HI 1020

Term Project (Final Project)

Fall 2023 – Due: December 12, 2023, 11:59pm EDT.

This document describes our HI 1020 term project. For HI 1020 term project, you will tackle common steps in predictive (supervised) machine learning model(s) using a health-related dataset.

Step (1): Finding a publicly available dataset in any healthcare setting

In this step, you need to find and grab a publicly available dataset in any healthcare setting, such as cancer, orthopedic, and cardiovascular. "Kaggle" could be a good source to find and grab a relevant dataset. In your dataset selection, please select a dataset that suits "supervised" and "predictive" machine learning models. That being said, the dataset should have a list a variable (x values) and an outcome (y value).

Examples of those datasets include, but not limited to:

[1] https://www.kaggle.com/datasets/uciml/breast-cancer-wisconsin-data

(Outcome: diagnosis [M: malignant AND B: benign])

[2] https://www.kaggle.com/datasets/johnsmith88/heart-disease-dataset

(Outcome: target [0: no disease AND 1: disease]

*Note: If you want, you can use one of these datasets for your term project.

Step (2): Loading the dataset into your Python IDE

Step (2) involves loading, reading and exploring the dataset. You may need to save the dataset in either your google drive or your local machine (e.g., your PC). Your data exploration may include understanding your dataset size, dataset shape, eliminating irrelevant attributes/features, etc.

Step (3): Data Visualization

In step (3), you need to draw three to four types of data visualization to better understand the dataset. Your data visualization may include bar chart, scatter chart, pie chart, etc.

Step (4): Making the dataset available for "Supervised" Machine Learning algorithms

In this step, you need to select your variables (x values) and the outcome (y value). You may also need to do data encoding (turning all data into numeric values), and for sure data normalization. Furthermore, you also need to split your data into train/test sets using K-fold cross validation.

Step (5): Building and evaluating machine learning predictive models

In this step, you will build, train, and test two predictive models, using **Decision Tree** and **Naïve Bayes**. The evaluation metrics should include -at least- **Accuracy**, **Precision**, and **Recall** for every target class.

Step (6): Comparing the predictive models you built using Decision Tree and Naïve Bayes

You need to compare the accuracy performance of your Decision Tree model with the Naïve Bayes model.

Step (7): Draw conclusion

This step should briefly conclude the term project and discuss its current limitations and shortcomings.

Implementation, Code, Documentation, deliverables, and Teamwork

- **Mandatory [implementation]:** You will create a GitHub ID/account, and make a private GitHub repository, namely "HI1020_Final_Project". You will then put all implementation and documentation there. You are free to choose either Python or R as the programming language. You will then add your team members to your GitHub repository, and add me (Ahmad P. Tafti) there too. My GitHub ID is "aptafti".
- **Mandatory [deliverables]:** You will need to upload:
  1) Your R or Python implementation
  2) Dataset (e.g., .csv file)
  3) Readme (in your GitHub Repository)

Rubric

Approx. % of Grade	Excellent (100%)	Adequate (80%)	Poor (60%)	Not Met (0%)
15% Design and Implementation	No errors, it works correctly and meets the specifications.	Minor details of the term project specification are violated, project functions incorrectly for some inputs/cases.	Significant details of the specification are violated, project often exhibits incorrect behavior/functionality.	Project only functions correctly in extremely limited cases or not at all.
5% Readme	Well-organized description and documentation	Minor issues in documentation	Significant issues in documentation.	No Readme.