04-638 A: Programming for Data Analytics

Final Project: Customer Segmentation and Classification using Machine Learning

Release Date: 6th December 2023 **Due Date:15th December 2023, 11.59pm CAT**

Points Total: 300 pts

To submit:

1. **Part 1:** Jupyter Notebook file, csv file, zip file of model deployment code. **Part 2**: PDF report
2. Zip all the files under the name **yourAndrewID.zip** and submit the zip file on Canvas.

Submission Mechanism: Submit on Canvas.

Task Context: You will work individually, prepare a dataset for analysis, perform exploratory data analysis, perform feature engineering, and feature selection, label the data using clustering, evaluate the clustering model, use the labeled data to train a classification model, perform hyperparameter tuning on the model, and evaluate machine learning models.

TASK OBJECTIVES

a) Perform data preprocessing.
b) Perform feature selection.
c) Label the dataset.
d) Build tuned ML models.
e) Build and evaluate ML models.
f) Deploy an ML model.
g) Write a technical report.

Toolkit:

- numpy, pandas, and matplotlib, sklearn, etc.

TASK BACKGROUND

Businesses can use customer segmentation to develop custom products or services that target specific customer segments. The task will involve three activities, i.e., a) cluster the data to identify customer segments; b) use the identified customer segments to label the data; and c) build and deploy a classification model that can be used to identify a customer's segment given an instance. Accurate identification of customer segments will ensure that the credit card issuer can push the most relevant products or services to the right customers.

This assignment will involve building and evaluating a) unsupervised machine learning (ML) models; and b) supervised ML models.

TASK DESCRIPTION

Dataset: Consider the credit card dataset available here. The dataset can be used for customer segmentation of credit card customers. The dataset is made up of eighteen(18) variables.

Required

Note:

- You should include Markdown cells and comments to describe the tasks you are performing. The markdown cells should also include a summary of your observations as you perform the tasks.
- Include comments in your ***Code Cells*** to explain your codes.

A. PART 1: JUPYTER NOTEBOOK (yourAndrewID_ics.ipynb)

1) **[15 Pts]** *Data Preparation*: Download the dataset, load into a pandas data frame, and prepare the datasets for analysis.
2) **[40 Pts]** *Exploratory Data Analysis (EDA)*: Use numpy, pandas, and matplotlib to perform EDA on the data. This should include identifying the presence of missing values (if any), creating visualizations with the data, and identifying presence of outliers (if any).
3) *Preprocessing*: This should make your data ready for building machine learning models. Using pandas and scikit-learn:
   a) **[10 Pts]** Handle missing values and outliers (if any).
   b) **[15 Pts]** Perform appropriate label encoding for categorical attributes (if needed) and data scaling using the standard scaler (if needed). Save the standard scaler if used so that it can be retrieved during model deployment. (*Hint*: You can serialize an object using pickle).
4) *Unsupervised model creation and evaluation*:
   a) **[16 Pts]** Use scikit-learn to build and evaluate a clustering model. Provide justification for the evaluation metric(s).
   b) **[4 Pts]** Based on the clusters in a) above, label the entire dataset and save the labelled dataset as *yourandrewid-cc-labeled.csv*.
5) Supervised model creation and evaluation:
   a) **[15 Pts]** Use scikit-learn to build and evaluate a classification model based on cross validation. Provide justification for the evaluation metric(s). Save the model so that it can be retrieved if necessary.
   b) **[7 Pts]** Use learning curves to determine whether the model is overfitting or underfitting the data. Add mark down cells to comment on the results.
   c) **[3 Pts]** Provide a justification for the classification algorithm used in a) above.
6) *Feature Selection and Engineering*: This should reduce your data dimensions to determine if building models with reduced dimensions improves performance. Using pandas and scikit-learn:
   a) **[15 pts]** Perform feature selection on the dataset to determine new features or a feature subset to be used in model building. Provide justification for the feature selection approach used.

b) **[7 Pts]** Using scikit-learn and cross validation build and evaluate a new classification model based on the features identified in 6a) above. Save the model so that it can be retrieved if necessary.

c) **[3 Pts]** Compare the results with those in 5a) above. What did you observe?

7) *Hyper parameter tuning*: This should make it possible to identify optimal hyperparameters for building a machine learning model. Using pandas and scikit-learn:

a) **[20 Pts]** Select the best model (either the one with or without feature selection) as your *benchmark model*. The *benchmark model* should be the model with the best performance overall (this should be clear by comparing the models in 5a) and 6b) based on the performance metrics). You may use visualizations to highlight the comparison. Using hyperparameter tuning (with Grid Search), and based on cross-validation, identify the combination of hyperparameters that gives the best estimator for the model on the dataset (you can only use *all features* or *selected features* depending on the one that resulted in the *benchmark model*). Your search space for hyperparameters must include at least two hyperparameters. Save the tuned model so that it can be retrieved if necessary.

b) **[5 Pts]** Provide a justification for the selected hyperparameters.

c) **[5 Pts]** Compare the performance of the tuned model in 7a) against that of the *benchmark model* selected. You may use visualizations to highlight the comparison. The best performing model should be designated the *final benchmark model*. Save the benchmark model so that it can be retrieved if necessary.

8) *Model deployment*:

[20 Pts] Use the Flask web framework to deploy the *final benchmark model* and use it to identify the segment that a new customer belongs to. Implement a web application that receives input and displays the result, i.e., the customer's segment for the newly input data. Ensure that the input is scaled appropriately if the dataset was scaled in 3b). (*Hint*: You can read a serialized object using pickle)

B. Part 2: Technical Report[yourAndrewID.pdf]

9) **[100 Pts]** *Project Report*: In this section you will prepare a technical report. Produce a summary writeup that contains the following information:

- ***Metadata[3 pts]***: Course Name and Code; Instructor; Assignment Title (see top of this document); Report Title (Choose a title for your work); Your name and AndrewID; Submission Date.
- ***Summary of the completed tasks and the results that includes the following sections [Max (Strictly): 4 pages, 11pt Times New Roman. The references don't have to fit in the 4 pages]***:
    - Abstract **[9 pts]**: Maximum of 150 words. Includes context/background (1 pt), problem statement(1 pt), main purpose(1 pt), summary of approach(3 pts), results(2 pts) and conclusion(1 pt).
    - Background and Problem Description **[4 pts]**
    - Approach **[30 pts]**
        - Overall description of process including data preparation (3 pts), exploratory data analysis (9 pts), performance metrics (3 pts),

unsupervised model building(3pts), supervised model building (including preliminary data labelling)(6 pts), model debugging using learning curves(3 pts), and model deployment(3 pts).

- Results and Discussion **[45 pts]**: (Results include EDA results (5 pts), evaluation results for unsupervised model (5 pts), evaluation results for all supervised models (12 pts), results of model debugging (5 pts), screenshots of the web application's input and result pages (6 pts)). The discussion (12 pts) should be brief but highlight important observations from the analysis of results.
- Conclusion **[3 Pts]**, and
- References **[3 Pts]**. There should be at least 5 references.
- Overall quality of the report **[6 Pts]**: covers formatting, grammar, and organization.