**Section 3:** **How to Implement an RF**

* “bagging” is used in tandem with “random feature selection”
  + Bagging means randomly selecting a subset of instances (usually 2/3rds of the dataset) to train each tree with
    - We track which instances are considered Out-of-Bag (OOB) for a given tree by recording which instances were used to train the tree
    - Any instances from the overall training dataset that aren’t in a given tree’s list of instances are considered OOB for that tree
    - These OOB instances are used to test that tree and provide an OOB estimate of the tree’s error rate
    - This removes the need to completely set aside a “test” dataset and exclude instances from contributing to the learning algorithm
    - The author wrote a 1996 paper that proves this method of data segmentation performs just as good as the old Train/Test splits
* Each individual tree is not pruned, meaning no leaf nodes should be ambiguous

**Section 4: Random Feature Selection**

* It seems that the author found that randomly choosing features to split on at each node of a tree didn’t perform significantly worse than always finding the best split considering all the features
* This section introduces the 1st method of growing an RF that takes advantage of this finding: **Forest-RI**
  + This method randomly selects a preset number of features (hyper-parameter named *F*) to exclusively split a node on, disregarding the possibility of any other features offering a split with higher information gain

**Section 5: Forest-RC, A 2nd Method**

* Randomly generating a new feature to split on by choosing some combination of existing features, adding them together, and then adding noise of either +1 or -1.
* Typically, just combining 2 features together is enough to see good overall performance in the RF
  + Datasets with a large number of instances or features (not sure what the author means by “large” in this context) may significantly benefit from combining more features together
* To allow the adding together of categorical features, the author recommends randomly choosing some of the categorical values to be replaced with 1 while the rest are replaced with 0.
  + If most or all of the features in the dataset are categorical, the author recommends combining at least features; where M = number of features.

**Section 8**

* RF are shown to be relatively insensitive to noise, meaning they are better predictors than prior ensemble ML methods

**Section 9**

* Describes how RF performs when none of the features strongly point to any classification
* Using Forest-RI and higher values of *F* will improve the performance of the RF

**Section 10: Conjecture on how RFs work**

* The author posits a way to show how important a given feature is to forming the classifier

**Section 11: RFs for Regression**

**Appendix:**

* See this section for how to calculate the RF’s *strength, correlation, variance, and std. dev.*