**IPHONE PURCHASE PREDICTION USING MACHINE LEARNING**

**Prepared by: Tarun Mondal**

**Table of Contents**

[Objective](#_n2yopppgmbr5) 3

[Methodology](#_4al1ey1rmq13) 3

[End-to-end process with solution architecture](#_8n8seygnqb6h) 3

[Importing libraries in Python](#_j4er0b6hqjd3) 3

[Data exploration](#_nq6j5jgfuivr) 4

[Data preprocessing](#_5bdlpe9xqeme) 5

[Exploratory data analysis (EDA)](#_4j1he0cm1jgd) 6

[Model development](#_xxsr8bt48lqc) 9

[Challenges faced](#_n66ncrpgyp64) 13

[Complexity level](#_dhl7r1wlxkiy) 13

# Objective

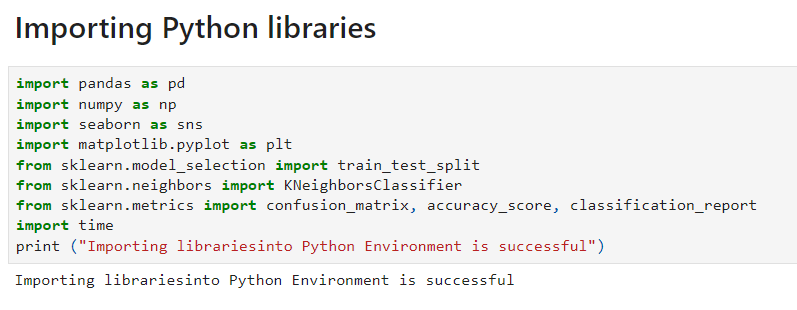
The objective of this project is to predict iPhone purchases based on age, gender and salary using the KNN Classifier.

# Methodology

The primary purpose of this project is to predict whether an individual will purchase an iPhone based on several factors such as age and salary. The dataset consists of 380 observations and 3 key variables: Age, Salary, and Purchase of iPhone (the target variable). Given that the target variable is categorical (binary: 0 or 1), a classification model is appropriate for this task. K-Nearest Neighbours (KNN) classification model has been implemented to predict iPhone purchases based on the input features. The model was developed using the Python programming language within the Jupyter Notebook environment.

# End-to-end process with solution architecture

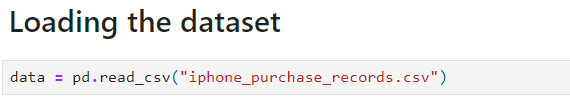
## Importing libraries in Python

****

**Figure 1: Importing libraries in Python**

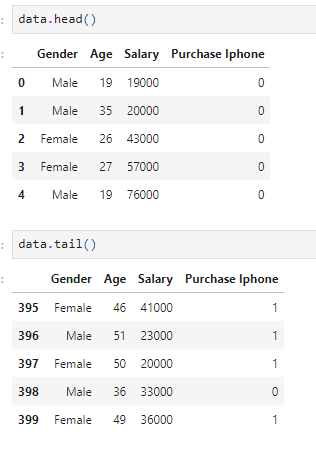
Pandas library has been imported into Python for data loading and manipulation, whereas Seaborn and Matplotlib libraries have been used for data visualisations. For developing machine learning models (like K-Nearest Neighbors), the KNeighborsClassifier module has been imported from the scikit-learn framework. Additionally, essential modules for splitting the dataset (train\_test\_split) and evaluating the model using metrics such as the confusion matrix, accuracy score, and classification report have also been imported from the sklearn.metrics module.

## Data exploration

****

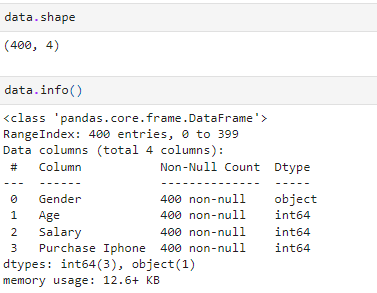
**Figure 2: Loading the dataset**

The dataset has been loaded in Python (Jupyter Notebook Environment) using the ‘read’ function from the pandas library. Try-catch block has been used for better handling of errors.

****

**Figure 3: Head and tail of the dataset**

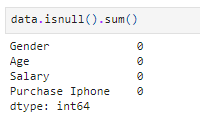
Figure 3 displays the ‘iphone\_purchase\_records’ dataset, retrieved using the head () and tail () functions from the Pandas library. These functions provide a quick overview of the dataset, showing the first and last few records for a better understanding of the data structure.

****

**Figure 4: Shape and info of the dataset**

The shape and info of the dataset revealed that the dataset contains 400 valid observations and 4 variables (**Refer to Figure 4**).

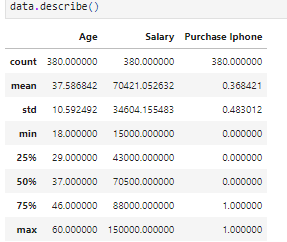
## Data preprocessing

****

**Figure 5: Data preprocessing**

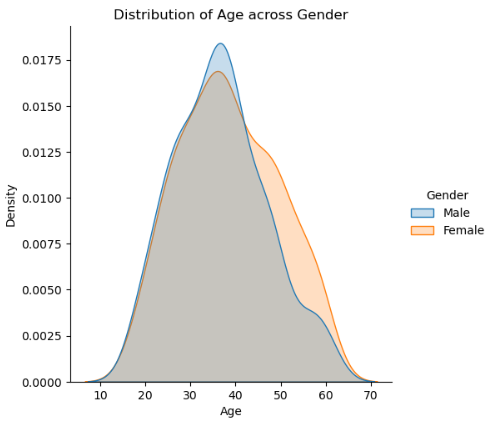
The dataset has been checked for missing values using the isnull(). sum () function, as shown in Figure 5. The output indicates that there are no missing values in any of the columns: ‘Gender’, ‘Age’, ‘Salary’, and ‘Purchase iPhone’. This suggests that the dataset is clean, and no further data preprocessing steps, such as handling missing values, are required at this stage.

## Exploratory data analysis (EDA)

****

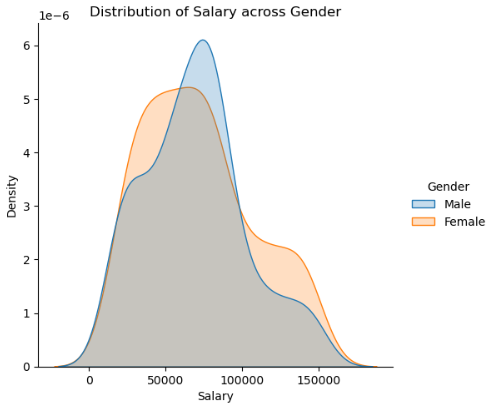
**Figure 6: Summary statistics**

The summary statistics in Figure 6 provide insights into the central tendencies and variability of the dataset. It shows that the average age of individuals is 37.59 years, with a standard deviation of 10.59, indicating moderate variability in the age distribution. The average salary is approximately 70,421, with a significant standard deviation of 34,604, reflecting a wide range of salaries. Additionally, 36.84% of the individuals in the dataset have purchased an iPhone, as indicated by the mean of the ‘Purchase iPhone’ variable.

****

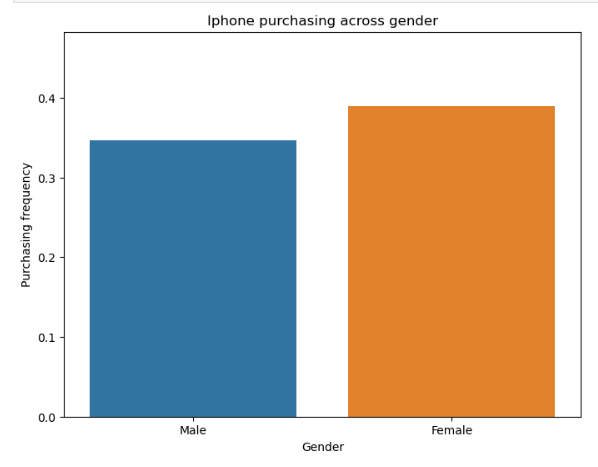
**Figure 7: Distribution plot for Age Across Gender**

Figure 7 shows a density distribution plot of age across gender. The plot compares males and females with overlapping age distributions, centred around ages 30-40. Both genders show a similar overall distribution shape.

****

**Figure 8: Distribution plot for Salary across Gender**

**Figure 8** presents a density distribution plot of salary across gender. It compares males and females. Male salary distribution peaks around 50,000, while female distribution peaks slightly lower, showing differences in salary ranges between genders.

****

**Figure 9: Bar Plot for iPhone purchasing across gender**

Bar plot shows comparing iPhone purchasing frequency across gender. Females have a higher purchasing frequency, around 0.4, while males have a slightly lower frequency, indicating gender differences in iPhone purchases (**Refer to Figure 9**).

## Model development

****

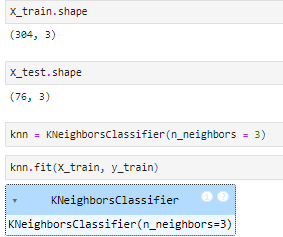
**Figure 10: Feature and target**

Features Age, Salary, and Gender are selected as X, and y represents the target, Purchase iPhone **(Refer to Figure 10).**

****

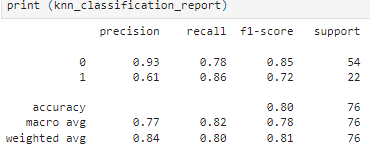
**Figure 11: train-test splitting**

The data is split into 80% training and 20% testing using train\_test\_split, with random state 1234 ensuring reproducibility.

****

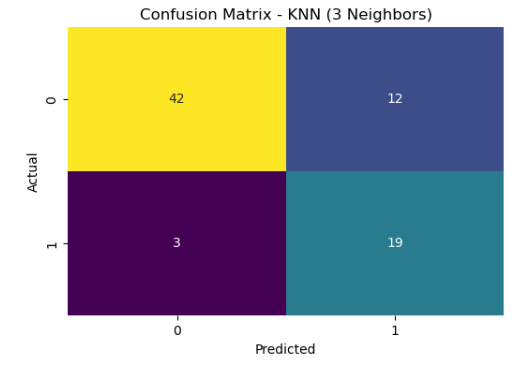
**Figure 12: Model architecture**

**Figure 12** shows that the K-Nearest Neighbors model is trained with 304 samples and 3 features, using 3 neighbours to predict the target variable.

****

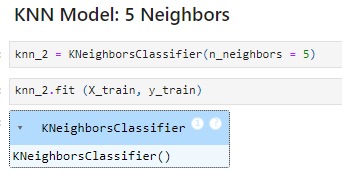
**Figure 13: Classification Report**

The model has performed better in prediction of Class 0 instances with 0.90 precision and 0.85 recall. Class 1 has lower precision (0.68) and higher recall (0.77). Overall accuracy is 83%.

****

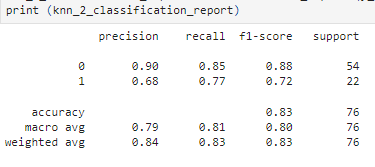
**Figure 14: Confusion matrix**

This confusion matrix (KNN with 3 neighbours) shows 42 true negatives, 19 true positives, 12 false positives, and 3 false negatives **(Refer to Figure 14).**

****

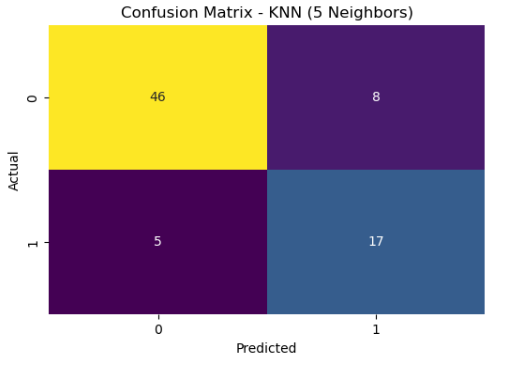
**Figure 15: Model Architecture**

Figure 15 initialises a K-Nearest Neighbours classifier with 5 neighbours and trains it using the X\_train and y\_train datasets **(Refer to Figure 15).**

****

**Figure 16: Classification Report**

Figure 16 shows accuracy (83%), precision, recall, and F1 scores for a KNN classifier, indicating performance on two classes.



**Figure 17: Confusion matrix**

The confusion matrix shows that the KNN model (with 5 neighbours) correctly predicted 46 true negatives, 17 true positives, with 8 false positives and 5 false negatives **(Refer to Figure 17).**

# Challenges faced

One of the challenges faced in this project was determining the optimal data-splitting ratio for training and testing the model. Ensuring the right balance between training and test data was crucial to avoid overfitting or underfitting the K-Nearest Neighbours (KNN) model when predicting iPhone purchases based on age and salary.

# Complexity level

The dataset is moderately sized, containing 380 records and 3 variables: Age, Salary, and Purchase iPhone. There are no data quality issues such as missing values, duplicates, or outliers, making the project relatively simple. However, the use of a robust classification model (K-Nearest Neighbours) and a well-structured machine learning workflow ensures that the project can be scaled to handle larger datasets with more features while maintaining accurate predictions.