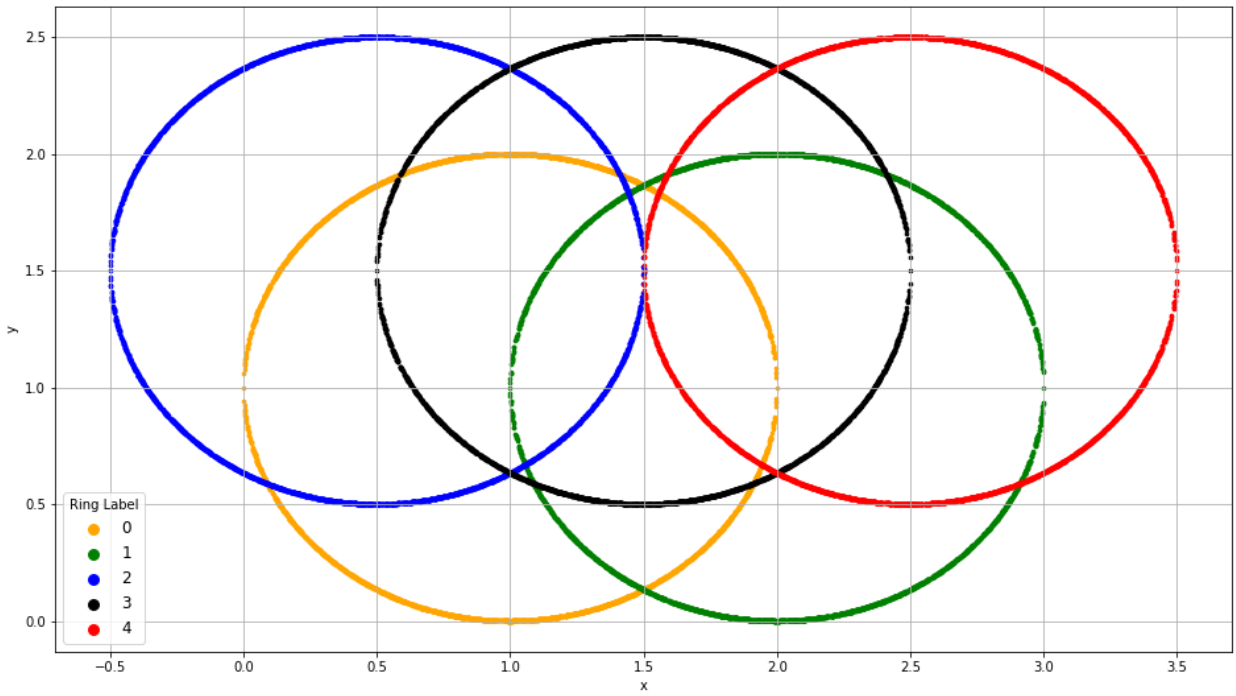
CS 584: Machine Learning

Pradyot Mayank(A20405826)

Spring 2019 Assignment 5

The Five Rings Data

1. You will analyze the FiveRing.csv data for all the questions.
2. This data has 20,010 observations and three numeric fields, namely, x, y, and ring.
3. The fields x and y are the x-coordinate and the y-coordinate of the rings respectively.
4. The field ring indicates to which ring the coordinates belong.
5. The rings are labelled 0, 1, 2, 3, and 4.
6. The graph below shows the five rings.



Misclassification Rate

Let $p_{ij}, j = 0,1,2,3,4$ be the predicted probability that the *i*-th observation will belong to the *j*-th ring. The predicted ring for the *i*-th observation is the smallest ring label which has the highest predicted probability. The following examples illustrate how the predicted ring is determined.

- Suppose $p_{i0} = 0.25,\ p_{i1} = 0.1,\ p_{i2} = 0.2,\ p_{i3} = 0.3,\ p_{i4} = 0.15$, then the predicted ring is 3 because p_{i3} is the highest value among the five probabilities.
- Suppose $p_{i0} = 0.25,\ p_{i1} = 0.1,\ p_{i2} = 0.25,\ p_{i3} = 0.25,\ p_{i4} = 0.15$, then the predicted ring is 0. Although p_{i0}, p_{i2}, p_{i3} are tied for the highest probability, the smallest ring label is 0.

An observation is *misclassified* if the predicted ring label is different from the observed ring label. The Misclassification Rate is the proportion of all the observations which are misclassified.

Root Average Squared Error (RASE)

The Root Average Squared Error is

$$\text{RASE} = \sqrt{\frac{1}{2n}\sum_{i=0}^{n-1}\sum_{j=0}^{4}\left(\delta_{ij} - p_{ij}\right)^2}$$

where

- $n = 20010$ is the number of observations.
- $\delta_{ij} = 1$ if the ring label of the *i*-th observation is *j*. Otherwise, $\delta_{ij} = 0$.

Question 1 (100 points)

You will build the multinomial logistic model according to the specifications below. You will use the Misclassification Rate and the Root Average Squared Error to assess the performance of your model.

- The nominal target variable is ring
- The predictors are x and y.
- The model will have the Intercept terms.
- The maximum number of iterations is 1000.

Build and assess the multinomial logistic model using all 20,010 observations **without** bagging and answer the following questions.

a) (10 points). List the parameter estimates (round to four decimal places) in a table. The rows are the Intercept, the predictor x, and the predictor y. The columns are the ring labels.

	0	**1**	**2**	**3**
const	-4.4355	0.1877	-2.7014	-8.7026
x	2.9533	-1.4679	1.4554	4.4382
y	0.0007	0.7201	0.6947	0.7194

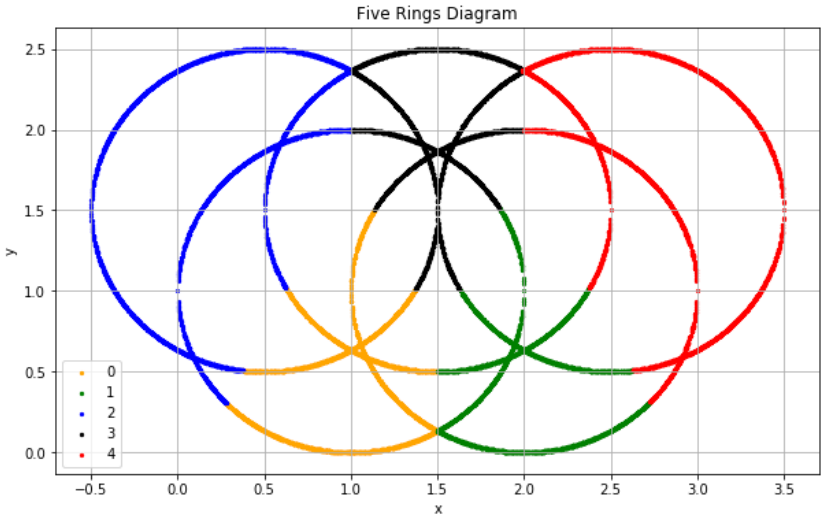
b) (10 points). What is the Misclassification Rate?

Misclassification Rate: 0.58806.

c) (10 points). What is the Root Average Squared Error?

Root Average Squared Error: 0.5581952.

d) (10 points). Redraw the above picture (i.e., the field y on the vertical axis and the field x on the horizontal axis), however, use the predicted ring label for coloring. The coloring scheme is 0 = orange, 1 = green, 2 = blue, 3 = black and 4 = red.



Apply the Bagging technique, build and assess the multinomial logistic model using all 20,010 observations. The initial random seed is 20190430. Try number of bootstraps equals to 10, 20, 30, 40, 50, 60, 70, 80, 90 and 100.

Note: you are not allowed to use any functions (e.g., BaggingClassifier) in the sklearn.ensemble module to perform Bagging. Instead, you must write your Python codes to implement the Bagging algorithm.

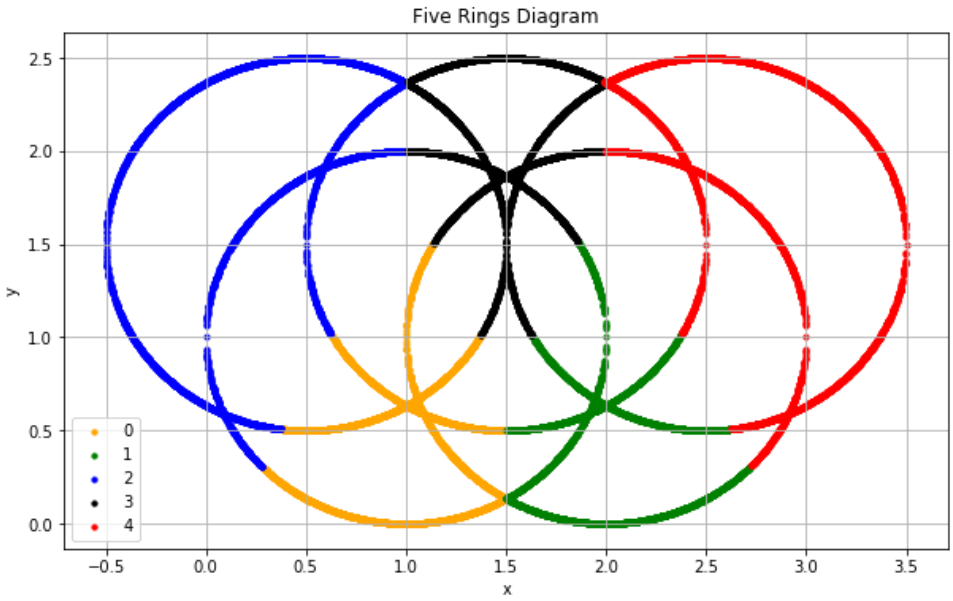
e) (40 points). List the Misclassification Rate and the Root Average Squared Error of the bootstrap results. The columns are the two metrics. The rows are the number of bootstraps. Also, include the no-bootstrap (i.e., zero number of bootstrap) metrics.

Number of Bootstraps	Misclassification Rate	Root Average Squared Error
0(No Bagging)	0.588006	0.5581952
10	0.588061	0.5582412
20	0.5878886	0.5582039
30	0.5877894	0.5581898
40	0.5877849	0.5581915
50	0.5878191	0.5581952
60	0.5878236	0.5581997

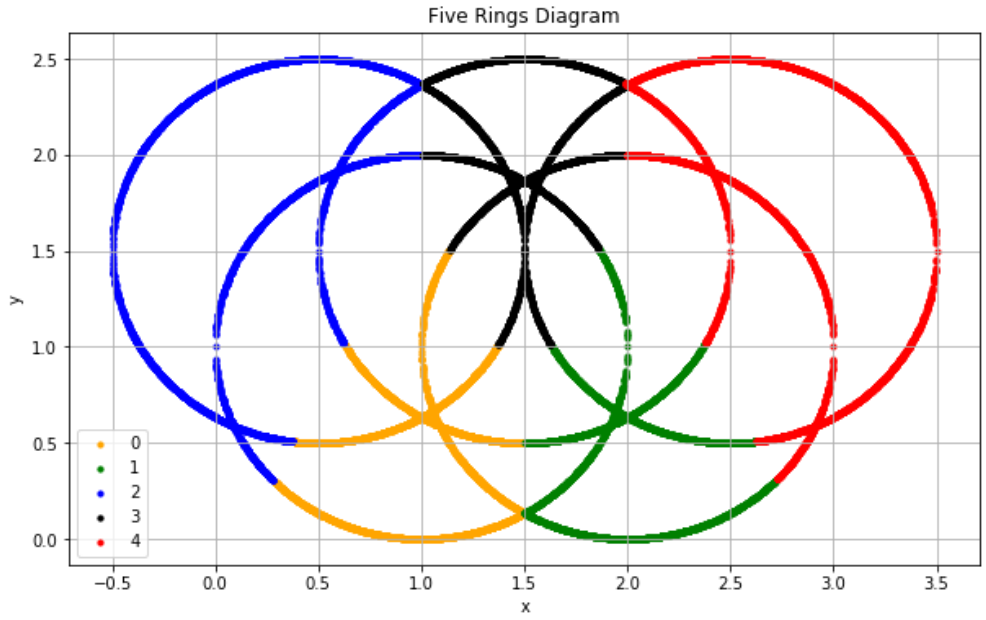
70	0.5878261	0.5582007
80	0.5878480	0.5582044
90	0.5878278	0.5582037
100	0.5878131	0.5582024

f) (10 points). Redraw the above picture (i.e., the field y on the vertical axis and the field x on the horizontal axis), however, use the predicted ring label for coloring. The coloring scheme is 0 = orange, 1 = green, 2 = blue, 3 = black and 4 = red. There should be ten pictures, one for each set of bootstraps.

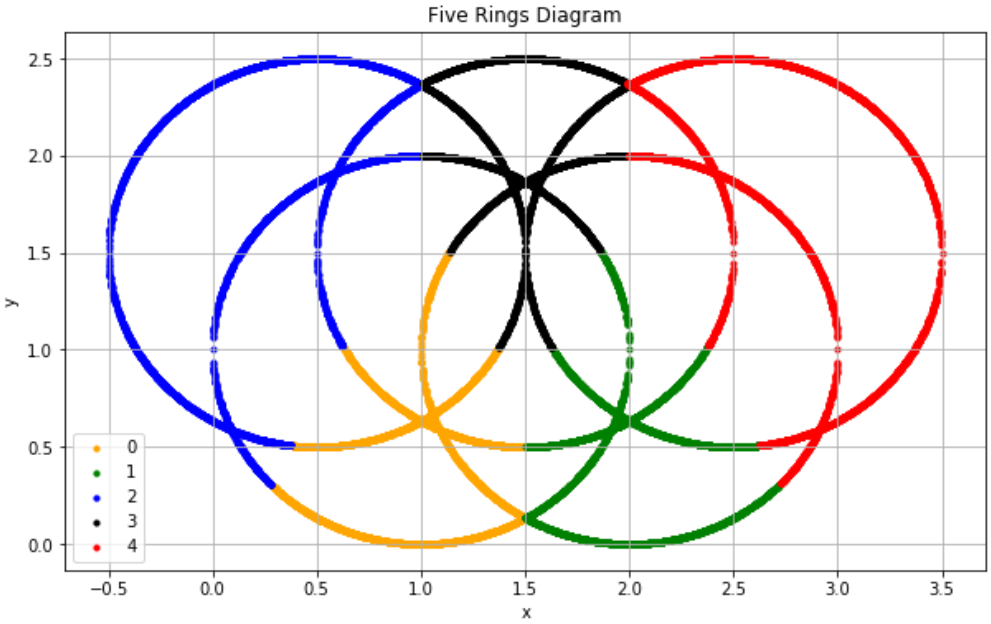
For 10 bags:



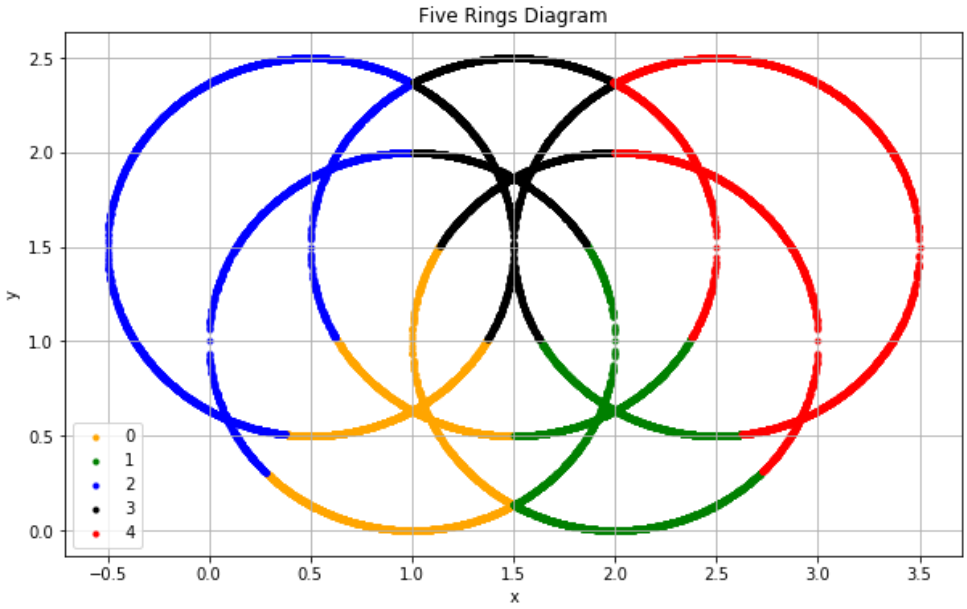
For 20 bags:



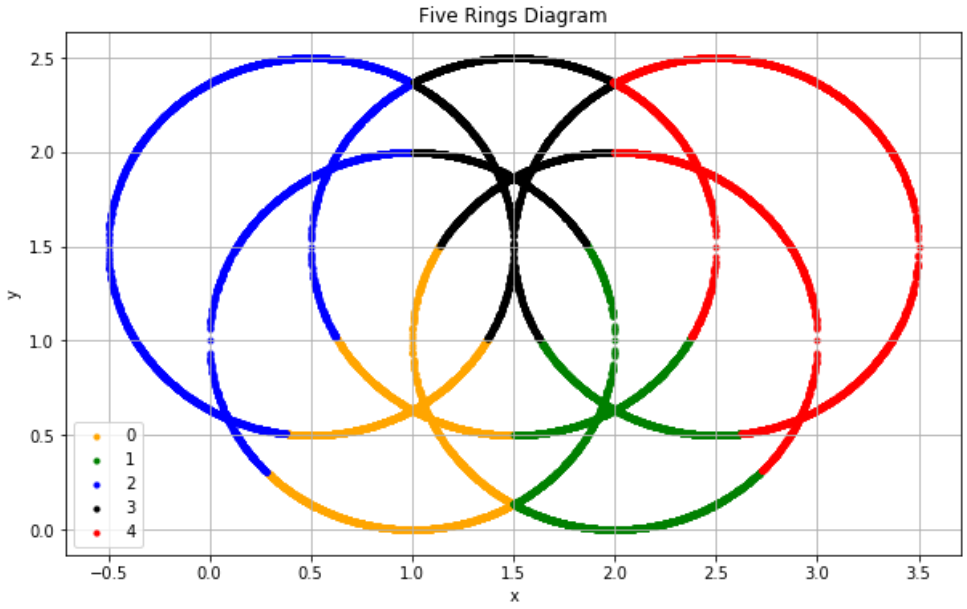
For 30 Bags:



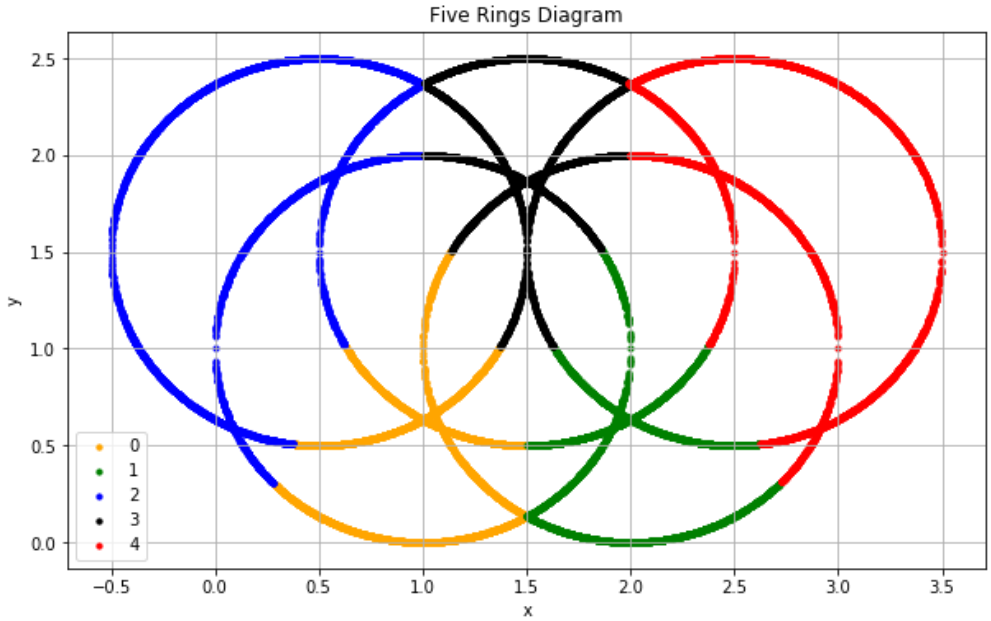
For 40 Bags:



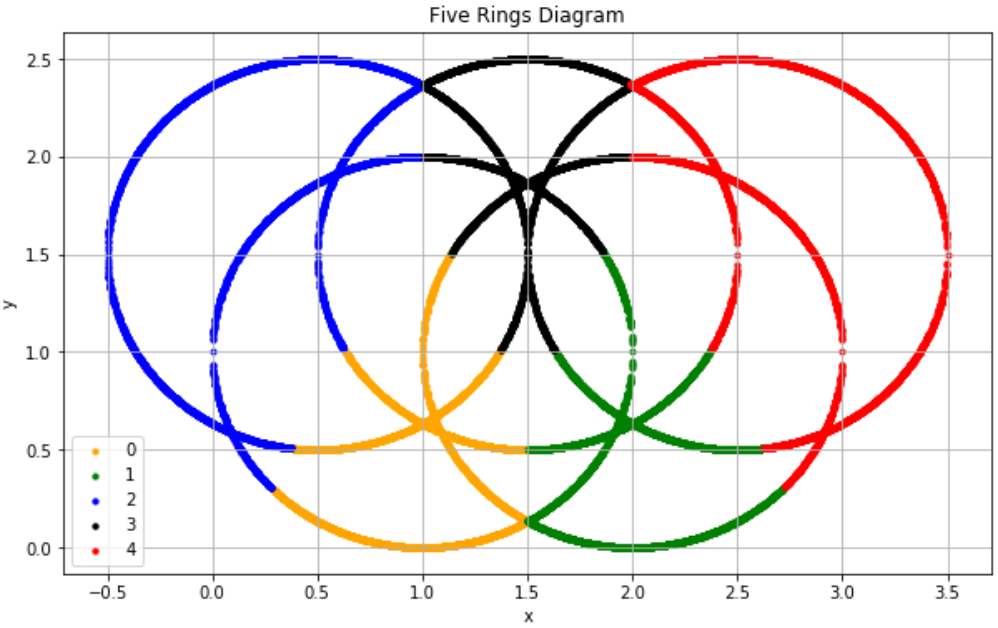
For 50 bags:



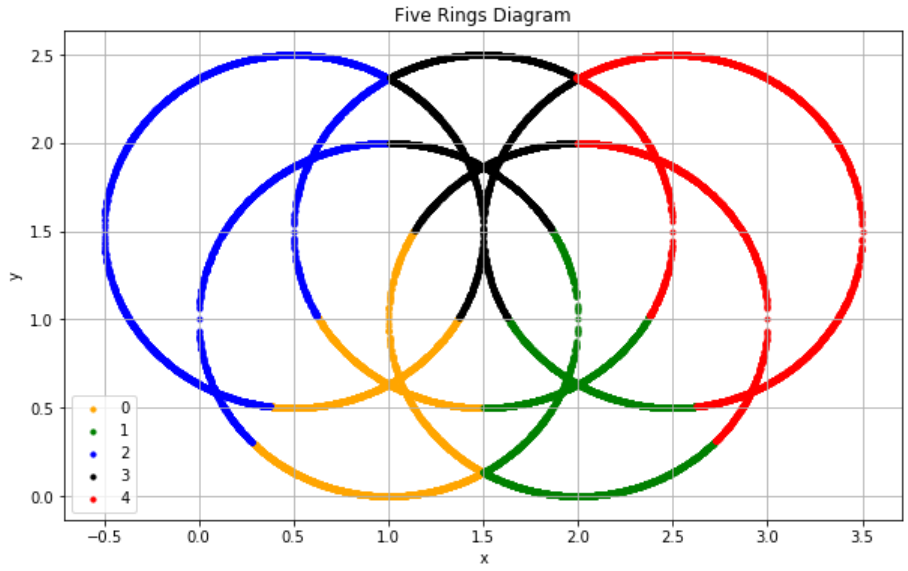
For 60 bags:



For 70 bags:

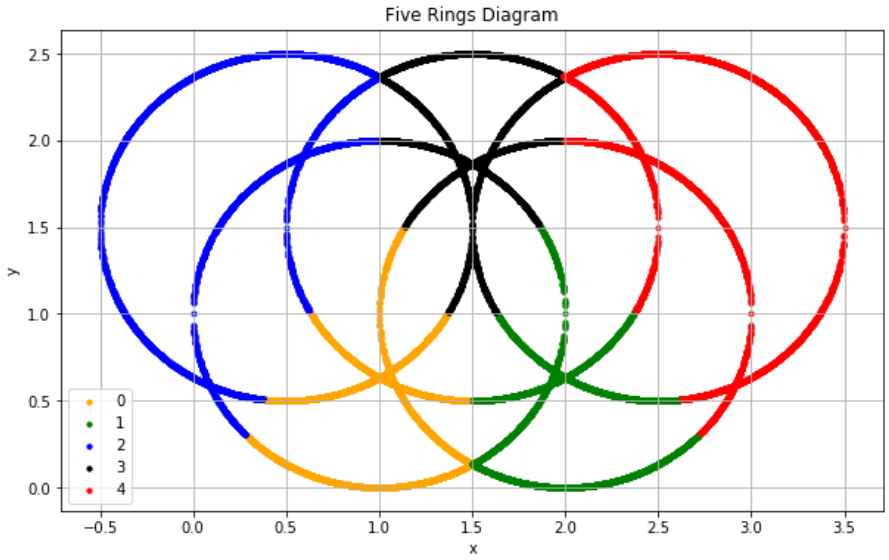


For 80 bags:



For 90 bags:

For 100 bags:



g) (10 points). Compare the results between the bagging results and the non-bagging results. Briefly comment on the comparison.

As we can observe, bagging produces more accurate results if compared to non- bagging results. Since, in bagging refers to Bootstrap Aggregation i.e. random sampling with replacement is done form the original data set. Then multiple classifier (Multinomial Regression) is trained on each bag of samples of the same size independently and output is aggregated at the end without preference to any model.

We can see that on increasing the number of bags the misclassification rate is decreasing very slowly and same we can say for the RASE.

Question 2 (100 points)

You will build the classification tree model and then apply the Adaptive Boosting technique. You will use the Misclassification Rate and the Root Average Squared Error to assess the performance of your model. The classification tree model should be built according to the specifications below.

- The nominal target variable is Ring
- The predictors are x and y.
- The splitting criterion is Entropy
- The maximum depth is 2.
- The random state value is 20190415

Build and assess the classification tree model using all 20,010 observations **without** boosting and answer the following questions.

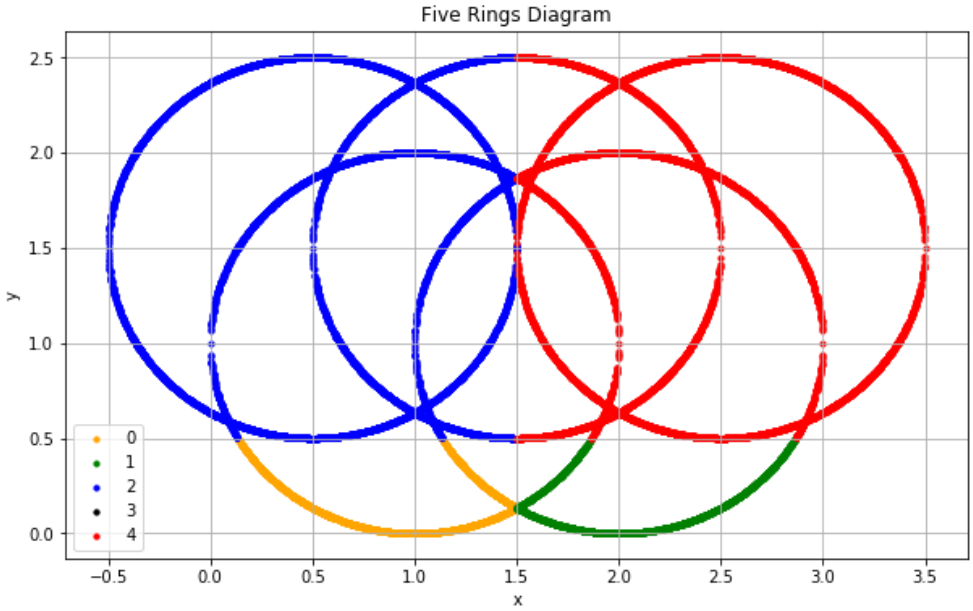
a) (10 points). What is the Misclassification Rate?

Misclassification Rate: 0.4641179

b) (10 points). What is the Root Average Squared Error?

Root Average Squared Error: 0.5507919

c) (10 points). Redraw the above picture (i.e., the field y on the vertical axis and the field x on the horizontal axis), however, use the predicted ring label for coloring. The coloring scheme is 0 = orange, 1 = green, 2 = blue, 3 = black and 4 = red.



Build and assess the classification tree model using all 20,010 observations **with** boosting with initial random seed 20190430. Try the maximum number of iterations equals to 100, 200, 300, 400, 500, 600, 700, 800, 900, and 1000. The case weights are determined as follows.

- The case weights are initialized to 1.
- After each iteration, the case weight is the $1 + \frac{1}{5}\sum_{j=0}^{4}\left|\delta_{ij} - p_{ij}\right|$ for a misclassified observation. The case weight is $\frac{1}{5}\sum_{j=0}^{4}\left|\delta_{ij} - p_{ij}\right|$ for a correctly classified observation.

The iteration stops if either the Misclassification Rate is zero or the maximum number of iterations is reached. The aggregated predicted probabilities from a set of boosting are the weighted mean of the predicted probabilities of the iterations. The weights are the accuracy (i.e., one minus the misclassification rate) of the iterations.

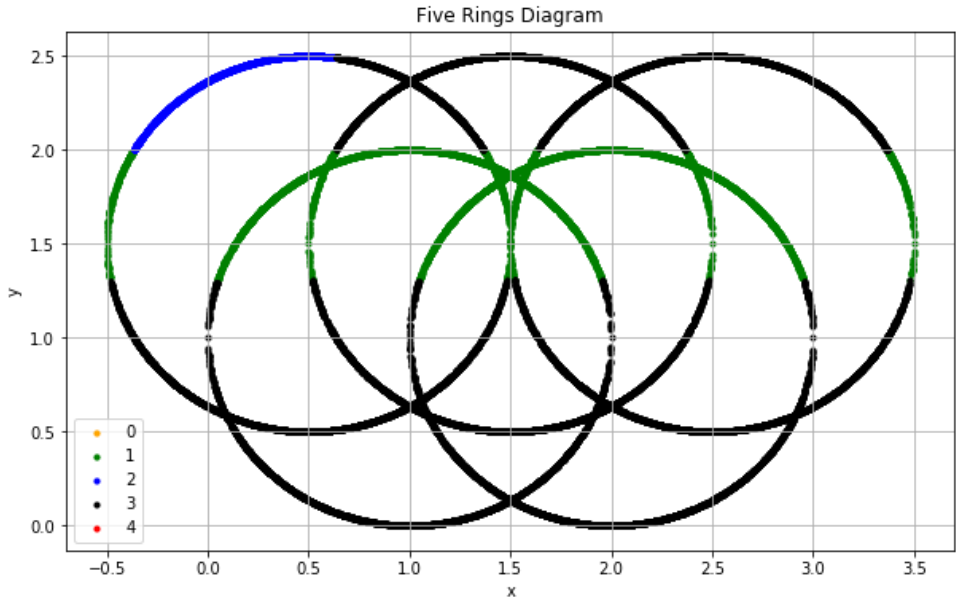
Note: you are not allowed to use any functions (e.g., AdaBoostClassifier) in the sklearn.ensemble module to perform boosting. Instead, you must write your Python codes to implement the Boosting algorithm.

d) (50 points). List the Misclassification Rate and the Root Average Squared Error of the aggregated boosting results. The columns are the number of iterations performed and the two metrics. The rows are the maximum number of iterations. Also, include the no-boosting metrics.

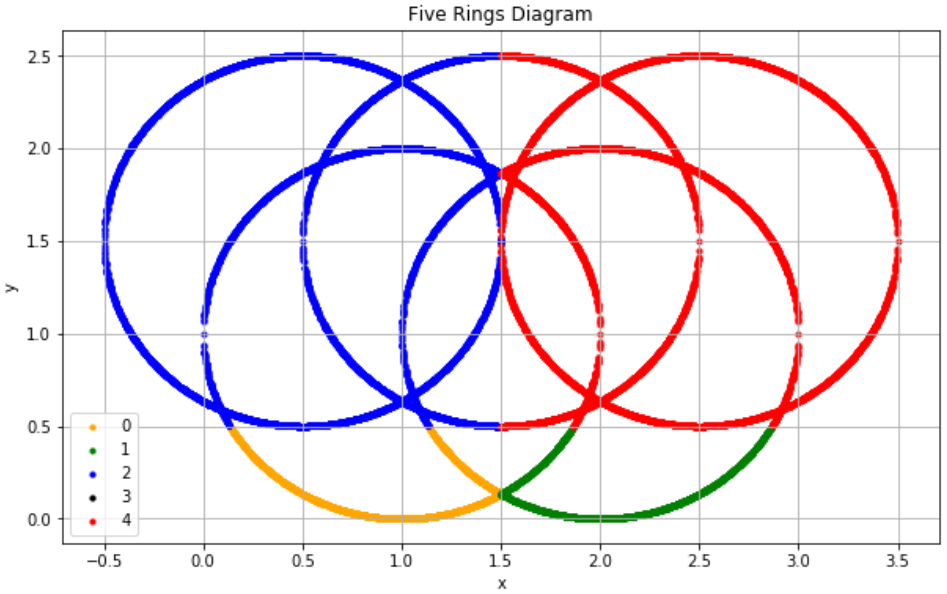
Number of Iterations	Misclassification Rate	Root Average Squared Error
0(Without Boosting)	0.4641179	0.5507919
100	0.4465839	0.6419789
200	0.3131376	0.5800995
300	0.4465839	0.6419789
400	0.3131376	0.5800995
500	0.4465839	0.6419789
600	0.3131376	0.5800995
700	0.4465839	0.6419789
800	0.3131376	0.5800995
900	0.4465839	0.6419789
1000	0.3131376	0.5800995

e) (10 points). Redraw the above picture (i.e., the field y on the vertical axis and the field x on the horizontal axis), however, use the predicted ring label for coloring. The coloring scheme is 0 = orange, 1 = green, 2 = blue, 3 = black and 4 = red. There should be ten pictures, one for each set of boosting.

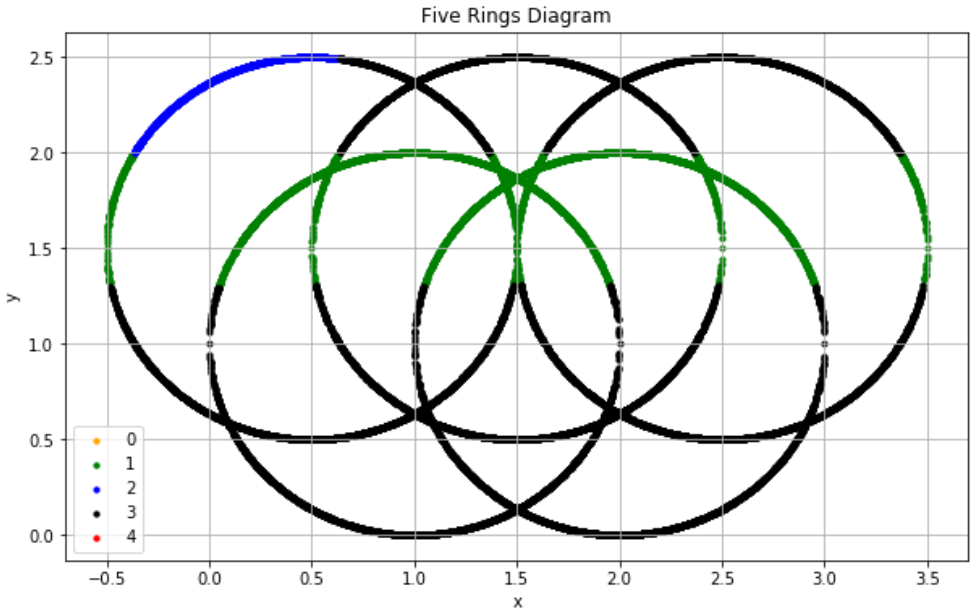
For 100 iterations:



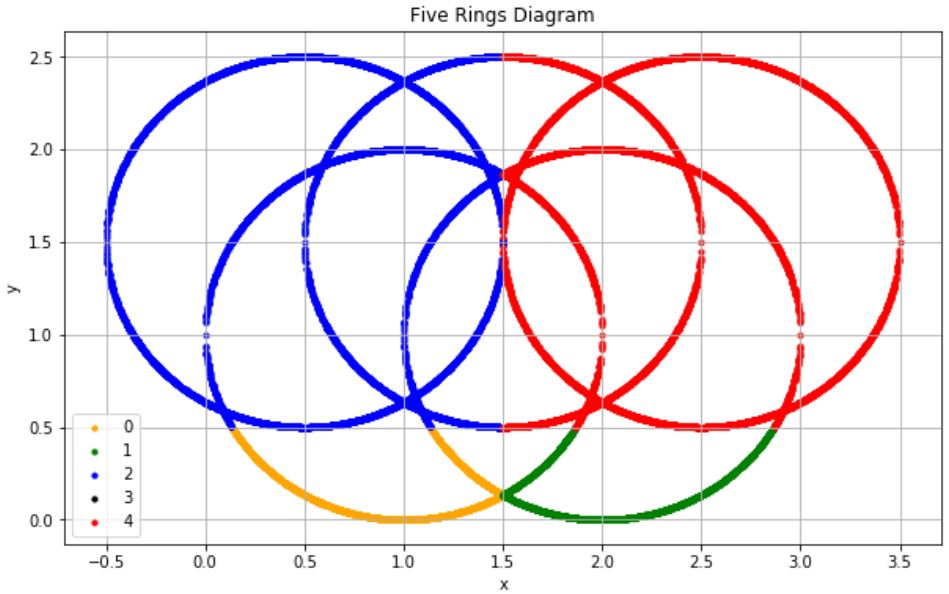
For 200 iterations:



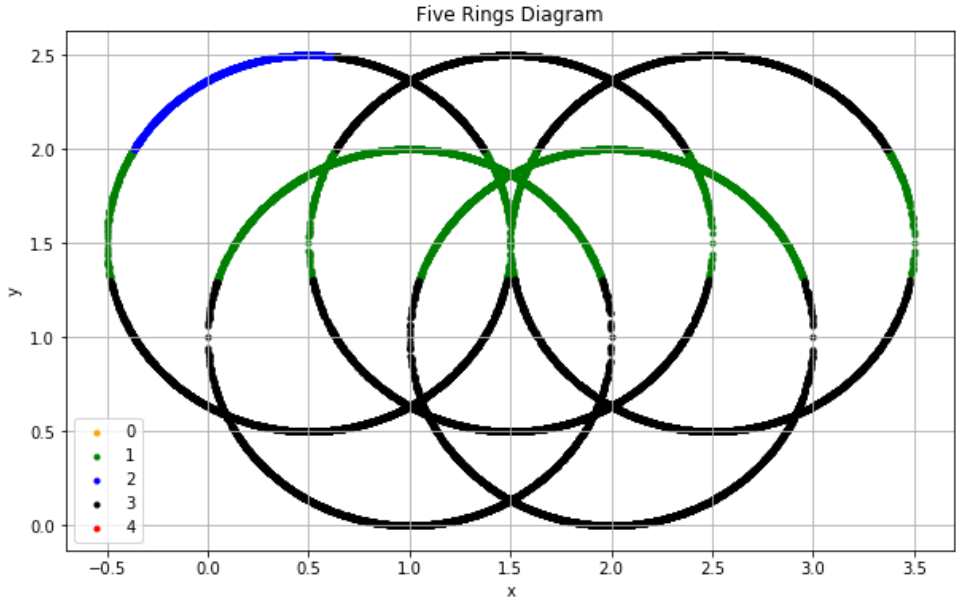
For 300 iterations:



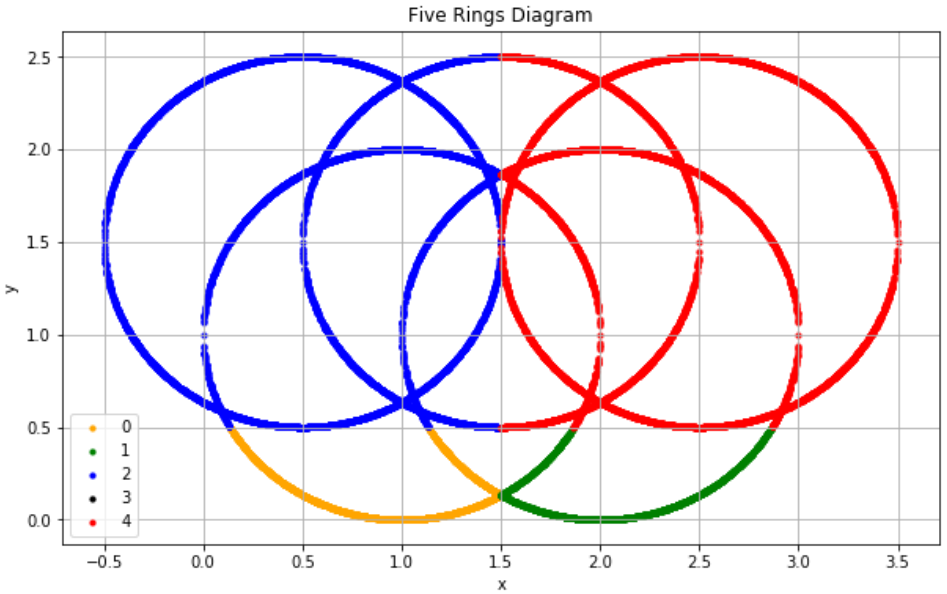
For 400 iterations:



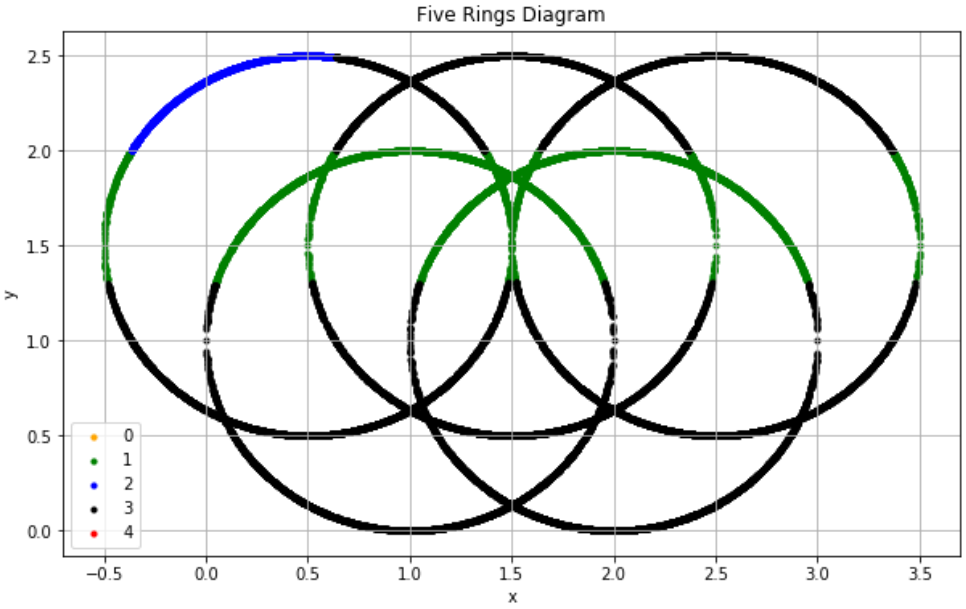
For 500 iterations:



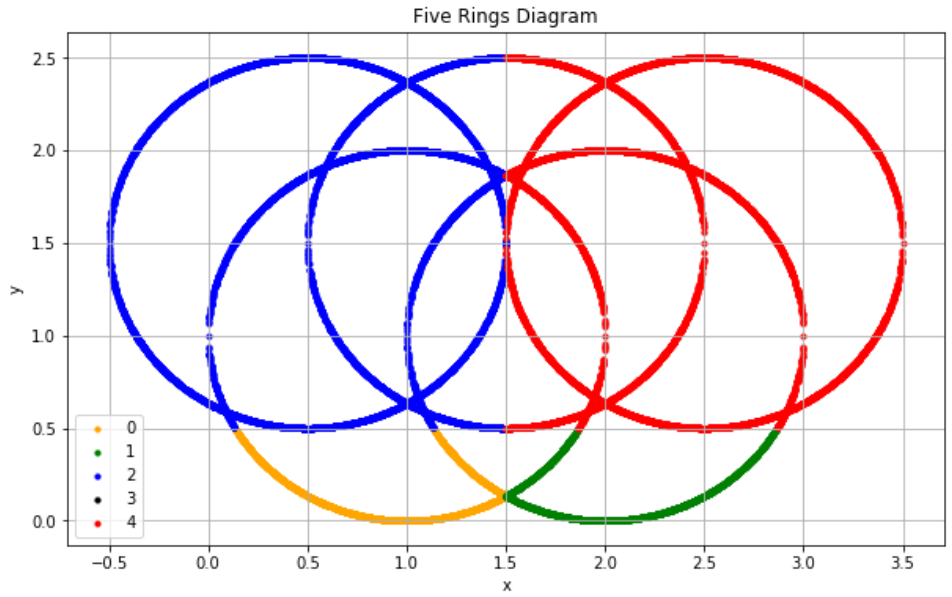
For 600 iterations:



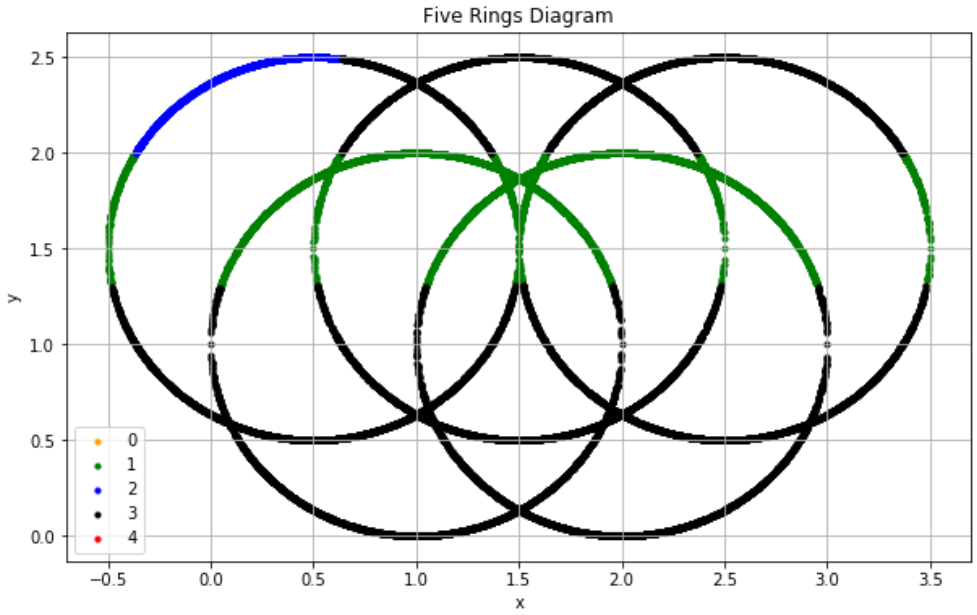
For 700 iterations:



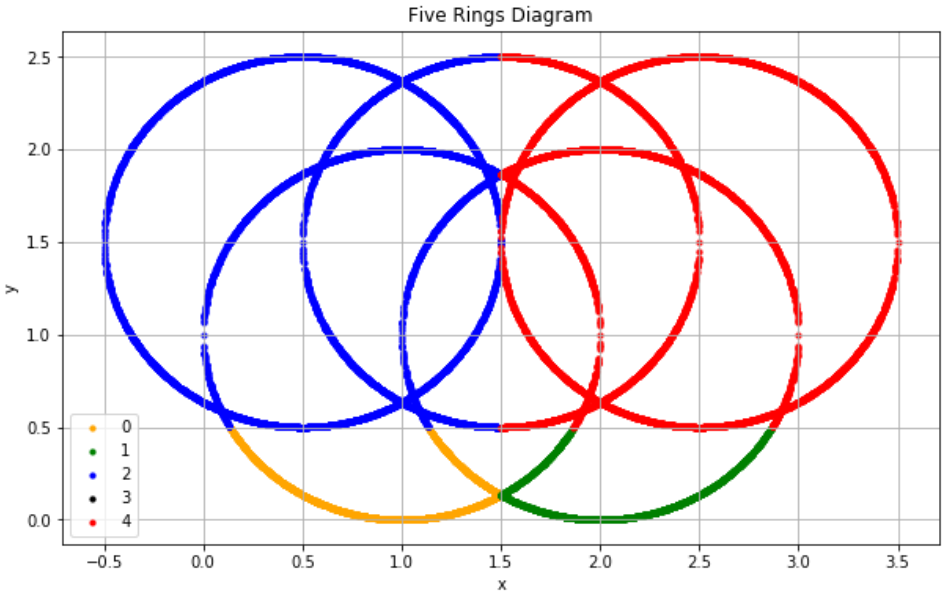
For 800 iterations:



For 900 iterations:



For 1000 iterations:



f) (10 points). Compare the results between the boosting results and the non-boosting results. Briefly comment on the comparison.

By observing boosting and non-boosting results we can say that our misclassification and RASE is oscillating between the two values after every iteration but at the end we get an increased accuracy i.e. lesser misclassification rate but hardly any improvement in RASE as compared to the non- boosting model, so we can say adaptive boosting can come in handy for classification Tree model. Also, the maximum depth of the tree is 2 so we are not getting good boosting results. By definition, the adaptive boosting tends to combine weighted averages of many weak learners to create a highly accurate prediction of the given data.