**IPHONE PURCHASE PREDICTION USING MACHINE LEARNING: DECISION TREE CLASSIFIER**

**Prepared by: Tarun Mondal**

**Table of Contents**

[Objective 2](#_n2yopppgmbr5)

[Methodology 2](#_4al1ey1rmq13)

[End-to-end process with solution architecture 2](#_8n8seygnqb6h)

[Importing libraries in Python 2](#_j4er0b6hqjd3)

[Data exploration 3](#_nq6j5jgfuivr)

[Data preprocessing 4](#_5bdlpe9xqeme)

[Exploratory data analysis (EDA) 5](#_4j1he0cm1jgd)

[Model development 8](#_xxsr8bt48lqc)

[Challenges faced 11](#_n66ncrpgyp64)

[Complexity level 11](#_dhl7r1wlxkiy)

# Objective

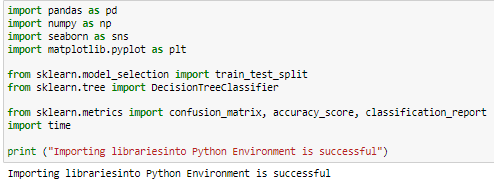
The objective of this project is to predict iPhone purchases based on age, gender and salary using the Decision Tree Classifier.

# 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. A Decision Tree Classifier 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 IDE.

# End-to-end process with solution architecture

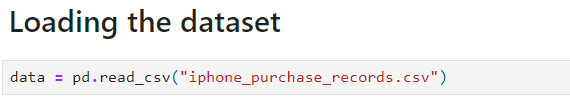
## Importing libraries in Python

****

**Figure 1: Importing libraries in Python**

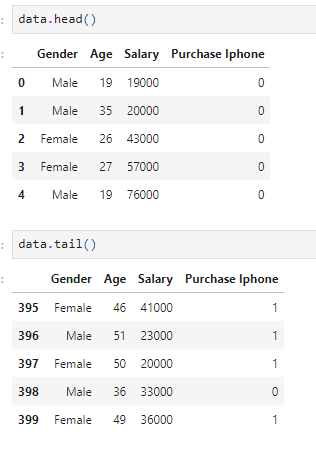
Pandas library has been imported into Python for data loading and manipulation, whereas Seaborn and Matplotlib libraries have been used for data visualisations. For developing machine learning models, the ‘DecisionTreeClassifier ()’ module has been imported from the scikit-learn framework. Additionally, 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

****

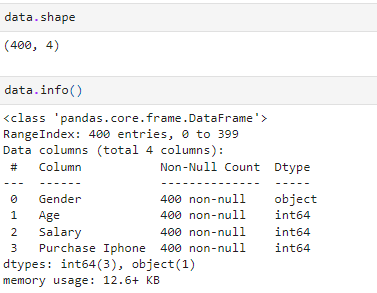
**Figure 2: Loading the dataset**

The dataset has been loaded in Python (Jupyter Notebook Environment) using the ‘read’ function from the Pandas library.

****

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

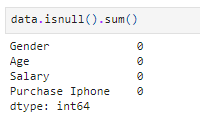
**Figure 3** displays the ‘iphone\_purchase\_records’ dataset, retrieved using the head () and tail () functions from the Pandas library. These functions provide a quick overview of the dataset, showing the first and last few records for a better understanding of the data structure.

****

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

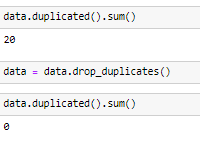
The shape and info of the dataset revealed that the dataset contains 400 valid observations and 4 variables (**Refer to Figure 4**). From the data info, it can be observed that the dataset contains 1 categorical feature (Gender).

## Data preprocessing

****

**Figure 5: Data preprocessing**

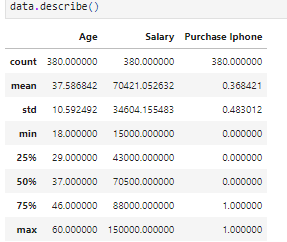
The dataset has been checked for missing values using the isnull().sum() function, as shown in **Figure 5**. The output indicates that there are no missing values in any of the columns: ‘Gender’, ‘Age’, ‘Salary’, and ‘Purchase Iphone’. This suggests that the dataset is clean, and no further data preprocessing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

****

**Figure 6: Duplicate values in the dataset**

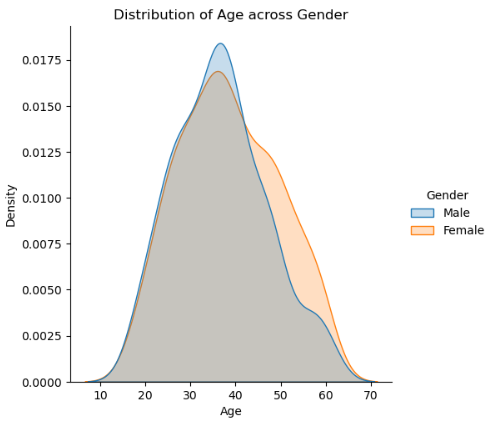
Duplicate values in the dataset have been checked using the ‘duplicated (). sum ()’ function, from which the observed duplicate values are 20 (**Refer to Figure 6**). These observed duplicate values have been removed from the data frame by using ‘drop\_duplicates ()’ function.

## Exploratory data analysis (EDA)

****

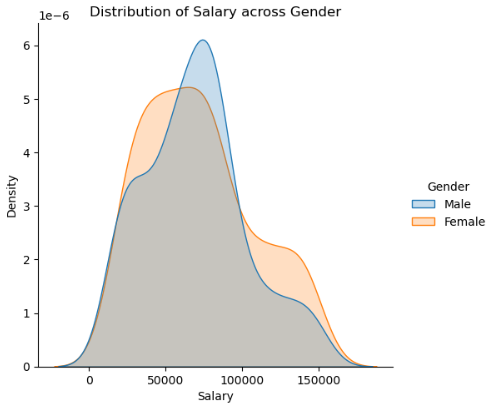
**Figure 7: Summary statistics**

The summary statistics in **Figure 7** 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.

****

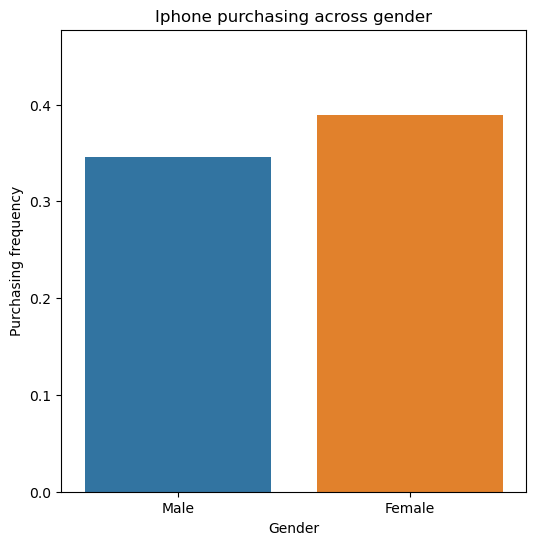
**Figure 8: Distribution plot for Age Across Gender**

**Figure 8** 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 9: 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 10: Bar Plot for iPhone purchasing across gender**

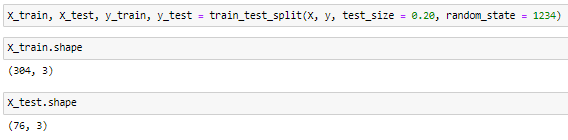
Bar plot shows comparing iPhone purchasing frequency across gender. Females have a higher purchasing frequency, around 0.4, while males have a slightly lower frequency, indicating gender differences in iPhone purchases (**Refer to Figure 10**).

## Model development

****

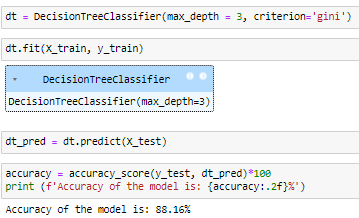
**Figure 11: Feature and target**

Features such as Age, Salary, and Gender are selected as X, and y represents the target, purchase iPhone (**Refer to Figure 11**).

****

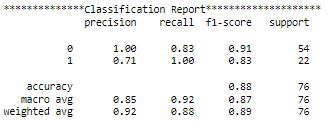
**Figure 12: train-test splitting**

The data is split into 80% training and 20% testing using train\_test\_split, with random state 1234 ensuring reproducibility. The shape of train and test set are respectively 304 and 76, indicating 304 instances are used to train the model, whereas, 76 instances are used to evaluate the predictive accuracy of the model (**Refer to Figure 12**).

****

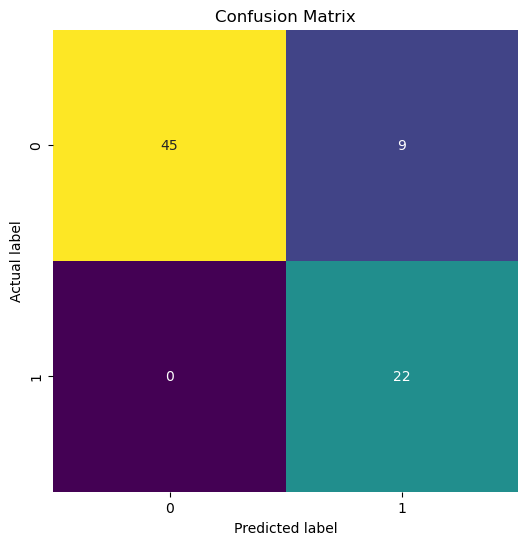
**Figure 13: Model architecture**

**Figure 13** shows the architecture of the Decision Tree classifier model, where the maximum accuracy has been obtained for a max\_depth = 3 and criteria = ‘gini’. The obtained accuracy of the model is 88.16%, indicating that the model has accurately predicted approximately 88% of the instances.

****

**Figure 14: Classification Report**

**Figure 14** shows the classification report of the model, from which it can be observed that the Decision tree model has performed significantly well in predicting both Class 0 and Class 1 instances in the target variable (iPhone Purchase Status).

****

**Figure 15: Confusion matrix**

The true negative and true positive cases of the model are 45 and 22 respectively, whereas the false positive and false negative instances are 9 and 0, indicating a significantly low level of misclassification (**Refer to Figure 15**).

# 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 Decision Tree Classification 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 of 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 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.