Pradip Dhakal

Data Mining Final Project

Introduction

In this project, breast cancer (diagnostic) dataset from the Kaggle website is used to make statistical models using different supervised and unsupervised algorithms. This project uses classification algorithms such as logistic regression and the k-nearest neighbor algorithm. Also, decision tree and random forest algorithms are used, and all the results are compared. Next, principal component analysis is performed, and the logistic regression and k-nearest neighbor algorithms are again used to see how the results will differ after using principal component analysis. The k-means clustering is also used to see if the data can be clustered into separate groups.

Data

This data is extracted from the website kaggle.com. It is a binary classified data where we can predict whether nuclei of the breast are benign or malignant to discover if someone has breast cancer or not. The attribute information of the dataset is given below.

Attribute Information:

1) ID number
2) Diagnosis (M = malignant, B = benign)
3-32)

Ten real-valued features are computed for each cell nucleus:

a) radius (mean of distances from the center to points on the perimeter)
b) texture (standard deviation of gray-scale values)
c) perimeter
d) area
e) smoothness (local variation in radius lengths)
f) compactness (perimeter^2 / area - 1.0)
g) concavity (severity of concave portions of the contour)
h) concave points (number of concave portions of the contour)
i) symmetry
j) fractal dimension ("coastline approximation" - 1)

The mean, standard error and “worst” or largest (mean of the three largest values) of these features were computed for each image, resulting in 30 features. For instance, field 3 is Mean Radius, field 13 is Radius SE, field 23 is Worst Radius.

All feature values are recoded with four significant digits.

Missing attribute values: none

Class distribution: 357 benign, 212 malignant

The diagnosis attribute is the dependent variable, and it is predicted using all other independent variables. The dimension of the data is 569 rows and 32 columns.

First, the dataset is checked to see if it has any missing values. Since this dataset does not have any missing value, it is ready for data analysis.

To explore more about the dataset, the correlation between the independent variables is observed. There is not a strong correlation between the independent variables except the correlation between radius_mean and area_mean, radius_mean and perimeter_mean, concave.points_means and radius_mean, area_se and radius_mean, and so on, which make sense because area, perimeter, and radius are directly proportional to each other.

Classification Algorithms

Under classification, logistic regression and k-nearest neighbor algorithms are used to make a model and predict the result. First, the logistic regression is used, and four independent variables: radius_mean, texture_se, perimeter_worst, and area_mean, are used to make the model. The summary of the logistic model is given below:

```
Call:
glm(formula = diagnosis ~ radius_mean + texture_se + perimeter_worst +
    area_mean, family = binomial, data = BCData)

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-2.83525  -0.18563  -0.06220   0.00416   3.09118

Coefficients:
                Estimate Std. Error z value Pr(>|z|)
(Intercept)     -7.04679    8.85946  -0.795   0.4264
radius_mean     -3.73179    1.34426  -2.776   0.0055 **
texture_se       1.75908    0.43476   4.046 5.21e-05 ***
perimeter_worst  0.41170    0.05139   8.011 1.14e-15 ***
area_mean        0.02151    0.01423   1.512   0.1305
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 751.44  on 568  degrees of freedom
Residual deviance: 151.94  on 564  degrees of freedom
AIC: 161.94

Number of Fisher Scoring iterations: 9
```

From the summary, it is observed that the p-value of area_mean is greater than the significance level, so area_mean is not statistically significant in predicting the diagnosis while taking the above four independent variables.

The diagnosis outcome is predicted using the above model, and the error rate is calculated.

The confusion matrix for logistic regression:

```
          diagnosis
Prediction   B   M
         B 347  19
         M  10 193
```

Among 569 data, 10 of them were misclassified as malignant, and 19 were misclassified as benign. The error rate for our model is 5%.

Next, the k-nearest neighbor algorithm is performed with a k value from 1 to 25. The same four independent variables were used to calculate the error rate. Half of the overall data is split into the training dataset and another half into the testing dataset. The diagnosis outcome is predicted, and the error value for each case is given below.

```
Error[ 1 ]:  0.1021127
Error[ 2 ]:  0.09859155
Error[ 3 ]:  0.08098592
Error[ 4 ]:  0.08802817
Error[ 5 ]:  0.07746479
Error[ 6 ]:  0.07394366
Error[ 7 ]:  0.07394366
Error[ 8 ]:  0.08450704
Error[ 9 ]:  0.09507042
Error[ 10 ]:  0.08098592
Error[ 11 ]:  0.0915493
Error[ 12 ]:  0.08802817
Error[ 13 ]:  0.08802817
Error[ 14 ]:  0.07746479
Error[ 15 ]:  0.08450704
Error[ 16 ]:  0.08450704
Error[ 17 ]:  0.08450704
Error[ 18 ]:  0.08450704
Error[ 19 ]:  0.08450704
Error[ 20 ]:  0.08098592
Error[ 21 ]:  0.08098592
Error[ 22 ]:  0.08450704
Error[ 23 ]:  0.08098592
Error[ 24 ]:  0.07746479
Error[ 25 ]:  0.08098592
```

From the observation, it is observed when k = 6 and 7, the error rate is least, which is 0.07394 or 7.94%.

On comparing the above two cases, logistic regression was better in predicting the result. There is a difference in an error rate of around 3% when comparing the above two results.

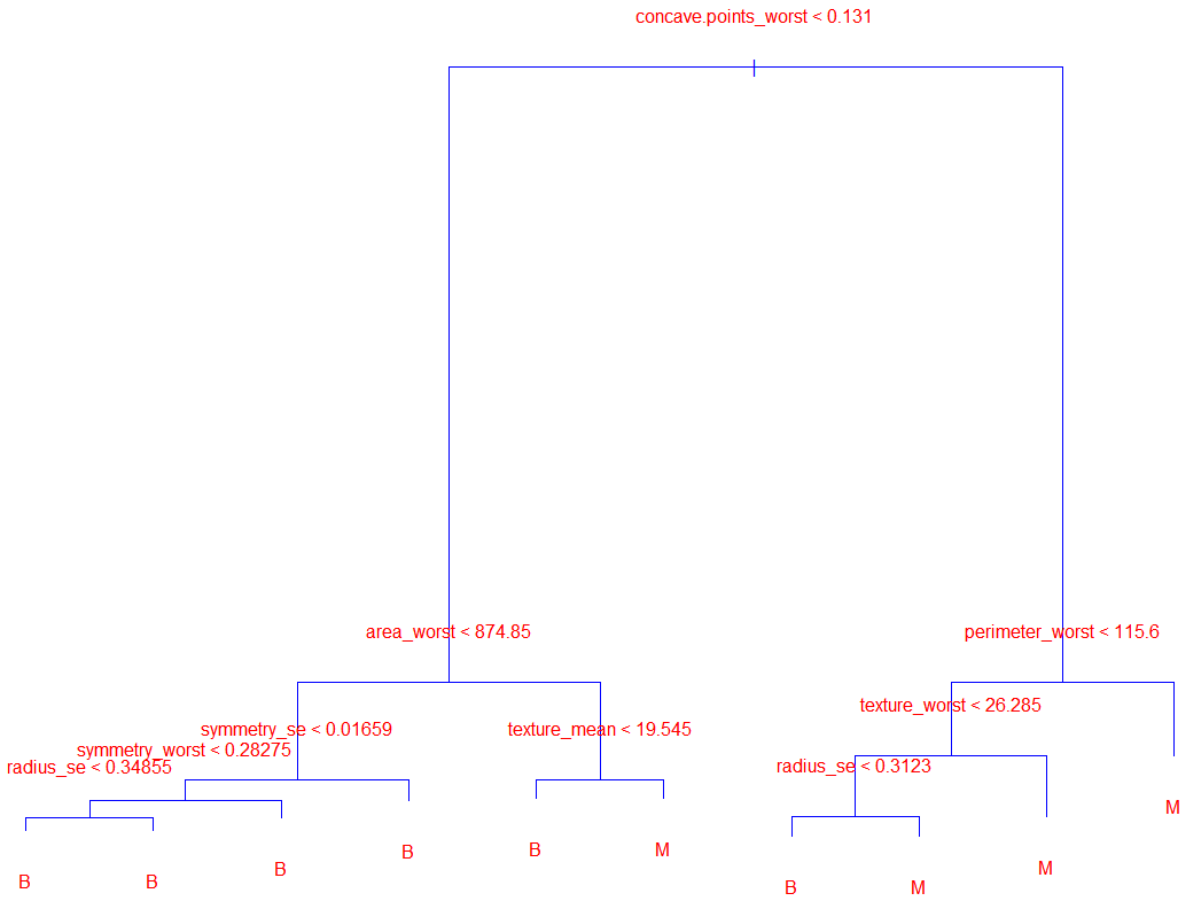
Decision Tree

The decision tree is performed using all the independent variables of the datasets, and the error rate is calculated. Later, the decision tree is pruned and again calculates the error rate.

The summary of the decision tree using all the variables:

```
Classification tree:
tree(formula = diagnosis ~ . - id, data = BCData1, subset = train)
Variables actually used in tree construction:
[1] "concave.points_worst" "area_worst"           "symmetry_se"          "symmetry_worst"       "radius_se"
[6] "texture_mean"         "perimeter_worst"      "texture_worst"
Number of terminal nodes:  10
Residual mean deviance:  0.1163 = 31.88 / 274
Misclassification error rate: 0.03169 = 9 / 284
```

From the summary, it is observed that even though all the independent variables from the dataset were used to create the model, only eight among those are used to construct the tree. The variables that are used in the tree construction are concave.point_worst, area_worst, symmetry_se, symmetry_worst, radius_se, texture_mean, perimeter_worst, and texture_worst. Also, there is a total number of 10 terminal nodes in the decision tree.



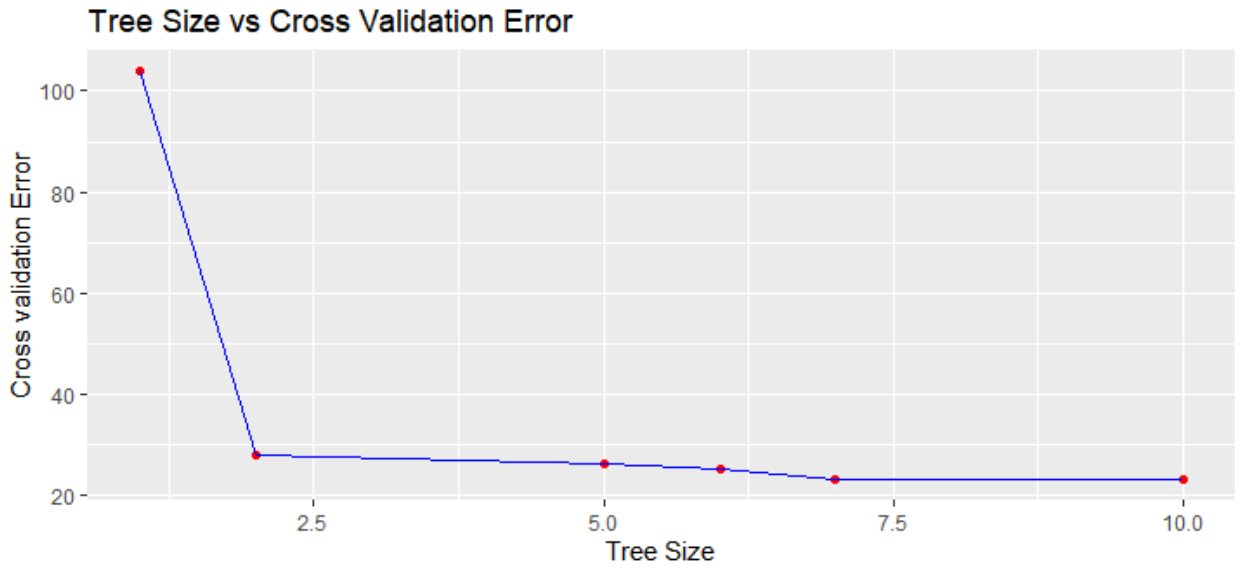
From the above decision tree, it is observed that when the area_worst is smaller than 874.85, the cancer is benign unless the value of texture_mean is greater than 19.545. Similarly, a higher value of concave.point_worst, perimeter_worst, texture_worst leads to the result that the cancer is malignant.

The confusion matrix for the unpruned tree:

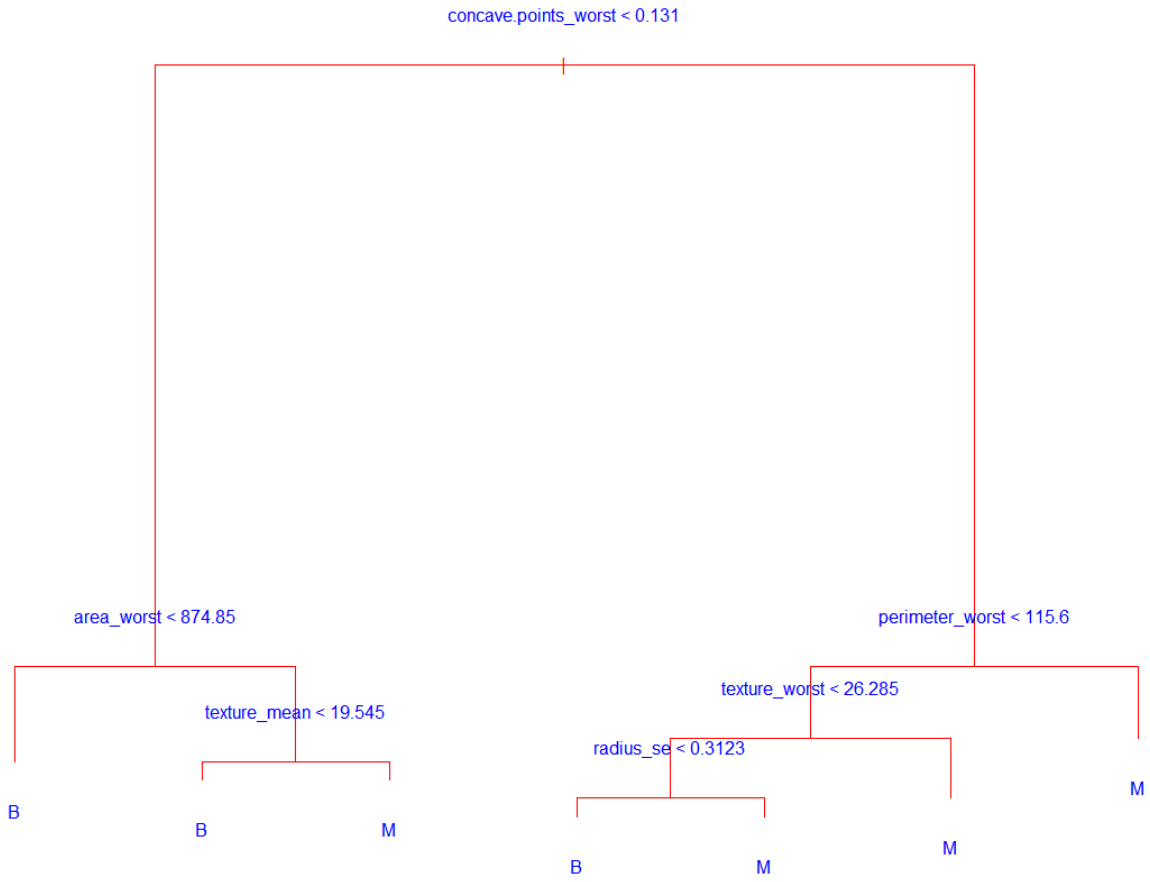
BCTreePred	B	M
B	168	6
M	8	102

On predicting the result and calculating the error rate, it was 4.9%. This is the same as the error rate of the logistic regression. The result from the decision tree when using all the independent variables as predictors is better than the k-nearest neighbor algorithm when only using four of the predictors.

Next, the decision tree is pruned at the best size value, and the result is predicted again. First, the cross-validation method is performed to check the best size to cut the tree. The best size was obtained when the size of the tree is equal to 7 and 10, which can be observed from the graph given below.



The tree is pruned at size = 7. The pruned tree is given below.



The confusion matrix for the pruned tree:

```
PrunedTreePred   B   M
             B 168   6
             M   8 102
```

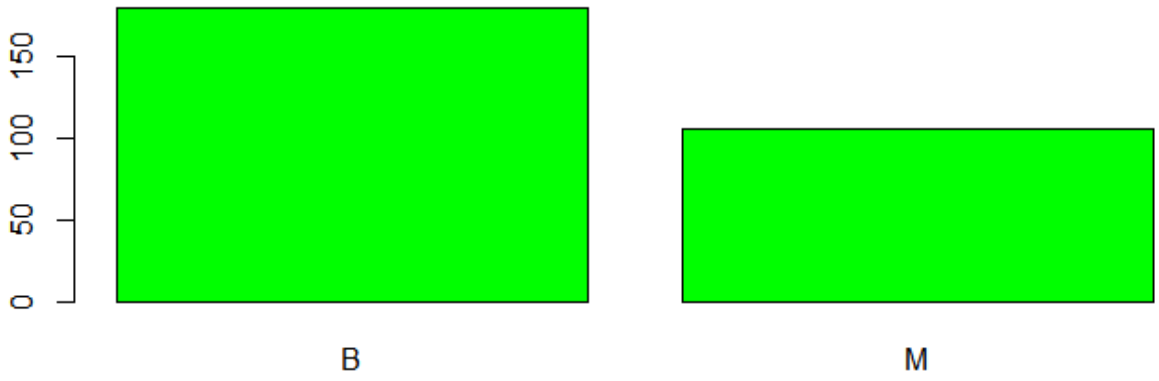
After pruning the tree at size = 7, the result is again predicted. The new error rate after pruning the tree is 4.9%, which is the same as an unpruned tree. It is because the error rate at size = 7 is equal to the error at size = 10. Although the error rate was not improved by pruning the tree, it improved the time efficiency. This cannot be observed in this dataset since it is not very large in size. When the size of the dataset is very large, it will certainly make a big difference.

Random Forest

Inside the Random Forest algorithm, bagging and boosting are performed, and the error rate is calculated. A random forest model is created using all the predictors, and the result is predicted using that model.

Pradip Dhakal

First, we perform bagging and the result. The plot of the predicted result using bagging is given below.



From the plot, it is observed that there are more benign cases as compared to the malignant cases. Let's check the mean error and figure out if it is indeed the case.

The confusion matrix for bagging:

BagPrediction	B	M
B	172	7
M	5	101

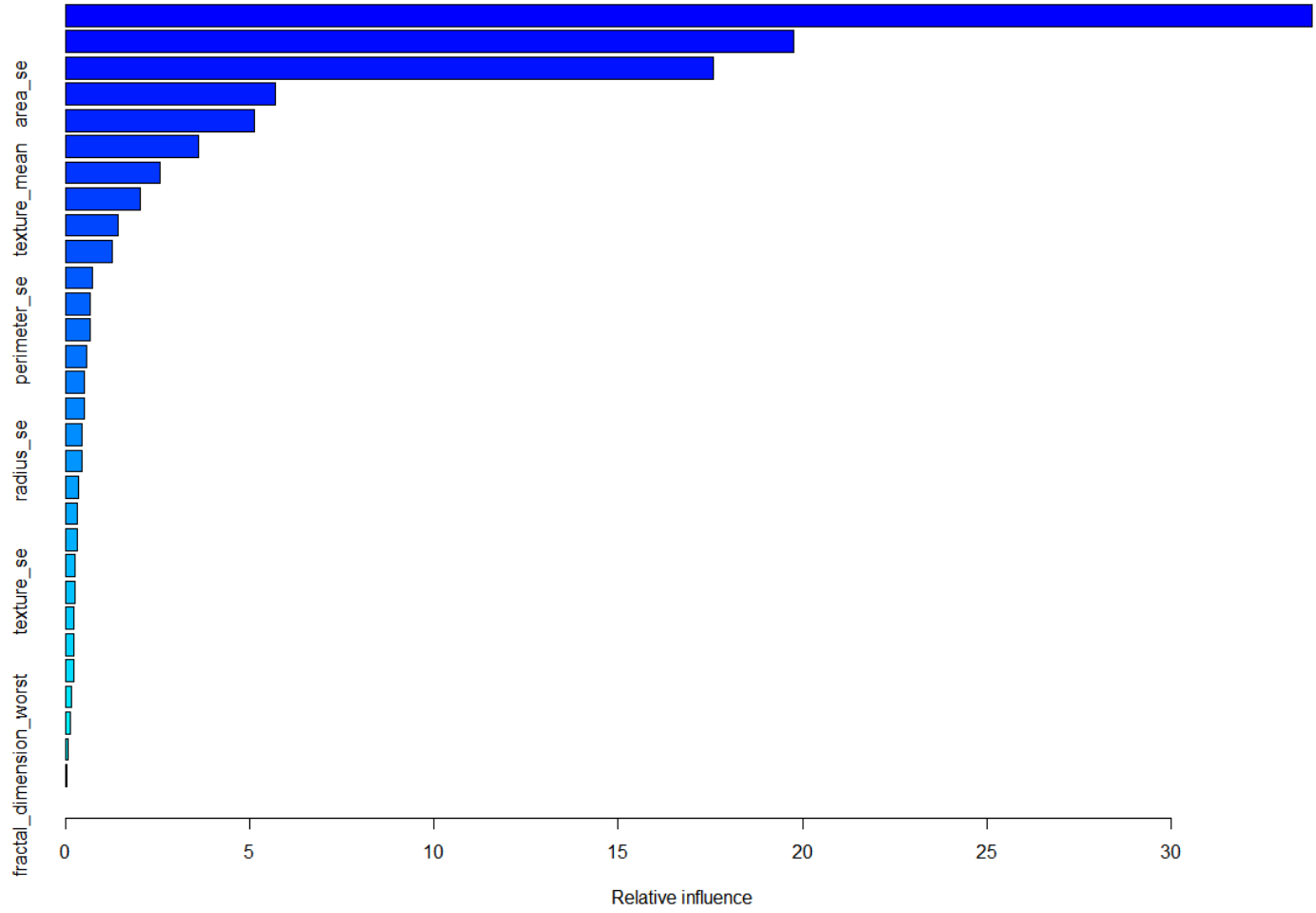
The error rate for bagging is equal to 4.2%, which is slightly less than the error rate obtained from the decision tree. Also, bagging is more accurate as compared to logistic regression and the k-nearest algorithm.

Next, we are going to perform boosting and obtain the error rate. The relative influence plot of the boosting model is given below.

	var	rel.inf
perimeter_worst	perimeter_worst	33.83524304
concave.points_worst	concave.points_worst	19.77023543
concave.points_mean	concave.points_mean	17.56943724
area_se	area_se	5.69864192
radius_worst	radius_worst	5.13480862
texture_worst	texture_worst	3.61502495
area_mean	area_mean	2.57037360
texture_mean	texture_mean	2.02215552
area_worst	area_worst	1.42038032
concavity_worst	concavity_worst	1.25798177
symmetry_worst	symmetry_worst	0.72141195
perimeter_mean	perimeter_mean	0.67085623
perimeter_se	perimeter_se	0.66776068
smoothness_worst	smoothness_worst	0.56421678
symmetry_se	symmetry_se	0.50146354
smoothness_se	smoothness_se	0.49766797
compactness_mean	compactness_mean	0.45768742
radius_se	radius_se	0.44167876
symmetry_mean	symmetry_mean	0.34749178
concavity_mean	concavity_mean	0.33367000
radius_mean	radius_mean	0.31937628
fractal_dimension_mean	fractal_dimension_mean	0.27054003
texture_se	texture_se	0.25495311
fractal_dimension_se	fractal_dimension_se	0.23340292
compactness_se	compactness_se	0.22091604
compactness_worst	compactness_worst	0.21565673
concave.points_se	concave.points_se	0.16886012
concavity_se	concavity_se	0.12636694
smoothness_mean	smoothness_mean	0.06105195
fractal_dimension_worst	fractal_dimension_worst	0.03068834

From the summary above, we observed that perimeter_worst, concave.points_worst, and concave.points_mean are by far the most important variables as they have the highest value of relative influence.

The automatic generated plot for the relative influence plot is also given below.



The confusion matrix of boosting:

BoostPrediction	0	1
0	165	18
1	1	100

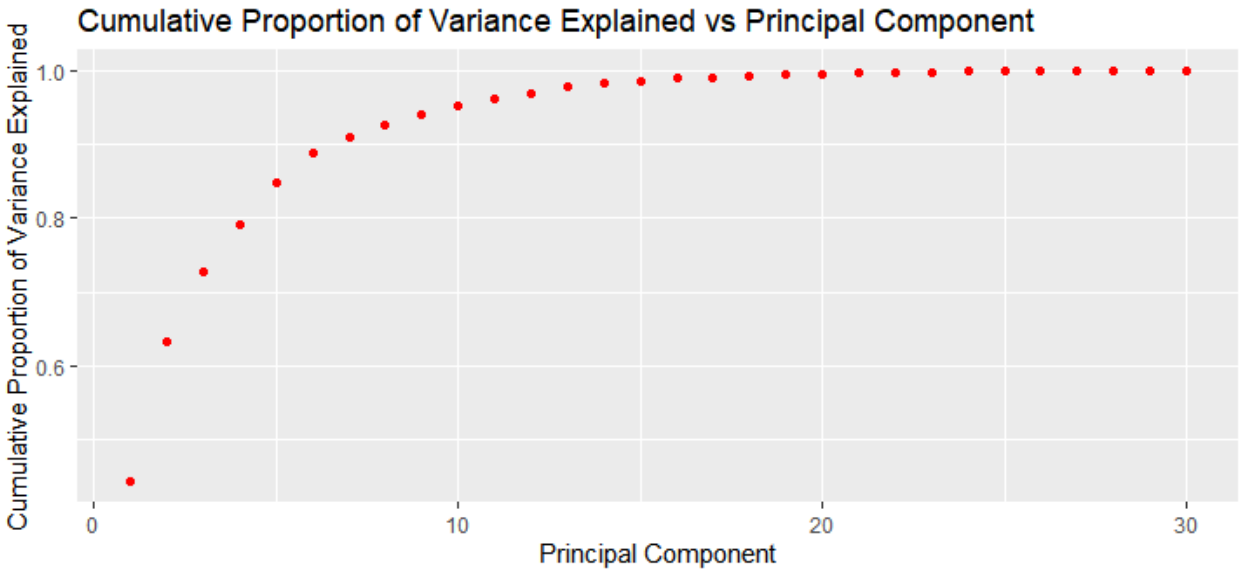
The error rate of 6.7% is obtained while using boosting on our dataset. The accuracy obtained from the boosting is better than the k-nearest algorithm but slightly less as compared to logistic regression, decision tree, and bagging.

Pradip Dhakal

Principal Component Analysis

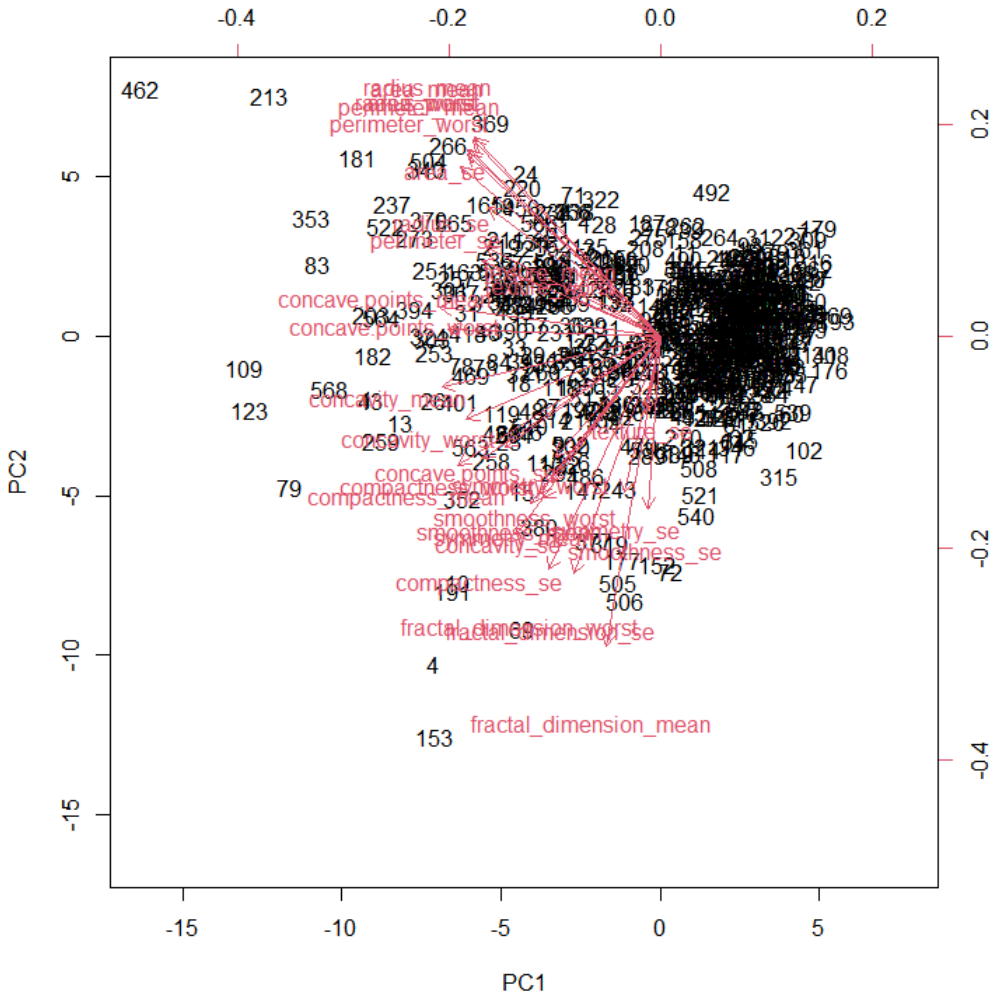
The dataset has 32 columns altogether, so PCA can be used to reduce the dimensions for computation purposes. We will perform PCA in our dataset and make a new dataset. Later, we will run the classification algorithms in this new dataset. We will use four columns of the dataset obtained after performing PCA: PC1, PC2, PC3, and PC4, and we will compare the results.

The Principal Component Analysis is performed, and we created the plot of Principal Component vs. the Cumulative Proportion of Variance Explained, which is given below.



From the plot above, it is observed that around 95% of the variance is explained by the first ten principal components. If we take the first four principal components, it has explained around 80% of the variance.

Also, let’s look at the biplot:



From the biplot, we can see that area_se, perimeter_worst, perimeter_mean_radius_worst, radius_mean, area_mean compactness_se, fractal_dimension_worst fractal_dimension_se, fractal_dimension_mean, and so on has more impact on PC2 while variables such as concave.points_mean, concave,points_worst, concaviity_mean, compactness_means have more impact on PC1.

Now, a new dataset is made by using all the columns of the principal component. Next, we performed logistic regression and k-nearest neighbor algorithm by taking the first four principal components.

Logistic Regression After PCA

As mentioned before, we will use the first four columns of PCA and predict the result.

The summary of the logistic model is given below.

```
Call:
glm(formula = diagnosis ~ PC1 + PC2 + PC3 + PC4, family = binomial,
    data = Data2)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.0469  -0.0765  -0.0095   0.0032   3.7311

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  -0.8889     0.2730  -3.256 0.001129 **
PC1          -2.4656     0.3312  -7.445 9.70e-14 ***
PC2           1.3191     0.2269   5.812 6.17e-09 ***
PC3          -0.6232     0.1640  -3.800 0.000145 ***
PC4          -0.8002     0.2035  -3.933 8.40e-05 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 751.44  on 568  degrees of freedom
Residual deviance: 104.50  on 564  degrees of freedom
AIC: 114.5

Number of Fisher Scoring iterations: 9
```

In this case, we can see that all the four principal component columns are significant to predict the result. The result is predicted, and the confusion matrix is given below.

```
            diagnosis
Prediction2   B   M
          B 351  10
          M   6 202
```

We can see that the result has been improved. Before performing PCA, we get the error rate of 5%, while after performing PCA, the error is reduced to 2.8%.

K-Nearest Neighbor After PCA

We use the same number of dimensions as before and calculate the result using the KNN algorithm. We run this algorithm from for K equals 1 to 25.

```
Error2[ 1 ]:  0.06338028
Error2[ 2 ]:  0.0528169
Error2[ 3 ]:  0.05633803
Error2[ 4 ]:  0.05985915
Error2[ 5 ]:  0.0528169
Error2[ 6 ]:  0.04577465
Error2[ 7 ]:  0.04577465
Error2[ 8 ]:  0.04225352
Error2[ 9 ]:  0.03873239
Error2[ 10 ]:  0.03521127
Error2[ 11 ]:  0.03873239
Error2[ 12 ]:  0.03521127
Error2[ 13 ]:  0.03873239
Error2[ 14 ]:  0.04577465
Error2[ 15 ]:  0.04225352
Error2[ 16 ]:  0.04225352
Error2[ 17 ]:  0.04225352
Error2[ 18 ]:  0.03873239
Error2[ 19 ]:  0.04577465
Error2[ 20 ]:  0.03521127
Error2[ 21 ]:  0.04225352
Error2[ 22 ]:  0.04225352
Error2[ 23 ]:  0.04577465
Error2[ 24 ]:  0.04577465
Error2[ 25 ]:  0.04929577
```

From the observation, it is observed that when k = 10 and 20, we have the least error of 0.035, which is 3.5%. Before using PCA on the dataset, we have the least error of 7.94%, which is now reduced to 3.5%.

K-means Clustering

In the original data, the dataset leads to two categories: either benign or malignant. Now we will try k-means clustering to see if we divide the clusters into two separate groups.

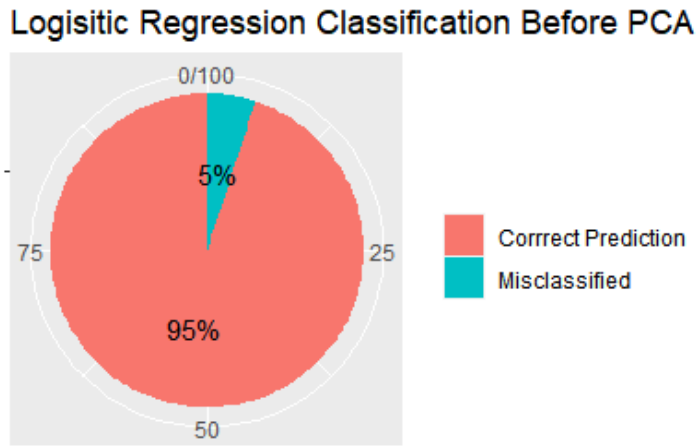
In the original dataset, 212 observations among 569 observations have malignant breast cancer, while the remaining 357 observations had benign breast cancer.

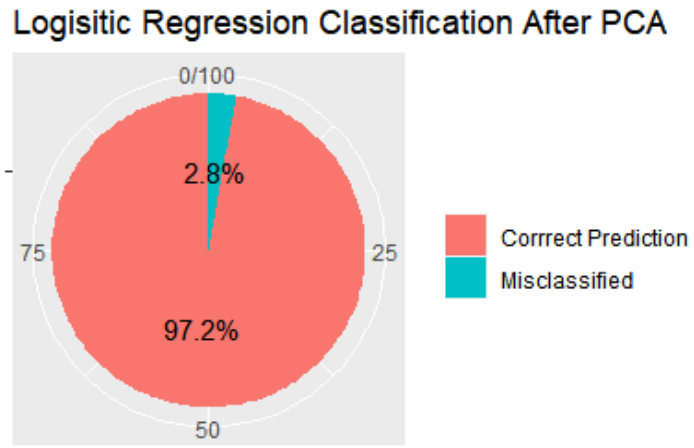
On performing the k-means clustering using k = 2, we got two different clusters with 189 and 380 observations. We observe that 23 observations were misclassified into different clusters while comparing it with the original dataset.

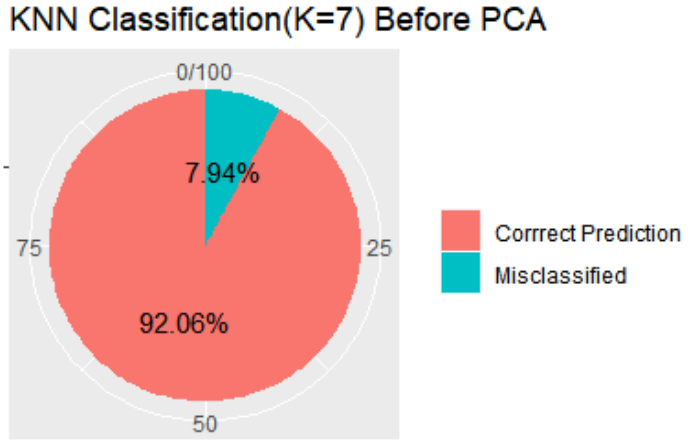
If we calculate the error of this misclassification, we get an error rate of 4%.

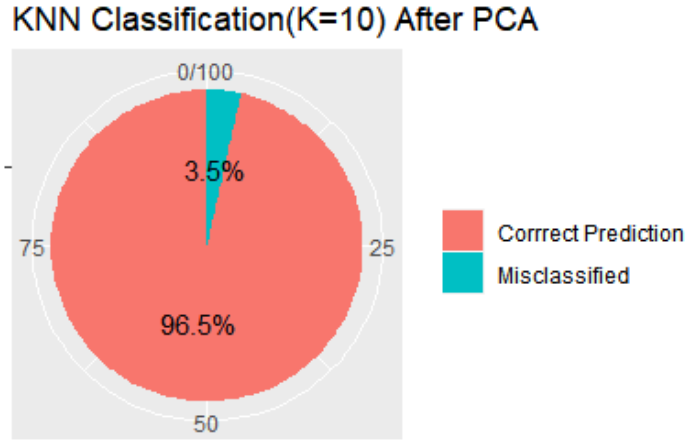
Pradip Dhakal

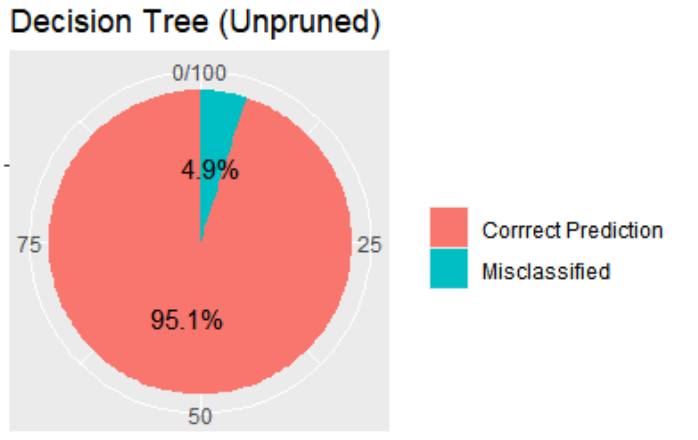
Results

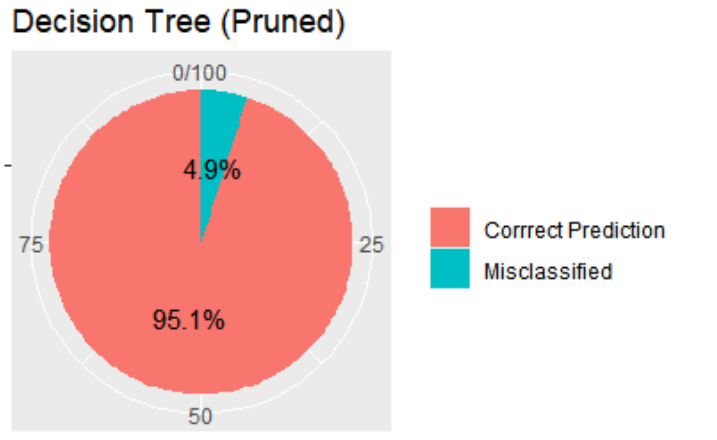


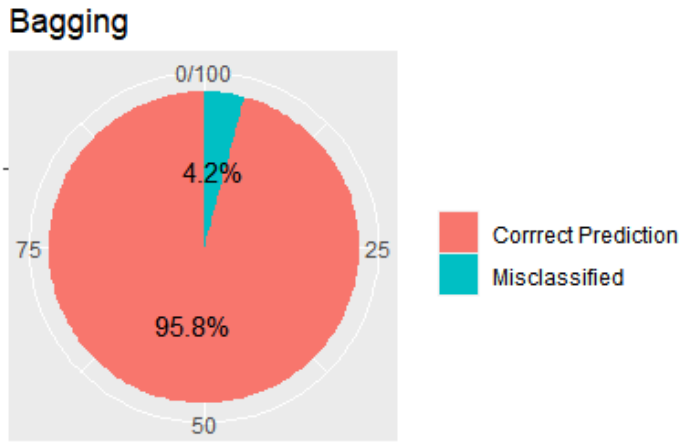


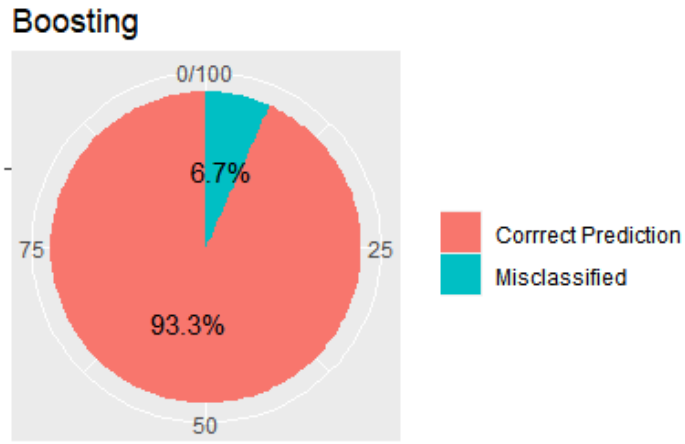












Methods	**Error Rates**
Logistic Regression Before PCA	5%
Logistic Regression After PCA	2.8%
KNN Before PCA	7.94%
KNN After PCA	3.5%
Decision Tree (Unpruned)	4.9%
Decision Tree (Pruned)	4.9%
Bagging	4.2%
Boosting	6.7%

Conclusion

In this project, various algorithms were implemented to see how accurately they can predict the result. We saw the importance of training various models and applying different algorithms to perform the data mining task because we do not have the universal best algorithm for predicting the result. We also observed that performing principal component analysis has a huge advantage in data mining tasks since it reduces the high dimensional data into low dimensions and increases the accuracy rate. In our project, we obtained the best accuracy when we perform principal component analysis and applied logistic regression to predict whether the cancer is benign or malignant. Also, performing k-means clustering on the data obtained after principal component analysis successfully separated the dataset into two different clusters with a very less error rate.

References

James, Gareth, et al. “Introduction to Statistical Learning with Applications in R.” *Introduction to Statistical Learning*, Springer Science + Business Media New York, 2013, faculty.marshall.usc.edu/gareth-james/ISL/.

Learning, UCI Machine. “Breast Cancer Wisconsin (Diagnostic) Data Set.” *Kaggle*, 25 Sept. 2016, www.kaggle.com/uciml/breast-cancer-wisconsin-data.