STAT-515- FINAL PROJECT REPORT (GROUP 12)
Finding Maternal Risk Factors for Low-Birth Weight Delivery Using Machine Learning

Introduction

A low birth weight is a public health issue that can be avoided. It is a significant predictor of child survival and development, in addition to lengthy implications such as the emergence of noncommunicable illness later in life. Tackling the causes linked to low gestational age can avert a huge number of deaths and morbidities. The primary purpose of this research was to discover risk variables for birth weight anomalies utilizing a machine learning approach.

Methods

This study implemented predictive LBW models. The study was employed to compare and identify the best-suited classifier for predictive classification among Logistic Regression, Decision tree, bagged-tree, and Random Forest (RF).

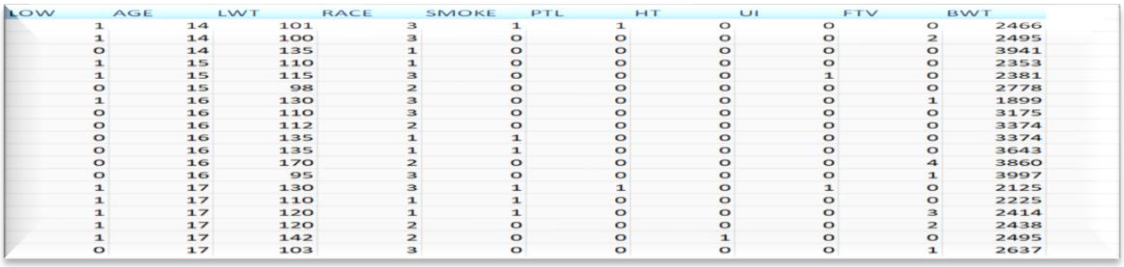
Research Questions

1. What is the most important factor influencing the low birth rate?
2. Which ML classifier is most appropriate for predictive classification among Generalized Linear Models & Nonlinear Models?

Data Exploration

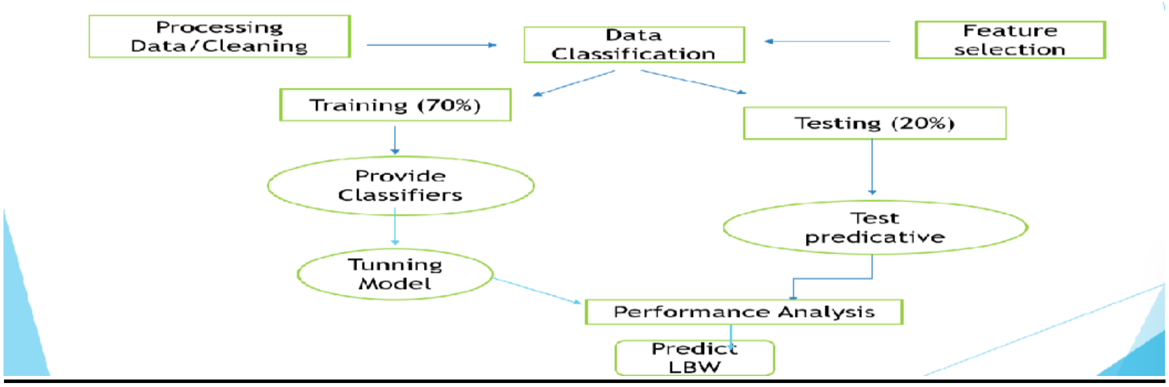
The data has 180 cases with 10 predictors with 2 classes (Low birth "0", High Birth "1")
The dependent variable are:
LOW: Birth weight is low.
BWT: Actual infant birth weight (BWT)
The predictor variables are:
AGE: The mother's age.
LWT: Last Mother's Weight RACE: 1 denotes white, 2 denotes black, and 3 denotes other.
SMOKE: Maternal cigarette condition: 1 = Yes, 0 = No
PTL: Premature labor background: 0 = None, 1 = One, 2 = Two, 3 = Three
HT: Hypertension history: 1 = Yes, 0 = No
Uterine irritation (UI): 1 = Yes, 0 = No
FTV: First pregnancy physician visits: 0 = none, 1 = one, 6 = six

Cleaned Data: Dropping NA Values



LOW	AGE	LWT	RACE	SMOKE	PTL	HT	UI	FTV	BWT
1	14	101	3	1	1	0	0	0	2466
1	14	100	3	0	0	0	0	2	2495
0	14	135	1	0	0	0	0	0	3941
1	15	110	1	0	0	0	0	0	2353
1	15	115	3	0	0	0	1	0	2381
0	15	98	2	0	0	0	0	0	2778
1	16	130	3	0	0	0	0	1	1899
0	16	110	3	0	0	0	0	0	3175
0	16	112	2	0	0	0	0	0	3374
0	16	135	1	1	0	0	0	0	3374
0	16	135	1	1	0	0	0	0	3643
0	16	170	2	0	0	0	0	4	3860
0	16	95	3	0	0	0	0	1	3997
1	17	130	3	1	1	0	1	0	2125
1	17	110	1	1	0	0	0	0	2225
1	17	120	1	1	0	0	0	3	2414
1	17	120	2	0	0	0	0	2	2438
1	17	142	2	0	0	1	0	0	2495
0	17	103	3	0	0	0	0	1	2637

Flowchart for Data Classification

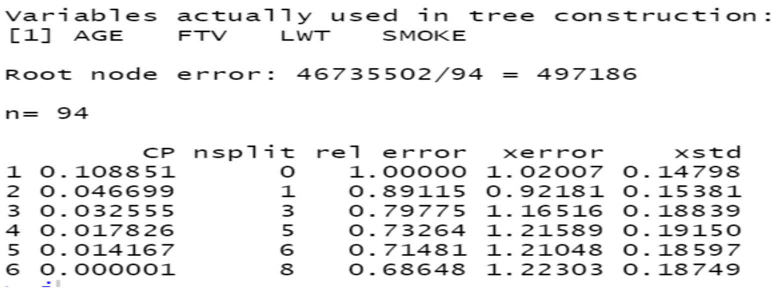


Once the data is cleaned and proceeded, it is classified into 70 per of the training set and

20 % of the testing set. Then performance analysis is done to provide predicated Low-Birth rate.

Decision Trees

We begin by constructing a huge initial regression tree. By setting a minimal benefit for cp=0.000001, which stands for "complexity parameter," we can assure that the tree is huge. This means that we will add additional splits to the regression tree as long as the total R-squared of the model rises by at least the cp value.



```
Variables actually used in tree construction:
[1] AGE   FTV   LWT   SMOKE

Root node error: 46735502/94 = 497186

n= 94

          CP nsplit rel error  xerror    xstd
1 0.108851      0   1.00000 1.02007 0.14798
2 0.046699      1   0.89115 0.92181 0.15381
3 0.032555      3   0.79775 1.16516 0.18839
4 0.017826      5   0.73264 1.21589 0.19150
5 0.014167      6   0.71481 1.21048 0.18597
6 0.000001      8   0.68648 1.22303 0.18749
```

Pruning the Tree

That the ideal value for cp is the one that results in the lowest xerror in the preceding output, which indicates the error on the cross-validation data sets. Plotting the best Cp values for min error.

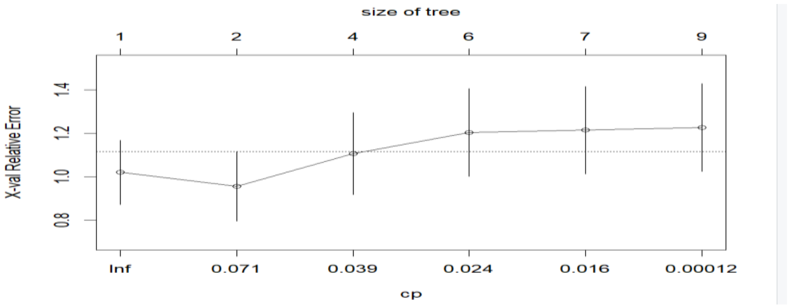


Fig-Cp vs error

Seven terminal nodes make up the pruned tree that is produced, and its cp value is close to 0.019(min error). Each terminal node indicates the number of discoveries from the dataset that are related to the comment Low Birth Weight as well as the reason for the low birth weight. It demonstrates that a mother who weighs less than 109 pounds gives birth to a 2531-gram child. Additionally, we may observe that mothers who smoke and who are smaller in stature also have babies with low birth weights (less than 2500 grams). The factors of less smoking, being at least 20 years old, and the mother's weight being larger than 164 pounds all increase the likelihood of having a baby with a healthy weight.

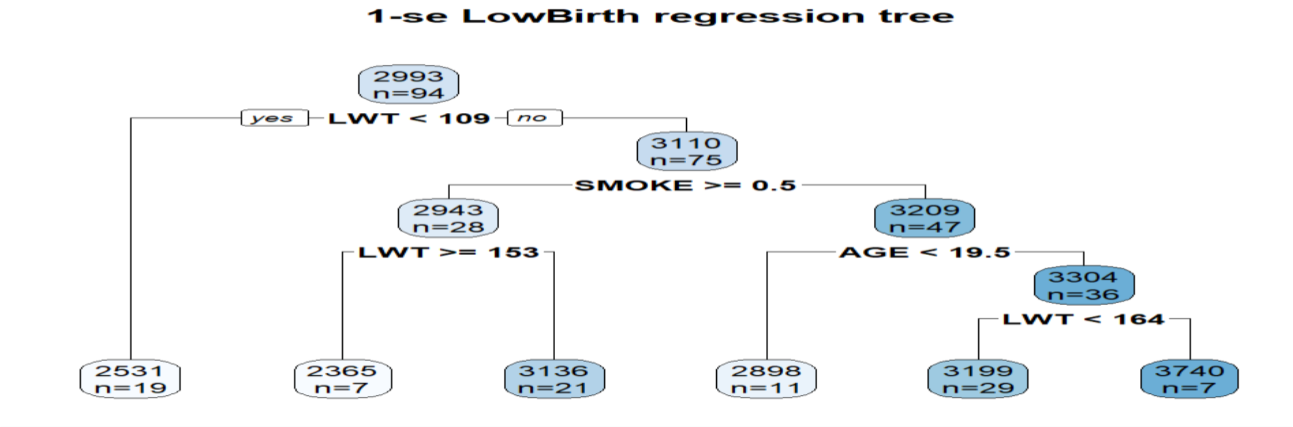


Fig: -Low Birth regression Tree

Logistic Regression:

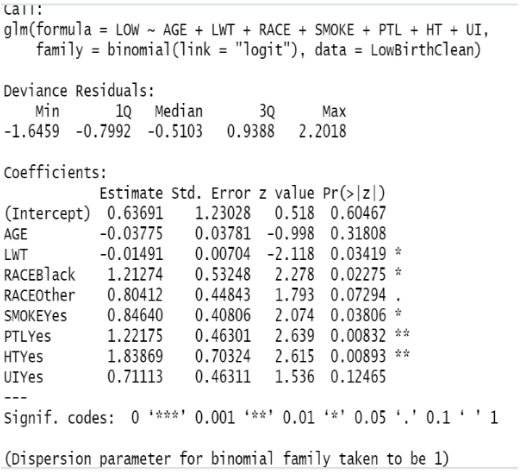
The Logistic Regression is a regression model where the response variable consists of categorical values such as True or False else 0 or 1. It is used to measure the binary response to the value of response based on the formula representing the predictor variables.

The general expression used for logistic regression is y = 1/ (1+e^-(a+b1x1+b2x2+b3x3+...)) where y is the response variable, x is the predictor variable and a and b are the coefficients which are numeric constants.

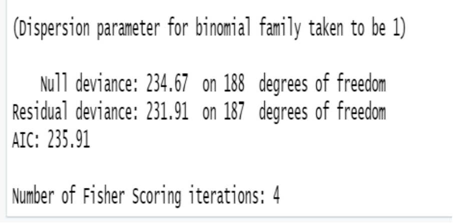
The function used to measure the logistic regression is glm () where glm is generalized linear model. This is like lm (), but the only exception is that here the argument family must be passed has family = binomial so that the R can recognize it as logistic regression rather than the general regression.

Logistic Regression for Simple models

The Low birth rate dataset consists of the following variables namely, LOW, AGE, LWT, RACE, SMOKE, PTL, HI AND UI. A logistic regression for each of the variable is performed.

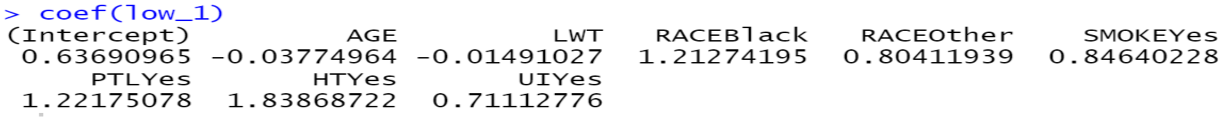


```
Call:
glm(formula = LOW ~ AGE + LWT + RACE + SMOKE + PTL + HT + UI,
    family = binomial(link = "logit"), data = LowBirthClean)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.6459  -0.7992  -0.5103   0.9388   2.2018

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  0.63691    1.23028   0.518  0.60467
AGE         -0.03775    0.03781  -0.998  0.31808
LWT         -0.01491    0.00704  -2.118  0.03419 *
RACEBlack    1.21274    0.53248   2.278  0.02275 *
RACEOther    0.80412    0.44843   1.793  0.07294 .
SMOKEYes     0.84640    0.40806   2.074  0.03806 *
PTLYes       1.22175    0.46301   2.639  0.00832 **
HTYes        1.83869    0.70324   2.615  0.00893 **
UIYes        0.71113    0.46311   1.536  0.12465
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)
```



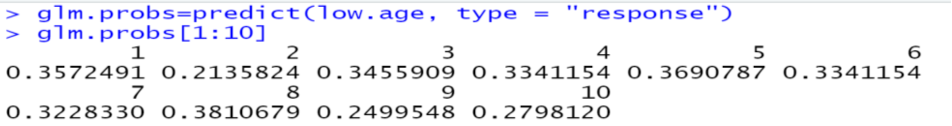
```
(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 234.67  on 188  degrees of freedom
Residual deviance: 231.91  on 187  degrees of freedom
AIC: 235.91

Number of Fisher Scoring iterations: 4
```

The summary () function is used to get the aspects of the fitted model such as min, max, median, etc. The coef () function is used to access the coefficients of the fitted model



```
> coef(low_1)
(Intercept)         AGE         LWT   RACEBlack   RACEOther    SMOKEYes
 0.63690965 -0.03774964 -0.01491027  1.21274195  0.80411939  0.84640228
     PTLYes       HTYes       UIYes
 1.22175078  1.83868722  0.71112776
```

The predict () function is used to predict the probabilities that will raise by giving the type as “response”. The first 10 probability values that might go up has been predicted.



```
> glm.probs=predict(low.age, type = "response")
> glm.probs[1:10]
        1         2         3         4         5         6
0.3572491 0.2135824 0.3455909 0.3341154 0.3690787 0.3341154
        7         8         9        10
0.3228330 0.3810679 0.2499548 0.2798120
```

A plot using the GGally library having the ggscatmat () function is used to show the scatterplot and correlations between each variable.

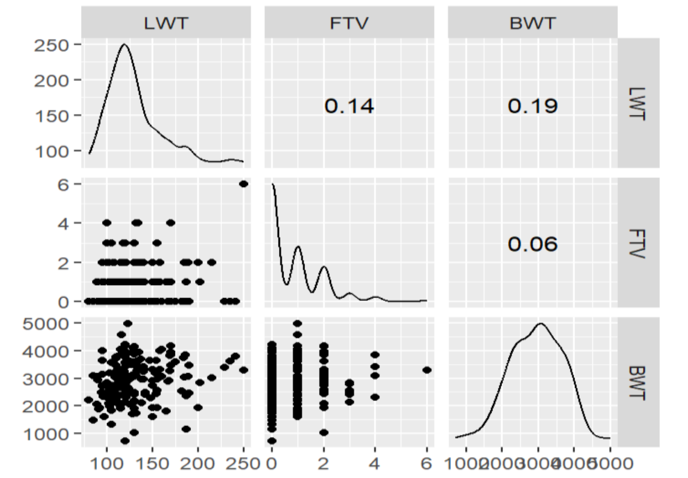


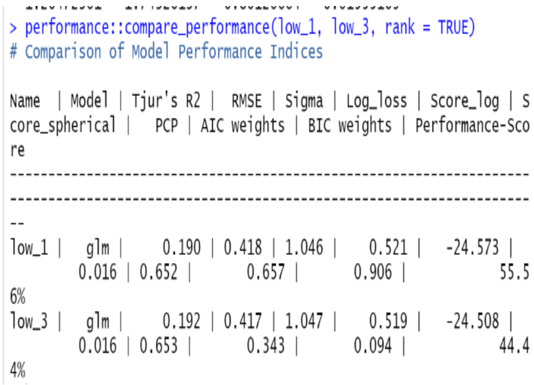
Fig-Scatmat plot

Logistic Regression for Multivariate models

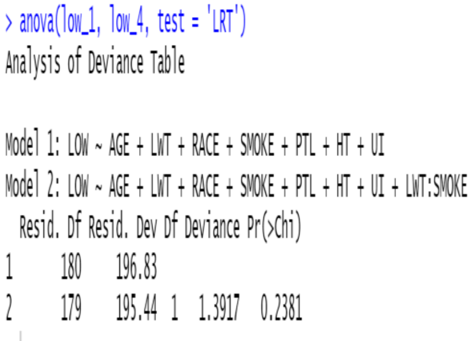
Few Logistic comparisons are made with the main variables after removing the minor effects that don’t affect the birth rate much and then comparison is done between them. Here the factors Age and Weight, Age and Smoking, Weight and Smoking are used, and the fit models are compared and tested using Likelihood Ratio Test.

Anova is a statistical test function used for estimating how a quantitative dependent variable changes according to the levels of one or more categorical variable. Here it is used to test the level of changes between the original fit model and the model with Weight and Smoking.

The performance () function is used to check and evaluate the performance of each of the fit models. Here two fit model’s performance are compared.



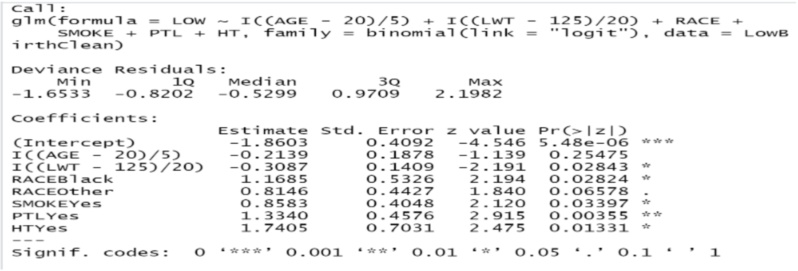
```
> performance::compare_performance(low_1, low_3, rank = TRUE)
# Comparison of Model Performance Indices

Name  | Model | Tjur's R2 |  RMSE | Sigma | Log_loss | Score_log | S
core_spherical |   PCP | AIC weights | BIC weights | Performance-Sco
re
--------------------------------------------------------------------
--------------------------------------------------------------------
--
low_1 |   glm |     0.190 | 0.418 | 1.046 |    0.521 |   -24.573 |
          0.016 | 0.652 |       0.657 |       0.906 |             55.5
6%
low_3 |   glm |     0.192 | 0.417 | 1.047 |    0.519 |   -24.508 |
          0.016 | 0.653 |       0.343 |       0.094 |             44.4
4%
```



```
> anova(low_1, low_4, test = 'LRT')
Analysis of Deviance Table

Model 1: LOW ~ AGE + LWT + RACE + SMOKE + PTL + HT + UI
Model 2: LOW ~ AGE + LWT + RACE + SMOKE + PTL + HT + UI + LWT:SMOKE
  Resid. Df Resid. Dev Df Deviance Pr(>Chi)
1       180     196.83
2       179     195.44  1   1.3917   0.2381
```

Logistic Regression for Final model

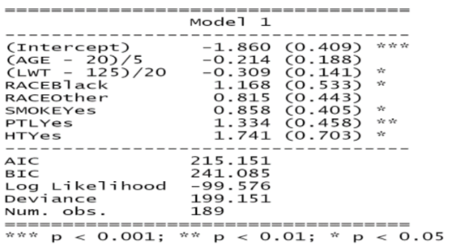
After comparing the fit models for each variable such as age, weight, smoke, etc., the mother’s age is the main factor, so a basic interpretation is done by keeping the age as 5 and weight as 20 lbs and the summary is obtained.



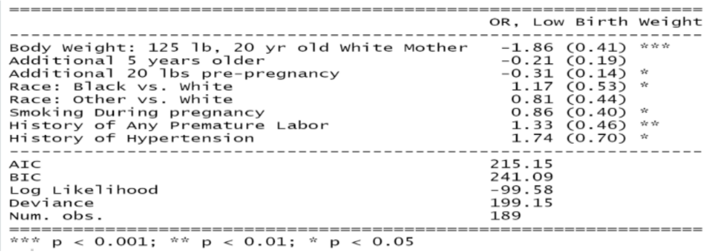
```
Call:
glm(formula = LOW ~ I((AGE - 20)/5) + I((LWT - 125)/20) + RACE +
    SMOKE + PTL + HT, family = binomial(link = "logit"), data = LowB
irthClean)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.6533  -0.8202  -0.5299   0.9709   2.1982

Coefficients:
                  Estimate Std. Error z value Pr(>|z|)
(Intercept)        -1.8603     0.4092  -4.546 5.48e-06 ***
I((AGE - 20)/5)    -0.2139     0.1878  -1.139  0.25475
I((LWT - 125)/20)  -0.3087     0.1409  -2.191  0.02843 *
RACEBlack           1.1685     0.5326   2.194  0.02824 *
RACEOther           0.8146     0.4427   1.840  0.06578 .
SMOKEYes            0.8583     0.4048   2.120  0.03397 *
PTLYes              1.3340     0.4576   2.915  0.00355 **
HTYes               1.7405     0.7031   2.475  0.01331 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Parameter Estimates

The texreg package in R uses an intermediate function called extract () to extract the information for the model and place them in the right place.



```
======================================
                     Model 1
--------------------------------------
(Intercept)        -1.860 (0.409) ***
(AGE - 20)/5       -0.214 (0.188)
(LWT - 125)/20     -0.309 (0.141) *
RACEBlack           1.168 (0.533) *
RACEOther           0.815 (0.443)
SMOKEYes            0.858 (0.405) *
PTLYes              1.334 (0.458) **
HTYes               1.741 (0.703) *
--------------------------------------
AIC               215.151
BIC               241.085
Log Likelihood    -99.576
Deviance          199.151
Num. obs.         189
======================================
*** p < 0.001; ** p < 0.01; * p < 0.05
```



```
=====================================================================
                                              OR, Low Birth Weight
---------------------------------------------------------------------
Body Weight: 125 lb, 20 yr old White Mother   -1.86 (0.41) ***
Additional 5 years older                      -0.21 (0.19)
Additional 20 lbs pre-pregnancy               -0.31 (0.14) *
Race: Black vs. White                          1.17 (0.53) *
Race: Other vs. White                          0.81 (0.44)
Smoking During pregnancy                       0.86 (0.40) *
History of Any Premature Labor                 1.33 (0.46) **
History of Hypertension                        1.74 (0.70) *
---------------------------------------------------------------------
AIC                                           215.15
BIC                                           241.09
Log Likelihood                                -99.58
Deviance                                      199.15
Num. obs.                                     189
=====================================================================
*** p < 0.001; ** p < 0.01; * p < 0.05
```

Model Plot focusing on the mother's weight and smoking habit

The function effect () is used to show the relation between the Mother's Weight and Smoking status along with premature labor and hypertension record. From the model, the factors premature labor and hypertension are also one of the factors in affecting low birth weight. The effect function can be used with any regression model which creates effects to display one or more conditions.

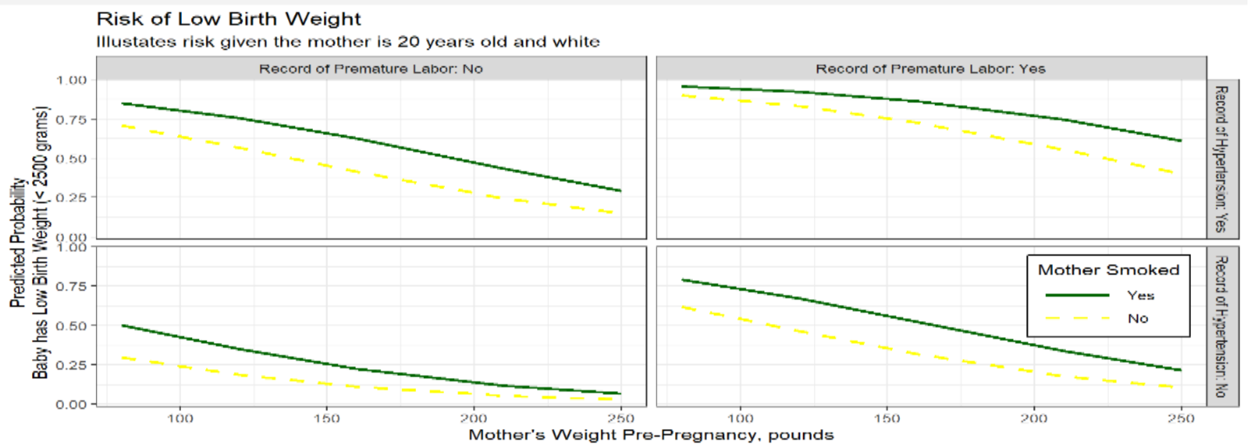
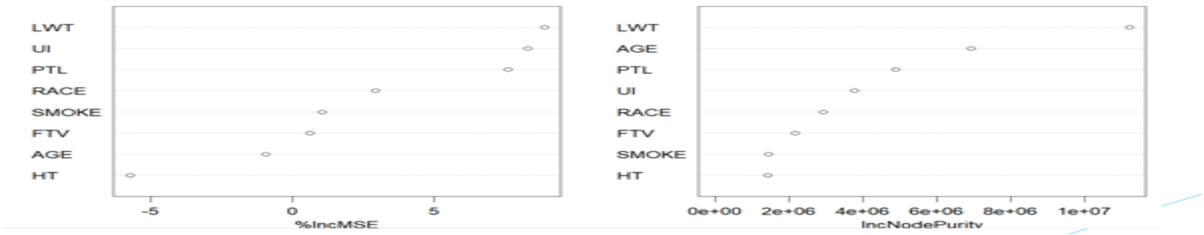


Fig-Risk of low birth weight

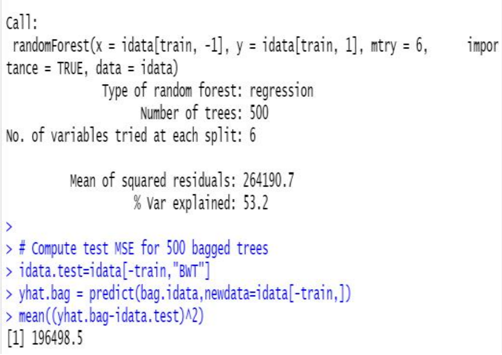
For Bragged and RF we are using both the dependent variable LOW and BWT to find the mean square residual, and mean of test and train a critical variable to determine the risk of LBW.

Bagged trees

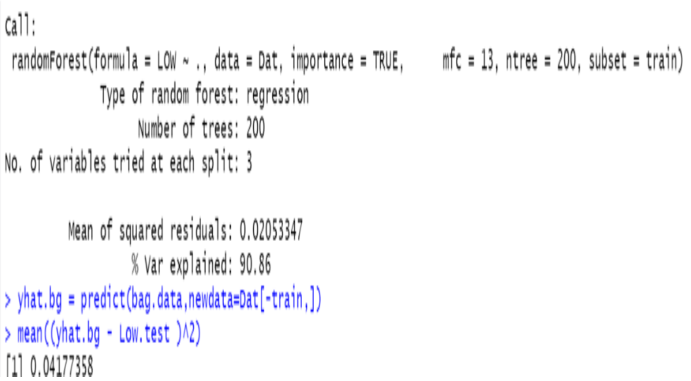


Bagged trees are also called Bootstrap aggregation. This is also termed as drawing random samples. They are nothing but tree like models which are used mainly for prediction purpose.

This could be of best use in cases of non-linear data. The basic concept is to take samples and get average for them and use it. Here, the predictors are LWT, AGE, PTL and HT.



```
Call:
 randomForest(x = idata[train, -1], y = idata[train, 1], mtry = 6,      impor
tance = TRUE, data = idata)
               Type of random forest: regression
                     Number of trees: 500
No. of variables tried at each split: 6

          Mean of squared residuals: 264190.7
                    % Var explained: 53.2
>
> # Compute test MSE for 500 bagged trees
> idata.test=idata[-train,"BWT"]
> yhat.bag = predict(bag.idata,newdata=idata[-train,])
> mean((yhat.bag-idata.test)^2)
[1] 196498.5
```



```
Call:
 randomForest(formula = LOW ~ ., data = Dat, importance = TRUE,      mfc = 13, ntree = 200, subset = train)
               Type of random forest: regression
                     Number of trees: 200
No. of variables tried at each split: 3

          Mean of squared residuals: 0.02053347
                    % Var explained: 90.86
> yhat.bg = predict(bag.data,newdata=Dat[-train,])
> mean((yhat.bg - Low.test )^2)
[1] 0.04177358
```

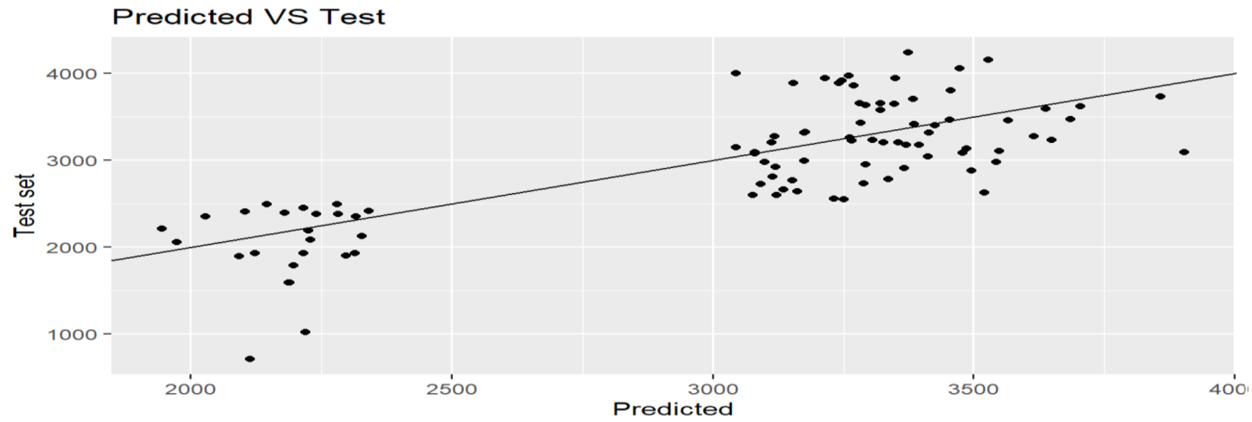


Fig- Predicted VS Test data

This is the plot we got for our data, and this is partially linear. The basics are to create samples which can be used as decision trees later, with that different models are generated and, in the end, the mean of them is used in place of decision trees.

Random Forest:

Creating a decision tree is one such way. Nevertheless, the disadvantage of employing a single decision tree is that it has a high variance. That instance, if we divide the dataset in half and apply the decision tree to each half, the outcomes may be very distinct.

By plotting Error vs Tree helps to find the trees with minimum error. The out-of-bag (OOB) error for a particular tree is the model error in charge of determining that was not included in the training dataset for that tree. OOB is a fairly simple method for determining the test error of a bagged model without using cross-validation or the model of society method. The estimated OBB error is 2.14%.

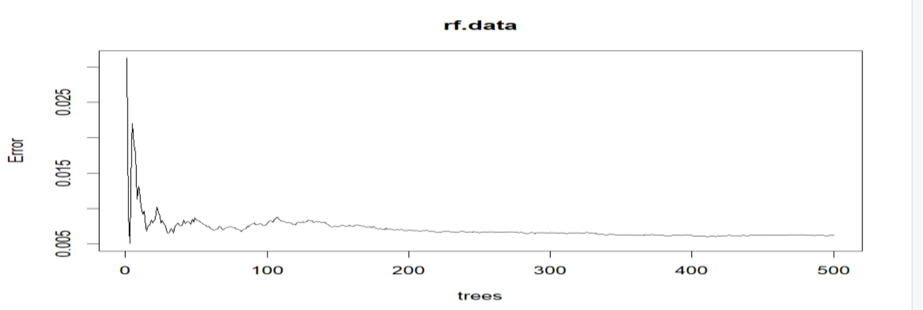
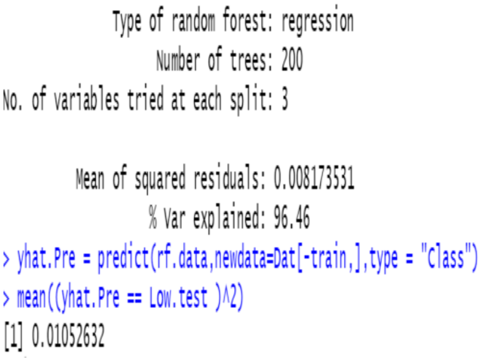
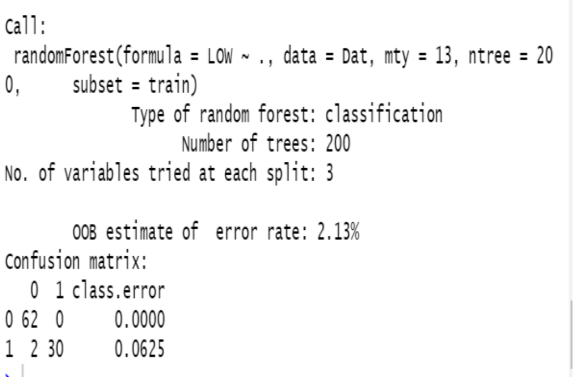


Fig: -trees vs Error



```
        Type of random forest: regression
              Number of trees: 200
No. of variables tried at each split: 3

          Mean of squared residuals: 0.008173531
                    % Var explained: 96.46
> yhat.Pre = predict(rf.data,newdata=Dat[-train,],type = "Class")
> mean((yhat.Pre == Low.test )^2)
[1] 0.01052632
```

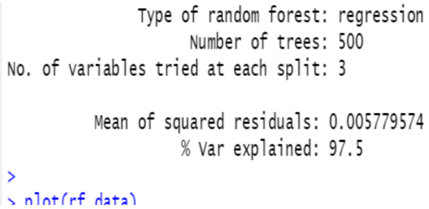


```
Call:
 randomForest(formula = LOW ~ ., data = Dat, mty = 13, ntree = 20
0,      subset = train)
               Type of random forest: classification
                     Number of trees: 200
No. of variables tried at each split: 3

        OOB estimate of  error rate: 2.13%
Confusion matrix:
   0  1 class.error
0 62  0      0.0000
1  2 30      0.0625
```

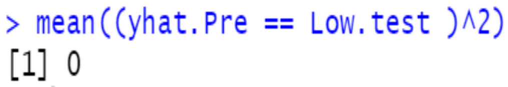
Fig:- N-tree 300's MSE and Test mean **Fig:- OOB and confusion matrix**

Tune

According to the results, the model with the lowest test mean squared error (MSE) utilized 499 trees.We also recognize that the model's root mean squared error was 0.057. This can be thought of as the difference in average between both the anticipated and observed values for low birth weight. And comparing the test and train mean provide training as 0.0057 and test error to be 0. By increasing the number of trees we could reduce the MSE value.

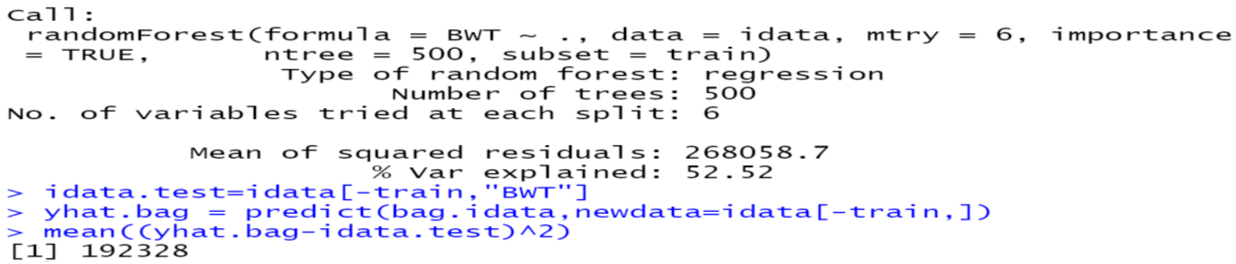


```
               Type of random forest: regression
                     Number of trees: 500
No. of variables tried at each split: 3

          Mean of squared residuals: 0.005779574
                    % Var explained: 97.5
>
> plot(rf.data)
```



```
> mean((yhat.Pre == Low.test )^2)
[1] 0
```

Fig: - Train and test mean for tree 500



```
Call:
 randomForest(formula = BWT ~ ., data = idata, mtry = 6, importance
 = TRUE,        ntree = 500, subset = train)
               Type of random forest: regression
                     Number of trees: 500
No. of variables tried at each split: 6

          Mean of squared residuals: 268058.7
                    % Var explained: 52.52
> idata.test=idata[-train,"BWT"]
> yhat.bag = predict(bag.idata,newdata=idata[-train,])
> mean((yhat.bag-idata.test)^2)
[1] 192328
```

Here, we are predicting the most important Variable affecting high birth weight. The Four most important variables are LWT, AGE, PTL

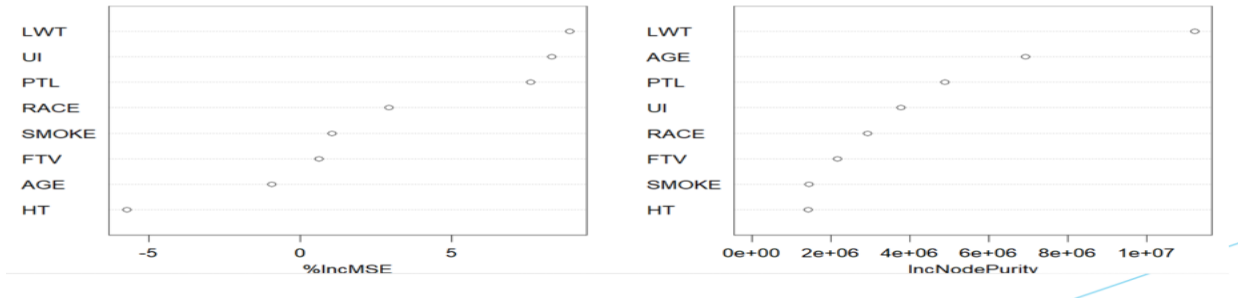


Fig-Important variables

Inference For Random Forest: The tree of 500 has even more less MSE in training and testing compared to tree of 200. Moreover, and AGE are most influential variable.

Conclusion:- Analyzing logistic regression, Bragg tree and Random Forest we could determine that the **Age of the mother** and the **Mother's Weight,** are the very high-risk factor for Low-birth rates. And we overcome the logistic regression's difficulty of prediction using the p-value for each variable by Decision tree, bagged and Rf performance. In case of logistic regression almost most of the P value > 0.55, so the degree of predication went more complex. The Random Forest is the most accurate method compared to all the methods of regression. Because The MSE and mean difference of the Train and test got reduced even more compared to all the methods. To obtain reliable estimates of the training and testing error, there is no requirement for cross-validation or a distinct test set.

Reference

[1] Logistic regression model [2022] Available: https://www.tutorialspoint.com/r/r_logistic_regression.htm [Accessed on 11/12/2022].

[2] Anova in R [2022] Available: https://www.scribbr.com/statistics/anova-in-r/

[Accessed on 8/12/2022].

[3] Referred class notes.