**Title**: Deep Learning for Heart Disease Prediction

**Team Members**: Yashashwini Devineni

Sai Venkata Anil Thota

**Date**: 07/26/2024

Heart disease is the main problem everywhere. Human beings are collapsing unexpectedly irrelevant of their age, fitness, and living habits. In such a scenario it is really very important for a system which can predict if a person is suffering from heart disease or not.

Deep learning methods such as decision tree classifiers, random forest techniques can analyze complex medical data and predict the presence of heart disease, helping doctors make informed decisions.

Setup Instructions:

1) Prerequisites:

- Python 3.x  
- Jupyter Notebook  
- Pandas  
- NumPy  
- Matplotlib  
- Scikit-learn

Or

- Google Colaboratory

Installation:

If not using Google Colaboratory follow the following steps or else just open the ipynb file in google colab:

2) Setting up the virtual environment:  
```bash  
python -m venv myenv  
source myenv/bin/activate # On Windows use Scripts in place of bin   
```  
  
3) Install the Dependencies

```bash  
pip install jupyter pandas numpy matplotlib scikit-learn  
```

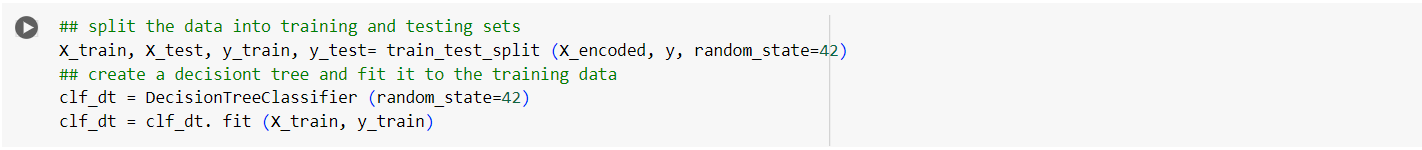
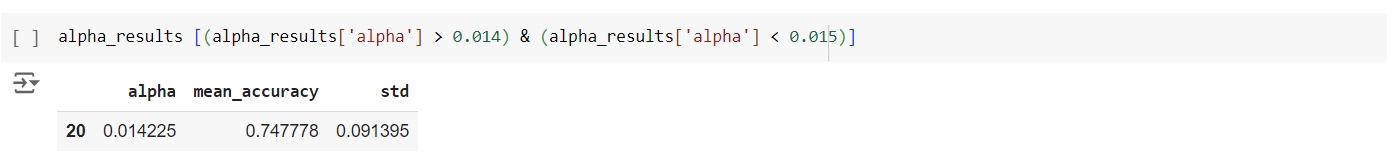
Running Program:

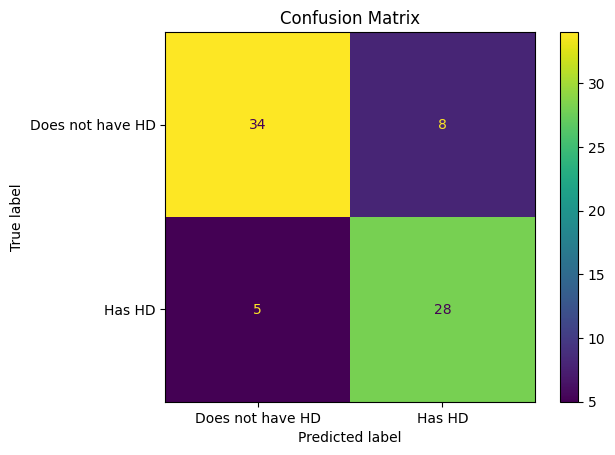
- Open the ipynb file in google colab and run all code snippets or follow below steps if using jupyter notebook.

1. Open Jupyter Notebook  
```bash  
jupyter notebook  
```  
2. Open the Notebook File  
- Navigate to the directory containing the `.ipynb` file.  
- Open the `Classification\_Trees.ipynb` notebook.  
  
3. Run the Notebook  
- Execute the cells sequentially to load data, preprocess it, train the decision tree model, and visualize the results.

# Screenshots of Experiment Results:

After running the notebook, you should observe the following key outputs:  
  
1. Model and Accuracy:

  
  
  
  
  
3. Confusion Matrix



This screenshot displays the confusion matrix for the model, indicating true positives, false positives, true negatives, and false negatives.

4. Pruned Decision Tree  
A diagram of a tree

Description automatically generated  
  
Accuracies:

A screenshot of a computer program

Description automatically generated

Comparision of two methods using confusion metrix:

A screenshot of a graph

Description automatically generated

Comparison between decision tree and random forest using performance metrics:

A screenshot of a graph

Description automatically generated

A graph of a comparison of a model

Description automatically generated with medium confidence

Code Explanation:

The notebook performs the following tasks:  
  
1. Importing Libraries:  
- Pandas, NumPy, Matplotlib, and Scikit-learn for data handling and machine learning.  
  
2. Loading and Preprocessing Data:  
- Loading the heart disease dataset from the UCI Machine Learning Repository.  
- Handling missing values and encoding categorical variables.  
  
3. Train Decision Tree Model:  
- Splitting data into training and testing sets.  
- Training decision tree classifier using the training data.

4. Evaluate Model:  
- Evaluating model using the performance metrics such as accuracy, precision, recall, and confusion matrix.  
- Visualizes the decision tree and confusion matrix.  
  
5. Prune Decision Tree:  
- Applied cost complexity pruning to optimize the decision tree.  
- Evaluated and visualized the pruned decision tree.

6. Code to compare random forest and decision tree with best parameters

7. Printing accuracies for unpruned and pruned decision trees, and random forest.

Conclusion:

Based on our understanding pruned decision trees and random forests are most relevant to predict the heart disease where random forest eliminate the challenge of overfitting. Where Random forest give best results.