Overview

The dataset and use case (for Task 1, 2 and 3) are referenced from Kaggle. The dataset is uploaded on the classroom page.

Problem Statement:
Dream Housing Finance company deals in all home loans. They have a presence across all urban, semi-urban, and rural areas. Customer-first applies for a home loan after that company validates the customer eligibility for a loan.

The company wants to automate the loan eligibility process (real-time) based on customer detail provided while filling the online application form. These details are Gender, Marital Status, Education, Number of Dependents, Income, Loan Amount, Credit History, and others. To automate this process, they have given you a task to build a model that can identify the customer's segments, those eligible for loan amount so that they can specifically target these customers.

Data set dictionary

Key Name	**Description**
Loan_ID	Unique Loan ID
Gender	Female/Male
Married	Applicant married (Y/N)
Dependents	Number of Dependents
Education	Applicant Education Graduate/Not Graduate
Self_Employed	Y/N
ApplicantIncome	Applicant Income
Co Applicant Income	Co Applicant Income
Loan Amount	Loan Amount in thousands
Loan_Amount_Term	Term of a loan in months
Credit_History	credit history meets guidelines
Property_Area	Urban/ Semi-Urban/ Rural
Loan_Status	Loan approved (Y/N)

Task 1 - Decision Trees with Hyper parameter Tuning

1. Start by building and evaluating a default decision tree model using appropriate metrics and a confusion matrix.
    a. Separate the data into training and test sets (use random_state=42 in the train_test_split for reproducibility).
    b. Create a model pipeline (for feature transformations)
2. Then, use GridSearchCV to tune the model. Be sure to change the default scoring to "recall_macro" when you instantiate the GridSearch. (Be sure to explain the GridSearchCV technique)
3. Visualize the trees from both the default and the tuned model versions.
4. Make sure you have evaluated both the default and the tuned versions using appropriate metrics and a confusion matrix.
5. Which combination of hyperparameters led to the best-tuned model?

Task 2- SVM Algorithm

1. From first principles, explain the SVM algorithm, highlight the pros and cons of this algorithm.
2. Build an SVM classifier
    a. Separate the data into training and test sets (use random_state=42 in the train_test_split for reproducibility).
    b. Create a model pipeline (for feature transformations)
3. Evaluate your model using a classification report and a Confusion Matrix display
4. Describe your results

Task 3 - Random Forest

1. From first principles, explain the Random Forest, highlight the pros and cons of this algorithm.
2. Build a Random Forest classifier
    a. Separate the data into training and test sets (use random_state=42 in the train_test_split for reproducibility).
    b. Create a model pipeline (for feature transformations)
3. Evaluate your model using a classification report and a Confusion Matrix display.
4. Describe your results.

Task 4 - Logistic regression (Use your own dataset)

Part A)

Identify your own dataset and define the problem

Build the best model you can based on your dataset and then report on your results. For this task, optimize 'recall_macro' as your scoring metric with GridSearchCV. Because GridSearchCV can take a longtime to run, you may want to use 3 folds instead of the default 5.

Part B)

a) Start by creating and evaluating a default logistic regression model using appropriate metrics and a confusion matrix
b) Then use GridSearchCV to tune the solver, penalty type and C values (inverse regularization strength), along with trying 'balanced' or None for class weight. Be sure to change the default scoring to 'recall_macro' when you instantiate the GridSearch.
    i) Your C values should be logarithmic: i.e., [.0001, .001, .01, .1, 1, 10, 100, 1000, 10000]
    ii) With elasticnet, remember to tune the ratio of l1 to l2, not C.
c) Make sure you have evaluated both the default and the tuned versions using appropriate metrics and a confusion matrix.
d) In a text cell, address these questions for your logistic regression models:
    i) Which combination of hyperparameters led to the best-tuned model?
    ii) Share any two relevant insights from the data.

Task 5 - Unsupervised learning

Your stakeholder is a credit card company (dataset attached) that wants to market new credit cards. They have asked you to segment their potential customers to determine how and what kind of cards they should market to each group

1. Use K Means to create various customer segments.
2. Create analytical visualizations that explore statistics for each feature for each cluster.
3. Write a description of each cluster based on the visualizations that you created. Do more than describe the numbers; try to see past the numbers and describe what kinds of people are represented by each cluster. Include at least one insight for each cluster.
4. Create one or two recommendations for your stakeholders (the credit card company) regarding how they should market credit cards differently or which cards they should market to each cluster based on your data and insights.
5. Generate another set of segments using hierarchical clustering.
6. Visualize and describe these clusters.

Submissions

1. Well documented notebook. The redundant code should be cleaned out prior to submission.
2. Video recording
    a. The video should be clear.
    b. Turn on your video and share your screen while presenting.
    c. The video **maximum** duration is **40 minutes** (**~ 8 minutes per task**)

3. **Deadline: Friday 8th August 23:59 hrs EAT**