Random Forest Package and Its Usage

Feature Description:

In this experiment, we will be using Scikit-learn's Random Forest model. Random forest calculates the average detection depth of each feature among all the trees in the forest. The controls on the tree contribute to the final forecasting decision for the bulk of the input sample. Thus, a feature with high importance is considered high priority by the algorithm.

Hyperparameter:

n_estimators: Number of tree.

criterion: Function to measure the quality of a split ('gini' value/ 'entropy').

max_depth: Maximum depth.

min_samples_split: Minimum number of samples required to split an internal node.

min_samples_leaf: Minimum number of samples required to be at a leaf node.

1. n_estimators:

This parameter indicates the number of trees in the forest. In general, larger numbers lead to more robust models at the expense of increased computation. However, setting too many trees may significantly increase the computation time with only little improvements on the accuracy. In the data presented, we observe the performance of the model at three values: 50, 100, and 200. In some cases, 200 trees may be too much. In most cases, the best benefit is a balance between accuracy improvement and computational efficiency.

2. max_depth:

This parameter represents the maximum tree depth. A tree that is too deep may overfit the training data, resulting in poor generalization to other unseen data. Conversely, a tree that is too shallow will not capture the underlying dimensions of the data. Values of None, 10, and 20 have been used. None allows a tree to spread until all leaves have been expanded or less than the minimum number of samples needed for split. Using a max_depth of 10 or 20 prevents the tree from growing too deep and can prevent overfitting.

3. **criterion:**

The choice between them often doesn't significantly impact the model's accuracy, but entropy might be a bit slower to compute than gini because it involves logarithmic calculations.

A function is used to measure the quality of the distribution. The two most common are Gini and entropy. Although both aim to match children's root results, they do so in slightly different ways:

Gini: Measures variation in aggregate. A Gini Impurity of 0 indicates that all items in the set are identical, while a high value indicates a mixture of classes.

Entropy: Another measure of disorder or randomness. Like the Gini, the entropy is 0 all things being same.

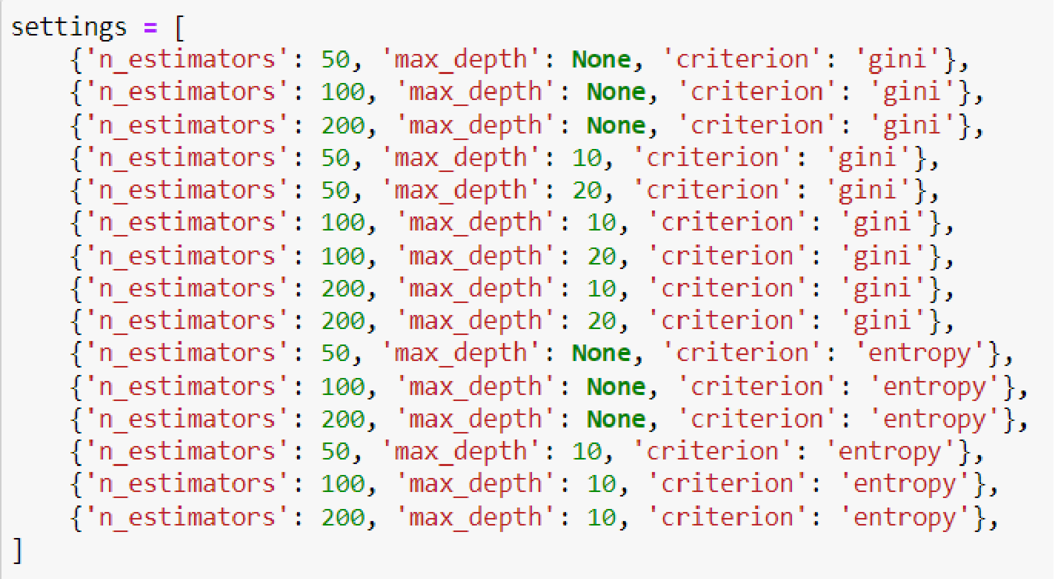
In general, the choice between them does not significantly affect the accuracy of the model, but the entropy estimate can be slower than the Gini because it involves logarithmic estimates.

Experimental Design:

Number of Estimators: The default value is 100. Increasing this number might result in better performance but it also increases computational cost. We will experiment from values of 50, 100, and 200 to observe any differences.

Max Depth: A deeper tree captures more nuanced patterns but might result in overfitting. We'll test with None (unlimited depth), 10, and 20.

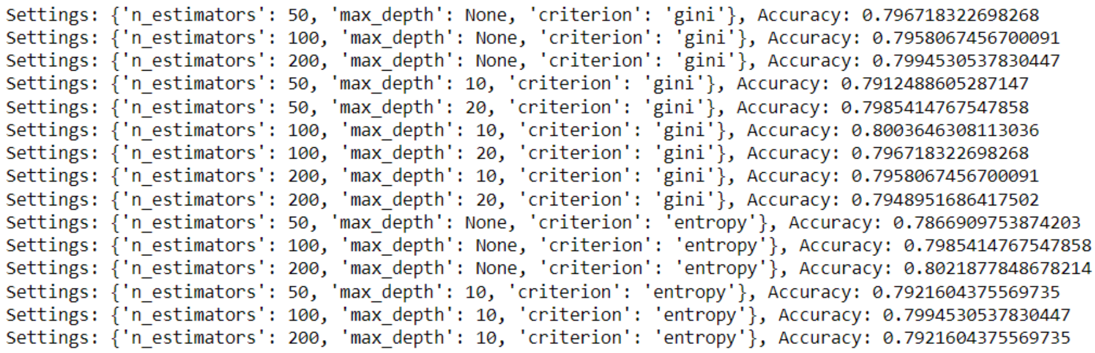
Criterion: We will use both 'gini' and 'entropy'.



```
settings = [
    {'n_estimators': 50, 'max_depth': None, 'criterion': 'gini'},
    {'n_estimators': 100, 'max_depth': None, 'criterion': 'gini'},
    {'n_estimators': 200, 'max_depth': None, 'criterion': 'gini'},
    {'n_estimators': 50, 'max_depth': 10, 'criterion': 'gini'},
    {'n_estimators': 50, 'max_depth': 20, 'criterion': 'gini'},
    {'n_estimators': 100, 'max_depth': 10, 'criterion': 'gini'},
    {'n_estimators': 100, 'max_depth': 20, 'criterion': 'gini'},
    {'n_estimators': 200, 'max_depth': 10, 'criterion': 'gini'},
    {'n_estimators': 200, 'max_depth': 20, 'criterion': 'gini'},
    {'n_estimators': 50, 'max_depth': None, 'criterion': 'entropy'},
    {'n_estimators': 100, 'max_depth': None, 'criterion': 'entropy'},
    {'n_estimators': 200, 'max_depth': None, 'criterion': 'entropy'},
    {'n_estimators': 50, 'max_depth': 10, 'criterion': 'entropy'},
    {'n_estimators': 100, 'max_depth': 10, 'criterion': 'entropy'},
    {'n_estimators': 200, 'max_depth': 10, 'criterion': 'entropy'},
]
```

Fig 1. Experiment Settings

Observations:



```
Settings: {'n_estimators': 50, 'max_depth': None, 'criterion': 'gini'}, Accuracy: 0.796718322698268
Settings: {'n_estimators': 100, 'max_depth': None, 'criterion': 'gini'}, Accuracy: 0.7958067456700091
Settings: {'n_estimators': 200, 'max_depth': None, 'criterion': 'gini'}, Accuracy: 0.7994530537830447
Settings: {'n_estimators': 50, 'max_depth': 10, 'criterion': 'gini'}, Accuracy: 0.7912488605287147
Settings: {'n_estimators': 50, 'max_depth': 20, 'criterion': 'gini'}, Accuracy: 0.7985414767547858
Settings: {'n_estimators': 100, 'max_depth': 10, 'criterion': 'gini'}, Accuracy: 0.8003646308113036
Settings: {'n_estimators': 100, 'max_depth': 20, 'criterion': 'gini'}, Accuracy: 0.796718322698268
Settings: {'n_estimators': 200, 'max_depth': 10, 'criterion': 'gini'}, Accuracy: 0.7958067456700091
Settings: {'n_estimators': 200, 'max_depth': 20, 'criterion': 'gini'}, Accuracy: 0.7948951686417502
Settings: {'n_estimators': 50, 'max_depth': None, 'criterion': 'entropy'}, Accuracy: 0.7866909753874203
Settings: {'n_estimators': 100, 'max_depth': None, 'criterion': 'entropy'}, Accuracy: 0.7985414767547858
Settings: {'n_estimators': 200, 'max_depth': None, 'criterion': 'entropy'}, Accuracy: 0.8021877848678214
Settings: {'n_estimators': 50, 'max_depth': 10, 'criterion': 'entropy'}, Accuracy: 0.7921604375569735
Settings: {'n_estimators': 100, 'max_depth': 10, 'criterion': 'entropy'}, Accuracy: 0.7994530537830447
Settings: {'n_estimators': 200, 'max_depth': 10, 'criterion': 'entropy'}, Accuracy: 0.7921604375569735
```

Fig 2. Experiment Results

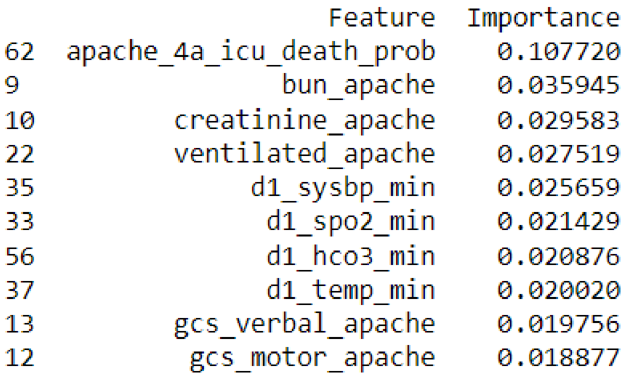
Best Settings: {'n_estimators': 200, 'max_depth': None, 'criterion': 'entropy'}, Best Accuracy: 0.8021877848678214

Generally, the Random Forest model performs better with higher numbers of trees (n_estimators).

Limiting the max_depth of the trees (to either 10 or 20) yields lower accuracy compared to None (which allows the trees to expand fully).

The entropy criterion, especially when max_depth is set to None, gives better results than the gini in some cases.

List of important features:

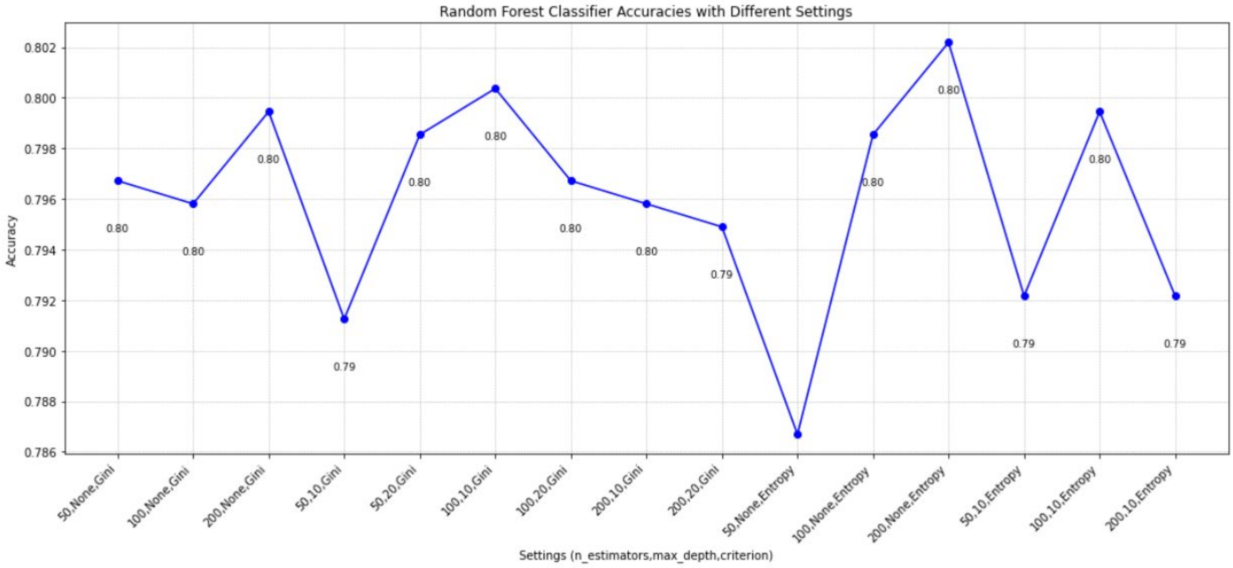


```
                    Feature  Importance
62  apache_4a_icu_death_prob    0.107720
9                bun_apache    0.035945
10         creatinine_apache    0.029583
22         ventilated_apache    0.027519
35              d1_sysbp_min    0.025659
33               d1_spo2_min    0.021429
56               d1_hco3_min    0.020876
37               d1_temp_min    0.020020
13         gcs_verbal_apache    0.019756
12          gcs_motor_apache    0.018877
```

Fig 3. List of important features

The important features consist of a combination of vital signs, laboratory test results, and neurological assessments. These are all the essential elements ICU which should be a huge determining factor. The data emphasizes the significance of early detection and monitoring of these important factors for better patient management.

Graphical Representation:


Random Forest Classifier Accuracies with Different Settings
Accuracy
50,None,Gini
100,None,Gini
200,None,Gini
50,10,Gini
50,20,Gini
100,10,Gini
100,20,Gini
200,10,Gini
200,20,Gini
50,None,Entropy
100,None,Entropy
200,None,Entropy
50,10,Entropy
100,10,Entropy
200,10,Entropy
Settings (n_estimators,max_depth,criterion)