| | |
|---|---|
| **Code** | **RANDOM_FOREST.PY** |
| **Author** | Nathaniel Heatwole, PhD (heatwolen@gmail.com) (GitHub) (LinkedIn) |
| **Summary** | Uses random forest to predict survival for passengers in the Titanic disaster |
| **Methods/ Process** | Random forest<br>- Supervised learning method that fits many decision trees ("forest") and aggregates the results<br>- Combines benefits of decision tree learning while mitigating their tendency to overfit to training data<br>- Each decision tree fitted on: 1) random subset of features; and 2) random selection of training data observations (with replacement)<br>- Randomly withholding some information (that would otherwise be available to fit the model) reduces correlations between trees<br>- Trees can be split using various measures, including entropy[1] or Gini impurity[2] (minimum sought in either case)<br>- *Root node* (top of tree): quantity/threshold yielding best split<br>- Predictions of many individual trees homogenized using plurality vote (classification) or average (regression)<br>- Out-of-bag testing can also be used (if entire dataset is not used to generate tree)<br>Steps<br>1. Import/clean training data<br>2. Feature engineering (prepare data for use in a random forest model, maximizing useful information that can be extracted from it)<br>3. Feature importance (correlation matrix, chi-squared, and coefficient of variation)<br>4. Fit random forest to training data to predict survival |
| **Training Data** | Titanic dataset – containing data for 891 Titanic passengers (from Kaggle) |
| **Results** | - High predicted survival probabilities for survivors (mean: 0.87), and opposite for non-survivors (mean: 0.09)<br>- IQRs for predicted survival probability for the two groups are non-overlapping<br>- Feature importance indicates most useful variables are: fare, male, and age |

[1] *Entropy* is a measure of disorder, defined as the expected value of information, equal to: sum[-p*ln(p)].

[2] *Gini impurity* measures how often a randomly chosen element would be incorrectly labeled (if group labels were assigned randomly, using the distribution of labels in the training set), equal to: sum[p*(1 - p)].