Birla Institute of Technology and Science Pilani, Hyderabad Campus

1st Semester 2024-25, BITS F464: Machine Learning

Assignment No: 2, Date Given: 02.09.2024 Date of Sub: 11.09.2024

(**Note: Decision Forests using TensorFlow library: Supervised Learning**) **Max. Marks: 10**

As discussed in the lecture class, in this assignment, you will experiment with TensorFlow's Decision Forest (TF-DF) library to train, test and interpret **Random Forests** and **Gradient Boosted Trees** in TensorFlow running over the **Google colab**. You will also run a hand-coded ID-3 decision tree classifier.

The overall goal in this assignment is to build a classification model (under Supervised learning) that predicts whether a patient will experience in-hospital mortality using various features from the MIMIC dataset, such as heart rate (HR), respiratory rate (RESP), oxygen saturation (SpO2), blood pressure (Systolic), blood pressure (diastolic), and pulse rate (PULSE). You are given with data of two patients in the attached assignment document (MIMIC dataset). The target variable (class) is a binary variable (Anomaly) defining in-hospital mortality. That is '1' indicates that the patient died during the hospital stay, and '0' indicates that the patient was discharged alive from the hospital.

After building the model (i.e. after completing the training and testing) you should be in a position to answer below type of queries:

Given a new patient's vital signs in the form of RESP, SpO2, Pulse, BP-S etc. during their stay in the ICU, can your model predict whether they will experience in-hospital mortality?

Note:

- Before starting the assignment, you will need to install tensorflow_decision_forests libraries for building the model. Also, you will need to install keras library.

Tasks:

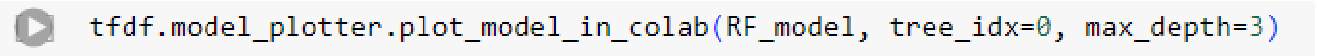
- Import OS environment variables and other supporting libraries like Pandas, NumPy, TensorFlow and TensorFlow-decision-forests.
- Prior to implementing a solution, conduct a thorough data exploration and preprocessing. Employ techniques such as data cleaning, normalization, feature engineering, and outlier detection to enhance data quality and identify relevant patterns. You are free to use the part of code that you might have developed in the previous assignment.

You are given here two csv files (vital signs for patient id:221 and 230), each representing a different patient.

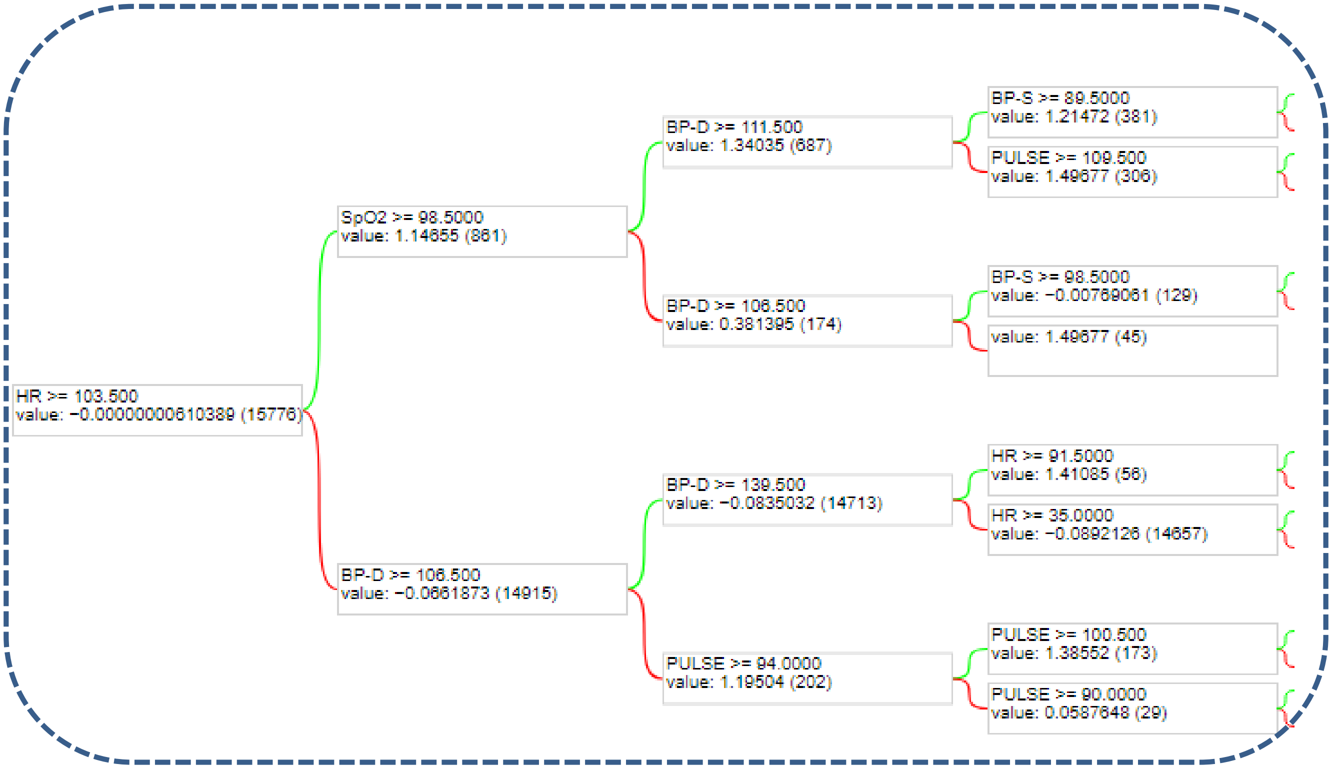
Your tasks are to:

- Implement the ID3 and CART decision tree algorithms on both datasets. (On patient-1 use the ID3 algorithm and on patient-2 use CART algorithm)

- Implement ID3 in two ways: One: Utilize Scikit-Learn's DecisionTreeClassifier: Train an ID3 model using this built-in implementation. And the other is to **Write a custom implementation:** Create your own Python function or class that implements the ID3 algorithm from scratch including functions for computing information gain, for finding the stopping condition etc. You may refer to the class materials to understand the flow or pseudo code.
- Analyze the performance of both implementations on your dataset.
- Implement CART in two ways: **Utilize Scikit-Learn's DecisionTreeClassifier:** Train a CART model using this built-in implementation. And **Utilize Tensorflow's CartModel:** Train a CART model using this built-in implementation.
- **Analyze** the performance of each algorithm on both datasets.
- **Compare** the results from ID3 and CART, identifying their strengths and weaknesses.
- **Write insights** based on your analysis, discussing the implications of your findings for decision tree applications.
- Train a Random Forest model using TensorFlow's Decision Forests library on the patient-1 training data. Evaluate the model's performance using the Patient-1 test data.
- Visualize the first tree in the trained model as discussed in the class (shown below diagram with depth = 3).



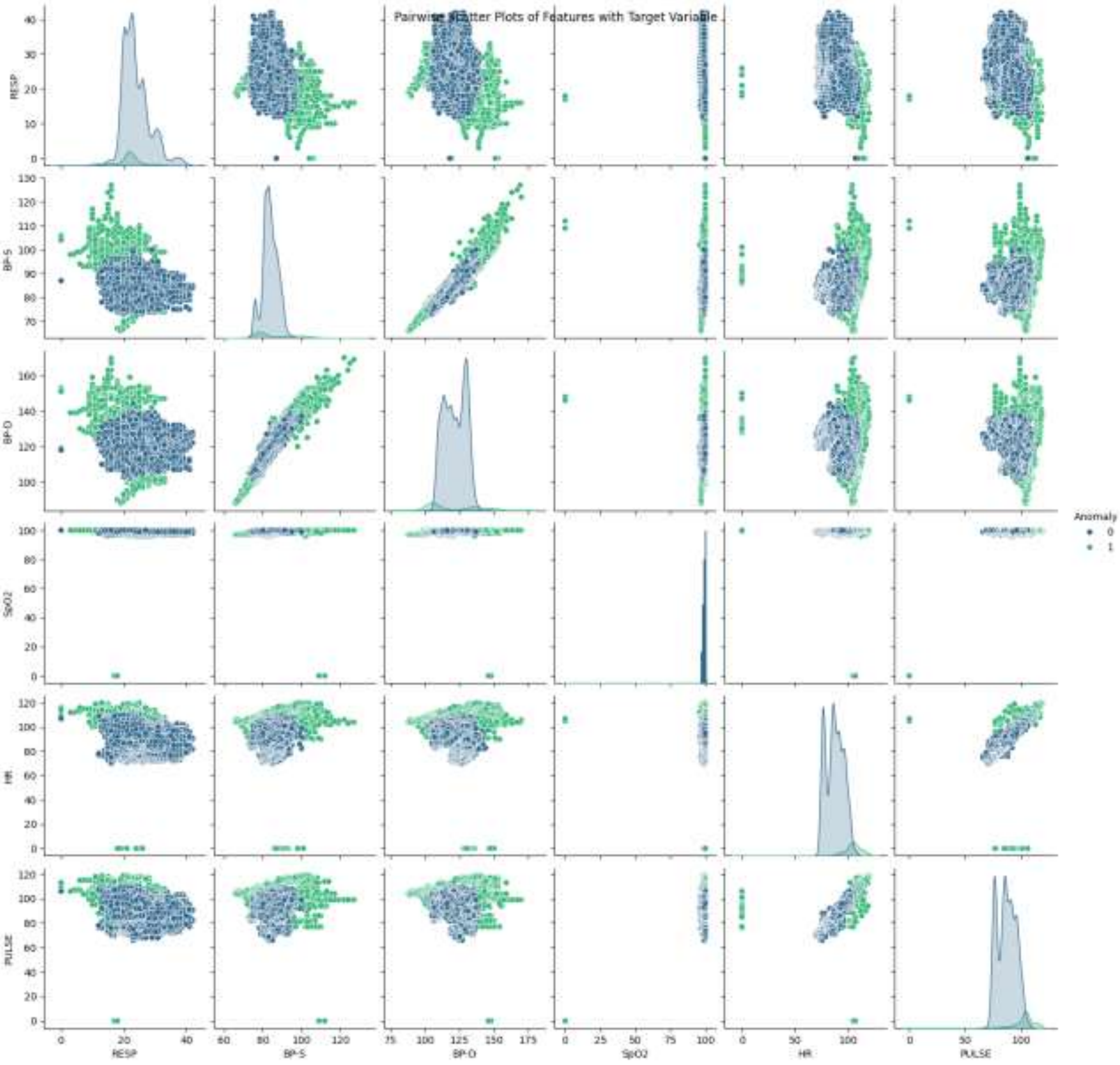
```
tfdf.model_plotter.plot_model_in_colab(RF_model, tree_idx=0, max_depth=3)
```



(Figure 1. Sample Run)

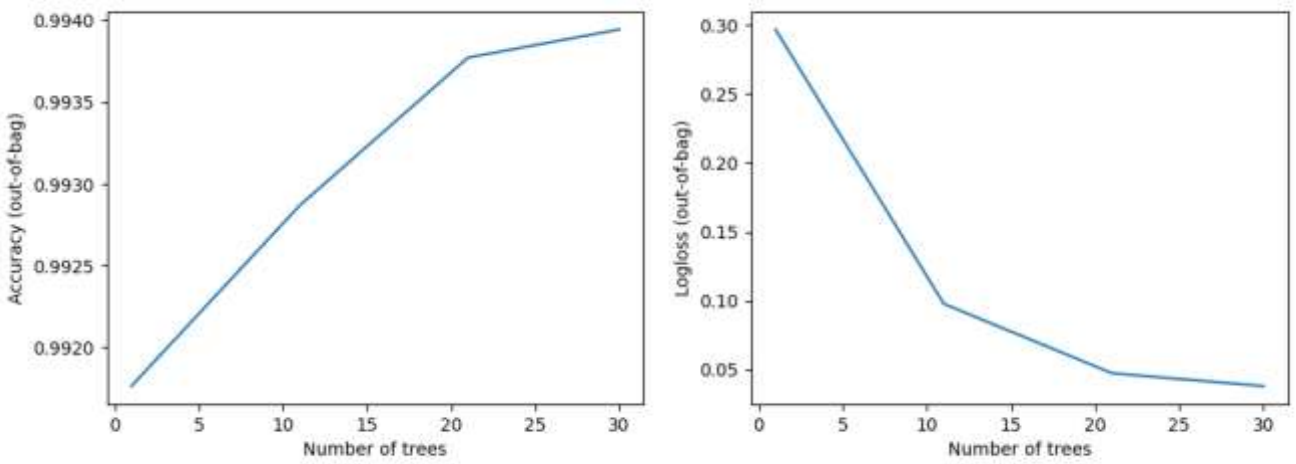
Additional Tasks:

- Visualize the Hypothesis space by producing the scatter plot of the dataset (MIMIC dataset given).



(Figure 2. Sample Run)

- Implement Gradient Boosted Decision Trees (GBDT) with various hyper-parameters on Patient-2 dataset. Compare the accuracy and log loss of these GBDT models to the Random Forest. Analyze the results and draw insights into the differences between the two approaches. Consider factors such as overfitting, underfitting, and the impact of hyper-parameter tuning on model performance.



(Figure 3. Accuracy of the models)

- As discussed in the class, Random Forest and Gradient Boosting models can also make it easy to measure the relative importance of each feature by using a variable called “feature_importances”, implement the same to find out their importance.

```
# Get feature importances for GBDT model
inspector = model_gbdt.make_inspector()
feature_importances = inspector.variable_importances()
print(feature_importances)

# Get feature importances for RF model
inspector = RF_model.make_inspector()
feature_importances = inspector.variable_importances()
print(feature_importances)
```

Submission Instructions:

Maintain the same grouping as that of the first assignment. No new groups are allowed at this stage. Clean your Notebook code (ipynb) and name your submission file as IDNo.ipynb. Write a readme.text with your group members name and ID numbers. Compress these two files into a single Zip, and name your Zip file using your idno (in lowercase). Make only one submission on behalf of your group in google class assignment submission page. Deadline for submitting your work is 11th September 2024 midnight.

Note: Any clarification on this coding assignment may be emailed to I/C or f20210564@hyderabad.bits-pilani.ac.in.

References:

1. https://www.tensorflow.org/decision_forests/tutorials/beginner_colab
2. https://www.kaggle.com/code/fareselmenshawii/decision-tree-from-scratch

-------------------- End of text! -------------------