Name: Vipin Singh
Email: vipsingh@iu.edu

Applied Data Science

Linear Regression, K-Means, KNN and Decision trees

Assignment 8

1. Part 1

Linear Regression

- How can we determine if a given model is overfitting or under fitting the data? (Hint: Explain with reference to bias and variance)

A model is under fitting the training data when it performs poorly on the training data. This is because the model is unable to capture the relationship between the input examples and the target values. A model is overfitting the data when it performs well on the training data but does not perform well on the test data. This happens because the model is memorizing the data it has been trained on and is unable to generalize to unseen test data.

If a model is under fitting or overfitting we can easily find whether the problem is due to bias or variance. The training data error and cross validation errors are calculated and compared. If the value of training data error and cross validation error is higher and at the almost same level, then it is a high bias problem. The high bias happens in a model where the function is of low polynomial degree. If the value of training data error is lower and cross validation error is higher, then this is the scenario of model overfitting and the function used in this type of model is of higher polynomial degree.

As per the Figure 1 below, the left most end of the two lines shows the high bias (underfit) and the right most end shows the high variance (overfit)

Name: Vipin Singh
Email: vipsingh@iu.edu

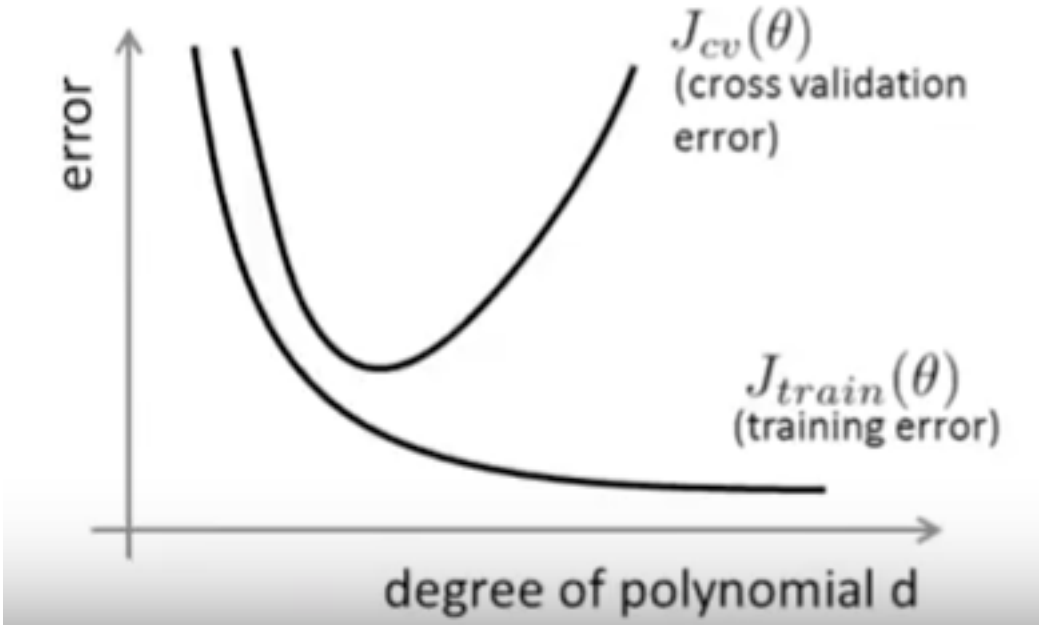


Figure 1: Reference - https://www.youtube.com/watch?v=6MBUIUjRri0

K Means

- Cluster the following set of points in @-Dimensional space into 2 groups using the K-Means algorithm: {(2, 5); (1, 5); (22, 55); (42, 12); (15, 16)}. Use the given distance metric to calculate the distance between the centroid and the points: d((x1; y1); (x2; y2)) = sqrt[sqr(x1 - x2) + sqr(y1 - y2))]

.

Data Set = {(2, 5); (1, 5); (22, 55); (42, 12); (15, 16)}

We need to create 2 groups, hence the value of K = 2. Let's assume the first two centroids as follows. We can choose any two points as the centroids to begin the algorithm.

Iteration 1

C1 = (2,5)
C2 = (22,55)

Using the above two centroids, let's build the two groups by using the distance between all the points in the data sets and the above centroids. The distance matrix is created as follows.

Name: Vipin Singh
Email: vipsingh@iu.edu

Distance Matrix

	(2,5)	(1,5)	(22,55)	(42,12)	(15,16)
(2,5)	0	1	53.8	40.6	17.02
(22,55)	53.8	23.25	0	47.4	39.6

If we observe the above distance matrix the points (1,5), (42,12) and (15,16) are near to C1. Hence (1,5), (42,12) and (15,16) becomes the member of G1.

In G2 only the assumed centroid C2 will remain.

G1 = {(2,5), (1,5), (42,12), (15,16)}
G2 = {(22,55)}

Iteration 2

New Centroid of Group G1 and G2 are –

C1 = ((2 + 1 + 42 + 15)/4, (5 + 5 +12+ 16)/4) = (15,9.5)
C2 = (22,55)

Calculating the distance of all the points from the new centroids as below in the distance matrix, the new groups G1 and G2 are created.

	(2,5)	(1,5)	(22,55)	(42,12)	(15,16)
(15,9.5)	13.75	14.7	46	27.11	6.5
(22,55)	53.8	23.25	0	47.4	39.6

The points (2,5), (1,5), (42,12), (15,16) are near to centroid C1. Hence these points form the group G1.

G1 = {(2,5), (1,5), (42,12), (15,16)}
G2 = {22,55}

If we observe closely, the groups G1, G2 in this iteration and previous iteration are same. Hence the new centroids will also be same. It means we need to stop algorithm to do the further iterations and the final groups are as shown below.

G1 = {(2,5), (1,5), (42,12), (15,16)}
G2 = {22,55}

Name: Vipin Singh
Email: vipsingh@iu.edu

.

- Does K-Means always converge to the Global minima ? Why/ why not?

  K-Means does not converge to Global minima, instead it converges to local minima. The reason is that each group has its own local minima and when multiple iterations are run, the data set tends to converge to the local minima. The error seems to go away if the algorithm is run for large number of iterations.

- Explain two drawbacks of using the K-Means clustering algorithm.

  The two drawbacks of using the K-Means algorithm are:

  1. The Value of K is assumed for the dataset which cannot possibly be right. There can be more clusters in the data than assumed.
  2. The algorithm can cluster the non-clustered data. If we run the K-means on uniform data you will still get clusters, it doesn't tell that the data does not have clusters.

K Nearest Neighbors

- Given data contains pairs of points and their corresponding class. {(2, 5),1}; {(1, 5),1}; {(48, 35),2}; {(42, 12),2}. Using KNN algorithm determine the class of the points (15, 16); (30,40); (0,0). Report the results for k = 1,2,3.

  Showing the points and the classes in the tabular form below. The Distance column is populated with the distance from point (15,16) with each point in the table below.

Name: Vipin Singh
Email: vipsingh@iu.edu

In the Table 1 below, the Euclidian distance of point (15,16) from all the given points is calculated and stored. The Rank is calculated based on the distance, if the distance is less, the rank is small. It increases with the increasing distance.

- ✓ When K = 1, the nearest neighbor of point (15,16) is (2,5) as per the Rank. Hence the class of point (15,16) will be the same as of point (2,5) which is 1.

- ✓ When K = 2, the nearest two neighbors of point (15,16) are (2,5) and (1,5) as per the ranks. Hence the class of point (15,16) in K= 2 scenario, will be the same as of points (2,5) and (1,5) which is 1.

- ✓ When K = 3, the nearest three neighbors of point (15,16) are (2,5), (1,5) and (42, 12) as per the ranks. The class of two neighbors is 1 and one neighbor is 2. Hence based on the 2/3 probability the class of point (15, 16) will be 1

Points	Class	Distance of (15,16) from points	Rank
(2,5)	1	17.02	1
(1,5)	1	17.80	2
(48,35)	2	38.07	4
(42,12)	2	27.29	3

Table 1

Please see the predicted class of point (15, 16) based on the KNN for K = 1,2,3

	Point	Class
K = 1	(15,16)	1
K = 2	(15,16)	1
K = 3	(15,16)	1

Name: Vipin Singh
Email: vipsingh@iu.edu

In the Table 2 below, the Euclidian distance of point (30, 40) from all the given points is calculated and displayed. The Rank is calculated based on the distance, if the distance is less, the rank is small. It increases with the increasing distance.

- ✓ When K = 1, the nearest neighbor of point (30,40) is (48, 35) as per the Rank. Hence the class of point (30, 40) will be the same as of point (48, 35) which is 2.

- ✓ When K = 2, the nearest two neighbors of point (30, 40) are (48,35) and (42, 12) as per the ranks. Hence the class of point (30,40) in K= 2 scenario, will be the same as of points (48,35) and (42,12) which is 2.

- ✓ When K = 3, the nearest three neighbors of point (30,40) are (48, 35), (42, 12) and (2, 5) as per the ranks. Hence the class of point (30,40) in K= 3 scenario, will be the same as of points (48,35), (42,12) and (2,5). The class of two neighbors is 2 and one neighbor is 1. Hence based on the 2/3 probability the class of point (30, 40) will be 2

Points	Class	Distance of (30,40) from points	Rank
(2,5)	1	44.82	3
(1,5)	1	45.45	4
(48,35)	2	18.6	1
(42,12)	2	30.46	2

Table 2

Please see the predicted class of point (30, 40) based on the KNN for K = 1,2,3

	Point	Class
K = 1	(30, 40)	2
K = 2	(30, 40)	2
K = 3	(30, 40)	2

Name: Vipin Singh
Email: vipsingh@iu.edu

In the Table 3 below, the Euclidian distance of point (0, 0) from all the given points is calculated and displayed. The Rank is calculated based on the distance, if the distance is less, the rank is small. It increases with the increasing distance.

- ✓ When K = 1, the nearest neighbor of point (0, 0) is (1, 5) as per the Rank. Hence the class of point (0, 0) will be the same as of point (1, 5) which is 1.

- ✓ When K = 2, the nearest two neighbors of point (0, 0) are (1,5) and (2, 5) as per the ranks. Hence the class of point (0,0) in K= 2 scenario, will be the same as of points (1,5) and (2,5) which is 1.

- ✓ When K = 3, the nearest three neighbors of point (0,0) are (1, 5), (2, 5) and (42, 12) as per the ranks. Hence the class of point (0,0) in K= 3 scenario, will be the same as of points (1,5), (2,5) and (42,12). The class of two neighbors is 1 and one neighbor is 1. Hence based on the 2/3 probability the class of point (0,0) will be 1

Points	Class	Distance of (0,0) from points	Rank
(2,5)	1	5.38	2
(1,5)	1	5.09	1
(48,35)	2	59.40	4
(42,12)	2	43.68	3

Table 3

Please see the predicted class of point (0, 0) based on the KNN for K = 1,2,3

	Point	Class
K = 1	(0, 0)	1
K = 2	(0, 0)	1
K = 3	(0, 0)	1

Name: Vipin Singh
Email: vipsingh@iu.edu

Decision Trees

- Define Entropy and Information gain.

  Entropy: In a decision tree algorithm, the data set can be split on various attributes to decide tree. Since there are lot of attributes in the data set, so based on different attributes split at each level, many decision trees are possible. To determine the best attribute to split the data set and create an efficient decision tree, entropy at each split is calculated to determine if the split on the respective attribute is better than the split on other attributes.

  In other words, we can say that Entropy is a measure of impurity or disorder for example if the split results in 50% - 50% then the set is impure and if the split result is 100% - 0% or 0% - 100% then the set is pure.

  Let S be a set, let p be the fraction of positive training examples and q be the fraction of negative training examples.

  Entropy (S) = $-p \log_2 (p) - q \log_2(q)$

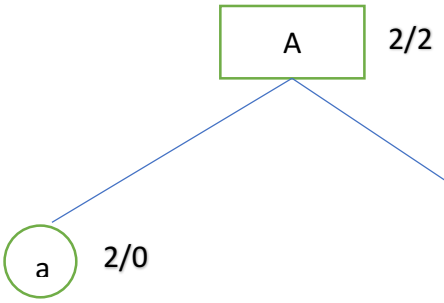
  Information gain: The entropy changes when we use a node in a decision tree to partition the training instances in to smaller subsets. Information gain is the measure of this change in the entropy.

- For the given data calculate the entropy for splitting based on the features A, B and C. Label gives the class of the input. (You DO NOT have to split based on information gain. It is enough to calculate the entropy for splitting the root node based on each of the 3 variables)

A	B	C	Label
a	a	a	1
b	b	a	2
a	a	b	1
b	b	a	2

Name: Vipin Singh
Email: vipsingh@iu.edu

Let's consider the Split by A



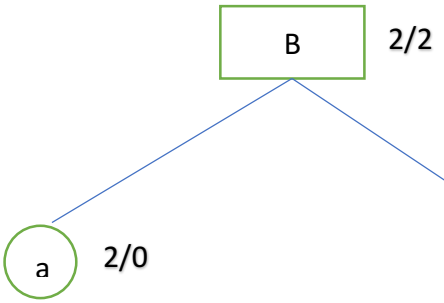
B	C	Label
a	a	1
a	b	1

B	C	Label
b	a	2
b	a	2

As per the above tree, the Entropy at each node are as follows at each split.

Entropy (A) = 1, The data set is divided in 50% with Label 1 and 50% with Label 2
Entropy (a) = 0, The Split is a pure set with 2 rows of Label 1 and 0 rows of label 2
Entropy (b) = 0, The Split is a pure set with 0 rows of Label 1 and 2 rows of label 2

Let's consider the Split by B



A	C	Label
a	a	1
a	b	1

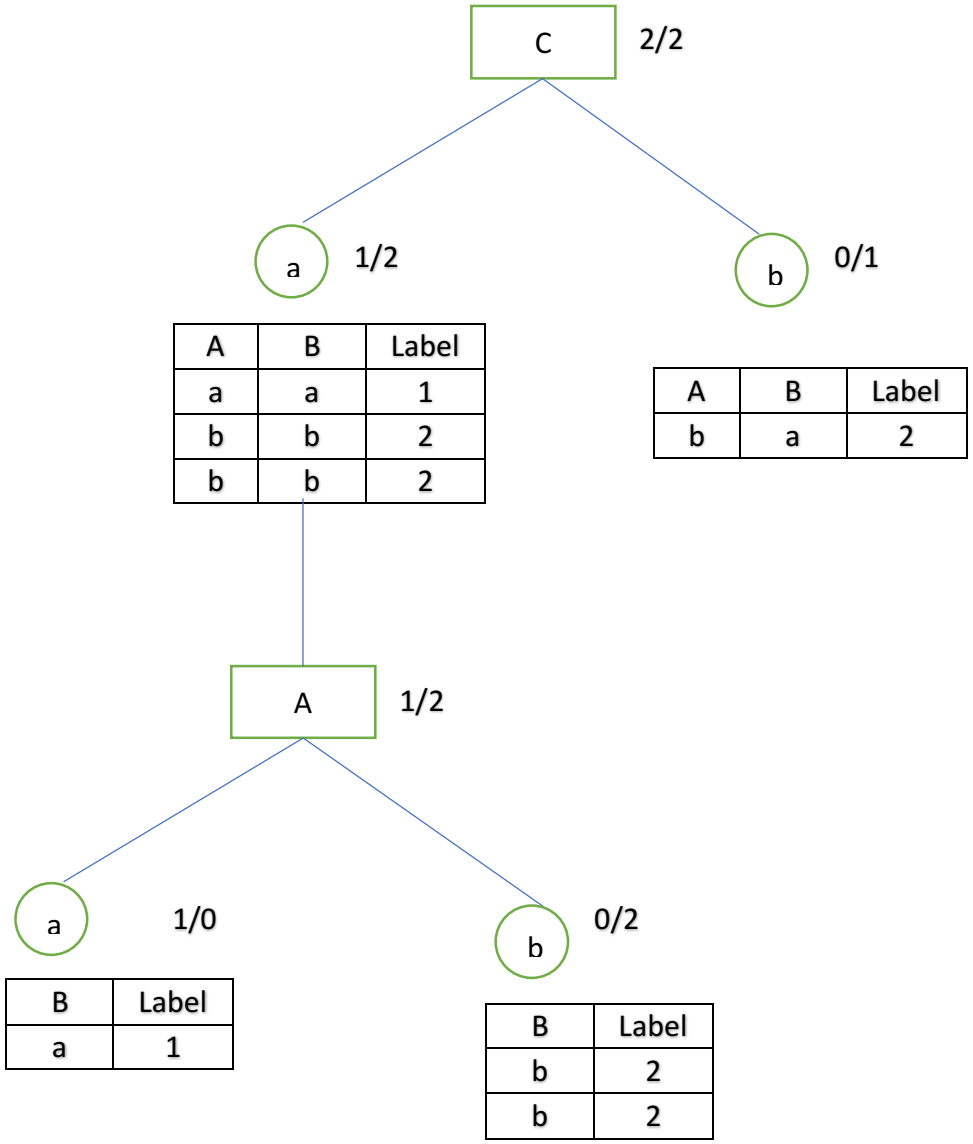
A	C	Label
b	a	2
b	a	2

As per the above tree, the Entropy at each node are as follows at each split.

Entropy (B) = 1, The data set is divided in 50% with Label 1 and 50% with Label 2

Name: Vipin Singh
Email: vipsingh@iu.edu

Entropy (a) = 0, The Split is a pure set with 2 rows of Label 1 and 0 rows of label 2
Entropy (b) = 0, The Split is a pure set with 0 rows of Label 1 and 2 rows of label 2
Let's consider the Split by C



A	B	Label
a	a	1
b	b	2
b	b	2

A	B	Label
b	a	2

B	Label
a	1

B	Label
b	2
b	2

As per the above tree, the Entropy at each node are as follows at each split.

Entropy (C) = 1, The data set has 50% with Label 1 and 50% with Label 2

P (Probability) at a = ¾, total rows are 4, at this path there are 3 rows out of 4. Hence 3/4
E(a) = P * Entropy (1/2) = P* (-1/3($\log_2$ (1/3)) – (2/3) ($\log_2$ (2/3)))
= ¾ (-0.3333 * log2(0.3333) – (0.6666) *($\log_2$(0.6666)))
= 0.75[(-0.3333 * -1.5851) – (0.6666) * (-0.5851)]

Name: Vipin Singh
Email: vipsingh@iu.edu

= 0.75 [0.5283 + 0.3896]
= 0.75*0.9179
= 0.6884
= 0.69

Entropy (a) = 0.69, The Split is divided in 1 row of Label 1 and 2 rows of Label 2, with a probability of ¾.
Entropy (A) = 0.69, same as parent a.
Entropy (a) = 0, The Split is a pure set with 1 rows of Label 1 and 0 rows of label 2
Entropy (b) = 0, The Split is a pure set with 0 rows of Label 1 and 2 rows of label 2

2. Part 2

Use each of the algorithms introduced in this module {regression, K-means, K nearest neighbors, and decision trees - to model the low birth rate dataset lbr-train.csv. For each algorithm, be sure to select which attribute is most relevant to predict (LOW or BWT) and which are relevant as features. You may wish to refer to your data exploration to assist in this. Apply each model to the low birth rate dataset lbr-test.csv and describe its performance. Compare the performance of each model and discuss.

Linear Regression

The Model is created on BWT ~ PTL + SMOKE + HT + AGE + RACE + LWT + UI
to predict the BWT (Birth Weight of Child in grams). The median difference in the predicted value and the test value of BWT is 473.6 grams.

This difference in the median will be compared below to other algorithms to evaluate the performance.

Name: Vipin Singh
Email: vipsingh@iu.edu

```
> setwd("~/Desktop/ADS/Assignment8")
> lbr_train_data <- read.csv("lbr-train.csv",header = TRUE, sep = ",")
> lbr_test_data <- read.csv("lbr-test.csv",header = TRUE, sep = ",")
> mylm <- lm(BWT ~ PTL + SMOKE + HT + AGE + RACE + LWT + UI , dat = lbr_train_data)
> summary(mylm)

Call:
lm(formula = BWT ~ PTL + SMOKE + HT + AGE + RACE + LWT + UI,
    data = lbr_train_data)

Residuals:
     Min       1Q   Median       3Q      Max
-1699.88  -430.56    20.14   497.66  1685.59

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 2901.3392   394.5752   7.353 1.37e-11 ***
PTL         -113.2267   115.2351  -0.983   0.3275
SMOKE       -297.1186   117.8601  -2.521   0.0128 *
HT          -608.6921   237.1512  -2.567   0.0113 *
AGE            0.1472    10.0460   0.015   0.9883
RACE        -147.4582    63.6321  -2.317   0.0219 *
LWT            4.4220     2.0322   2.176   0.0312 *
UI          -612.1365   150.4068  -4.070 7.75e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 637.1 on 143 degrees of freedom
Multiple R-squared:  0.2474,    Adjusted R-squared:  0.2106
F-statistic: 6.716 on 7 and 143 DF,  p-value: 6.98e-07

> new_BWT <- predict.lm(mylm, lbr_test_data)
> summary(abs(new_BWT - lbr_test_data$BWT))
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  17.07  134.50  473.60  567.20  906.00 1708.00
```

K-Means

As per the below screenshot, I loaded the entire data of low birth rate to the lbr_full_data. I created the two clusters using K-means for columns AGE and BWT. The cluster is then stored in the data frame and displayed.

Cluster

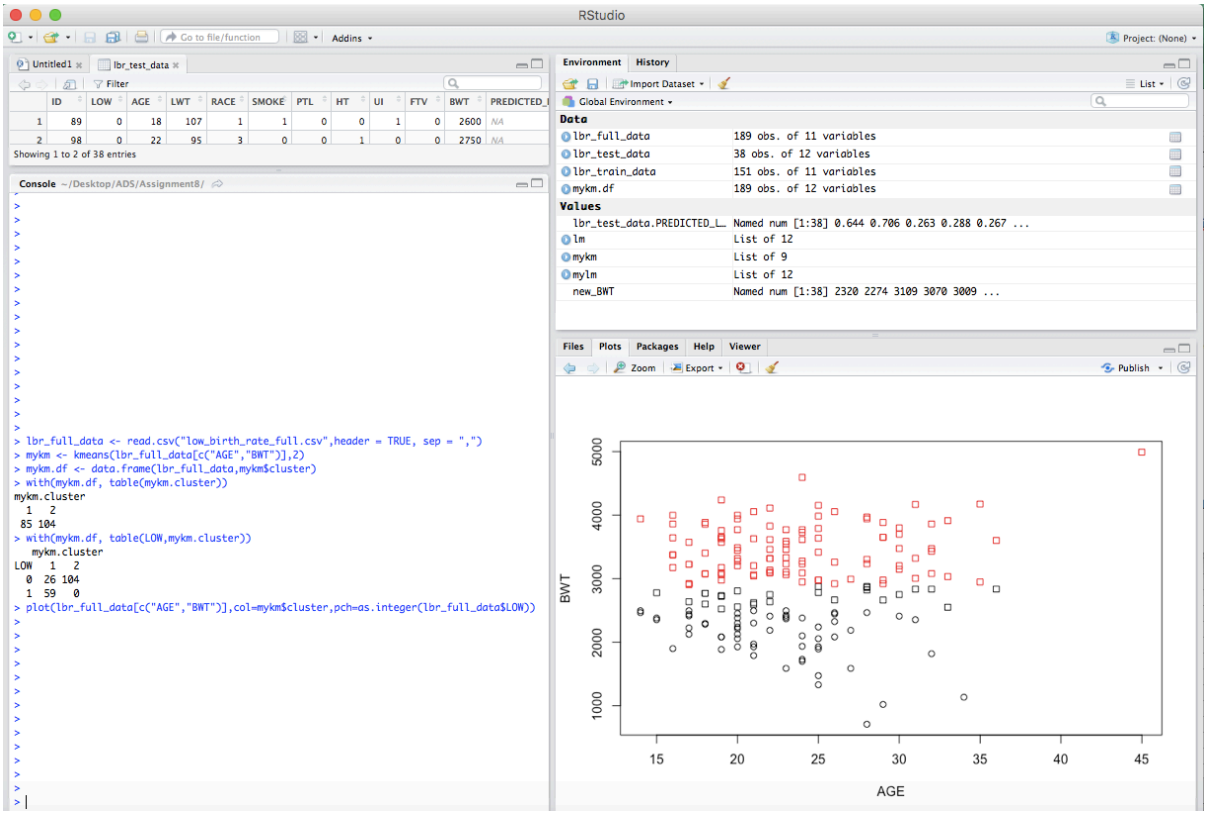
The data set has 189 data points with 130 as normal birth weight and 59 as low birth weight. The clusters are created as follows.

- Cluster1 – 85 Data Points
- Cluster2 – 104 Data Points

Observation

I Observed the value of LOW (Birth weight normal or low), 0 means normal and 1 means low. In our clusters, Normal birthweights are divided in to two clusters – cluster1 having 26 and cluster2 having 104. The low birthweights are in cluster1 only. It means 26 rows out of 130 went to wrong cluster, which means there is an error of 20% in the overall clustering if we compare it with the LOW data.

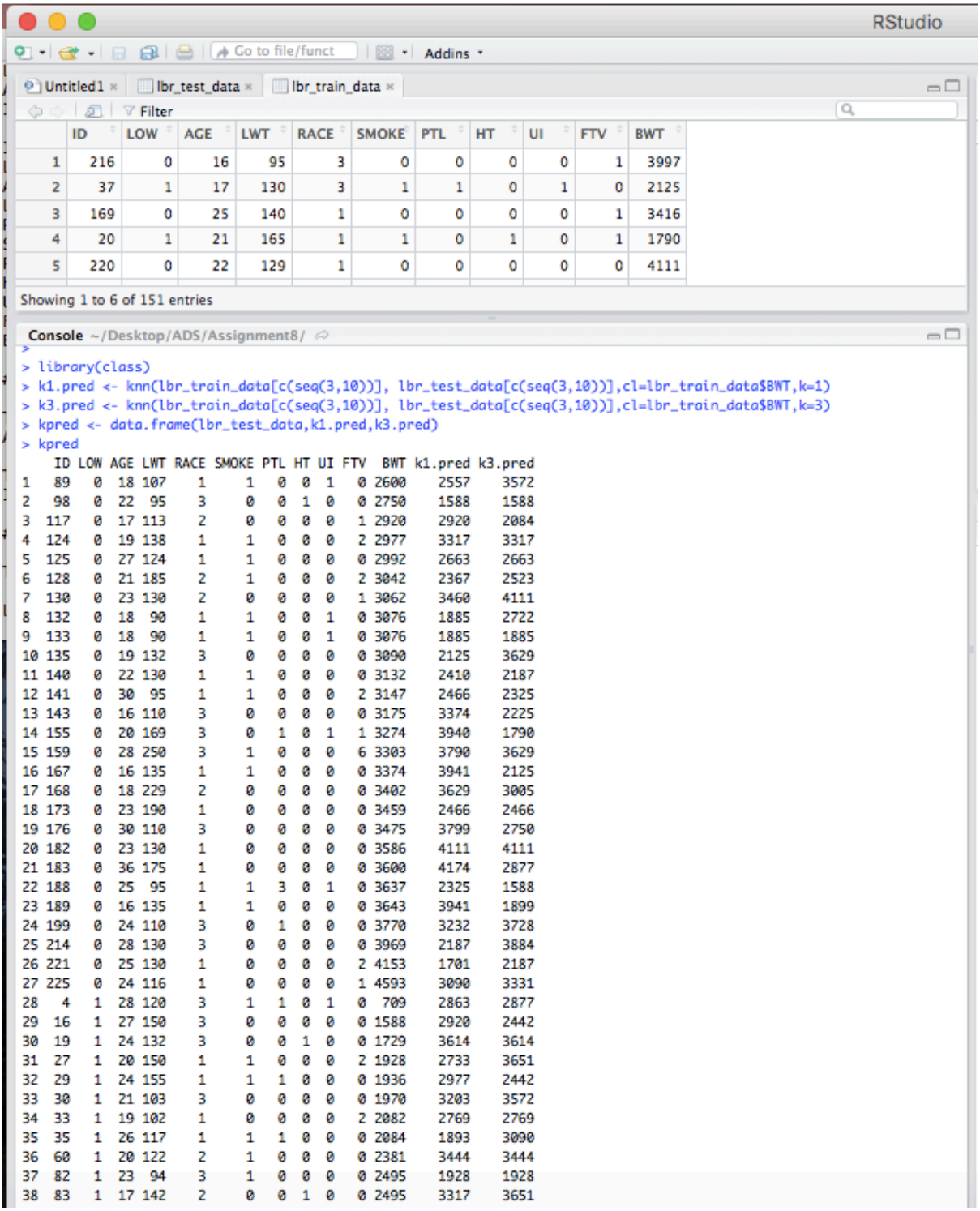
Name: Vipin Singh
Email: vipsingh@iu.edu



Name: Vipin Singh
Email: vipsingh@iu.edu

KNN

The below screenshot shows the predicted value of BWT on the test data set using the columns 3 to column 10 of the training and test data set. We are predicting the same column which we predicted in Linear regression. A comparison of the median error will be shown between KNN and Linear regression.



	ID	LOW	AGE	LWT	RACE	SMOKE	PTL	HT	UI	FTV	BWT
1	216	0	16	95	3	0	0	0	0	1	3997
2	37	1	17	130	3	1	1	0	1	0	2125
3	169	0	25	140	1	0	0	0	0	1	3416
4	20	1	21	165	1	1	0	1	0	1	1790
5	220	0	22	129	1	0	0	0	0	0	4111

	ID	LOW	AGE	LWT	RACE	SMOKE	PTL	HT	UI	FTV	BWT	k1.pred	k3.pred
1	89	0	18	107	1	1	0	0	1	0	2600	2557	3572
2	98	0	22	95	3	0	0	1	0	0	2750	1588	1588
3	117	0	17	113	2	0	0	0	0	1	2920	2920	2084
4	124	0	19	138	1	1	0	0	0	2	2977	3317	3317
5	125	0	27	124	1	1	0	0	0	0	2992	2663	2663
6	128	0	21	185	2	1	0	0	0	2	3042	2367	2523
7	130	0	23	130	2	0	0	0	0	1	3062	3460	4111
8	132	0	18	90	1	1	0	0	1	0	3076	1885	2722
9	133	0	18	90	1	1	0	0	1	0	3076	1885	1885
10	135	0	19	132	3	0	0	0	0	0	3090	2125	3629
11	140	0	22	130	1	1	0	0	0	0	3132	2410	2187
12	141	0	30	95	1	1	0	0	0	2	3147	2466	2325
13	143	0	16	110	3	0	0	0	0	0	3175	3374	2225
14	155	0	20	169	3	0	1	0	1	1	3274	3940	1790
15	159	0	28	250	3	1	0	0	0	6	3303	3790	3629
16	167	0	16	135	1	1	0	0	0	0	3374	3941	2125
17	168	0	18	229	2	0	0	0	0	0	3402	3629	3005
18	173	0	23	190	1	0	0	0	0	0	3459	2466	2466
19	176	0	30	110	3	0	0	0	0	0	3475	3799	2750
20	182	0	23	130	1	0	0	0	0	0	3586	4111	4111
21	183	0	36	175	1	0	0	0	0	0	3600	4174	2877
22	188	0	25	95	1	1	3	0	1	0	3637	2325	1588
23	189	0	16	135	1	1	0	0	0	0	3643	3941	1899
24	199	0	24	110	3	0	1	0	0	0	3770	3232	3728
25	214	0	28	130	3	0	0	0	0	0	3969	2187	3884
26	221	0	25	130	1	0	0	0	0	2	4153	1701	2187
27	225	0	24	116	1	0	0	0	0	1	4593	3090	3331
28	4	1	28	120	3	1	1	0	1	0	709	2863	2877
29	16	1	27	150	3	0	0	0	0	0	1588	2920	2442
30	19	1	24	132	3	0	0	1	0	0	1729	3614	3614
31	27	1	20	150	1	1	0	0	0	2	1928	2733	3651
32	29	1	24	155	1	1	1	0	0	0	1936	2977	2442
33	30	1	21	103	3	0	0	0	0	0	1970	3203	3572
34	33	1	19	102	1	0	0	0	0	2	2082	2769	2769
35	35	1	26	117	1	1	1	0	0	0	2084	1893	3090
36	60	1	20	122	2	1	0	0	0	0	2381	3444	3444
37	82	1	23	94	3	1	0	0	0	0	2495	1928	1928
38	83	1	17	142	2	0	0	1	0	0	2495	3317	3651

Name: Vipin Singh
Email: vipsingh@iu.edu

Summary for K1 and K3 KNN algorithm for predicted BWT. If we compare the K1 and K3, K3 seems to have less difference (3017) from the actual BWT value compared to K1. Hence K3 performs better than K1 in this data set.

```
> summary(abs(as.integer(k1.pred) - lbr_test_data$BWT))
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
    653    2435    3067    2882    3362    4527
> summary(abs(as.integer(k3.pred) - lbr_test_data$BWT))
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
    652    2424    3017    2886    3407    4515
```

Summary of Linear regression algorithm for predicted BWT

```
> summary(abs(new_BWT - lbr_test_data$BWT))
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  17.07  134.50  473.60  567.20  906.00 1708.00
```

The Linear regression shows better performance compared to KNN algorithm for this training and test data. The value 473.60 is far less than 3017 and 3067.

Decision Tree

Classification Tree

The Classification tree does not specifically predict any new value, it creates a path to traverse and identify the output for the test data based on the training data. Hence this classification tree answers the BWT prediction for test data with higher accuracy.

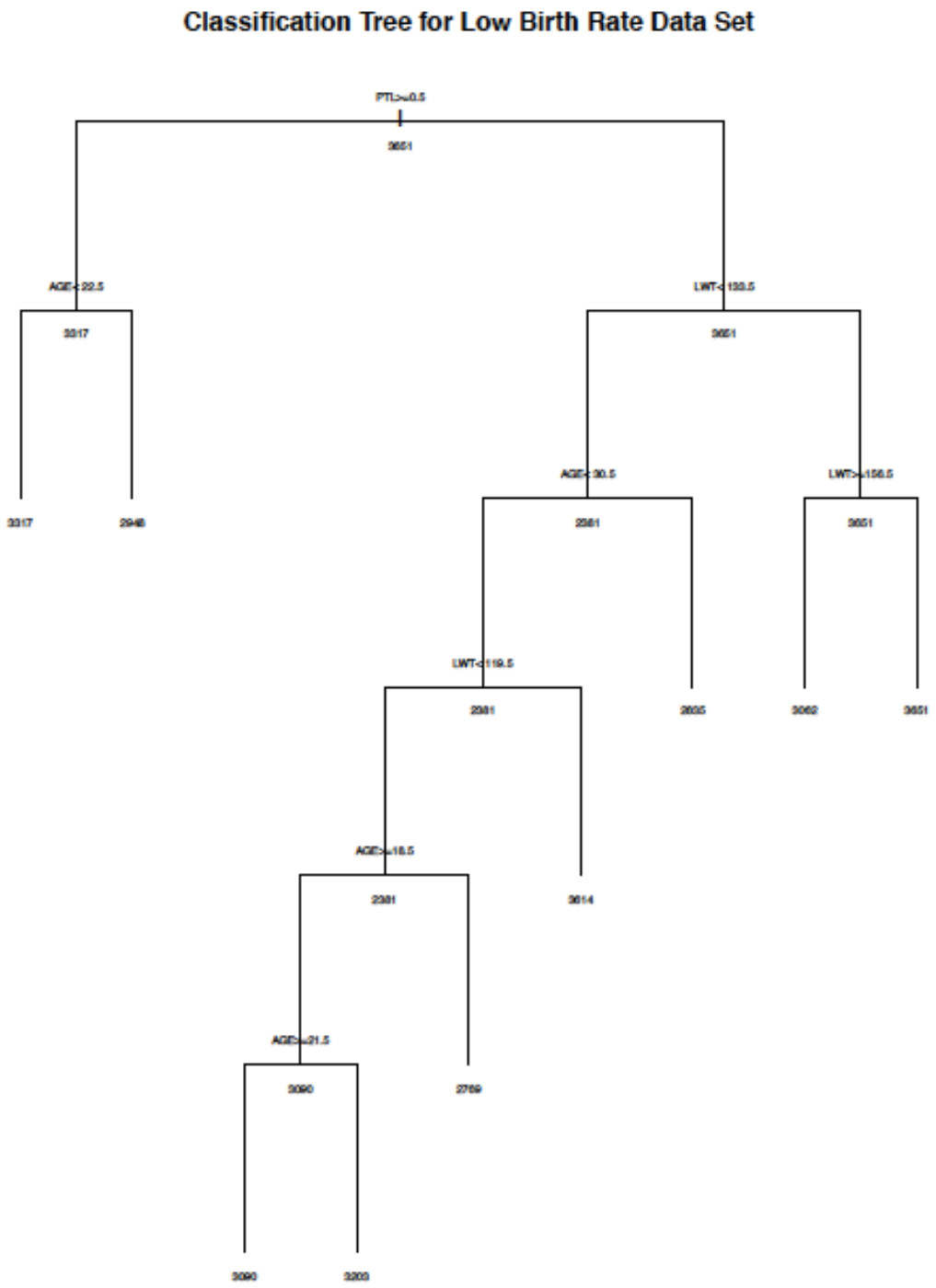
```
> library(rpart)
> ctree <- rpart(BWT ~ PTL + SMOKE + HT + AGE + RACE + LWT + UI,
+                data=lbr_train_data,
+                method = 'class')
> ctree.test <- data.frame(lbr_test_data,predict(ctree, lbr_test_data))
>
>
> ctree.test[which(ctree.test$BWT>=0),c(11)]
 [1] 2600 2750 2920 2977 2992 3042 3062 3076 3076 3090 3132 3147 3175 3274 3303 3374 3402 3459 3475 3586 3600
[22] 3637 3643 3770 3969 4153 4593  709 1588 1729 1928 1936 1970 2082 2084 2381 2495 2495
>
> summary(ctree.test[which(ctree.test$BWT>=2495),c(11)])
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   2495    3042    3175    3285    3586    4593
> summary(ctree.test[which(ctree.test$BWT< 2495),c(11)])
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
    709    1729    1936    1823    2082    2381
```

Name: Vipin Singh
Email: vipsingh@iu.edu

```
>
> lbr_test_bwt_0 <- subset(lbr_test_data, BWT >= 2495)
> lbr_test_bwt_1 <- subset(lbr_full_data, BWT < 2495)
>
> summary(lbr_test_bwt_0["BWT"])
      BWT
 Min.   :2495
 1st Qu.:3042
 Median :3175
 Mean   :3285
 3rd Qu.:3586
 Max.   :4593
> summary(lbr_test_bwt_1["BWT"])
      BWT
 Min.   : 709
 1st Qu.:1914
 Median :2187
 Mean   :2068
 3rd Qu.:2374
 Max.   :2466
```

The median differences are not much off, because the tree has a specific path to reach the predicted value for the BWT. Please see the screenshot of classification tree below.

Name: Vipin Singh
Email: vipsingh@iu.edu



Name: Vipin Singh
Email: vipsingh@iu.edu

Regression tree

The below screenshot shows how a regression tree is build and trained on the training data set. The regression tree model is then applied on the test data set. The Column 11 shows the test data BWT, and Column 12 is the predicted column for BWT. A difference in the prediction is shown below and then compared with the linear regression.

```
> rtree <- rpart(BWT ~ PTL + SMOKE + HT + AGE + RACE + LWT + UI,
+               data=lbr_train_data,
+               method = 'anova')
> rtree.test <- data.frame(lbr_test_data,predict(rtree, lbr_test_data))
> rtree.test
    ID LOW AGE LWT RACE SMOKE PTL HT UI FTV  BWT predict.rtree..lbr_test_data.
1   89   0  18 107    1     1   0  0  1   0 2600                       2707.556
2   98   0  22  95    3     0   0  1  0   0 2750                       2707.556
3  117   0  17 113    2     0   0  0  0   1 2920                       3056.697
4  124   0  19 138    1     1   0  0  0   2 2977                       3264.571
5  125   0  27 124    1     1   0  0  0   0 2992                       2957.643
6  128   0  21 185    2     1   0  0  0   2 3042                       2426.750
7  130   0  23 130    2     0   0  0  0   1 3062                       3056.697
8  132   0  18  90    1     1   0  0  1   0 3076                       2707.556
9  133   0  18  90    1     1   0  0  1   0 3076                       2707.556
10 135   0  19 132    3     0   0  0  0   0 3090                       3056.697
11 140   0  22 130    1     1   0  0  0   0 3132                       2957.643
12 141   0  30  95    1     1   0  0  0   2 3147                       2234.933
13 143   0  16 110    3     0   0  0  0   0 3175                       3056.697
14 155   0  20 169    3     0   1  0  1   1 3274                       2460.200
15 159   0  28 250    3     1   0  0  0   6 3303                       2426.750
16 167   0  16 135    1     1   0  0  0   0 3374                       3264.571
17 168   0  18 229    2     0   0  0  0   0 3402                       3056.697
18 173   0  23 190    1     0   0  0  0   0 3459                       3369.160
19 176   0  30 110    3     0   0  0  0   0 3475                       3056.697
20 182   0  23 130    1     0   0  0  0   0 3586                       3369.160
21 183   0  36 175    1     0   0  0  0   0 3600                       3786.444
22 188   0  25  95    1     1   3  0  1   0 3637                       2234.933
23 189   0  16 135    1     1   0  0  0   0 3643                       3264.571
24 199   0  24 110    3     0   1  0  0   0 3770                       3056.697
25 214   0  28 130    3     0   0  0  0   0 3969                       3056.697
26 221   0  25 130    1     0   0  0  0   2 4153                       3369.160
27 225   0  24 116    1     0   0  0  0   1 4593                       3369.160
28   4   1  28 120    3     1   1  0  1   0  709                       2460.200
29  16   1  27 150    3     0   0  0  0   0 1588                       3056.697
30  19   1  24 132    3     0   0  1  0   0 1729                       3056.697
31  27   1  20 150    1     1   0  0  0   2 1928                       3264.571
32  29   1  24 155    1     1   1  0  0   0 1936                       2426.750
33  30   1  21 103    3     0   0  0  0   0 1970                       2707.556
34  33   1  19 102    1     0   0  0  0   2 2082                       2707.556
35  35   1  26 117    1     1   1  0  0   0 2084                       2957.643
36  60   1  20 122    2     1   0  0  0   0 2381                       3264.571
37  82   1  23  94    3     1   0  0  0   0 2495                       2234.933
38  83   1  17 142    2     0   0  1  0   0 2495                       3056.697
```

Name: Vipin Singh
Email: vipsingh@iu.edu

Summary of regression tree algorithm for predicted BWT

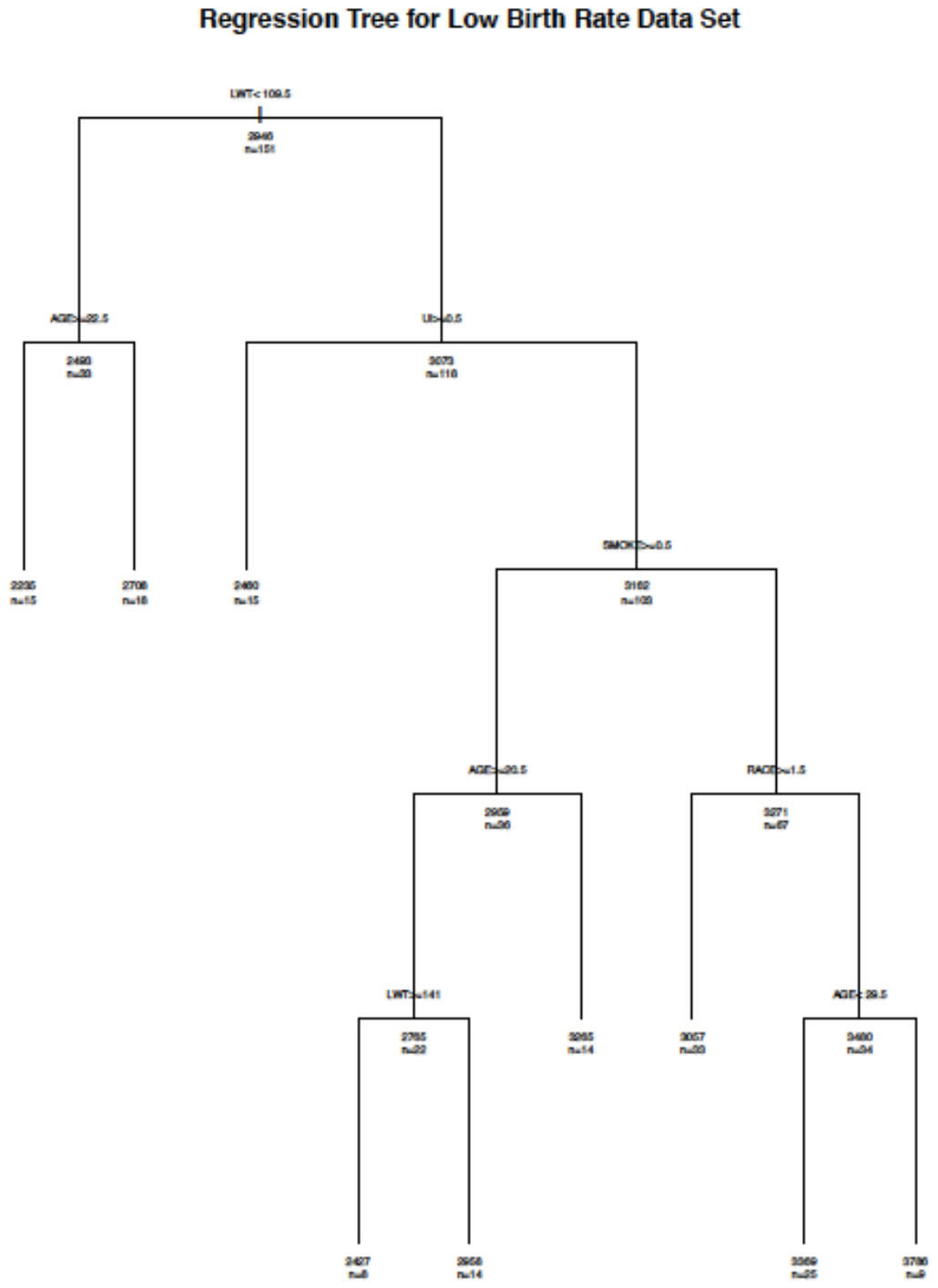
```
> summary(abs(rtree.test[11]-rtree.test[12]))
      BWT
 Min.   :   5.303
 1st Qu.: 177.379
 Median : 454.526
 Mean   : 578.713
 3rd Qu.: 875.598
 Max.   :1751.200
```

Summary of Linear regression algorithm for predicted BWT

```
> summary(abs(new_BWT - lbr_test_data$BWT))
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  17.07  134.50  473.60  567.20  906.00 1708.00
```

If we observe the above summary for the median differences in the prediction. The regression tree performs better than linear regression, and linear regression performs better than KNN, K= 1,2,3

Name: Vipin Singh
Email: vipsingh@iu.edu



Conclusion:

The Performance in terms of accuracy is in the following order from best to worst.

Classification Tree -> Regression Tree -> Linear Regression -> KNN.

K-Means is a clustering algorithm which is moving around 20% of data points in a wrong cluster.

Name: Vipin Singh
Email: vipsingh@iu.edu

3. Part 3

For each question, determine which algorithm is best. Discuss why you chose the algorithm you did and why you did not select the others.

- Given dog breed data (height, length, weight, ear length, fur length), can we classify new dogs?

  The dog breeds can be many, hence it seems to be a multi-class problem. KNN – K Nearest Neighbor is the best algorithm in this type of scenario. If we use KNN we can identify the nearest neighbor and can predict the breed of the new dog. Linear regression, clustering and Decision tree does not seem to be best for this multiclass problem.

- Given an unlabeled dataset of information (height, length, weight, ear length, fur length) about dogs, can we estimate four breed distinctions?

  As per the problem, we need 4 breed distinction, or 4 clusters. Hence this problem can be effectively solved by using K-Means algorithm. We need to use K=4, to create the 4 distinct clusters of the dataset. Linear Regression, KNN and Decision Tree are not best to solve this type of problem. Linear regression predicts any continuous value based on the model, KNN, as explained above, is good for multiclass problems and decision tree are useful for decision making scenarios like Yes or No, or the event can be predicted to happen or not etc.

- Given some information about a dog (height, length, weight, ear length, fur length, age, time at shelter) can we predict the amount someone will pay to adopt the dog?

  This scenario is trying to predict amount for each dog. The amount will be in a range of dollars. Hence, Linear Regression algorithms are best for this type of problem. Linear regression can easily predict the dollar amount based on the model. KNN can also be used here, but I think Linear regression will be a better choice in terms of accuracy. We don't need to do any clustering here; hence clustering algorithm is not required. Decision tree also is not the best but can be used.

- Given some information (breed, color, age category, size category, house-trained, good with kids) about a dog at a shelter, can we predict if that dog will be taken home or not?

  In this problem, a decision needs to be made based on the data – whether the dog will be taken home or not. Hence a Decision Tree algorithm is the best in this scenario. A decision tree can predict accurately yes and no, if dog will be taken home or not. The algorithms like KNN and clustering can also be used, but I think Decision Tree will be the best.