Report of HW2 - Diabetes Prediction

Student ID: 109062211 Name: 張惇媛

I. The attributes setting of the random forest model

1. The maximum depth = 7
2. The minimum samples split = 10
3. The number of trees you used = 31
4. The number of features you used = 11
5. The number of instances you used to build each tree

   To avoid sample data being imbalanced, I use 2*n_samples instances (half with label0 & half with label1) in each tree.

   n_samples = int(0.5 * min(#label1, #label0)) with # means the number of
6. (optional) any other settings
   - random.seed (use for random.sample features and instances) = 0(a constant)
     Fix random seed with a constant 0 to make the output reproducible
   - n_dimensions (use for feature_selection) = 18 which is 2/3*24(initial dimensions)
     Drop out some features which are useless and leave the top 18 helpful ones for further actions.
     Examples: values of "gcs_unable_apache" are all 0 in advanced input data

II. The difficulty you encountered

1. Find the best threshold

   Spend time on finding how to generate the threshold to split the data into two groups.
2. Set the attributes of the random forest model

   The attributes will affect the performance and the execution time of the model.

III. Summary: The Solution of the Hardness and Reflections

First, to find the threshold, I use the concept of Gaussian distribution. Since the data near the mean are more frequent in occurrence, I start at the point of mean – 1.5σ and continue adding the delta of 0.05σ until the point of mean + 1.5σ to get the threshold that obtain the best information gain.

Second, with the observation that some features are useless in decision tree, I select features based on each fisher score[1] before repeatedly building trees. Then, use random to collect the features from previous selected ones in each tree. In addition, since simply use random to collect instances may cause the problem of imbalanced data[2] (like only small portion of data with label1 while others are label0), I separate data into two groups by its label and then respectively choose instances from both. Finally, repeatedly build trees to make the random forest model.

In this assignment, it is important to take some common problems that may happened into consideration and try to solve them to optimize the random forest model. There are a lot of papers that illustrate many clever ways to handle these problems. It is helpful to make use of them when we have difficulty implementing some machine learning projects.

---

[1] Reference: *Generalized Fisher Score for Feature Selection* https://arxiv.org/abs/1202.3725

[2] Reference: Wikipedia oversampling and undersampling - two solution for imbalanced data https://en.wikipedia.org/wiki/Oversampling_and_undersampling_in_data_analysis