CIS 662: Assignment 4

Instructions: This is an individual assignment. You are welcome to look anything up online but do NOT collaborate with your classmates or get significant help from anyone outside. The due date is a week from when this is assigned. You need to: (i) answer the questions by going to blackboard; and (ii) upload a PDF of the python code which includes the run outputs (make sure the code is fully visible in the output). **For all questions, if float then round off to 2 decimal places.**

1. **Get data and pre-process**: From http://www.howstat.com we have downloaded and pre-processed some data on cricket batting performances of various male players in various batting positions. As a result of the pre-processing, you would not be able to tell if the batsman was not out when they made their highest score. Also, if a player played for multiple countries, we only use the first country they played for. The resulting dataset is `cricket_batting_data.xlsx`. Make sure to use `pd.read_excel` while reading the data, and there are no date columns to "parse_dates". The goal is to predict the highest score for a player (this column in the dataset is `HS` and it is the dependent variable).

   (a) To give a quick description of the columns: `Player` is the name of the player (and there could be duplicates because they may have played for different countries or in different positions); `Country` is the country the player first represented; `Years` tells us from which year to which year the player played; `Inns` is the number of innings played; `NO` is the number of not-outs for that player in that position; `Runs` is the total runs scored; `Ducks` is the number of zeros; `50s` is the number of innings they scored 50-99; `100s` is the number of innings they scored 100 or above; `Avg` is the average runs scored per inning they were out; and `Position` takes values 0, 3, 4, 5, 6, 7, 8 which respectively denote Opener, 1st down, 2nd down, 3rd down, 4th down, 5th down, and 6th down. *How many rows of data are in the data frame? Also, are there any NaN values*?

   (b) To that data frame, we are going to add a few columns and subtract a few. Create a column called `Debut` to denote the year the player started (this can be obtained using the first four characters in the column `Years`). Create another column called `Tenure` to denote the number of years the player played (this is one more than the difference between the last 4 characters and the first four characters in the column `Years`). Make sure to convert `Debut` and `Tenure` columns as `int64`. As a check, see if A.N. Cook has a debut year as 2006 and tenure of 13 years (this is the first row). *The second row is that of S.M. Gavaskar. What did you get for debut year and tenure for Gavaskar?*

   (c) Take that data frame, drop three columns `Player`, `Years`, and `Avg` (note that the `Avg` is linearly dependent on other columns as it is just the number of runs divided by innings minus not-outs). *How many columns of data are in the resulting data frame?* Display the header.

   (d) Convert the categorical column `Country` into a quantitative one by using `get_dummies` and be sure to drop first. Split the rows of data so that 80% is randomly selected as train data and the remaining as test data. Use 'HS' as the response variable (y). Make sure you use a random state equal to 35. Use a standard scaler to scale the data. *What are the lengths of the train and the test data?*

2. **Multi-method Regression**: Our goal is to compare 5 methods. The first two are what you did before the midterm and the last three are based on ensemble of trees. *In all five parts below, the metric "error ratio" denotes the ratio of the mean absolute error to the mean "HS" for the test data.*

   (a) Perform a linear regression using the training data and in the code output display R^2 and the intercept for the training. Make predictions using the test data. *What was the "error ratio"?*

   (b) Perform a Lasso regression with alpha of 0.05, and obtain the score and the intercept value for the train data. *What was the "error ratio"?*

   (c) Use a bagging regressor with random state equal to 50 and maximum samples of 100. Display the mean absolute error for the test data. *What was the "error ratio"?*

   (d) Use a random forest regressor with random state equal to 50, square root for the maximum features, n_estimators as 200, and minimum samples leaf of 2. Display the mean absolute error for the test data. *What was the "error ratio"?*

   (e) Use a gradient boosting regressor with random state equal to 50, minimum samples split of 6, minimum sample leaf of 2, and maximum depth of 5. Display the mean absolute error for the test data. *What was the "error ratio"?*

3. **Alternative Metrics**: A natural question is whether there are other metrics to be considered while comparing algorithms.

   (a) When there is a large variability and some values are equal to zero, a metric called sMAPE (symmetric mean absolute percentage error) can be used. For the test data, if a_1, a_2, ..., a_n are actual ground truth values, and p_1, p_2, ..., p_n are the predicted values, then sMAPE is

   $$sMAPE = \frac{1}{n}\sum_{i=1}^{n}\frac{|a_i - p_i|}{|a_i| + |p_i|}$$

   in terms of the absolute values of the numbers. Use the above definition with the understanding that some people like to also multiply the above by 2, but we will not. *Write down the sMAPE for the test data predictions by comparing against the ground truth for all five methods.*

   (b) For the "error ratio" and the sMAPE we used the mean. However, in such kind of problems, some outliers could significantly skew the mean. *For all the five methods obtain "median error ratio" and the symmetric median absolute percentage error.*

   (c) We wish to ask if any of the methods dominates the others in all the metrics. *For this data set, which would be the preferred method among the five*?

4. **Further Improvement Attempts**: Hybrid method, feature selection, and parameter tuning.

   (a) Oftentimes we use hybrid methods for predictions. For the training data, say y_i is the actual and $\hat{y}_i$ is the prediction using gradient boosting. Then you can train a random forest model for which the `y-train` is the residuals, i.e. $y_i - \hat{y}_i$. Then when we make a

prediction for the test data using gradient boosting with `X-test` and add the predictions from the random forest model. *Write down the performance of this algorithm in terms of the symmetric median absolute percentage error, and say if it does better than the original five methods.*

(b) Recall that we used `get_dummies` and created a lot of columns for countries. Obtain the symmetric median absolute percentage error for all 5 methods (plus the hybrid method) by also dropping the 'Country' column when we drop 'Player', 'Years' and 'Avg'. *For all the five methods obtain symmetric median absolute percentage error using this reduced feature set. Do you see an improvement from when we had more features?*

(c) Pick 2-3 parameters for one of the tree-based ensemble methods. Try multiple values, select one set at a time, and check on the training data which one has the best result in terms of the sMAPE of `HS`. You may want to define a function to do this, and not manually try all the options (but that is just a suggestion and not a requirement). Make sure your code output has the details. *In the text box, write down what you tried and what you observed in terms of the performance using the best parameter set.*