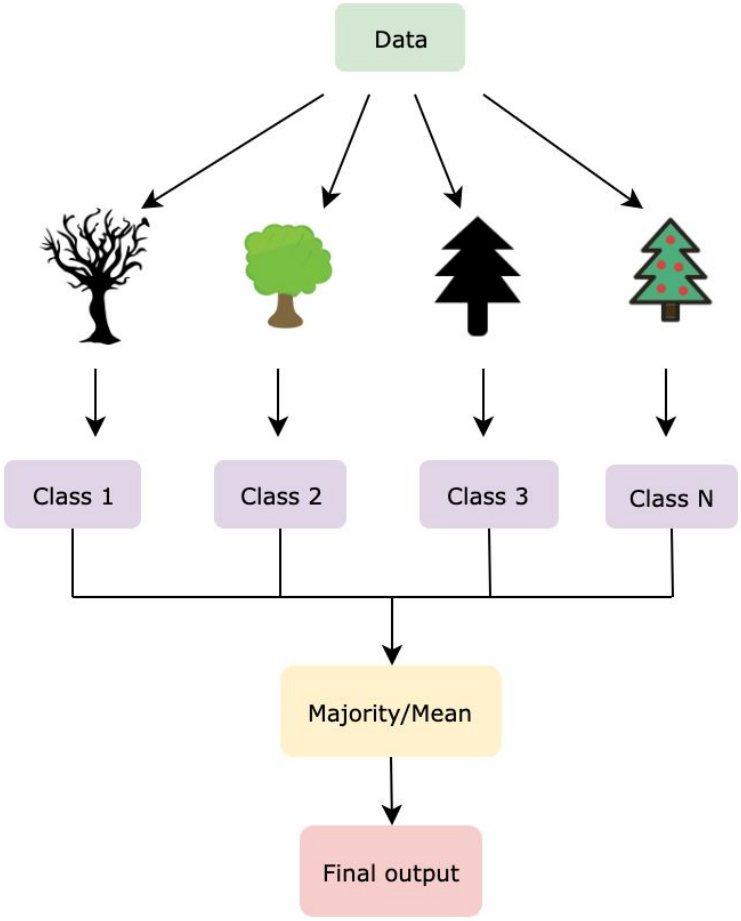
Vinoothna Bavireddy

Random Forest

Random Forests are considered advanced decision tree models where it creates several decision trees during the training process to make predictions. The final prediction is the majority/ mean of the several decision trees it creates. Random forest offers methods to balance errors between classes.

Major Hyperparameters in Random forest:

- **ntree :** Number of decision trees that the model should create. Typically values range between 100 and 1000.
- **mtry** : Number of features/variables selected based on feature importance (The feature importances indicates the sum of the reduction in Gini Impurity over all the nodes that are split on that feature) to split the data at each node.
- **replace**: To decide if sampling should be done with or without replacement.



Vinoothna Bavireddy

Steps:

- **Splitting:** Data is split into different subsets by selecting random features.
  Splitting the nodes follows the sqrt(n) method where if there are 9 features only 3 features will be used for splitting the nodes.
- **Bootstrapping**: Sampling random samples with replacement (meaning that some of the samples are used multiple times in a single tree).
- **Prediction:** The training is done on hundreds/thousands of decision trees where each node has limited specific features which are used for splitting. The final prediction is the average of the predictions of each decision tree.

Prediction for heart failure in a dataset using decision trees and Random forest:

- Importing packages

```
%matplotlib inline

import pandas as pd
import math
import numpy as np

import plotly.graph_objs as go
from matplotlib.pyplot import pie, axis
import matplotlib.pyplot as plt
import seaborn as sns

#import lightgbm
from scipy import stats
#import xgboost
from sklearn.model_selection import train_test_split
from sklearn.metrics import confusion_matrix, accuracy_score
from sklearn.ensemble import GradientBoostingClassifier
from sklearn.ensemble import RandomForestClassifier
from sklearn.tree import DecisionTreeClassifier
from sklearn.neighbors import KNeighborsClassifier
```

- Sample data:

```
df = pd.read_csv('hfcrd.csv')
df.head(10)
```

	age	anaemia	creatinine_phosphokinase	diabetes	ejection_fraction	high_blood_pressure	platelets	serum_creatinine	serum_sodium	sex	smoking
0	75.0	0	582	0	20	1	265000.00	1.9	130	1	0
1	55.0	0	7861	0	38	0	263358.03	1.1	136	1	0
2	65.0	0	146	0	20	0	162000.00	1.3	129	1	1
3	50.0	1	111	0	20	0	210000.00	1.9	137	1	0
4	65.0	1	160	1	20	0	327000.00	2.7	116	0	0
5	90.0	1	47	0	40	1	204000.00	2.1	132	1	1
6	75.0	1	246	0	15	0	127000.00	1.2	137	1	0
7	60.0	1	315	1	60	0	454000.00	1.1	131	1	1
8	65.0	0	157	0	65	0	263358.03	1.5	138	0	0
9	80.0	1	123	0	35	1	388000.00	9.4	133	1	1

Decision Tree classifier Model

```
dtc = DecisionTreeClassifier(max_leaf_nodes=10, random_state=30, criterion='entropy')
dtc.fit(x_train, y_train)
dtc_prediction = dtc.predict(x_test)
dtc_accuracy = dtc.score(x_test,y_test)
accuracy_scores.append(dtc_accuracy)
```

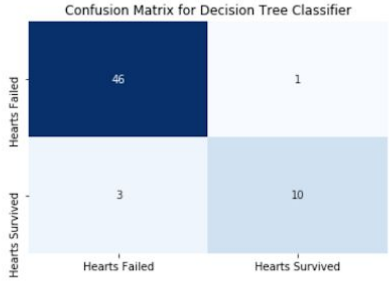
Check the Accuracy score

```
print("Accuracy score of Decision Tree Classifier is: ", dtc_accuracy)
```

```
Accuracy score of Decision Tree Classifier is:  0.9333333333333333
```

Confusion Matrix for Decision Tree Classifier

```
cm = confusion_matrix(y_test, dtc_prediction)
sns.heatmap(cm, cmap='Blues', annot=True, xticklabels=[ 'Hearts Failed','Hearts Survived'], \
            yticklabels= ['Hearts Failed', 'Hearts Survived'], cbar=False)
plt.title('Confusion Matrix for Decision Tree Classifier')
plt.show()
```



Random Forest Model

```
rfc = RandomForestClassifier(max_features=0.5, max_depth=15, random_state=1, n_estimators = 100)
rfc.fit(x_train, y_train)
rfc_prediction = rfc.predict(x_test)
rfc_accuracy = rfc.score(x_test,y_test)
accuracy_scores.append(rfc_accuracy)
```

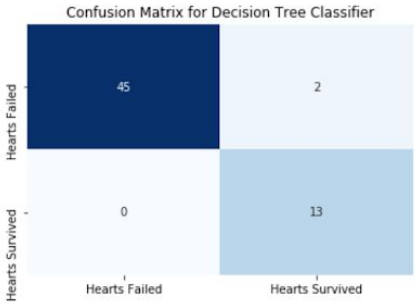
Check the Accuracy score

```
print("Accuracy score of Random Forst Classifier is: ", rfc_accuracy)
```

```
Accuracy score of Random Forst Classifier is:  0.9666666666666667
```

Confusion Matrix for Random Forest Classifier

```
cm = confusion_matrix(y_test, rfc_prediction)
sns.heatmap(cm, cmap='Blues', annot=True, xticklabels=[ 'Hearts Failed','Hearts Survived'], \
            yticklabels= ['Hearts Failed', 'Hearts Survived'], cbar=False)
plt.title('Confusion Matrix for Decision Tree Classifier')
plt.show()
```



From the above example we can see that the Random forest model outperformed the Decision Tree model.

Applications:

- Used for remote sensing/ pattern recognition
- Used in Gaming console
- Helpful in object recognition.

Advantages:

- Little training time
- High accuracy
- Using multiple trees reduces the risk of over-fitting
- Unlike decision trees, random forests can be used for large datasets.
- Even with missing data, random forest gives accurate predictions.

Disadvantages:

- Limited usability with regression models. Random Forest models tend to overfit continuous variables leading to inaccurate predictions.
- Very little user control and difficult to interpret in comparison to decision trees.
- Not good for predicting rare outcomes.

Conclusion:

Random forests perform better than decision trees by reducing the variance in the samples thus there is minimum risk of overfitting. The prediction capabilities increase because random forest uses the concepts of random sampling of observations, random sampling of features, and averaging predictions to provide more robust overall predictions than decision trees.

References:

1. Koehrsen, W. (2018, August 31). An implementation and explanation of the random forest in Python. Medium. https://towardsdatascience.com/an-implementation-and-explanation-of-the-random-forest-in-python-77bf308a9b76
2. Singh, H. (2019, March 24). Understanding random forests. Medium. https://medium.com/@harshdeepsingh_35448/understanding-random-forests-aa0ccecdbbbb
3. Koehrsen, W. (2020, August 18). Random forest simple explanation. Medium. https://medium.com/@williamkoehrsen/random-forest-simple-explanation-377895a60d2d

4. Aggiwal, R. (2017, February 28). Introduction to random forest. DIMENSIONLESS TECHNOLOGIES PVT.LTD. https://dimensionless.in/introduction-to-random-forest/