Project Proposal

Team Members: Kailey Carbone, Michaela Johnson, Tahseen Shaik, Aleid van der Zel
GitHub LINK
Google Folder LINK

Focus: Heart Disease Data
https://www.kaggle.com/datasets/johnsmith88/heart-disease-dataset

Project ideas/question to solve:
Sources of Data: Predicting Heart Disease Using Clinical Variables (kaggle.com)
Credit: Robert Hoyt MD

- What is our dataset?
    - 270 case studies of individuals classified as either having or not having heart disease based on results from cardiac catheterizations
    - 13 independent predictive variables

- Find the problem we want to solve
    - What are the best clinical variables for predicting the probability of heart disease?
    - The variables include, demographics (age, sex), EKG data, and other heart condition related variables
    - Inform patients what are the highest risk factors in developing heart disease
    - This data could be used to advice, provide interventions, treatment plans

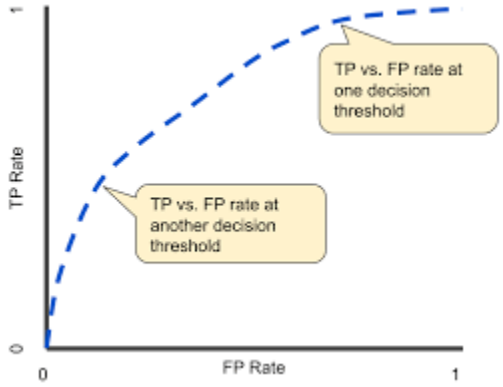
- How are we going to use ML to predict outcome
    - See steps below

- What technologies/ resources do we plan to use
    - Google Colab shared notebook LINK
    - Tableau
    - Spark SQL
    - Python

Steps:

1. Create visuals to see dataset and compare different health related values from people with heart disease to people without heart disease(Tableau)
2. Scale data (StandardScaler ?)
3. Create feature and label variables (X and y)
    a. X = Heart Disease
    b. y = Selected variables
4. Split, test, train data(.75/.25)
5. Create models/ Fit the model on training set
    a. Random Forest
    b. Neural Network
    c. Logistic Regression

    d. Xgboost
    e. SVM
6. Feature selection of what clinical conditions might be most important (might be a set to do before)
7. Get metrics of the models to select the best model to predict the probability of Heart Disease for testing set
    a. Accuracy
    b. AUC - to decide threshold



    c. Recall
    d. Precision
    e. Confusion matrix - willing to accept that level of prob
        i. FN would be worse
8. Plot AUC of models to compare performance

Distribution of Tasks (A, M, K, T)

Day 1(Thursday): Tableau exploration (A)/ ALL
Day1-2(Tuesday): To set up Notebook until model selection (Together)
Day 2(Tuesday): Selected Model
- Random Forest (MDJ)
- Neural Network
- Logistic Regression

Day 3 (Thursday): Evaluate metrics/ select best model /metrics
Day 4 (Monday): make graphs/ input new data/ intro to presentation
Day 5 (Tuesday): Create presentation/ practice
Day 6 (Thursday): Presentation

Project 4

Blueprint Published

Requirements

Data Model Implementation (25 points)

- A Python script initializes, trains, and evaluates a model (10 points)
- The data is cleaned, normalized, and standardized prior to modeling (5 points)
- The model utilizes data retrieved from SQL or Spark (5 points)
- The model demonstrates meaningful predictive power at least 75% classification accuracy or 0.80 R-squared. (5 points)

Data Model Optimization (25 points)

- The model optimization and evaluation process showing iterative changes made to the model and the resulting changes in model performance is documented in either a CSV/Excel table or in the Python script itself (15 points)
- Overall model performance is printed or displayed at the end of the script (10 points)

GitHub Documentation (25 points)

- GitHub repository is free of unnecessary files and folders and has an appropriate .gitignore in use (10 points)
- The README is customized as a polished presentation of the content of the project (15 points)

Group Presentation (25 points)

- All group members speak during the presentation. (5 points)
- Content, transitions, and conclusions flow smoothly within any time restrictions. (5 points)
- The content is relevant to the project. (10 points)
- The presentation maintains audience interest. (5 points)