**Assignment 4**

**Social Network Ads Classification using Naive Bayes**

***Name: Saurabh Isane***

***Roll No: 122B1F035***

Objective:

To apply the Naive Bayes Classification technique on a real-world dataset and predict whether a user will purchase a product based on their age and estimated salary.

What is Naive Bayes?

Naive Bayes is a simple yet powerful supervised machine learning algorithm used for classification. It is based on Bayes' Theorem and assumes that the features are independent of each other, which is often not the case in real data but works surprisingly well in practice.

Bayes' Theorem:  
P(A|B) = (P(B|A) \* P(A)) / P(B)

Key Assumptions:

* All features are independent.
* All features contribute equally to the outcome.

Types of Naive Bayes Classifiers:

* Gaussian Naive Bayes (for continuous features)
* Multinomial Naive Bayes (for text data)
* Bernoulli Naive Bayes (for binary features)

In this assignment, we used Gaussian Naive Bayes since our features (age, salary) are continuous.

Steps Performed in the Assignment

1. Importing Libraries:

We imported essential libraries like pandas, numpy, matplotlib, seaborn, and sklearn tools for preprocessing, model building, and evaluation.

2. Loading the Dataset:

We used the Social\_Network\_Ads.csv file and loaded it using pd.read\_csv().

3. Initial Exploration:

* Displayed first few rows using head()
* Verified data structure with info()
* Checked for null values with isnull().sum()

4. Preprocessing:

* Label encoded categorical variables (like Gender)
* Used StandardScaler to normalize the numerical features (Age, Estimated Salary)

5. Splitting the Dataset:

Split the dataset into training and testing sets using train\_test\_split() with a 70-30 ratio.

6. Model Building:

Used GaussianNB() from sklearn.naive\_bayes to create a Naive Bayes classifier and trained it using fit().

7. Prediction:

Predicted the target values on the test set using predict().

8. Evaluation:

Used various metrics to evaluate the model:

* Accuracy Score
* Confusion Matrix
* Classification Report
* F1 Score

9. Visualization (Optional):

Used seaborn and matplotlib to visualize data and evaluation results.

Results:

* The Naive Bayes model successfully predicted user purchases with good accuracy.
* Evaluation metrics like F1 score and precision helped us understand how well the model handled true positives and false positives.

Conclusion:

Naive Bayes is an efficient and fast classification technique that works well on small to medium datasets. Even with the assumption of feature independence, it can provide decent results and is ideal for baseline models.

Future Scope:

* Apply cross-validation to evaluate model stability.
* Use feature engineering to add more meaningful features.
* Compare performance with other classifiers like Logistic Regression, KNN, or Decision Trees.