Random Forest

Random forest builds multiple decision trees and merges them together to get a more accurate and stable prediction.

Random forest is a supervised learning algorithm.

Advantages

Random forest can solve both classification and regression problems.

It is ensemble method and prone to overfitting

It may help future selections by telling with future is more significant

It takes care of null values

It can balance the data set if a class more frequent than other classes

Disadvantages

Large numbers of trees can make algorithms too slow for real-time predictions.

Random forest is a predictive modeling tool and not a description tool

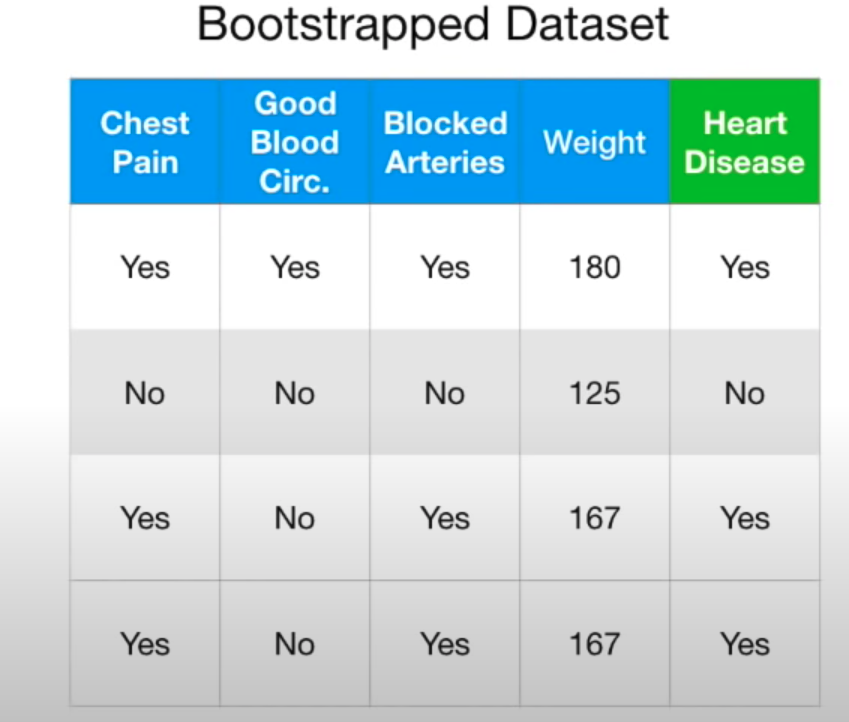
How does it work?



Step 1: Create a ‘bootstrapped’ dataset.

Randomly choose a subset of the original data without replacement. Which means some of the samples may repeat and some of the rows won’t be in the bootstrapped data.

Sample Bootstrapped



Step 2:

Create a decision tree using the bootstrapped dataset, but only use a random subset of variables (or columns) at each step.

Let’s say we picked good blood and blocked arteries. From this random sample we chose a root node based assuming that good blood circulation did the best job at separating samples.

We left chest pain, blocked, weight to choose as a chid node but we will chose that between chest pain and weight.

And make more bootstrapped dataset and choose random roots and build more trees.

How do we use it.

We get new patients and ask all trees in the forest and get total result from trees. Let's assume 5 trees said yes and 1 said no. Then decide based on majority vote that is yes.

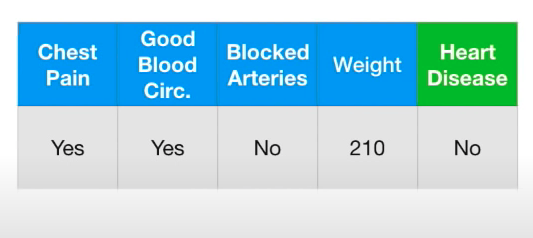
Bagging

Bootstrapping the data plus using the aggregate to make a decision is called ‘Bagging’. Simply combination of step 1 and step 2.

Measuring accuracy

Since we used bootstrapping method usually ⅓ of the data won't end up in the bootstrapped data which is called out of bag dataset.

Let’s assume below dataset is out of bad dataset.



We run this through all other trees that we built without the above sample.

Out samples said said 1 yes 3 no -- 4 yes 0 no, 3 yes 1 no

Then we can find the accuracy of how random forest classified out of bag samples.

Ultimately, we build random forest check accuracy and change variables to end up with the most accurate.

<https://towardsdatascience.com/random-forests-and-decision-trees-from-scratch-in-python-3e4fa5ae4249>

Python Sklearn Implementation

Import libraries

Load data

Separate futures and target

(Split data into train and test for some cases to find accuracy, Also we may need to normalize input for some ML algorithm)

Define the model

Fit the Model

Predict

Example

Random forest depth and number of esitmators

