- Briefly describe the attributes setting of the random forest model, including:

  1. The number of trees you used

     I use 100 trees for random forest. (n_trees = 100)

  2. The number of features you used

     For each iteration, I use 15 features to train a decision tree. (n_features = 15)

  3. The number of instances you used to build each tree

     I use 80 percent of training data for training model, and the rest 20 percent of data are for validation. Also, I set the sample ratio to 0.7 (sample_size = 0.7), thus there are 8500 * 0.8 * 0.7 = 4760 instances to build each tree.

- Briefly describe the difficulty you encountered

  In the beginning, I just followed the template to build the random forest, but I found that the f1 score is not good enough, approximately 0.68 for my first few tries. No matter how I finetuned my hyperparameters mentioned above, I still didn't get too much improvement. So, I wondered why my model's f1 score is so low.

- Summarize how you solved the difficulty and your reflections

  Then I realized that I didn't implement any extra function other than template. Accordingly, I applied some techniques. For example, when building a tree in each iteration, I calculated the accuracy of each tree on validation set, and set the tree's weight to be the accuracy of validation set. In the end, when deciding each testcase's prediction, it is based on the **weighted average** of the prediction from each tree. To be more specific, if the weighted average is >= 0.5, then predict 1; otherwise predict 0.