CSC 177-02 Data Warehousing and Data Mining

Spring 2020

SACRAMENTO
STATE

Assignment 2: Linear Regression and Classification Tree

Team: Data Pirates

Khushali Upadhyay

Surat Rawat

Yesha Shah

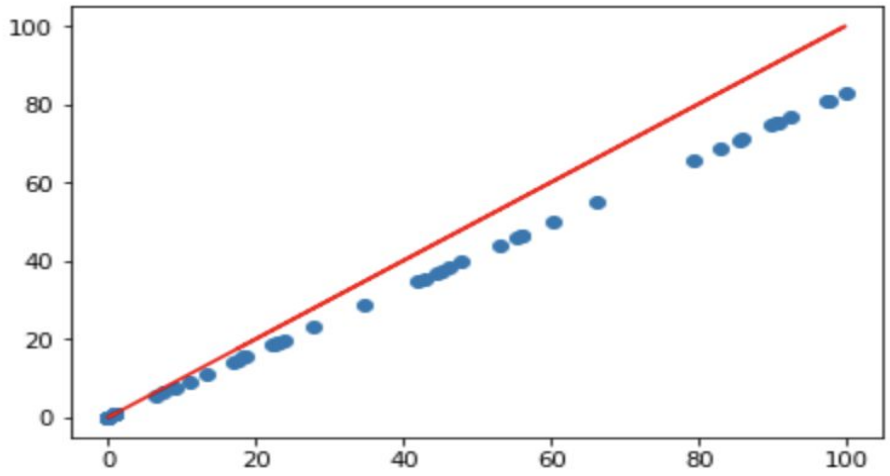
To

Prof. Jagannadha Chidella

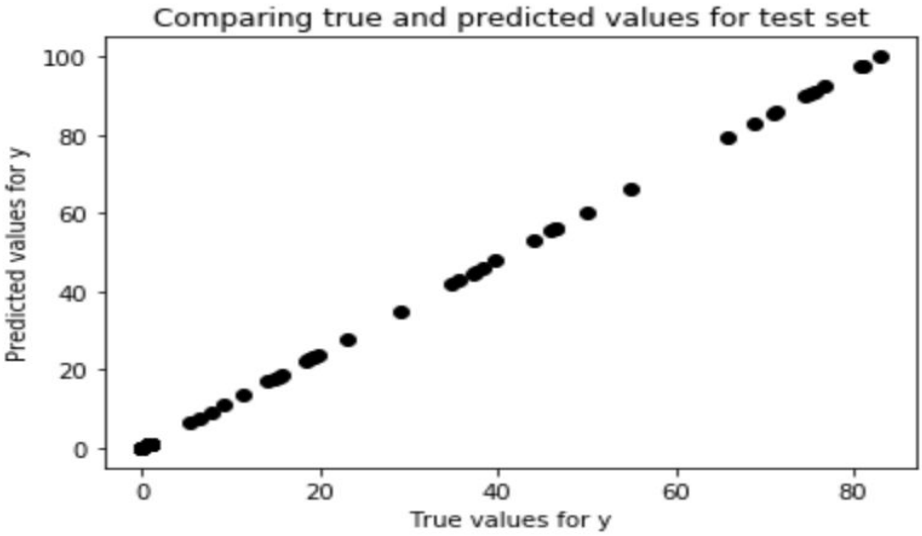
PART A: Regression

(i)The dataset that we have used from our previous assignment is 'sudeste.csv' and we have used our training and test data we got after splitting our dataset.Now we have performed 'Simple Linear Regression' and 'Multiple Linear Regression' on both the pair of training and test data.

(a) In Simple linear Regression, I have used two variables i.e x as 'smax' and y as 'smin' attribute of data.Then to get most accurate regression results and we have performed Normalisation process to above subset of data.Now we split it into variables 'x' and 'y'. We have also added data instances to ensure both interpolation and extrapolation of results.After applying Regression analysis we get a results as show below



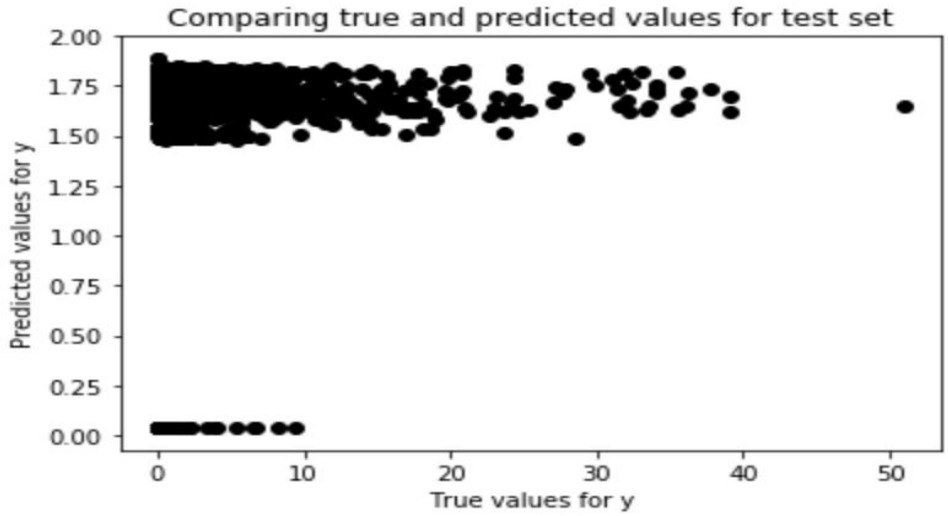

```
Root mean squared error = 1.1386
R-squared = 0.9572
```

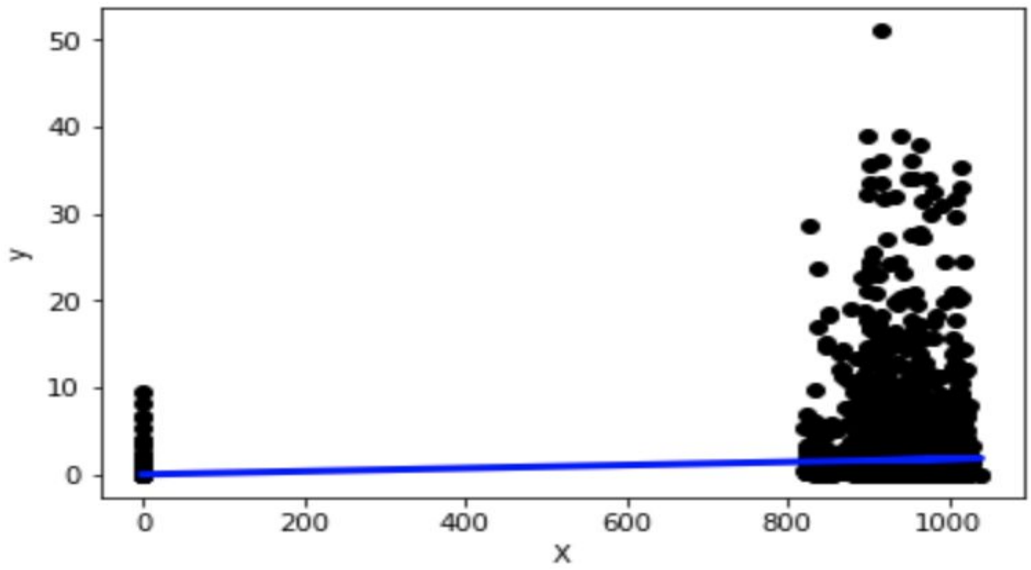


As we can say root mean squared error is 1.1386 and R-squared is 0.9572.Also notice how ‘Test’ data deviates from the ‘Trained’ data trend line.It is very close to actual or predicted value.Similarly in the second you get plotted true value and predicted value of y.

(b) In Multiple linear Regression, we used 3 variables i.e 'smax','smin','prcp' attribute of data.Then to get most accurate regression results and we have performed Normalisation process to above subset of data.Now we split it into variables ‘x’ and ‘y’.

We have also added data instances to ensure both interpolation and extrapolation of results.After applying Regression analysis we get results as shown below.





As we can see, the trend line ‘blue’ performs best for data values of x between 800 and 1000 and small values of y, as the majority of test data lies at that range.As we see data plotted on both sides of line.

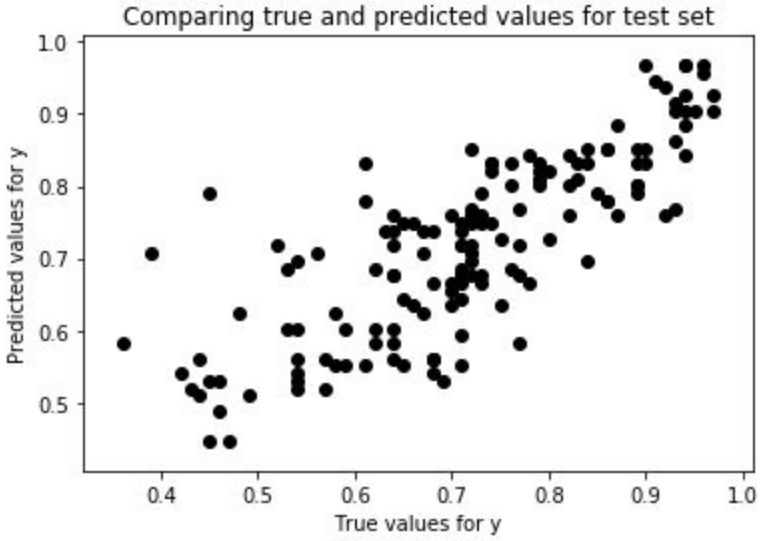
PART B: Regression and Classification

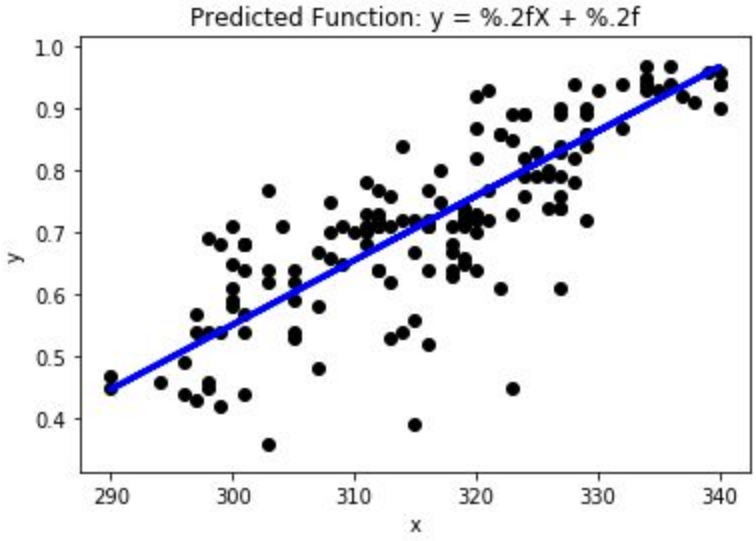
Regression:-

(ii)The dataset that we have used is 'Admission_Predict_Ver1.1_small_data_set_for_Linear_Regression.csv' and we have used our training and test data we got after splitting our dataset. Now we have performed 'Simple Linear Regression' and 'Multiple Regression' on one pair of training and test data.

(c) In Simple linear Regression, we have used one variable x i.e "GRE Score" to predict y i.e "Chance Of Admit". Now we split it into variables 'x' and 'y'.
After applying Regression analysis we get the results as shown below.

```
Root mean squared error = 0.0871
R-squared = 0.6357
```

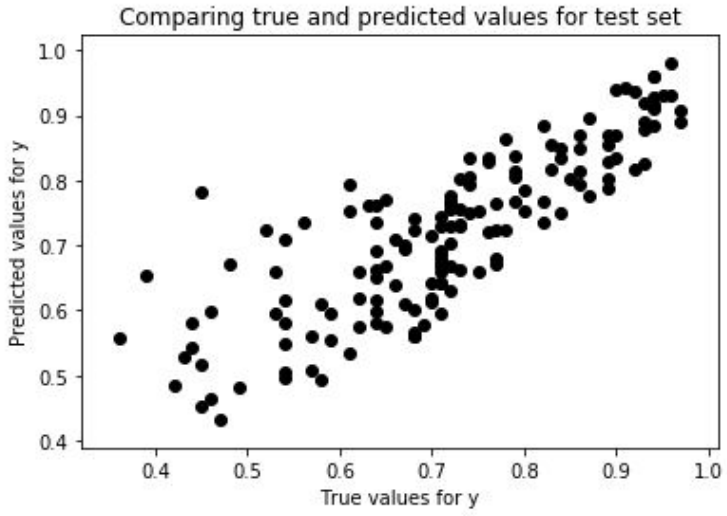


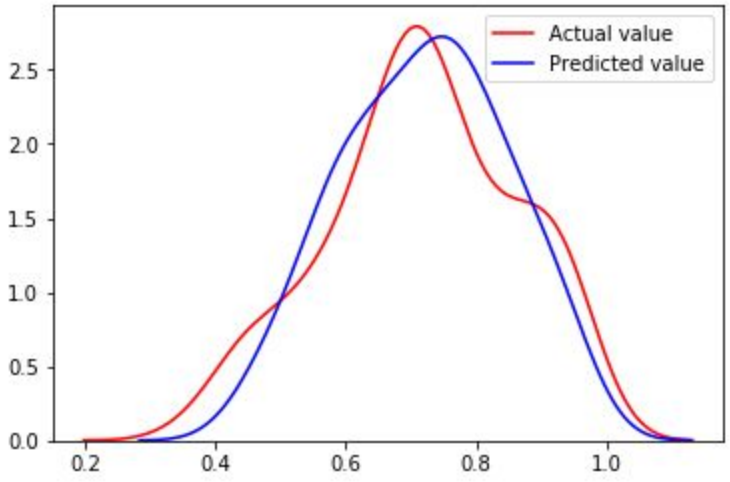


As shown above, in the 1 st figure the root mean squared error is 0.0871 and R-squared is 0.6357. You get a plot of true values and predicted values of y. In the 2nd figure for all values of x, it is predicting correct values of y.

(d) In Multiple linear Regression, we used 2 variables i.e “GRE Score” and “SOP” attributes of the dataset to predict “Chance of Admit”. Now we split it into variables ‘x’ and ‘y’. . After applying Regression analysis we get results as shown below.

```
Root mean squared error = 0.0794
R-squared = 0.6976
```

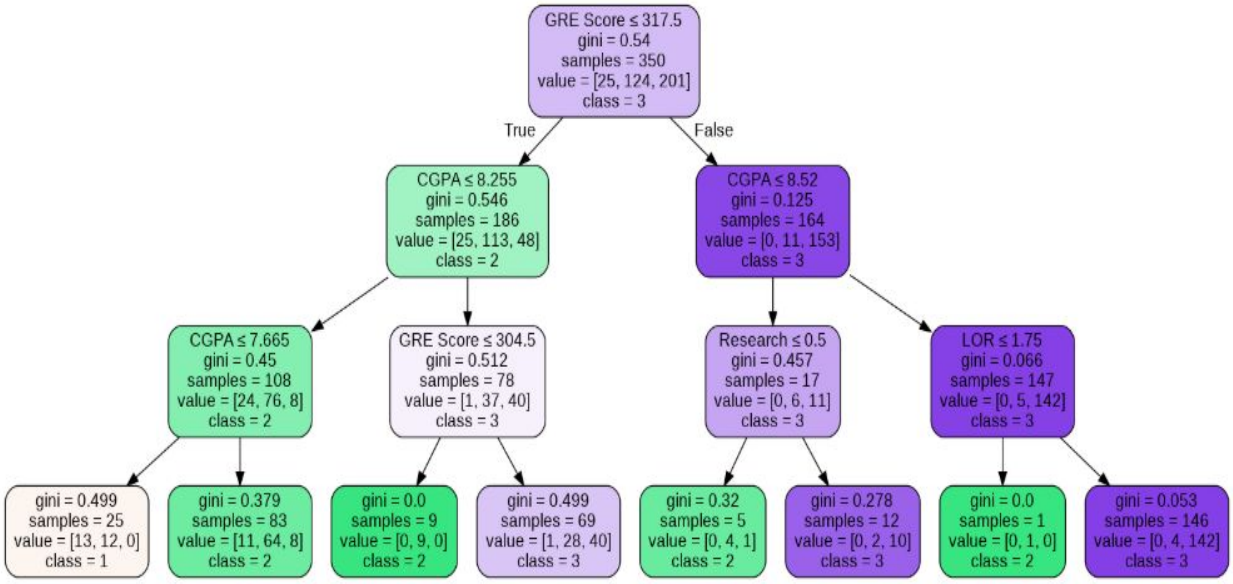




As shown above, in the 1st figure the root mean squared error is 0.0794 and R-squared is 0.6976. We get a plot of true values and predicted values of y. In the 2nd figure, we get a plot of the actual value and predicted value for "Chance of Admit."

Classification:

The dataset that we have used for classification is again 'Admission_Predict_Ver1.1_small_data_set_for_Linear_Regression.csv' and have classified the column "Chance of Admit" into 3 different classes. We used bins for creating different categories for "Chance of Admit". 3 bin ranges and bin names were defined to create 3 classes.GRE Score, TOEFL Score, SOP, LOR, CGPA, Research and University Ratings were defined in x and Chance of Admit as y. We split the dataset into the ratio of 70:30 and a decision tree classifier with maximum depth of 3 was created. The classifier was trained using xTrain_clf and was tested using xTest_clf. The figure for the decision tree is as follows:-



The following are some of the decision rules that we discovered:-

1. If CGPA <=8.255 and GRE Score <= 304.5 then the class is 2 which is medium possibility.
2. If CGPA <=8.52 and LOR > 1.75 then the class is 3 which is high possibility.
3. If CGPA <=8.52 and LOR <= 1.75 then the class is 2 which is medium possibility.
4. If CGPA <= 7.665 then the class is 1 which is low possibility.

We got an accuracy of 75 % for the classification.

PART C: Classification

(i) Solution

1. For the following data set, apply ID3 separately, and show all steps of derivation (computation, reasoning, developing / final decision trees, and rules).

	color	shape	size	class
1	red	square	big	+
2	blue	square	big	+
3	red	round	small	-
4	green	square	small	-
5	red	round	big	+
6	green	round	big	-

Calculating Entropy:

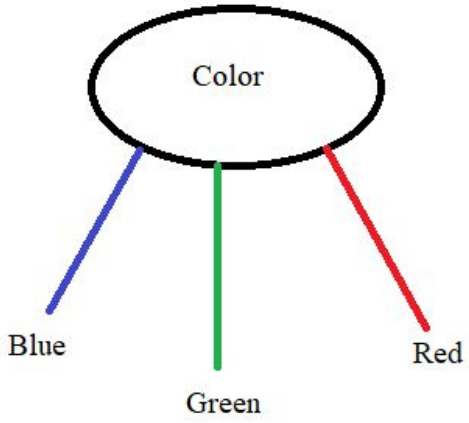
$$Entropy(t) = -\sum_{j} \mathrm{p}(j|t) \log_2 \mathrm{p}(j|t)$$

1. Calculating initial entropy:

   Probability of + = p(+) = 3/6 = 0.5
   Probability of - = p(-) = 3/6 = 0.5
   Entropy(class) = - 0.5 * log(0.5) – 0.5 * log(0.5)
   = - 2 * 0.5 * log(0.5)
   = -log(0.5) = - (-1)
   = 1

2. Every feature will calculate entropy and information gain i.e, Gain(class, color), Gain(class, shape), and Gain(class, size).

As per the calculation provided in Entropy_ID3_Excercise.pdf we know that out of color, shape, and size, color having the highest Gain it provides with maximum information. So, feature colo will be the root node of the tree as follows:



- Exploring dataset only in terms of attribute color:
    - We can see that if the color **blue** is chosen then, the decision will always be '+'.
    - If the color **green** is chosen then, the decision will always be '-'.
    - For the color **red**, there are 3 instances.
        - Probability for '+' would be ⅔%.
        - Probability for '-' would be ⅓%.

Calculating Gain(class, shape):

Probability of + = p(+) = 2/3 = 0.66
Probability of - = p(-) = 1/3 = 0.33
Entropy(shape) = - 0.33 * log2(0.33) – 0.66 * log2(0.66)
= 0.9234

Gain (class, shape) = Entropy(shape) - ∑ [p (class | shape) * Entropy (class | shape)]

Gain (class , shape) = Entropy(class) – p(class| shape =square) * Entropy(class | shape = square) – p(class| shape=round) * Entropy(class | shape = round)
Entropy (class | shape = round) = - p(-)* log p(-) – p(+) * log p (+)
= - (1/2) * log (1/2) – (1/2) * log(1/2)
= 0.9183
Entropy (class | shape = square) = - p(-)* log p(-) – p(+) * log p (+)
= - (0/1) * log (0/1) – (1/1) * log(1/1) ~ 0

Gain (class, shape) = 0.9234 - (2/3) * 0.9183- 0-0
= 0.3112

- **Gain(color=red|shape) = 0.3112**

Calculating Gain(class, size):

Gain(class, size) = Entropy(size) - ∑ [p(class | size) * Entropy(class | size)]

Gain(class , size) = Entropy(class) – p(class| size =big) * Entropy(class | size =big) – p(class|shape=size) * Entropy(class | size = small)

Entropy (class | size = big) = - p(-)* log2 p(-) – p(+) * log2 p (+)
= - (0) * log2 (0) – (2/2) * log2 (2/2)
= 0
Entropy (class | size = small) = - p(-)* log2 p(-) – p(+) * log2 p (+)
= - (1/1) * log2 (1/1) – (0) * log2 (0)
= 0
Gain (class, size) = 0.9234 - (0) * 0- 0*0 = 0

- **Gain(color=red|size) = 0.9234**

For, **color** = **Red**

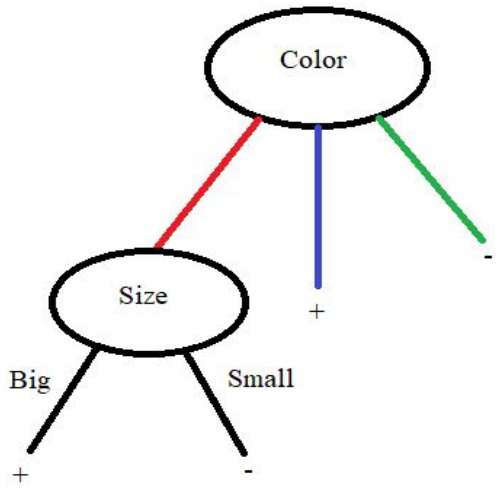
Now, if the decision is **size = big**.

- It produces the highest score if the **color = red.**
- And the **class** will always be '+'.

If the decision is **size = small.**

- The **class** chosen will always be '-'.

The final decision tree is:



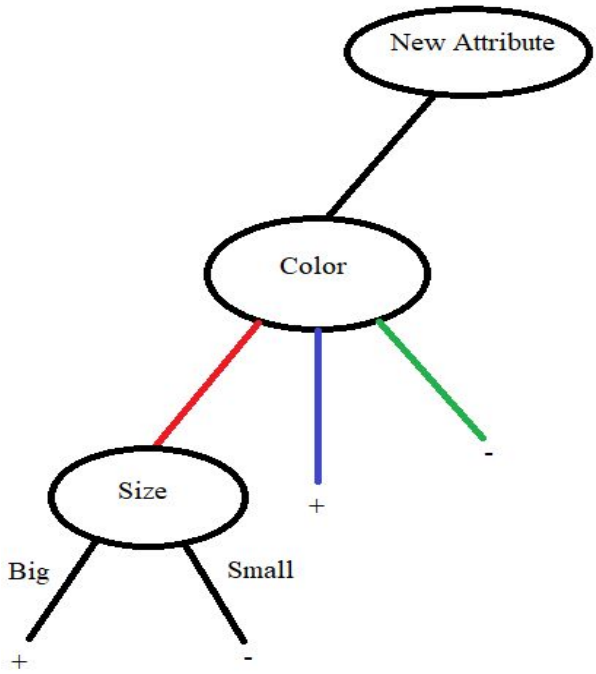
(ii) Decision tree with variations:

what impact may happen to your created tree, if you later add a new missing attribute after creating the tree?

Ans: Adding a new attribute. There are three possible cases in this scenario.

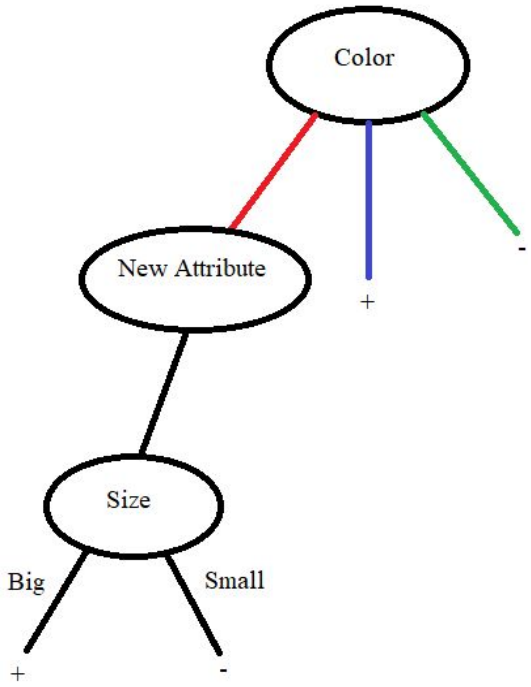
Case 1: The new attribute's Gain is > Color's Gain.

- The new attribute would become the root of the tree. One possible example is as follows:



Case 2: New Attribute's Gain is less than color's gain and greater than size's gain

- The newly added attribute can fit between color and size. Using that new attribute we could derive the classification either using size or without using the size. Following is one possible solution:



Case 3: The New Attribute's Gain is less than size and shape.

- If this is the case, then the decision tree would be the same as the original tree created because the new attribute's Gain would be least of all the other attributes.

What if a data scientist provided his or her results with high confidence, by missing this attribute altogether?

Ans: If the data scientist has provided results with missing values, and if it does not create any catastrophic outcome then it should be okay. If it does create an outcome that changes the whole idea of the project then it should result in a lower confidence level. This would be a very good learning experience.

What if his or her results are used for decision making on how many million more shirts to produce for the next year?

Ans: Based on the intensity of those results, their decision making could generate good or bad results. If missing the new attribute changes nothing, then the intention of predicting millions of

more shirts would result in high productivity. In contrast, they believed that missing the new attribute changes nothing but it reduces the sell-by predicting false decision.

Do you think the data scientist surprises the manager and CEO in case he or she discovers the new attribute and its influence in getting more reliable results valuable to the company?
Ans: If the company makes tons of money out of the result provided by the data scientist, then the manager and CEO would definitely be amazed and delighted. If not, then it definitely would surprise them with shock!!

Task Division:

- Part A: Combined Efforts
- Part B: Combined Efforts
- Part C: Combined Efforts
- Report: Combined Efforts

Things Learned During This Project

- Learning algorithms is always exciting but actually understanding an algorithm such as ID3. Got to learn how each decision is made in a tree training with manual calculations which was really interesting.
- Also, we learned how a new attribute in a decision tree makes a difference.
- Normalization is important as it allows numbers to be put in a standard form so that two values can be easily compared.

Challenges Encountered

- Not exactly a challenge but in Part C, thinking how a new attribute would make a difference was a brain teaser.