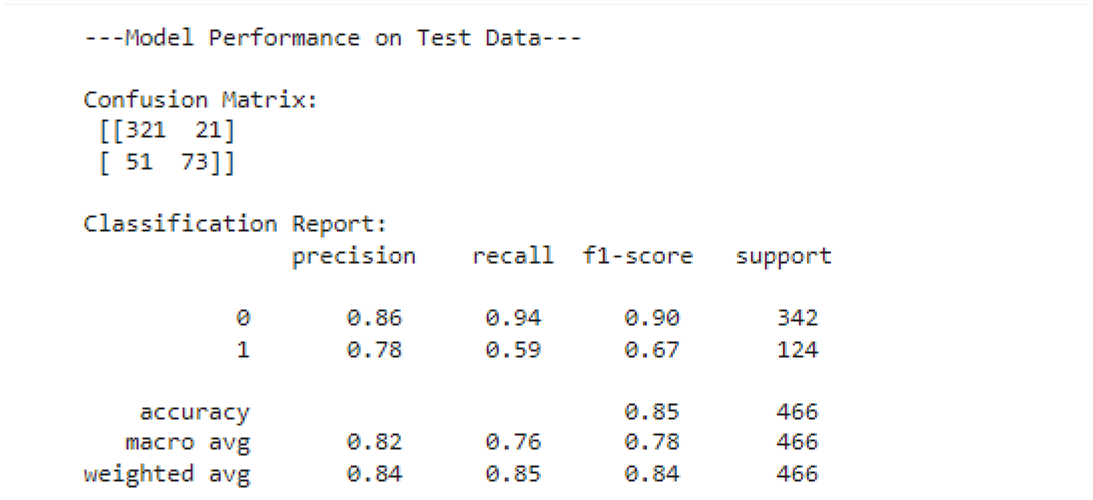
```
Accuracy on Test Data: 0.8454935622317596
Accuracy on Train Data: 0.8645808454740864
```

These comparisons indicate varying performance levels across the models, with MLP achieving the highest accuracy on the training data, while KNN exhibits the highest accuracy on the test data.

5.2 Interpretation of result

The interpretation of results for the project is conducted as follows:

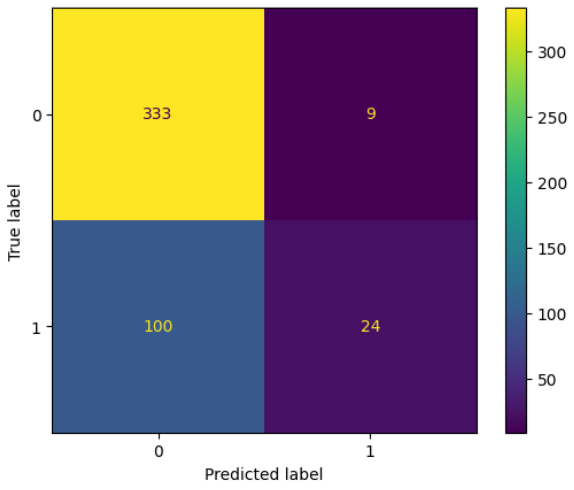
Analysis of Evaluation Metrics: The evaluation metrics, including accuracy, precision, recall, and F1-score, are examined to assess the performance of the machine learning models in predicting employee attrition. Higher values of these metrics indicate better predictive performance.

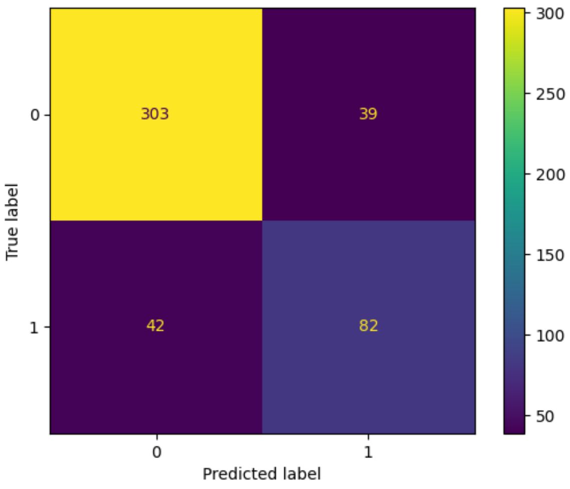


```
---Model Performance on Test Data---

Confusion Matrix:
 [[321  21]
 [ 51  73]]

Classification Report:
              precision    recall  f1-score   support

           0       0.86      0.94      0.90       342
           1       0.78      0.59      0.67       124

    accuracy                           0.85       466
   macro avg       0.82      0.76      0.78       466
weighted avg       0.84      0.85      0.84       466
```

Comparison of Model Performance: The performance of different machine learning models, such as K-Nearest Neighbors (KNN), Multilayer Perceptron (MLP) classifier, and Random Forest Classification, is compared based on accuracy scores, precision, and other evaluation metrics. This comparison helps identify the most effective model for predicting employee attrition.

Analysis of Confusion Matrix: The confusion matrix provides insights into the model's ability to correctly classify instances of attrition and no attrition. It helps identify the counts of true positive, true negative, false positive, and false negative predictions, facilitating a deeper understanding of the model's performance.





Interpretation of Feature Importance: Feature importance analysis identifies the most influential attributes or predictors contributing to the prediction of employee attrition. This analysis helps prioritize important factors affecting attrition and informs decision-making processes for employee retention strategies.

5.3 Insights for HR management:

Insights for HR management derived from this project are outlined below:

- **Attrition Prediction:** Machine learning models developed in this project offer insights into the likelihood of employee attrition based on various attributes such as age, experience, education level, and city of employment. By leveraging these predictive models, HR managers can anticipate potential attrition risks among employees and take proactive measures to mitigate turnover rates.

- **Identification of At-Risk Employees:** Through the analysis of feature importance and model performance metrics, HR managers can identify the most influential factors contributing to attrition. This enables them to pinpoint employees who are at a higher risk of leaving the organization and tailor retention strategies accordingly.

- **Understanding Employee Preferences**: Analysis of demographic attributes such as age, gender, and city of employment provides HR managers with insights into the preferences and characteristics of the workforce. Understanding these preferences helps in designing employee-centric policies and initiatives that enhance employee satisfaction and engagement.

- **Optimization of Recruitment and Retention Strategies:** By analyzing the performance of different machine learning models and their respective predictive capabilities, HR managers can optimize recruitment and retention strategies. For instance, they can prioritize the implementation of targeted recruitment efforts in regions with high attrition rates or focus on retention initiatives tailored to specific employee demographics.

- **Enhanced Decision-Making**: Insights derived from the project enable HR managers to make data-driven decisions regarding workforce planning, talent management, and organizational development. By leveraging predictive analytics, HR managers can allocate resources more effectively, identify talent gaps, and implement proactive measures to foster a positive work environment.

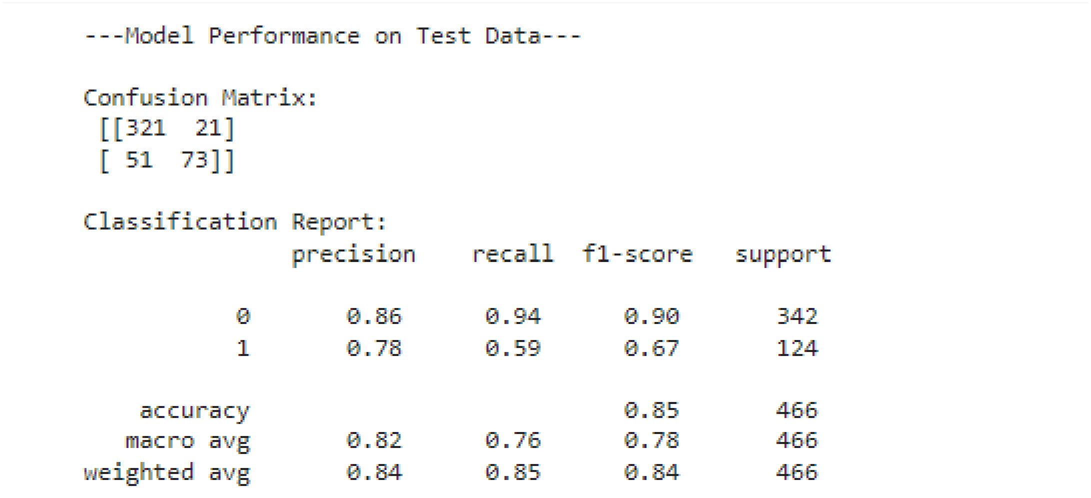
- **Continuous Monitoring and Adaptation**: The predictive models developed in this project provide HR managers with a framework for continuous monitoring of employee attrition trends. By regularly evaluating model performance and updating predictive algorithms, HR managers can adapt their strategies in response to evolving workforce dynamics and market conditions.

VI. CONCLUSION

6.1 Summary of findings

A summary of findings for this project is provided as follows:

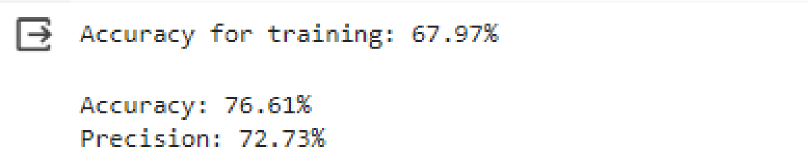
Predictive Modeling Performance: Machine learning models, including K-Nearest Neighbors (KNN), Multilayer Perceptron (MLP) classifier, and Random Forest Classification, were developed to predict employee attrition. Evaluation metrics such as accuracy, precision, recall, and F1-score were utilized to assess the performance of these models.



```
---Model Performance on Test Data---

Confusion Matrix:
 [[321  21]
 [ 51  73]]

Classification Report:
              precision    recall  f1-score   support

           0       0.86      0.94      0.90       342
           1       0.78      0.59      0.67       124

    accuracy                           0.85       466
   macro avg       0.82      0.76      0.78       466
weighted avg       0.84      0.85      0.84       466
```

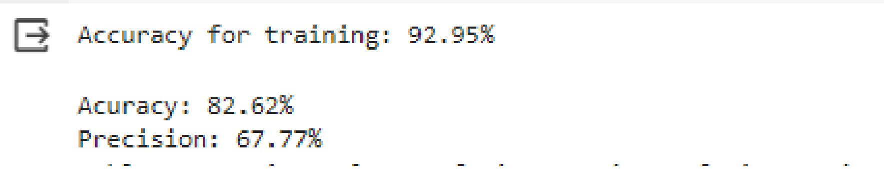
Model Comparison: The performance of different machine learning models was compared based on various evaluation metrics. Insights were drawn from the comparison to identify the most effective model for predicting employee attrition in the context of the dataset.

- **Multilayer Perceptron (MLP) classifier:**



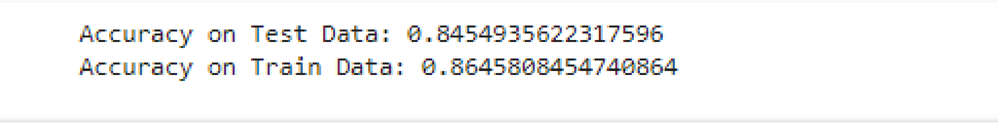
```
Accuracy for training: 67.97%

Accuracy: 76.61%
Precision: 72.73%
```

- **Random Forest Classification:**



```
Accuracy for training: 92.95%

Acuracy: 82.62%
Precision: 67.77%
```

- **K Nearest Neighbor (KNN):**



```
Accuracy on Test Data: 0.8454935622317596
Accuracy on Train Data: 0.8645808454740864
```

Feature Importance Analysis: Feature importance analysis was conducted to identify the most influential attributes contributing to employee attrition prediction. This analysis provided insights into the key factors driving attrition within the organization.

Insights into Employee Demographics: Analysis of demographic attributes such as age, gender, education level, and city of employment offered insights into the composition and characteristics of the workforce. Understanding these demographic trends facilitated the development of targeted retention strategies.

Recommendations for HR Management: Based on the findings, recommendations were made for HR management to optimize recruitment and retention strategies, enhance employee satisfaction and engagement, and foster a positive work environment conducive to employee retention.

6.2 Limitations

Limitations for this project are outlined as follows:

- **Data Quality:** The project's effectiveness may be constrained by the quality and completeness of the dataset. Inaccuracies, inconsistencies, or missing values within the data can impact the performance and reliability of the predictive models.
- **Sample Size:** The size of the dataset may limit the generalizability of the findings. A small sample size may not adequately capture the diverse range of factors contributing to employee attrition, leading to biased or unreliable results.
- **Feature Selection:** The selection of predictive features may influence the performance of the models. Limited availability of relevant features or incorrect assumptions about feature importance can affect the accuracy of attrition prediction.
- **Model Assumptions:** The machine learning models employed in the project are based on certain assumptions about the underlying data distribution and relationships between variables. Deviations from these assumptions may impact the models' predictive capabilities.
- **Temporal Dynamics:** The dataset's temporal dynamics may not be fully captured, potentially leading to model inaccuracies. Changes in workforce dynamics, market conditions, or organizational policies over time may not be adequately reflected in the data, affecting the models' predictive performance.
- **External Factors:** The predictive models may not account for external factors influencing employee attrition, such as industry trends, economic conditions, or competitive pressures. Failure to consider these external factors may limit the models' predictive accuracy and relevance.
- **Ethical Considerations**: The use of predictive models in HR management raises ethical considerations related to fairness, bias, and privacy. Inadequate consideration of these ethical concerns may result in unintended consequences or negative impacts on employees.

VII. FUTURE SCOPE

(i) **Expanded Data Collection:** Incorporating additional attributes or capturing data over a longer period can provide a more comprehensive understanding of attrition factors.

(ii) **Advanced Modeling Techniques:** Exploring advanced algorithms like gradient boosting or deep learning, and ensemble methods can improve prediction accuracy.

(iii) **Dynamic Modeling:** Developing models that adapt to changing workforce dynamics ensures relevance and effectiveness over time.

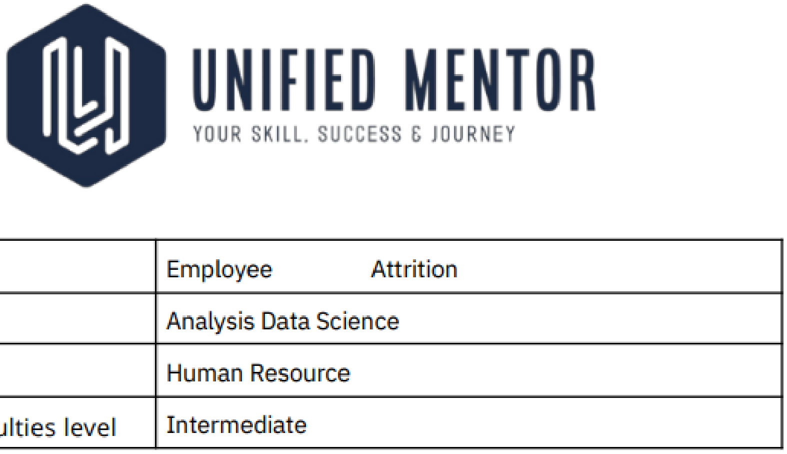
(iv) **Predictive Analytics Integration:** Integrating attrition prediction into HR systems facilitates real-time monitoring and proactive intervention.

VIII. APPENDIX

A. Data collection:

The dataset utilized in this project was provided by Unified Mentor, the company where the author is currently undertaking an internship. The dataset specifically pertains to employee attrition and contains 4653 instances and 9 features.[1]

The dataset extraction screenshot was illustrated below for easy reference:



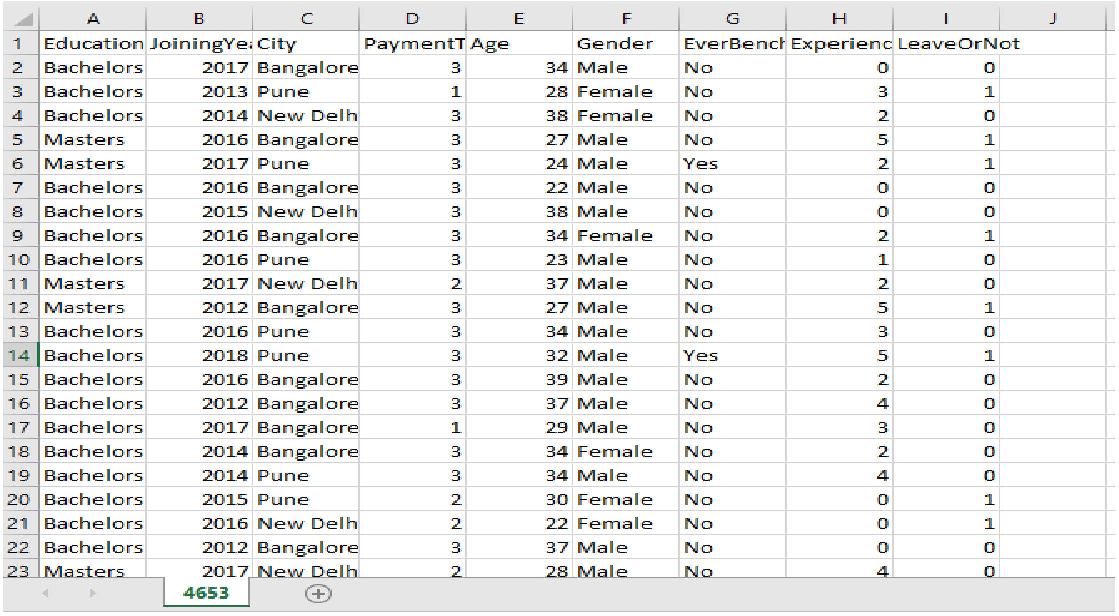
UNIFIED MENTOR
YOUR SKILL, SUCCESS & JOURNEY

Project Title	Employee Attrition
Technologies	Analysis Data Science
Domain	Human Resource
Project Difficulties level	Intermediate

Fig A. Data Collection

B. Data storage

The Employee Attrition dataset was extracted and stored it in CSV format for analysis



	A	B	C	D	E	F	G	H	I	J
1	Education	JoiningYe	City	PaymentT	Age	Gender	EverBench	Experienc	LeaveOrNot	
2	Bachelors	2017	Bangalore	3	34	Male	No	0	0	
3	Bachelors	2013	Pune	1	28	Female	No	3	1	
4	Bachelors	2014	New Delh	3	38	Female	No	2	0	
5	Masters	2016	Bangalore	3	27	Male	No	5	1	
6	Masters	2017	Pune	3	24	Male	Yes	2	1	
7	Bachelors	2016	Bangalore	3	22	Male	No	0	0	
8	Bachelors	2015	New Delh	3	38	Male	No	0	0	
9	Bachelors	2016	Bangalore	3	34	Female	No	2	1	
10	Bachelors	2016	Pune	3	23	Male	No	1	0	
11	Masters	2017	New Delh	2	37	Male	No	2	0	
12	Masters	2012	Bangalore	3	27	Male	No	5	1	
13	Bachelors	2016	Pune	3	34	Male	No	3	0	
14	Bachelors	2018	Pune	3	32	Male	Yes	5	1	
15	Bachelors	2016	Bangalore	3	39	Male	No	2	0	
16	Bachelors	2012	Bangalore	3	37	Male	No	4	0	
17	Bachelors	2017	Bangalore	1	29	Male	No	3	0	
18	Bachelors	2014	Bangalore	3	34	Female	No	2	0	
19	Bachelors	2014	Pune	3	34	Male	No	4	0	
20	Bachelors	2015	Pune	2	30	Female	No	0	1	
21	Bachelors	2016	New Delh	2	22	Female	No	0	1	
22	Bachelors	2012	Bangalore	3	37	Male	No	0	0	
23	Masters	2017	New Delh	2	28	Male	No	4	0	

Fig B. Data storage

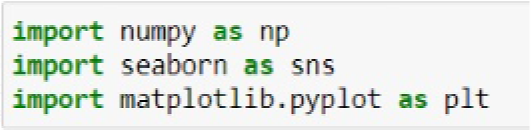
C. Source code in Python

The Python programming language is used for analyzing the Employee Attrition project.

➢ Importing necessary libraries

importing pandas as pd
importing numpy as np
importing seaborn as sns

```
import pandas as pd
```

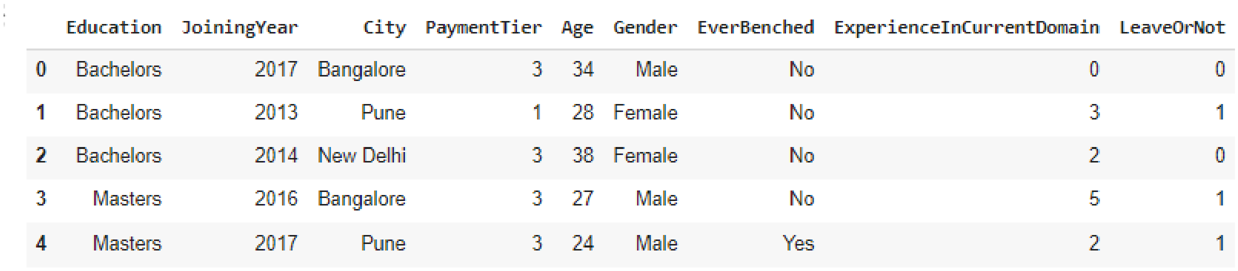


```
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
```

➢ Read the dataset

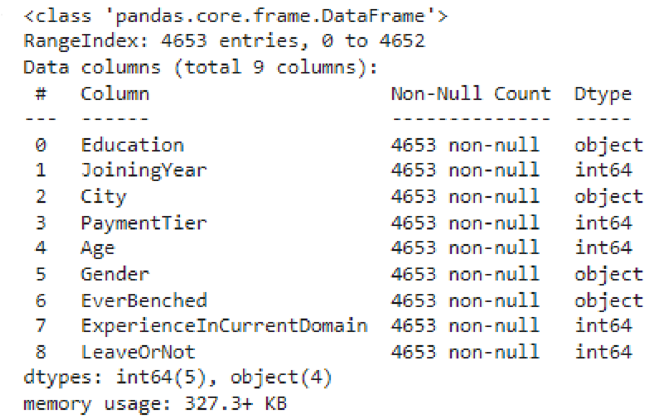
```
[ ]  #Dataset importion
     df = pd.read_csv('/content/Employee.csv')
     df.head()
```

➢ Visualizing the dataset



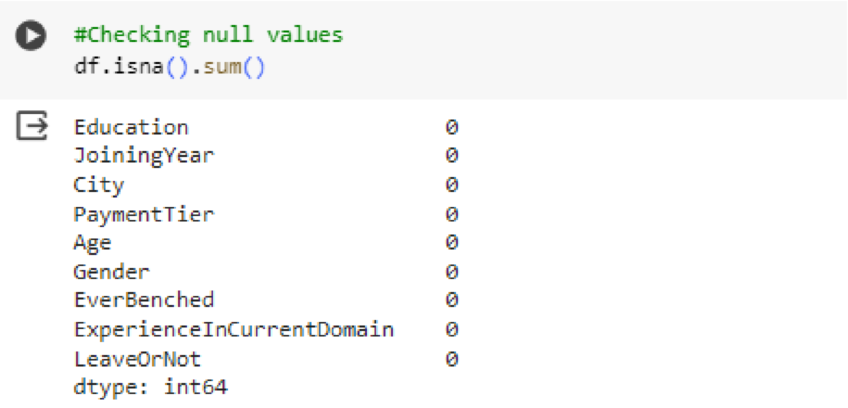
	Education	JoiningYear	City	PaymentTier	Age	Gender	EverBenched	ExperienceInCurrentDomain	LeaveOrNot
0	Bachelors	2017	Bangalore	3	34	Male	No	0	0
1	Bachelors	2013	Pune	1	28	Female	No	3	1
2	Bachelors	2014	New Delhi	3	38	Female	No	2	0
3	Masters	2016	Bangalore	3	27	Male	No	5	1
4	Masters	2017	Pune	3	24	Male	Yes	2	1

➢ Dataset information



```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4653 entries, 0 to 4652
Data columns (total 9 columns):
 #   Column                     Non-Null Count  Dtype
---  ------                     --------------  -----
 0   Education                  4653 non-null   object
 1   JoiningYear                4653 non-null   int64
 2   City                       4653 non-null   object
 3   PaymentTier                4653 non-null   int64
 4   Age                        4653 non-null   int64
 5   Gender                     4653 non-null   object
 6   EverBenched                4653 non-null   object
 7   ExperienceInCurrentDomain  4653 non-null   int64
 8   LeaveOrNot                 4653 non-null   int64
dtypes: int64(5), object(4)
memory usage: 327.3+ KB
```

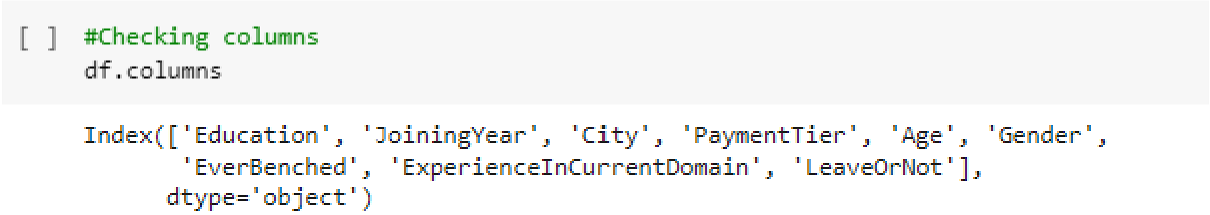
➢ Checking null values



```
#Checking null values
df.isna().sum()
```

```
Education                    0
JoiningYear                  0
City                         0
PaymentTier                  0
Age                          0
Gender                       0
EverBenched                  0
ExperienceInCurrentDomain    0
LeaveOrNot                   0
dtype: int64
```

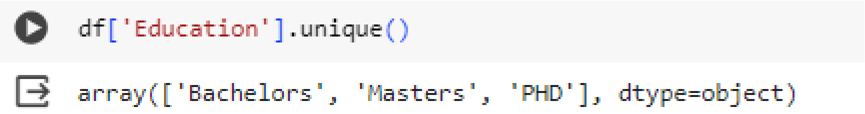
➢ Checking columns



```
[ ] #Checking columns
    df.columns
```

```
Index(['Education', 'JoiningYear', 'City', 'PaymentTier', 'Age', 'Gender',
       'EverBenched', 'ExperienceInCurrentDomain', 'LeaveOrNot'],
      dtype='object')
```

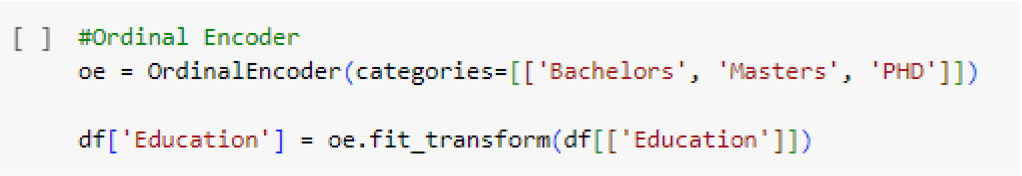
➢ Checking unique values



```
df['Education'].unique()
```

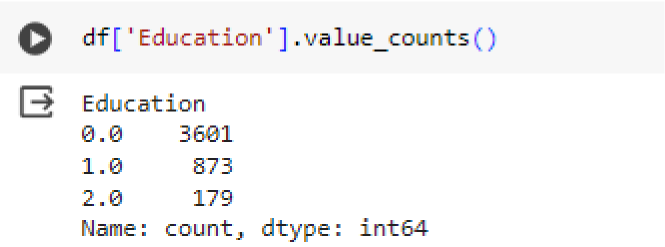
```
array(['Bachelors', 'Masters', 'PHD'], dtype=object)
```

➢ Ordinal Encoder



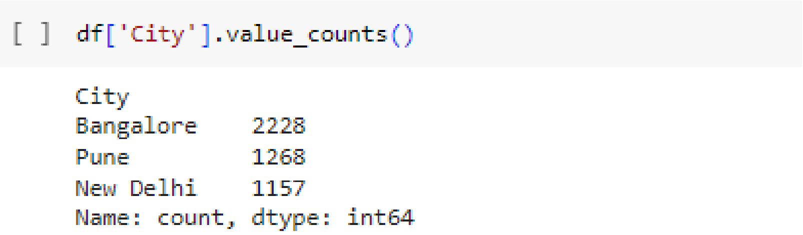
```
[ ] #Ordinal Encoder
    oe = OrdinalEncoder(categories=[['Bachelors', 'Masters', 'PHD']])

    df['Education'] = oe.fit_transform(df[['Education']])
```

➢ Checking value counts

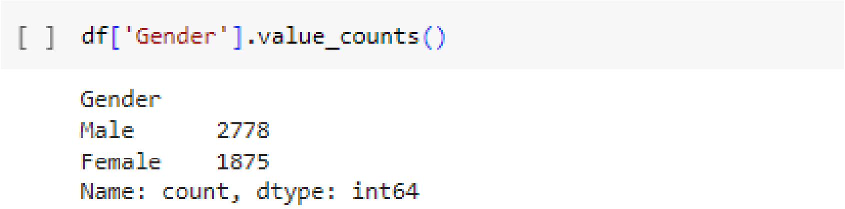


```
df['Education'].value_counts()
```

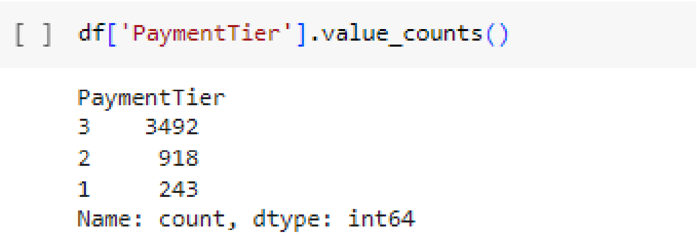
```
Education
0.0    3601
1.0     873
2.0     179
Name: count, dtype: int64
```



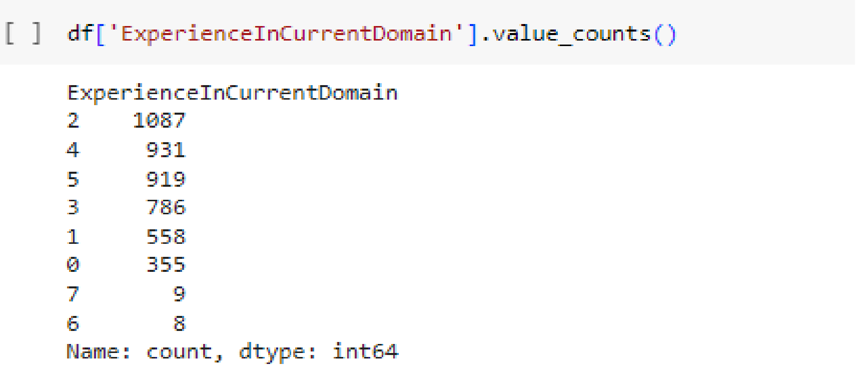
```
[ ]  df['City'].value_counts()

     City
     Bangalore    2228
     Pune         1268
     New Delhi    1157
     Name: count, dtype: int64
```



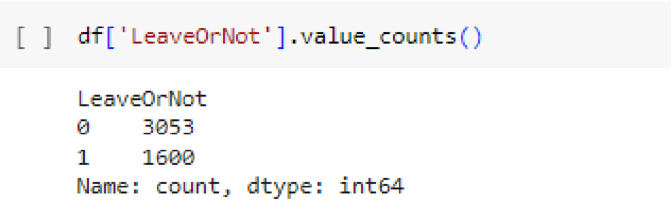
```
[ ]  df['Gender'].value_counts()

     Gender
     Male      2778
     Female    1875
     Name: count, dtype: int64
```



```
[ ]  df['PaymentTier'].value_counts()

     PaymentTier
     3    3492
     2     918
     1     243
     Name: count, dtype: int64
```

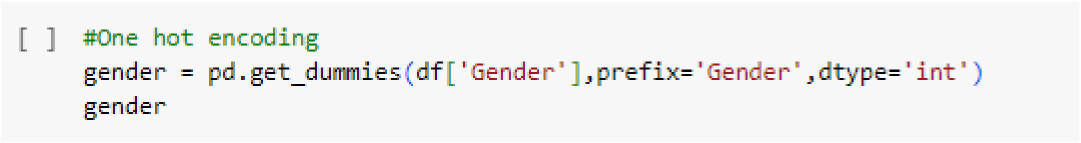


```
[ ]  df['ExperienceInCurrentDomain'].value_counts()

     ExperienceInCurrentDomain
     2    1087
     4     931
     5     919
     3     786
     1     558
     0     355
     7       9
     6       8
     Name: count, dtype: int64
```

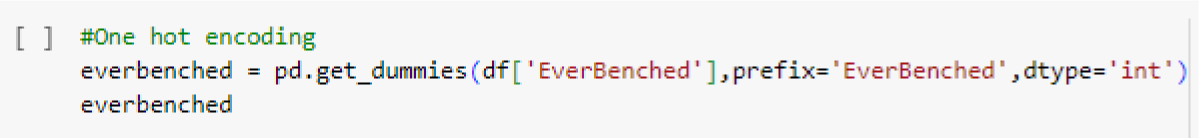


```
[ ]  df['LeaveOrNot'].value_counts()

     LeaveOrNot
     0    3053
     1    1600
     Name: count, dtype: int64
```

- **One-hot encoding:**

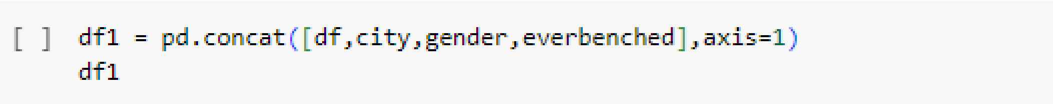


```
[ ]  #One hot encoding
     gender = pd.get_dummies(df['Gender'],prefix='Gender',dtype='int')
     gender
```



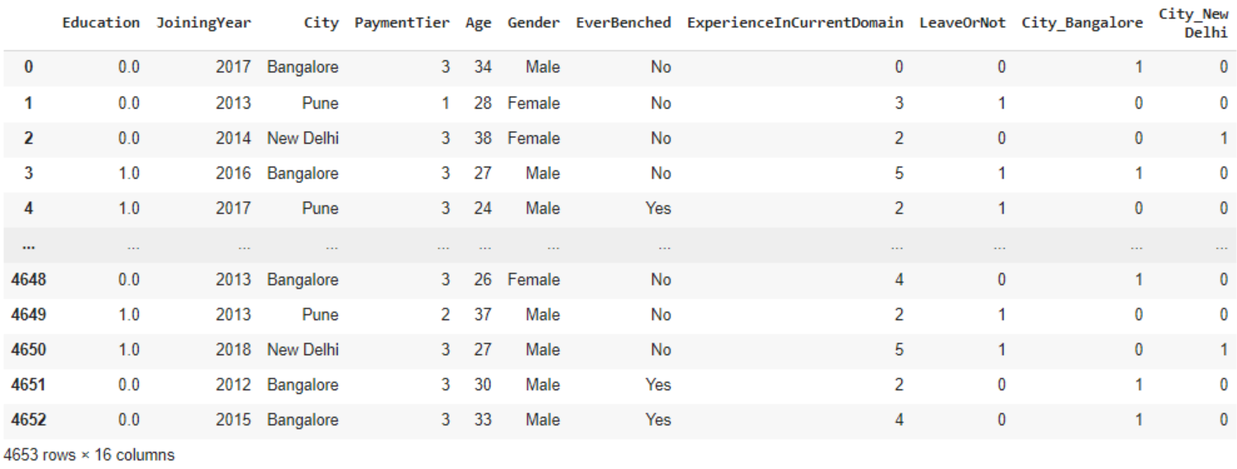
```
[ ] #One hot encoding
    everbenched = pd.get_dummies(df['EverBenched'],prefix='EverBenched',dtype='int')
    everbenched
```

➢ Concatenating Data Frame:



```
[ ] df1 = pd.concat([df,city,gender,everbenched],axis=1)
    df1
```

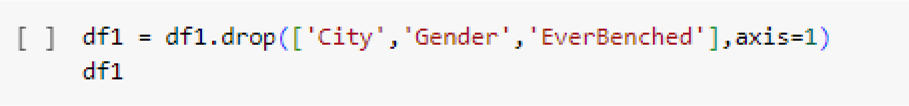
➢ Visualizing the concatenation:



	Education	JoiningYear	City	PaymentTier	Age	Gender	EverBenched	ExperienceInCurrentDomain	LeaveOrNot	City_Bangalore	City_New Delhi
0	0.0	2017	Bangalore	3	34	Male	No	0	0	1	0
1	0.0	2013	Pune	1	28	Female	No	3	1	0	0
2	0.0	2014	New Delhi	3	38	Female	No	2	0	0	1
3	1.0	2016	Bangalore	3	27	Male	No	5	1	1	0
4	1.0	2017	Pune	3	24	Male	Yes	2	1	0	0
...	...	...	...	...	...	...	...	...	...	...	...
4648	0.0	2013	Bangalore	3	26	Female	No	4	0	1	0
4649	1.0	2013	Pune	2	37	Male	No	2	1	0	0
4650	1.0	2018	New Delhi	3	27	Male	No	5	1	0	1
4651	0.0	2012	Bangalore	3	30	Male	Yes	2	0	1	0
4652	0.0	2015	Bangalore	3	33	Male	Yes	4	0	1	0

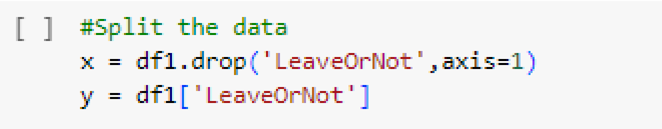
4653 rows × 16 columns

➢ Dropping the unnecessary columns



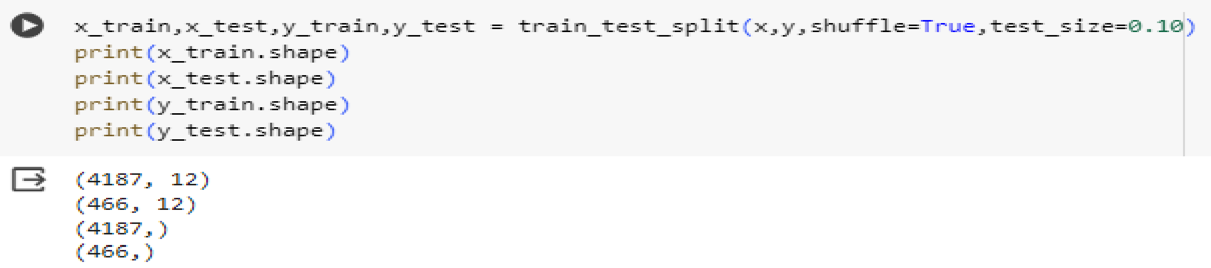
```
[ ] df1 = df1.drop(['City','Gender','EverBenched'],axis=1)
    df1
```

➢ Splitting the data



```
[ ] #Split the data
    x = df1.drop('LeaveOrNot',axis=1)
    y = df1['LeaveOrNot']
```

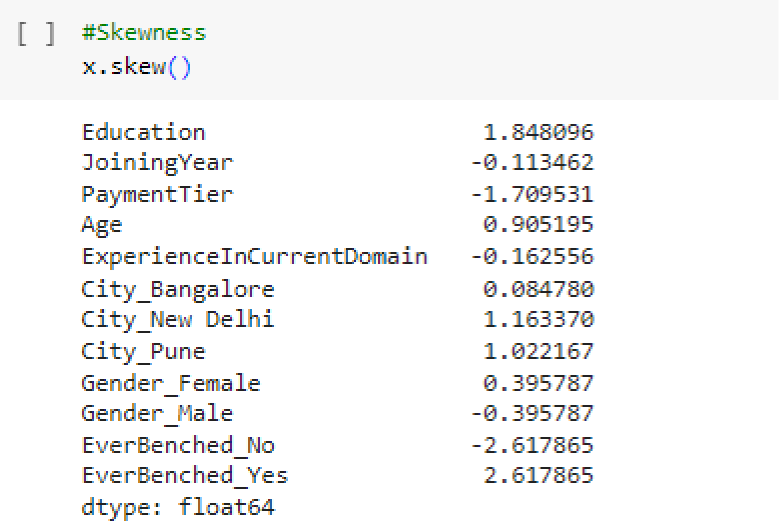
➢ Training the data



```
x_train,x_test,y_train,y_test = train_test_split(x,y,shuffle=True,test_size=0.10)
print(x_train.shape)
print(x_test.shape)
print(y_train.shape)
print(y_test.shape)
```

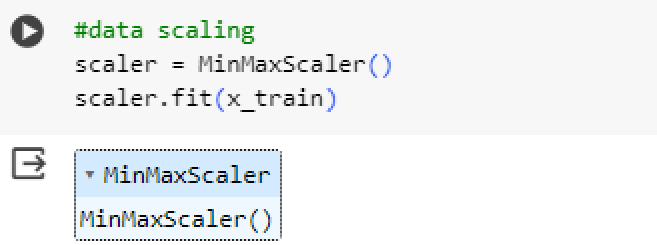
```
(4187, 12)
(466, 12)
(4187,)
(466,)
```

- **Skeweness**



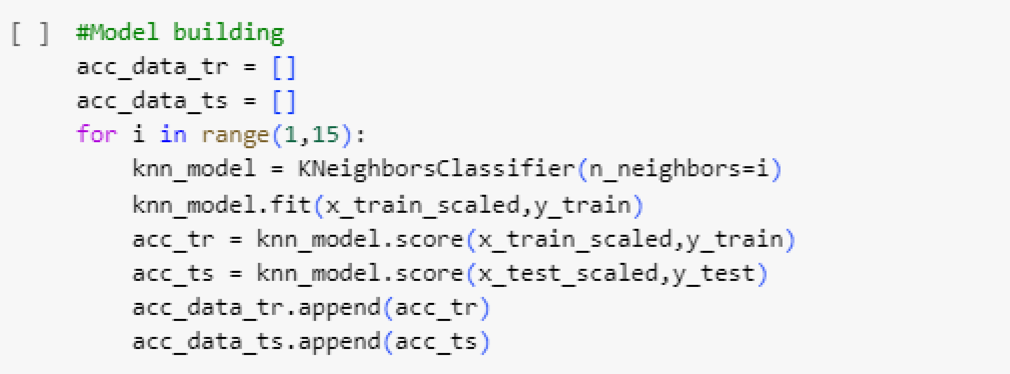
```
[ ] #Skewness
    x.skew()

    Education                    1.848096
    JoiningYear                 -0.113462
    PaymentTier                 -1.709531
    Age                          0.905195
    ExperienceInCurrentDomain   -0.162556
    City_Bangalore               0.084780
    City_New Delhi               1.163370
    City_Pune                    1.022167
    Gender_Female                0.395787
    Gender_Male                 -0.395787
    EverBenched_No              -2.617865
    EverBenched_Yes              2.617865
    dtype: float64
```

- **Data scaling**



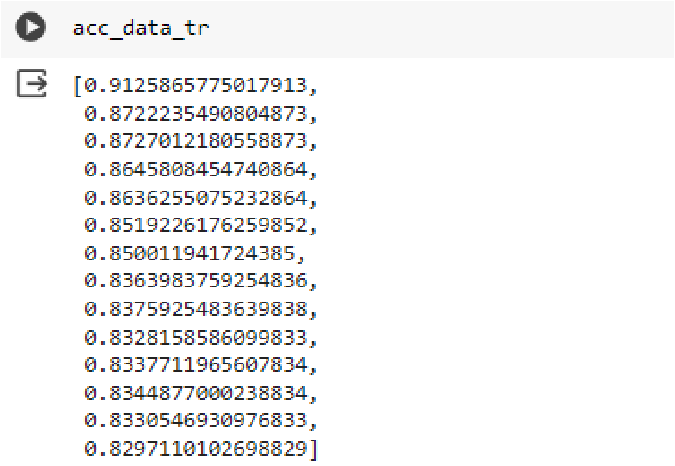
```
#data scaling
scaler = MinMaxScaler()
scaler.fit(x_train)

▾ MinMaxScaler
MinMaxScaler()
```

- **Model building**



```
[ ] #Model building
    acc_data_tr = []
    acc_data_ts = []
    for i in range(1,15):
        knn_model = KNeighborsClassifier(n_neighbors=i)
        knn_model.fit(x_train_scaled,y_train)
        acc_tr = knn_model.score(x_train_scaled,y_train)
        acc_ts = knn_model.score(x_test_scaled,y_test)
        acc_data_tr.append(acc_tr)
        acc_data_ts.append(acc_ts)
```

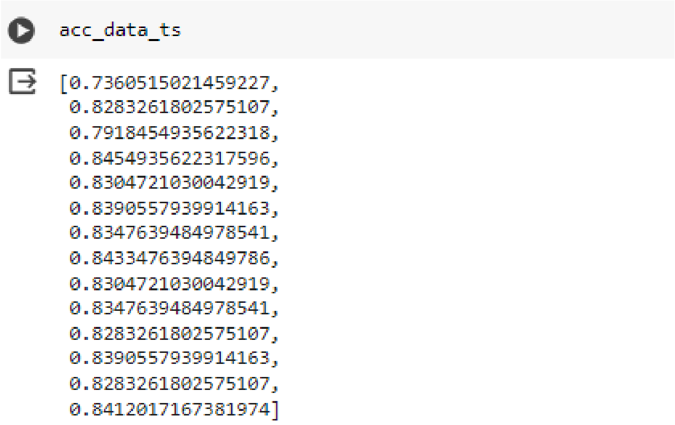
➢ Model training accuracy



```
acc_data_tr
```

```
[0.9125865775017913,
 0.8722235490804873,
 0.8727012180558873,
 0.8645808454740864,
 0.8636255075232864,
 0.8519226176259852,
 0.850011941724385,
 0.8363983759254836,
 0.8375925483639838,
 0.8328158586099833,
 0.8337711965607834,
 0.8344877000238834,
 0.8330546930976833,
 0.8297110102698829]
```

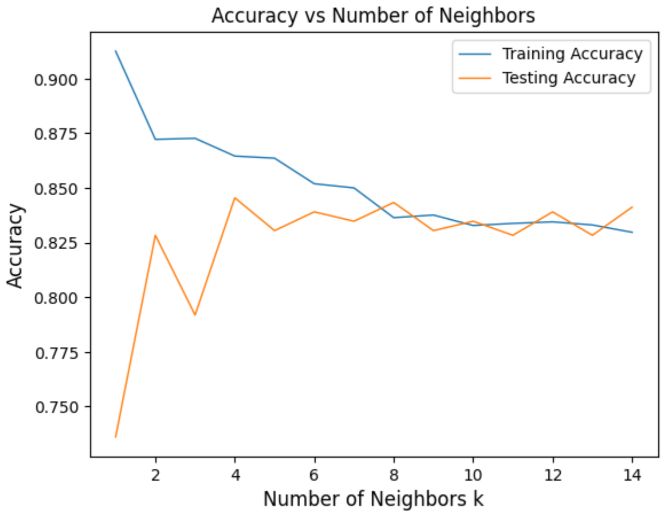
➢ Model testing accuracy



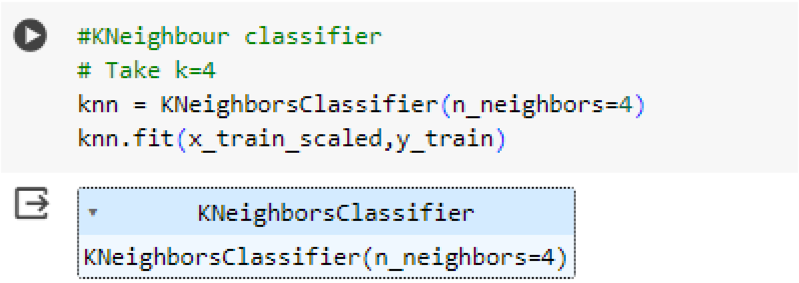
```
acc_data_ts
```

```
[0.7360515021459227,
 0.8283261802575107,
 0.7918454935622318,
 0.8454935622317596,
 0.8304721030042919,
 0.8390557939914163,
 0.8347639484978541,
 0.8433476394849786,
 0.8304721030042919,
 0.8347639484978541,
 0.8283261802575107,
 0.8390557939914163,
 0.8283261802575107,
 0.8412017167381974]
```

Training and testing accuracy plotting



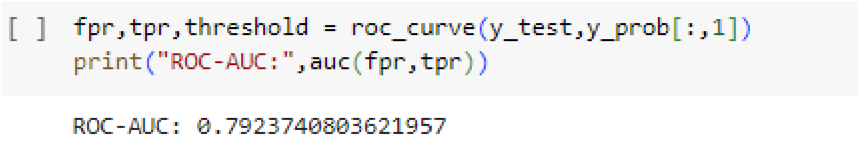
➢ KNeighbour Classifier



```
#KNeighbour classifier
# Take k=4
knn = KNeighborsClassifier(n_neighbors=4)
knn.fit(x_train_scaled,y_train)
```

```
▾        KNeighborsClassifier
KNeighborsClassifier(n_neighbors=4)
```

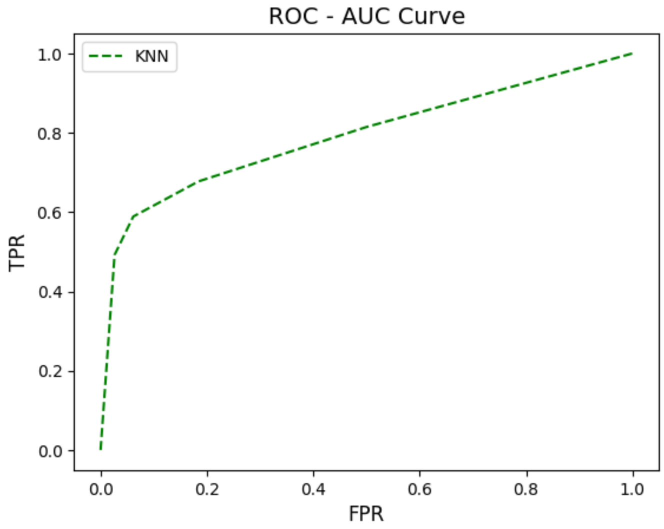
➢ ROC-Curve score



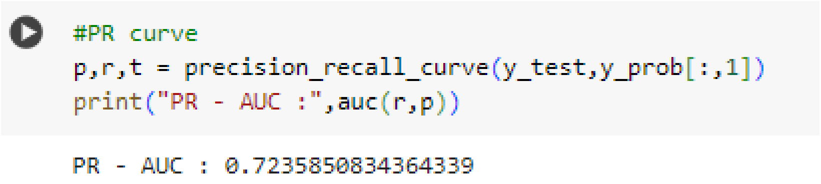
```
[ ]  fpr,tpr,threshold = roc_curve(y_test,y_prob[:,1])
     print("ROC-AUC:",auc(fpr,tpr))
```

```
ROC-AUC: 0.7923740803621957
```

➢ ROC-Curve plotted



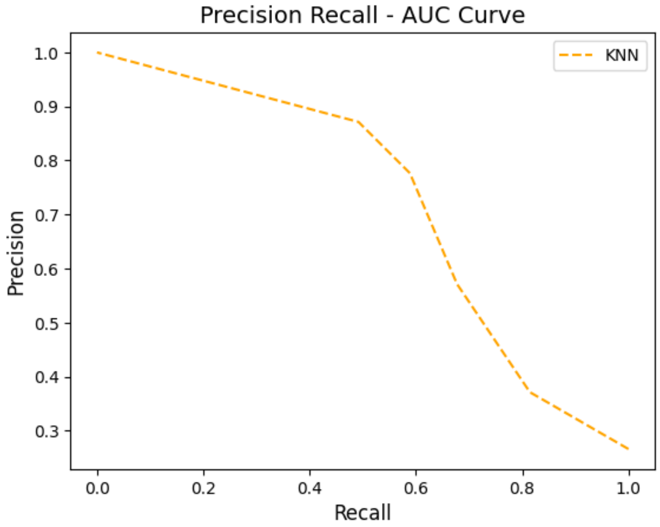
➢ PR-Curve score



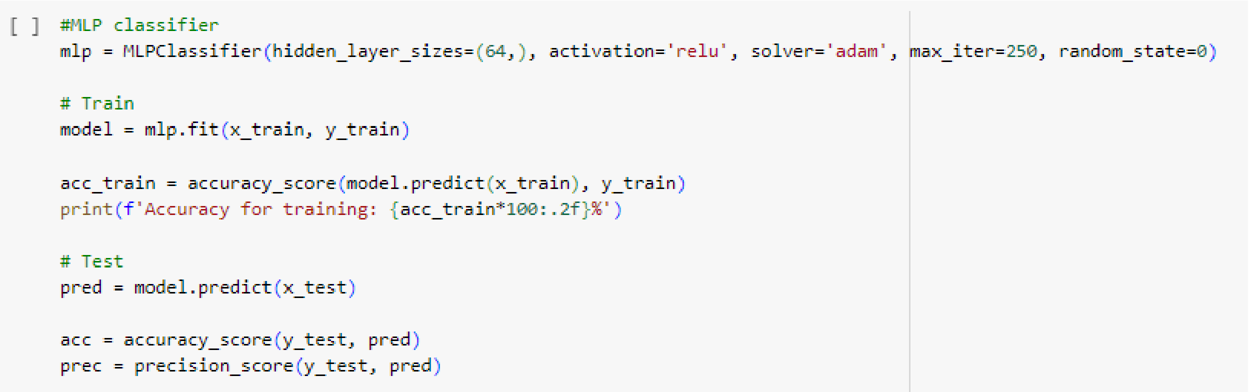
```
#PR curve
p,r,t = precision_recall_curve(y_test,y_prob[:,1])
print("PR - AUC :",auc(r,p))
```

```
PR - AUC : 0.7235850834364339
```

- **PR-Curve plotted**

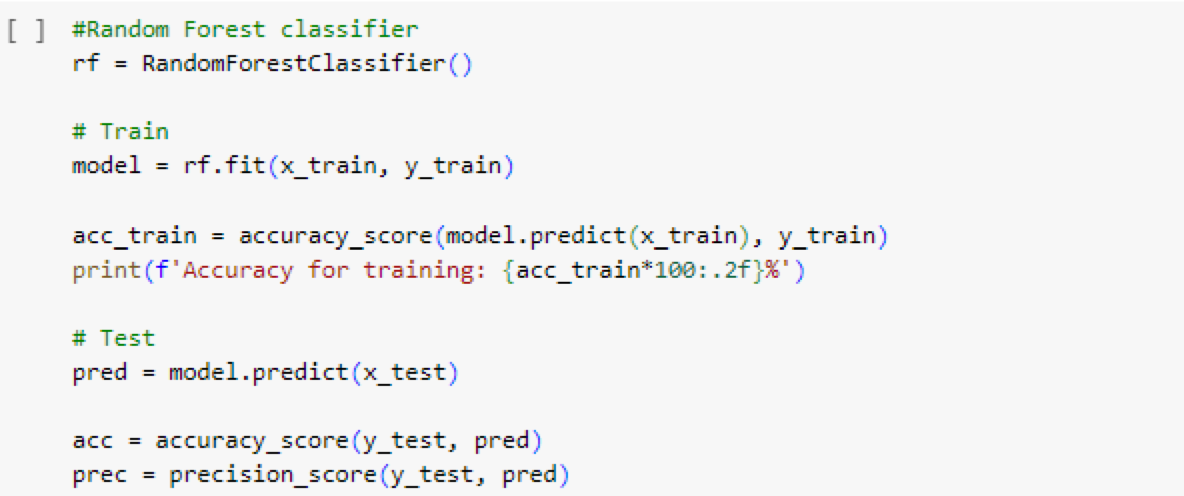


- **MLP Classifier**



```
#MLP classifier
mlp = MLPClassifier(hidden_layer_sizes=(64,), activation='relu', solver='adam', max_iter=250, random_state=0)

# Train
model = mlp.fit(x_train, y_train)

acc_train = accuracy_score(model.predict(x_train), y_train)
print(f'Accuracy for training: {acc_train*100:.2f}%')

# Test
pred = model.predict(x_test)

acc = accuracy_score(y_test, pred)
prec = precision_score(y_test, pred)
```

- **Random Forest classifier**



```
#Random Forest classifier
rf = RandomForestClassifier()

# Train
model = rf.fit(x_train, y_train)

acc_train = accuracy_score(model.predict(x_train), y_train)
print(f'Accuracy for training: {acc_train*100:.2f}%')

# Test
pred = model.predict(x_test)

acc = accuracy_score(y_test, pred)
prec = precision_score(y_test, pred)
```

IX. BIBLIOGRAPHY

[1] https://unifiedmentor.podia.com/view/courses/data-analytics-internship-8-weeks/2313051-project-allocation-day-1/7357012-project-selection-choose-projects-as-per-your-level

REFERENCES

- **https://scikit-learn.org/stable/modules/generated/sklearn.neural_network.MLPClassifier.html**
- **https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html**
- **https://www.ibm.com/topics/knn#:~:text=The%20k%2Dnearest%20neighbors%20(KNN)%20algorithm%20is%20a%20non,used%20in%20machine%20learning%20today.**