CSEN911: Data Mining
Winter 2025
Course Instructor: Dr. Ayman Alserafi

Classification Mini-Project

Due: 21 November, 2025 11:59 PM

Introduction & Motivation

In the financial sector, loan approval decisions are among the most critical and complex processes faced by lending institutions. Every loan application involves assessing multiple factors such as the applicant's income level, credit score, employment stability, and debt-to-income ratio to determine their creditworthiness. However, manual or rule-based evaluation can be time-consuming, inconsistent, and prone to human bias. By leveraging data-driven approaches and machine learning classification models, financial institutions can automate and enhance the accuracy of loan approval decisions. Predicting whether a loan should be approved or rejected based on historical applicant data not only minimizes default risks but also improves operational efficiency, transparency, and customer satisfaction.

Mini-Project Description & Requirements

You will be provided with a dataset containing financial and demographic information about loan applicants, including factors such as income, credit score, employment status, loan details, and other indicators of financial stability, along with the final decision showing whether each loan was approved or not.

Your task is to inspect the dataset, perform appropriate data cleaning and preprocessing to ensure accuracy and consistency, and then apply **a Decision Tree classification model** to predict the likelihood of loan approval based on the applicant's characteristics. It is also required that you answer the analytical questions below using suitable visualizations to identify key trends and patterns in applicant profiles, financial behavior, and approval outcomes.

Data Challenges

You are expected to perform several necessary data pre-processing and cleaning steps while **justifying** and **explaining** each step using appropriate markdown text, such as:

- Resolving inconsistencies in categorical & numerical columns
- Resolving null values using the most appropriate method for each column.
- Feature engineer "Monthly_Debt_To_Income_Ratio" column, which indicates the applicant's ratio of debt to income per month.
- Selecting suitable columns for the modeling step.
- Applying an appropriate categorical encoding method for the modeling step.

Analytical Questions

Answer the following questions in the notebook by using an appropriate visualization method for each question to find out the answer, while justifying and explaining each method using appropriate markdown text:

1. On average, which type of educational level has the highest approval rate? Order them on the graph.
2. How does the annual income vary among approved applicants? Interpret the values of the 3 quartiles.
3. How does the age of an applicant affect their credit score? (Hint: Use the line of best fit.)
4. Is the distribution of applicants' income per month normal or skewed?

5. Display the Decision Tree after modeling and comment briefly on how the rules are split. Write in English one of the rules of an approved loan in the tree. For simplicity, you may choose a leaf node that is quick to reach.

Info: Ensure your code, explanations, and outputs are clearly structured, properly labeled, and easy to follow. **Notebook organization will be graded**, so take care to keep it neat and logical.

Evaluation Criteria

Warning: Using AI to solve the project will result in a ZERO. You can use external sources to search for methods or syntax; However, all the explanations and comments must be your own.

Your Jupyter notebook submission will be graded with a total of **50 Points** on the following:

- Justifying and commenting on each step taken or decision made (submitting the code only will result in a **ZERO**).
- Notebook organization, applying and justifying appropriate data pre-processing and cleaning methods (data cleaning, column selection, data transformation, etc.) **(20 points)**.
- Answering each query in detail and displaying an appropriate visualization for each query **(20 points)**.
- Applying & evaluating the decision tree model and commenting on its performance and giving a final recommendation (an appropriate performance metric should be explained, used, and justified) **(10 Points)** .

Notice: If you make changes to the dataset, you are required to display the effect of your changes using an appropriate method. You are also required to comment your code, briefly explaining each step you are doing (necessary in data preprocessing, cleaning and feature engineering) You are not allowed to loop over the dataset rows/cells (unnecessary and inefficient); use built-in functions instead.

Submission

You can only work on the mini-project in pairs. To accept the assignment and gain access to the private repository for your team, use the following GitHub classroom invitation `https://classroom.github.com/a/Z506ZgfS`. Like mini-project 1, submissions are done via your mini-project repository, and no further action is required after saving your final version to the repo.

Warning: Your Jupyter notebook submission will be re-run after submission. Submitting a notebook that has errors mid-way through execution will result in losing the grade of all subsequent cells! The order of the notebook's code cells and the order in which you execute them matters. Please make sure your notebook runs and outputs the expected results by restarting the kernel and running all the cells before finalizing your submission.

Bonus (2.5 Points)

Apply a different classification algorithm, compare the models' performance using an appropriate metric, which of the two models that you applied will you choose? and why?

Note: Code related to this bonus task should be performed at the end of your mini-project notebook, not in-between the mini-project's requirements. Indicate the start of your bonus code, either with a comment at the start of the code cell, or add a new section/heading titled 'Bonus' for clarity (similar to the existing sections provided).