Janet Anagli, Raunak Sarbajna and Christoph F. Eick

COSC 3337: Data Science I in Fall 2023

ProblemSet1 Task2

Individual Task

**Task 2: Predicting Loan Approval using Decision Trees and Support Vector Machines**

**Submission deadline:** Sept. 29, 11:59p CST (electronic submission)

**Weight:** 19-23% of the points that are allocated to the Problem Set Tasks

Responsible TA: Janet

Last updated: Sept. 12, 8a.

**Learning Objectives**:

1. Understand how decision trees and support vector machines (SVMs) are used for classification tasks.
2. Learn to tune hyperparameters in decision trees and SVMs to avoid overfitting.
3. Gain proficiency in using recall, precision, and accuracy as metrics for evaluating classification model performance.
4. Learn to use cross-validation techniques to assess model performance.

In this assignment, you will work with a dataset containing information used to determine the eligibility of individuals or organizations for obtaining loans from a lending institution. Your goal is to build, evaluate and compare two models for predicting whether a loan is approved or not: one using Decision Tree and one using SVM.

Overview of a Loan Process:  
The loan process begins when an individual or entity requests funds from a lender, providing financial details and reasons for the loan. The lender assesses creditworthiness, considering factors like debt-to-income ratio. If approved, a contract is signed, and funds are disbursed. Loan terms, including interest rates and repayment periods, are agreed upon beforehand, and collateral may be required. Should a loan application be denied, the lender must provide a reason. This process ensures clarity and compliance before any funds are exchanged.**Dataset:**

**Download the “loan\_approval.csv” dataset** [**here**](https://uofh-my.sharepoint.com/:x:/r/personal/jyanagli_cougarnet_uh_edu/Documents/loan_approval.csv?d=w7456211cff4d43b9a4aa8aae17d21f4b&csf=1&web=1&e=BvA11K)**.**

The loan approval dataset is a collection of financial records and associated information used to determine the eligibility of individuals or organizations for obtaining loans from a lending institution. The dataset is a modification of the [original Loan Approval Prediction Dataset.](https://www.kaggle.com/datasets/architsharma01/loan-approval-prediction-dataset) The labels in the original dataset has been encoded to numerical values and loan\_id column has been dropped. More details about the attributes of this dataset are listed below:

1. no\_of\_dependents (Number of Dependents of the Applicant) - Domain {0,1,…, 5},
2. education (Education of the Applicant: Graduate-1/Not Graduate-0),
3. self\_employed (Employment Status of the Applicant: Yes-1/No-0),
4. income\_annum (Annual Income of the Applicant) - Domain [200k, 9.90m],
5. loan\_amount (Loan Amount) - Domain [300k, 39.5m],
6. loan\_term (Loan Term in Years) - Domain {2, 3,…,20},
7. cibil\_score (Credit Score) - Domain [300, 900],
8. residential\_assets\_value - Domain [-100k, 29.2m],
9. commercial\_assets\_value - Domain [0, 19.4m],
10. luxury\_assets\_value - Domain [300k, 39.2m],
11. bank\_asset\_value - Domain [0, 14.7m] and
12. loan\_status (Loan Approval Status: Approved-1, Rejected-0; class attribute that needs to be predicted using SVM and Decision Trees ).

The first 3 examples of the dataset are listed below:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| no\_of\_dependents | education | self\_employed | income\_annum | loan\_amount | loan\_term | cibil\_score |
| 2 | 1 | 0 | 9600000 | 29900000 | 12 | 778 |
| 0 | 0 | 1 | 4100000 | 12200000 | 8 | 417 |
| 3 | 1 | 0 | 9100000 | 29700000 | 20 | 506 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| residential\_assets\_value | commercial\_assets\_value | luxury\_assets\_value | bank\_asset\_value | loan\_status |
| 2400000 | 17600000 | 22700000 | 8000000 | 1 |
| 2700000 | 2200000 | 8800000 | 3300000 | 0 |
| 7100000 | 4500000 | 33300000 | 12800000 | 0 |

**Tasks**

1. A. Using all attributes, build a Decision Tree model to predict whether a loan is approved: Train the Decision Tree model using the given maximum depths (3, 7, 11, 15).  **8 points**
2. Perform 5-fold cross-validation for each of the 4 max depths and compute accuracy (mean of validation scores), precision and recall. Generate a table, as given below, for the obtained results. **5 points**

|  |  |  |  |
| --- | --- | --- | --- |
| **Decision Tree Experiments** | | | |
| Max Depths | Accuracy | Precision | Recall |
| 3 |  |  |  |
| 7 |  |  |  |
| 11 |  |  |  |
| 15 |  |  |  |

1. Explain how the tree size/depth affects model performance in the context of overfitting/underfitting. **3 points**
2. Explain the meaning of the difference in accuracy, precision and recall scores in relation to the task; only if there is a significant difference. **2 points**
3. A. Using all attributes, build an SVM Model to predict whether a loan is approved: Train the SVM model using the given kernel functions (linear, polynomial, sigmoid, sigmoid with different σ value). **10points**

**Note:**

Excluding the 4th kernel function where you use a different “coef0” value (By default, coef0 is set to 0.0) for the sigmoid kernel, use the default values for the parameters of the other 3 kernel functions.

Adjust the “coef0” parameter of the sigmoid kernel to fine-tune the SVM model for better model performance. “coef0” is the σ equivalent using scikit-learn.

B. Perform 5-fold cross-validation for each of the 4 kernel functions and compute accuracy (mean of validation scores), precision and recall. Generate a table, as given below, for the obtained results. **5 points**

|  |  |  |  |
| --- | --- | --- | --- |
| **SVM Experiments** | | | |
| **Kernel Function** | Accuracy | Precision | Recall |
| linear |  |  |  |
| polynomial |  |  |  |
| sigmoid |  |  |  |
| sigmoid with different σ value |  |  |  |

1. Discuss the impact of different kernels on model performance. **2 points**
2. Explain the meaning of the difference in accuracy, precision and recall scores in relation to the task. **2 points**
3. Interpret the tables you generated in questions 1B and 2B; compare the performance of the Decision Tree and SVM models. Which model performs better? Why do you think that is the case? What would you recommend to further improve each model’s performance? **5 points (and up to 3 extra points)**

## **Deliverables:**

1. A python/R file with your code and analysis.
2. A pdf copy of your report (1-2 pages) of your findings and conclusions for Task 2C, 2D, 3C, 3D and 4. Boldly state the task name before explanations.

**Submission Guidelines[[1]](#footnote-1):**

1. Name your python/R files to **COSC3337F23-PS1T2-Firstname-Lastname.ipynb** or any other appropriate format.
2. Name the pdf copy of your report **COSC3337F23-PS1T2-Report-Firstname-Lastname.pdf**
3. Create a folder and name it **COSC3337F23-PS1T2-Firstname-Lastname**.The folder should contain both python/R file and pdf copy of your report named correctly. Compress (zip) the folder and submit it to MS TEAMS.
4. Submit on the Assignment tab in MS Teams.

**Guideline for Code Grading:**

1. Code documentation and readability: Use of comments, proper indentation, clear notations, and simplicity.
2. Code completeness: Working code with no errors including instructions for running your code. Indicate any module(s) to be downloaded before code is run.

**References:**

Python:

1. <https://www.datacamp.com/tutorial/decision-tree-classification-python>
2. <https://www.datacamp.com/tutorial/svm-classification-scikit-learn-python>

R:

1. <https://www.datacamp.com/tutorial/decision-trees-R>
2. <https://www.datacamp.com/tutorial/support-vector-machines-r>

1. More detailed submission instructions for Task2 will be added to this specification by Sept. 23, 2023 the latest. [↑](#footnote-ref-1)