Breast Cancer Classification Using Different Classification Models

Cancheng Ji, Jiexi Zhou, Sean Cancino

Data Description

- **Data Source:** https://www.kaggle.com/uciml/breast-cancer-wisconsin-data and also https://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+%28Diagnostic%29
- **Description:** To predict whether the breast mass is malignant:
  - Attribute: Diagnosis (M=Malignant B=Benign)
  - Features: The mean, standard error and "worst" or largest of the **10 features**: radius, texture, perimeter, area, smoothness, compactness, concavity, concave points, symmetry, fractal dimension
  - So there are 30 features

Train Test Split

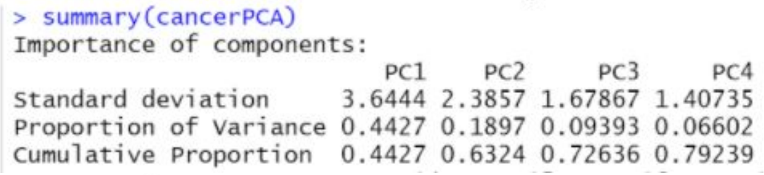
Used 70% training, 30% test, random seed = 202012

Training set contained 398 observations

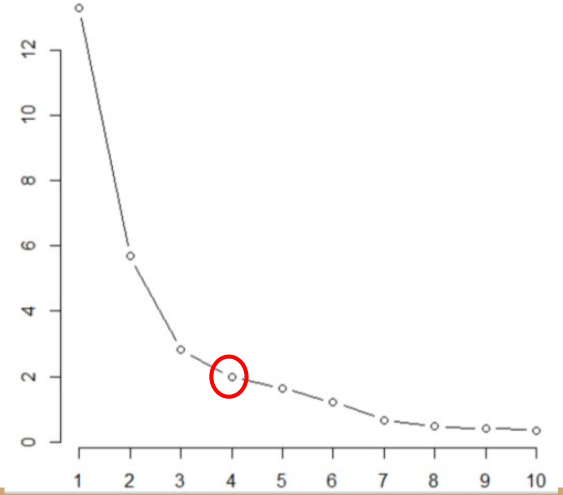
Test set contained 171 observations

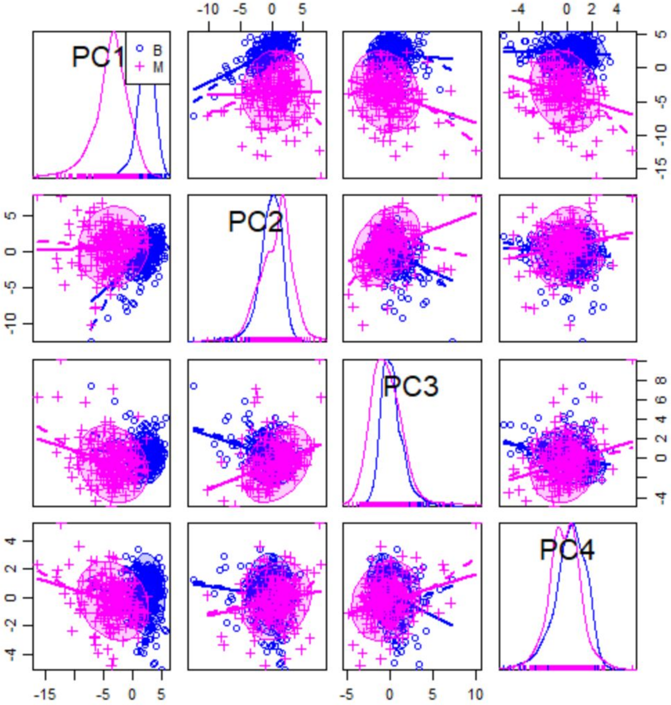
PCA

Used PCA like in midterm to choose optimal principal components. Decided to use the first 4 components.



```
> summary(cancerPCA)
Importance of components:
                          PC1    PC2     PC3     PC4
Standard deviation     3.6444 2.3857 1.67867 1.40735
Proportion of Variance 0.4427 0.1897 0.09393 0.06602
Cumulative Proportion  0.4427 0.6324 0.72636 0.79239
```

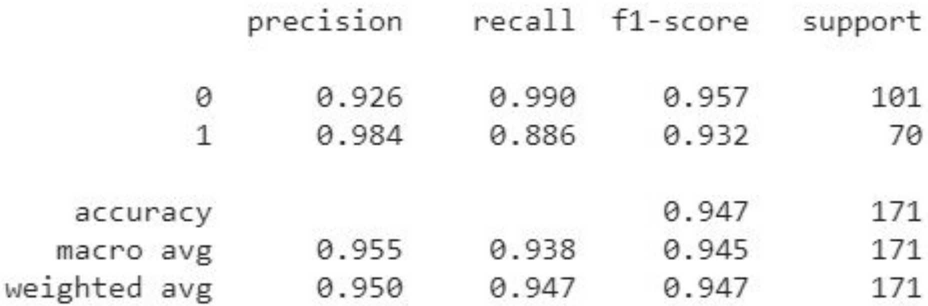




Classification Algorithms Used

- LDA/QDA
- Logistic Regression
- Classification Tree
- Random Forest
- Support Vector Machines
- Neural Network
  - 2 layers, 10 nodes per layer
- Clustering Analysis
  - GMM
  - Hierarchical Clustering
  - K Means

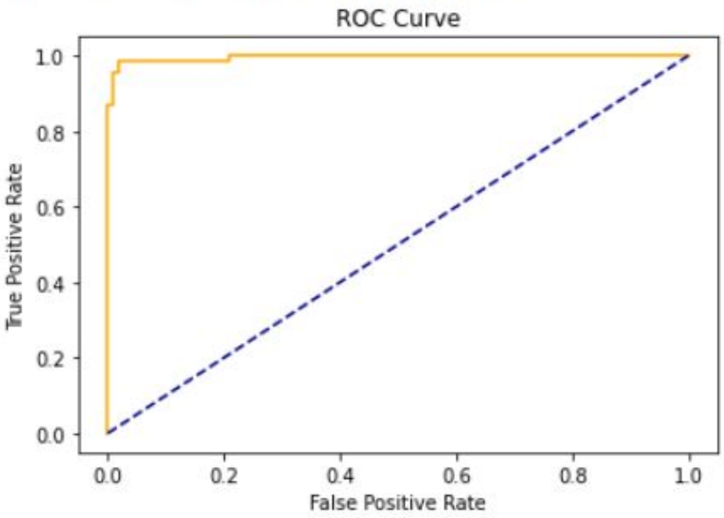
LDA

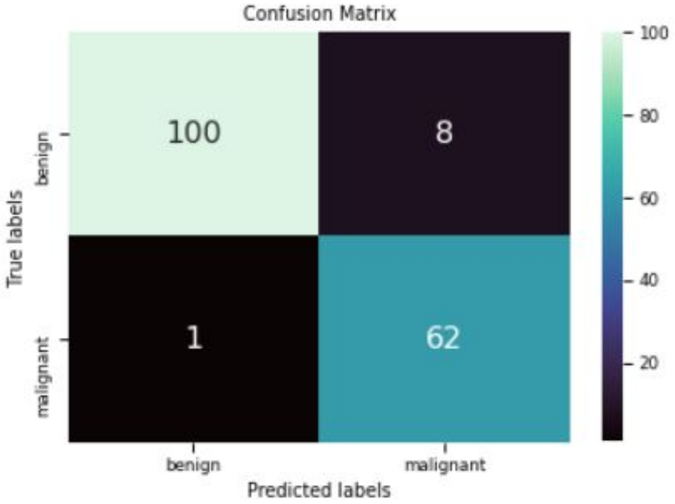


```
              precision    recall  f1-score   support

           0      0.926     0.990     0.957       101
           1      0.984     0.886     0.932        70

    accuracy                          0.947       171
   macro avg      0.955     0.938     0.945       171
weighted avg      0.950     0.947     0.947       171
```

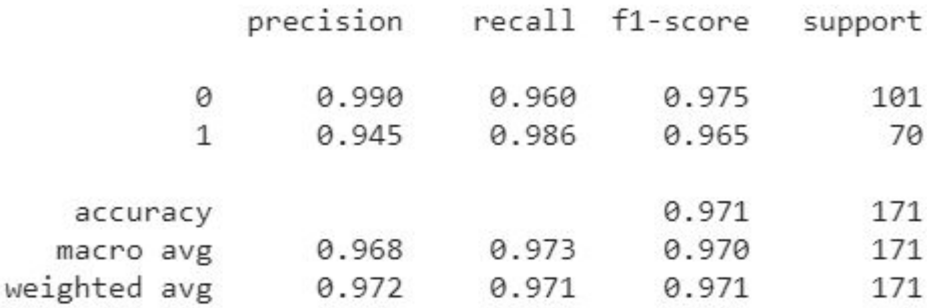
LDA misclassification error rate: 5.263

```
AUC of ROC Curve: 0.9956152758132957
```

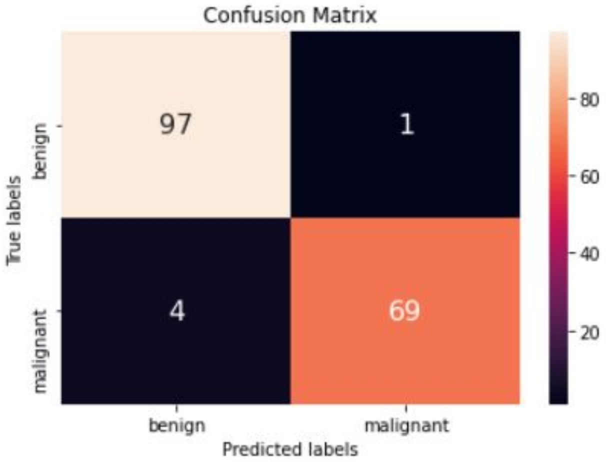




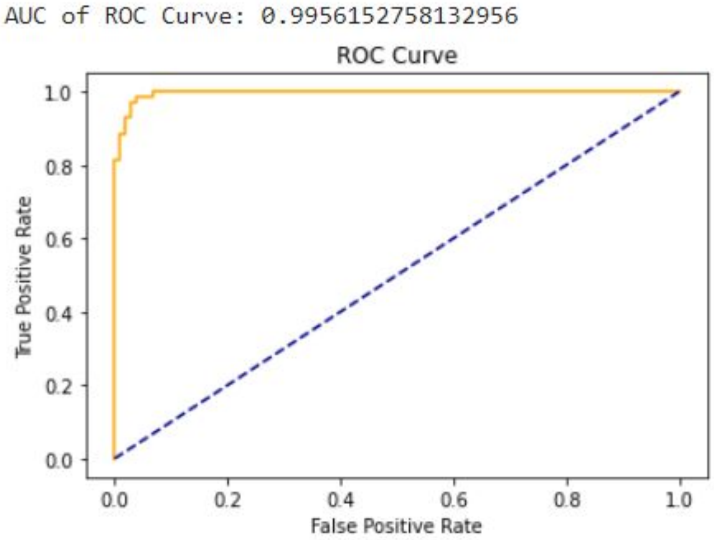
QDA



```
              precision    recall  f1-score   support

           0      0.990     0.960     0.975       101
           1      0.945     0.986     0.965        70

    accuracy                          0.971       171
   macro avg      0.968     0.973     0.970       171
weighted avg      0.972     0.971     0.971       171
```

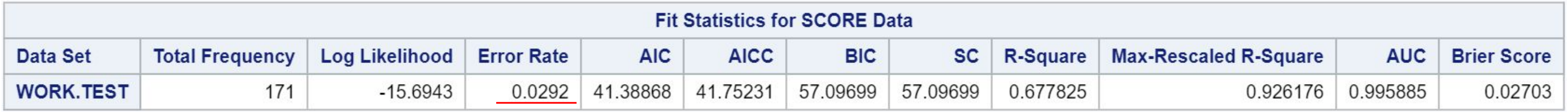


QDA Misclassification Error Rate: 2.92397

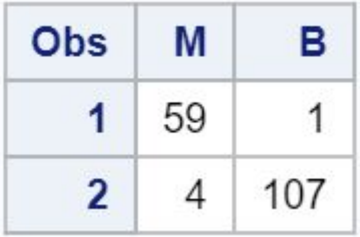


```
AUC of ROC Curve: 0.9956152758132956
```

Logistic Regression with PCA

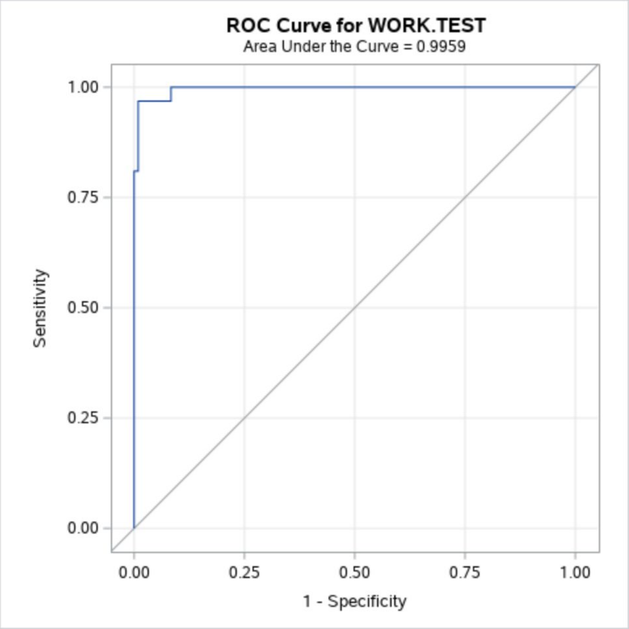


Fit Statistics for SCORE Data											
Data Set	Total Frequency	Log Likelihood	Error Rate	AIC	AICC	BIC	SC	R-Square	Max-Rescaled R-Square	AUC	Brier Score
WORK.TEST	171	-15.6943	0.0292	41.38868	41.75231	57.09699	57.09699	0.677825	0.926176	0.995885	0.02703



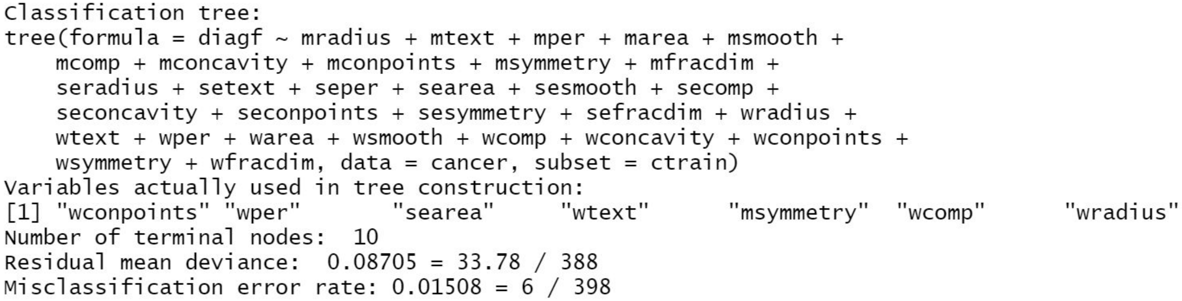
Test result	Obs	M	B
M	1	59	1
B	2	4	107

Misclassification error rate=5/171=0.0292

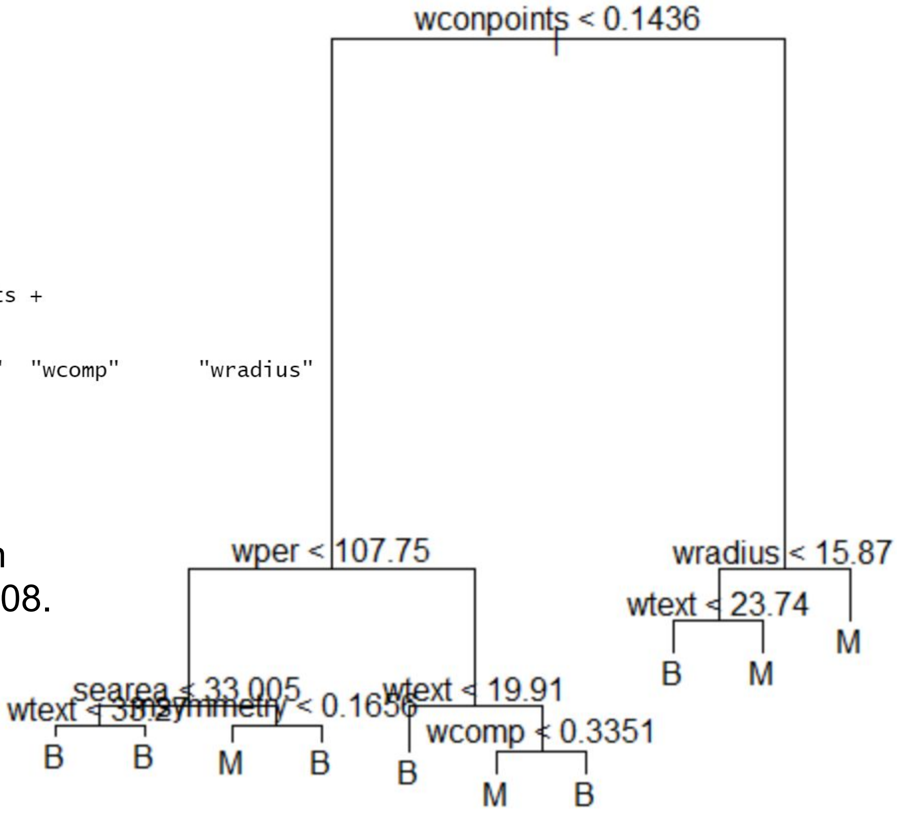


Classification Tree

We generated a classification tree first.



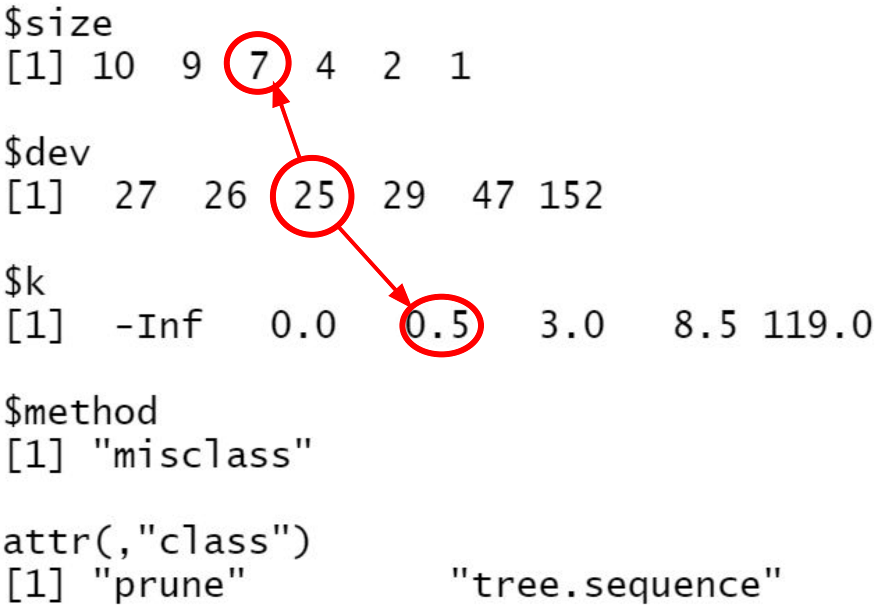
```
Classification tree:
tree(formula = diagf ~ mradius + mtext + mper + marea + msmooth +
    mcomp + mconcavity + mconpoints + msymmetry + mfracdim +
    seradius + setext + seper + searea + sesmooth + secomp +
    seconcavity + seconpoints + sesymmetry + sefracdim + wradius +
    wtext + wper + warea + wsmooth + wcomp + wconcavity + wconpoints +
    wsymmetry + wfracdim, data = cancer, subset = ctrain)
Variables actually used in tree construction:
[1] "wconpoints" "wper"       "searea"     "wtext"      "msymmetry"  "wcomp"      "wradius"
Number of terminal nodes:  10
Residual mean deviance:  0.08705 = 33.78 / 388
Misclassification error rate: 0.01508 = 6 / 398
```



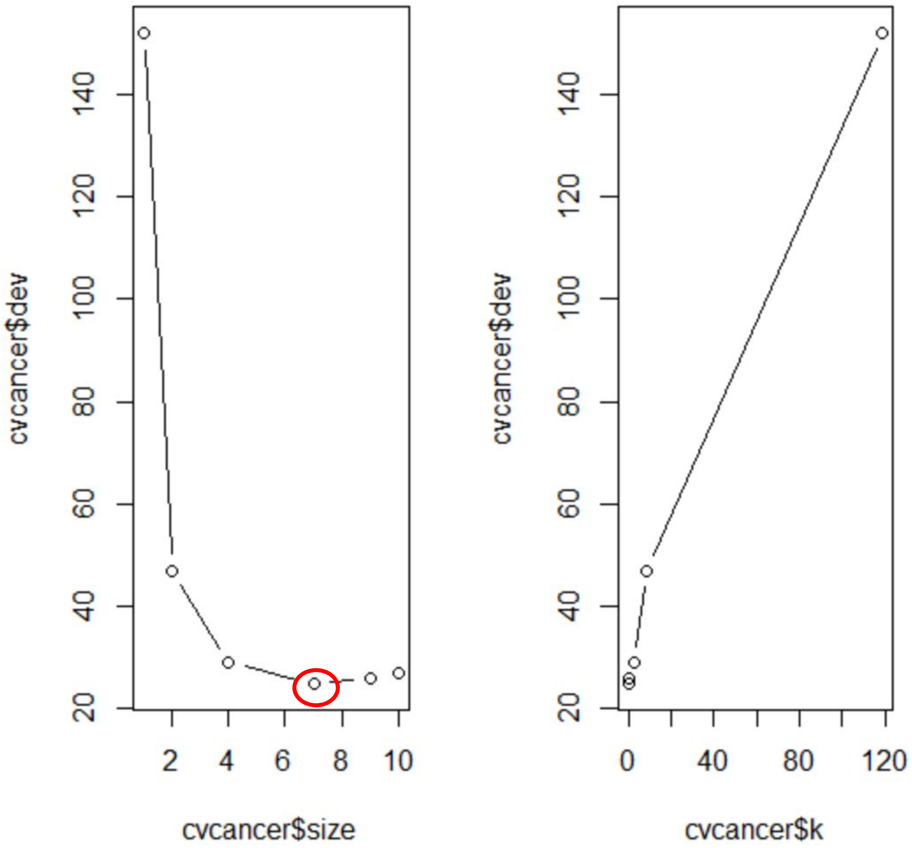
The tree has 7 variables and 10 terminal nodes. In training set, its misclassification error rate is 0.01508.

Classification Tree

Then we try to prune the tree:

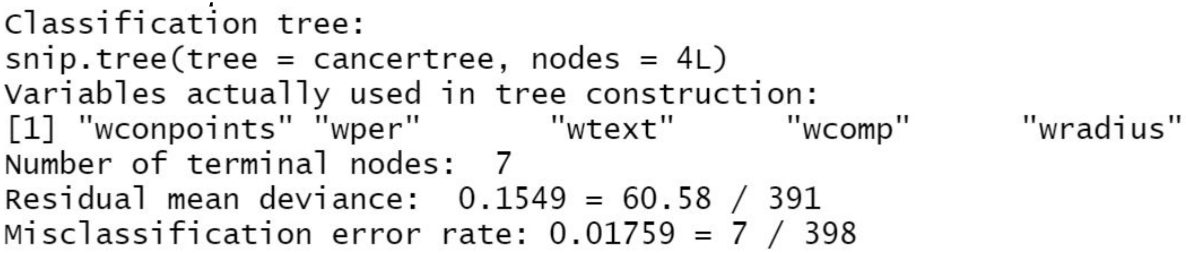


```
$size
[1] 10   9   7   4   2   1

$dev
[1]   27   26   25   29   47 152

$k
[1]  -Inf    0.0    0.5    3.0    8.5 119.0

$method
[1] "misclass"

attr(,"class")
[1] "prune"         "tree.sequence"
```

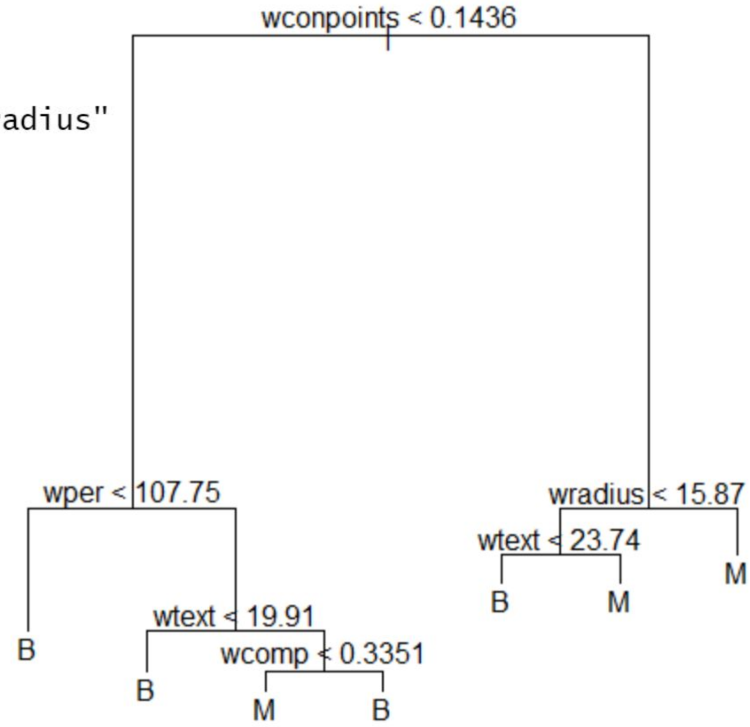


Classification Tree

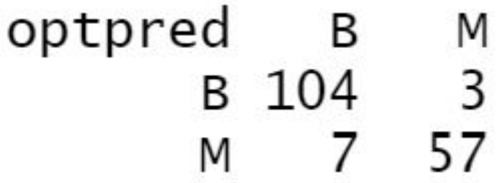
Then to build the optimal classification



```
Classification tree:
snip.tree(tree = cancertree, nodes = 4L)
Variables actually used in tree construction:
[1] "wconpoints" "wper"       "wtext"      "wcomp"      "wradius"
Number of terminal nodes:  7
Residual mean deviance:  0.1549 = 60.58 / 391
Misclassification error rate: 0.01759 = 7 / 398
```



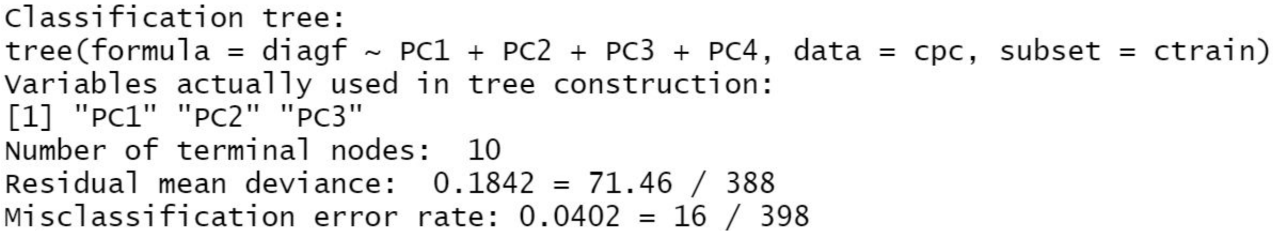
By cross-validation



```
optpred   B   M
      B 104   3
      M   7  57
```

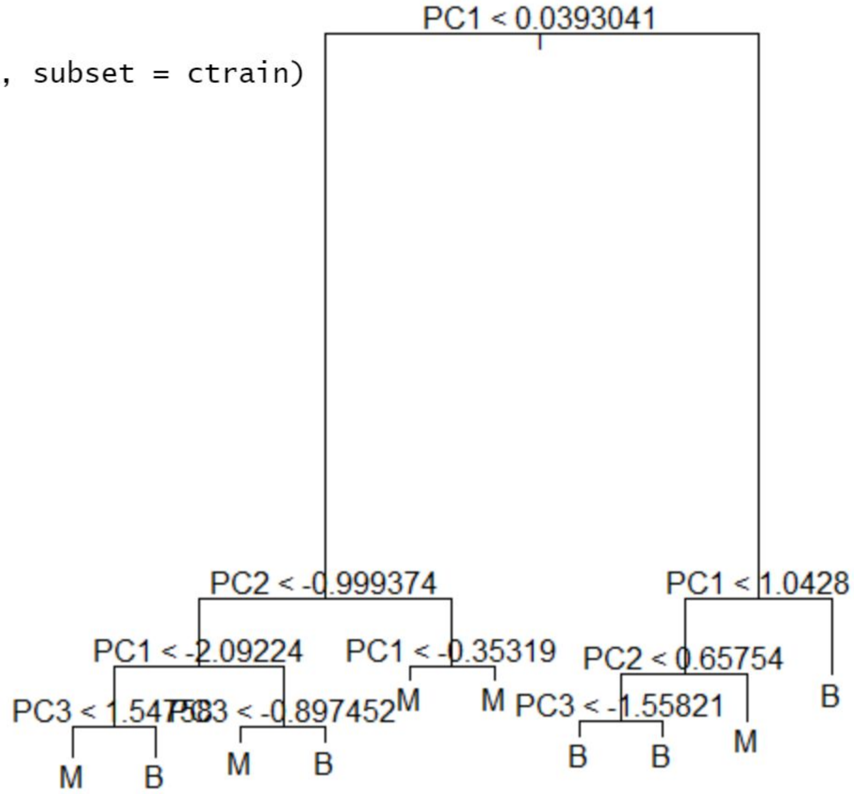
Misclassification error rate=10/171=0.0585

Classification Tree with PC



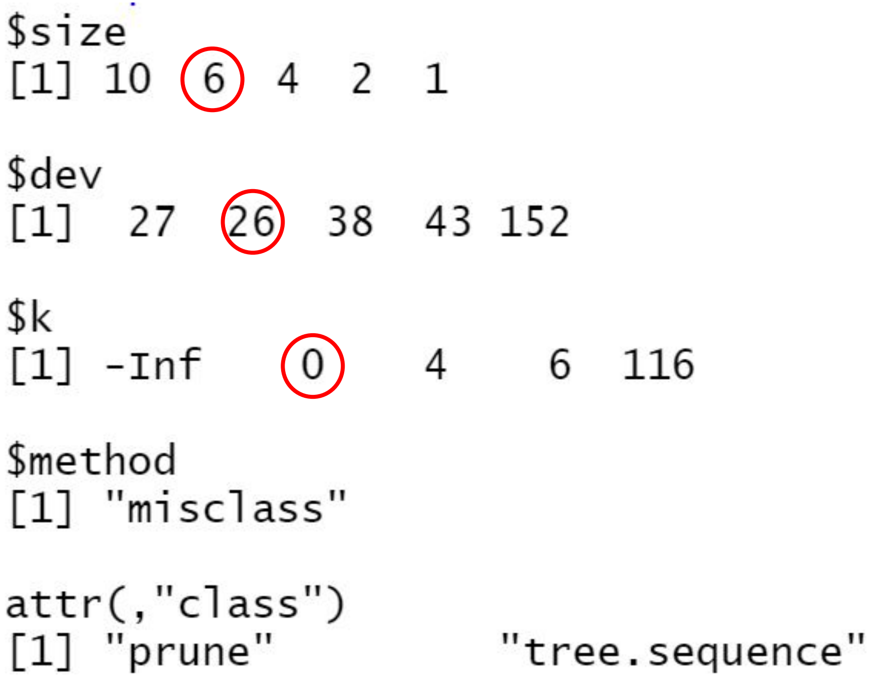
```
Classification tree:
tree(formula = diagf ~ PC1 + PC2 + PC3 + PC4, data = cpc, subset = ctrain)
Variables actually used in tree construction:
[1] "PC1" "PC2" "PC3"
Number of terminal nodes:  10
Residual mean deviance:  0.1842 = 71.46 / 388
Misclassification error rate: 0.0402 = 16 / 398
```

The tree has 3 variables and 10 terminal nodes. In training set, its misclassification error rate is 0.0402.

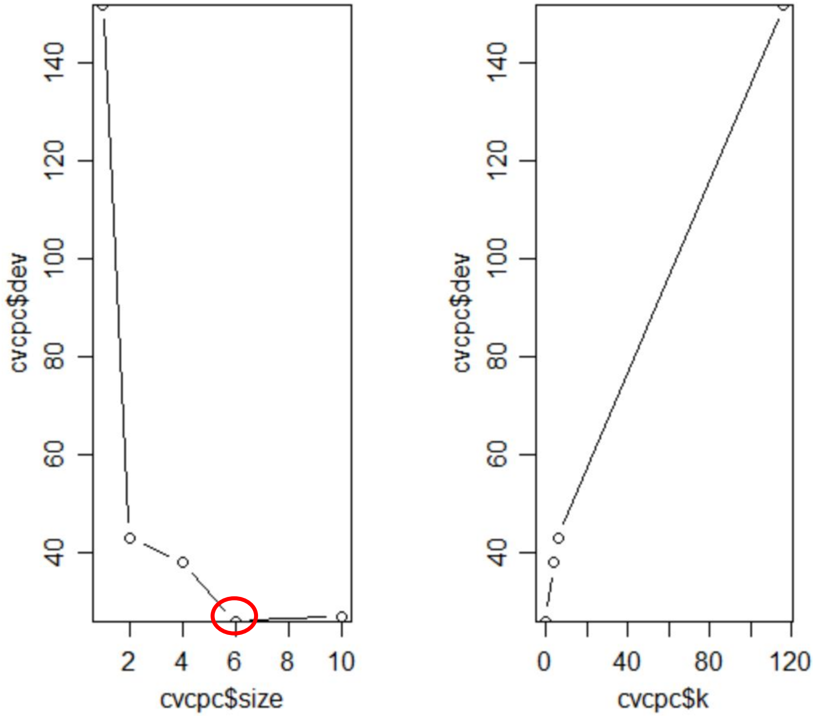


Classification Tree with PC

To prune the tree:



```
$size
[1] 10  6  4  2  1

$dev
[1]  27  26  38  43 152

$k
[1] -Inf    0    4    6  116

$method
[1] "misclass"

attr(,"class")
[1] "prune"         "tree.sequence"
```

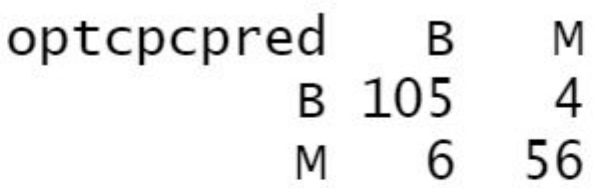


Classification Tree with PC

the optimal classification tree:

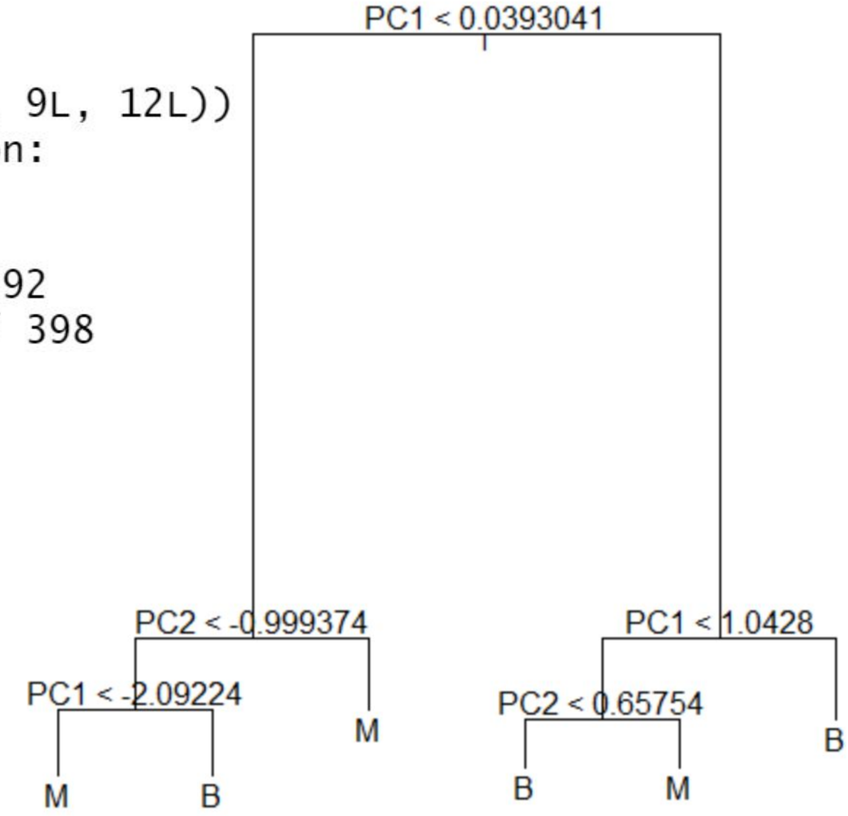
```
Classification tree:
snip.tree(tree = cpctree, nodes = c(5L, 8L, 9L, 12L))
Variables actually used in tree construction:
[1] "PC1" "PC2"
Number of terminal nodes:  6
Residual mean deviance:  0.2765 = 108.4 / 392
Misclassification error rate: 0.0402 = 16 / 398
```

By cross-validation



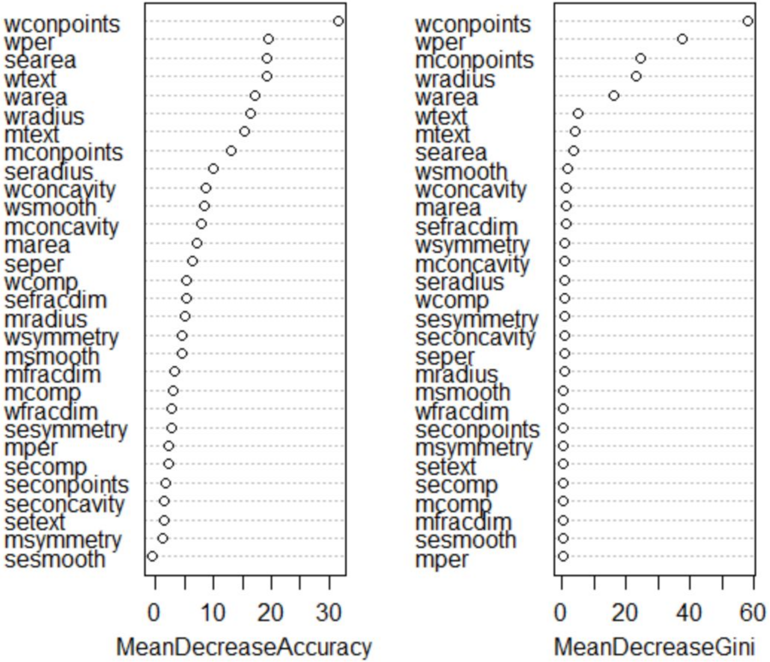
```
optcpcpred   B   M
         B 105   4
         M   6  56
```

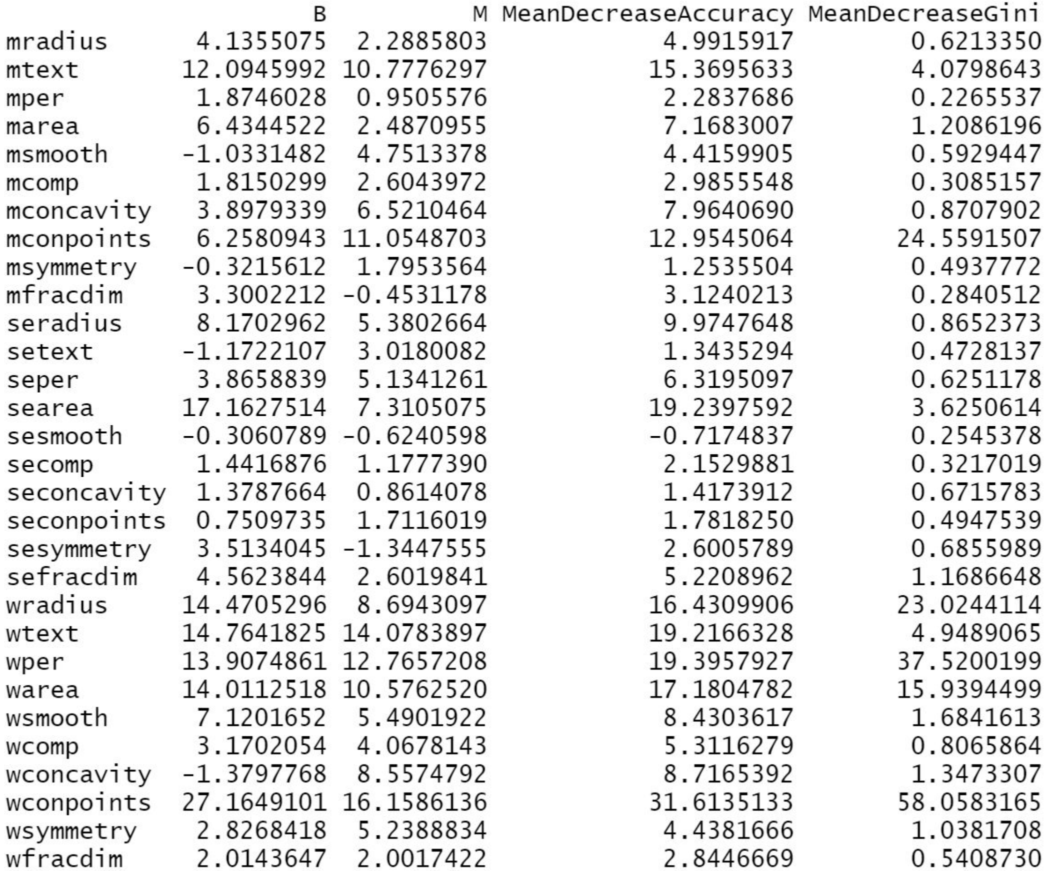
Misclassification error rate=10/171=0.0585



Random Forest

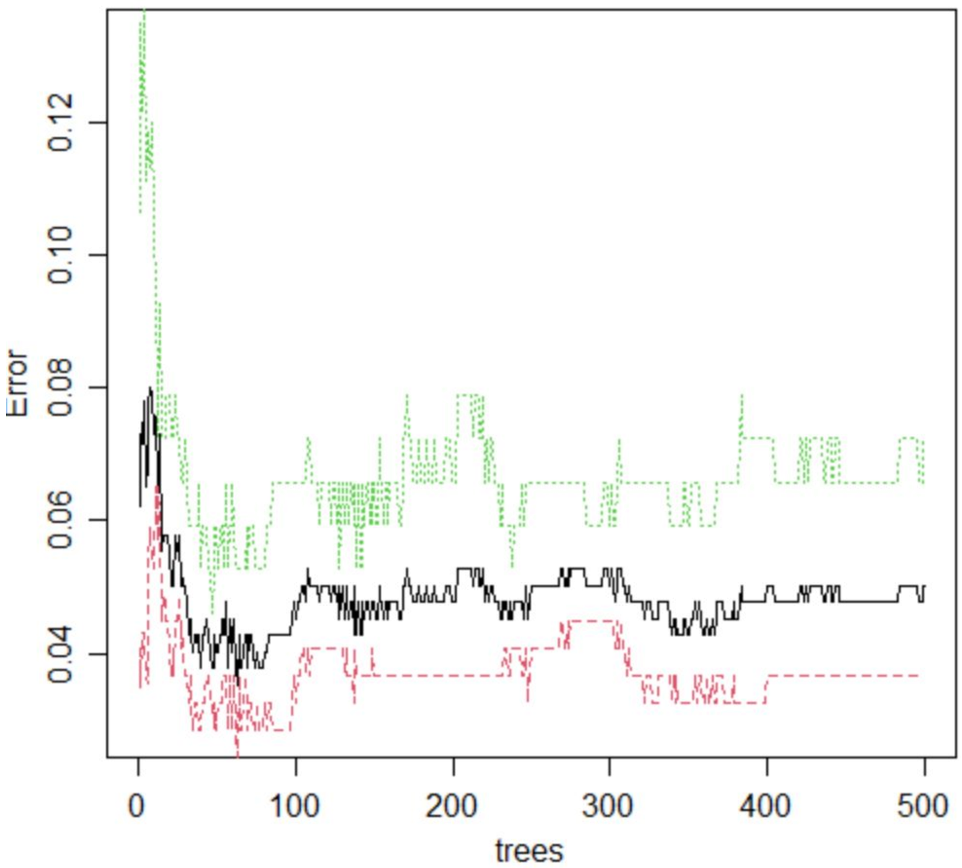
Checking the importance of variables:



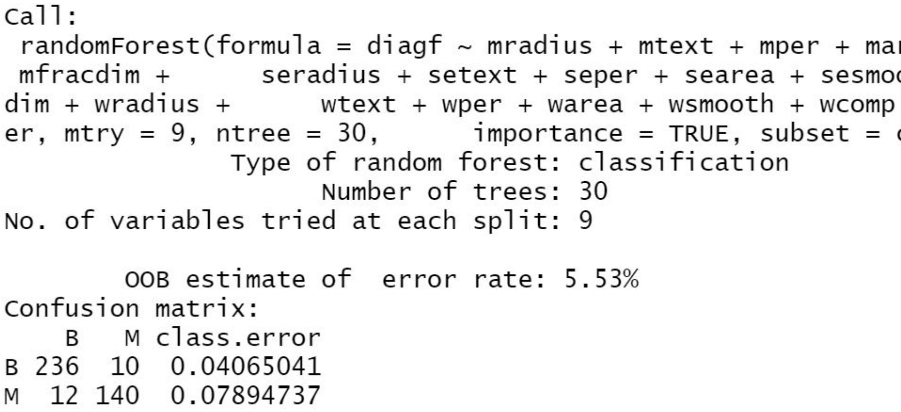

```
                     B           M MeanDecreaseAccuracy MeanDecreaseGini
mradius      4.1355075   2.2885803            4.9915917        0.6213350
mtext       12.0945992  10.7776297           15.3695633        4.0798643
mper         1.8746028   0.9505576            2.2837686        0.2265537
marea        6.4344522   2.4870955            7.1683007        1.2086196
msmooth     -1.0331482   4.7513378            4.4159905        0.5929447
mcomp        1.8150299   2.6043972            2.9855548        0.3085157
mconcavity   3.8979339   6.5210464            7.9640690        0.8707902
mconpoints   6.2580943  11.0548703           12.9545064       24.5591507
msymmetry   -0.3215612   1.7953564            1.2535504        0.4937772
mfracdim     3.3002212  -0.4531178            3.1240213        0.2840512
seradius     8.1702962   5.3802664            9.9747648        0.8652373
setext      -1.1722107   3.0180082            1.3435294        0.4728137
seper        3.8658839   5.1341261            6.3195097        0.6251178
searea      17.1627514   7.3105075           19.2397592        3.6250614
sesmooth    -0.3060789  -0.6240598           -0.7174837        0.2545378
secomp       1.4416876   1.1777390            2.1529881        0.3217019
seconcavity  1.3787664   0.8614078            1.4173912        0.6715783
seconpoints  0.7509735   1.7116019            1.7818250        0.4947539
sesymmetry   3.5134045  -1.3447555            2.6005789        0.6855989
sefracdim    4.5623844   2.6019841            5.2208962        1.1686648
wradius     14.4705296   8.6943097           16.4309906       23.0244114
wtext       14.7641825  14.0783897           19.2166328        4.9489065
wper        13.9074861  12.7657208           19.3957927       37.5200199
warea       14.0112518  10.5762520           17.1804782       15.9394499
wsmooth      7.1201652   5.4901922            8.4303617        1.6841613
wcomp        3.1702054   4.0678143            5.3116279        0.8065864
wconcavity  -1.3797768   8.5574792            8.7165392        1.3473307
wconpoints  27.1649101  16.1586136           31.6135133       58.0583165
wsymmetry    2.8268418   5.2388834            4.4381666        1.0381708
wfracdim     2.0143647   2.0017422            2.8446669        0.5408730
```

Random Forest

oob error versus number of trees

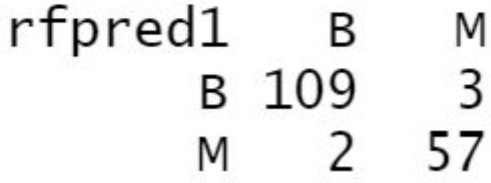


The final model



```
Call:
 randomForest(formula = diagf ~ mradius + mtext + mper + mar
 mfracdim +       seradius + setext + seper + searea + sesmoo
dim + wradius +       wtext + wper + warea + wsmooth + wcomp
er, mtry = 9, ntree = 30,       importance = TRUE, subset = c
               Type of random forest: classification
                     Number of trees: 30
No. of variables tried at each split: 9

        OOB estimate of  error rate: 5.53%
Confusion matrix:
    B   M class.error
B 236  10  0.04065041
M  12 140  0.07894737
```

It has 30 trees and 9 variables tried at each split.
And for the test group,

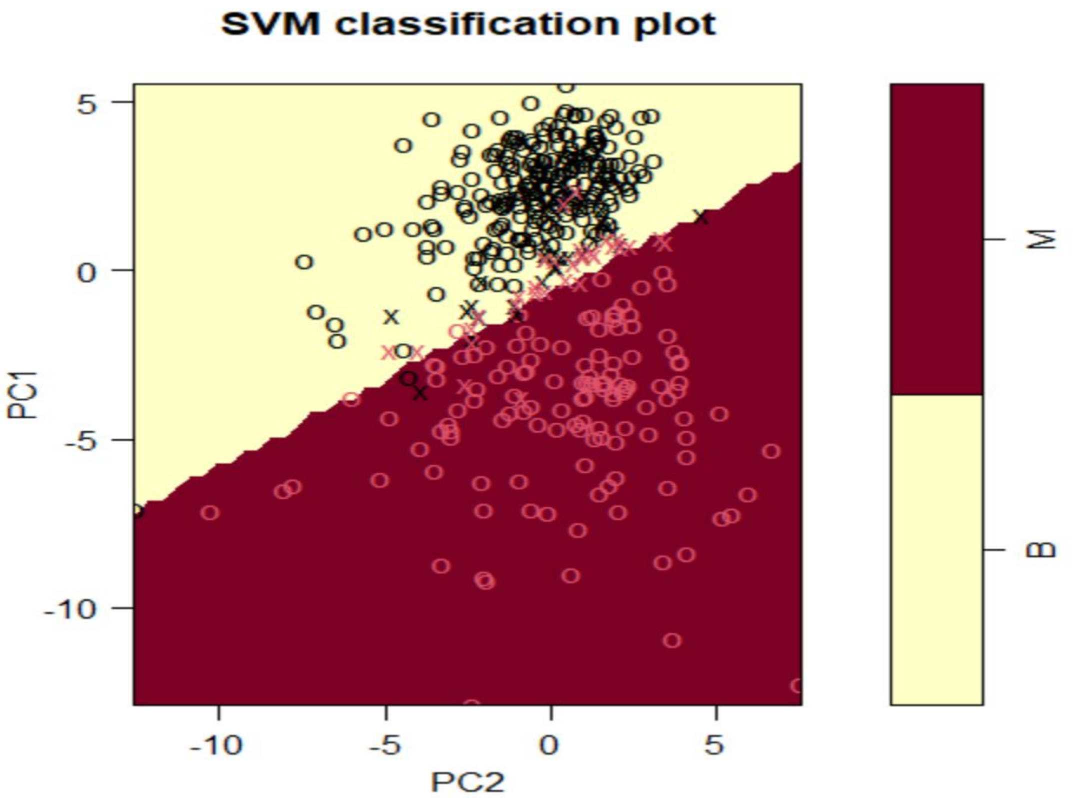


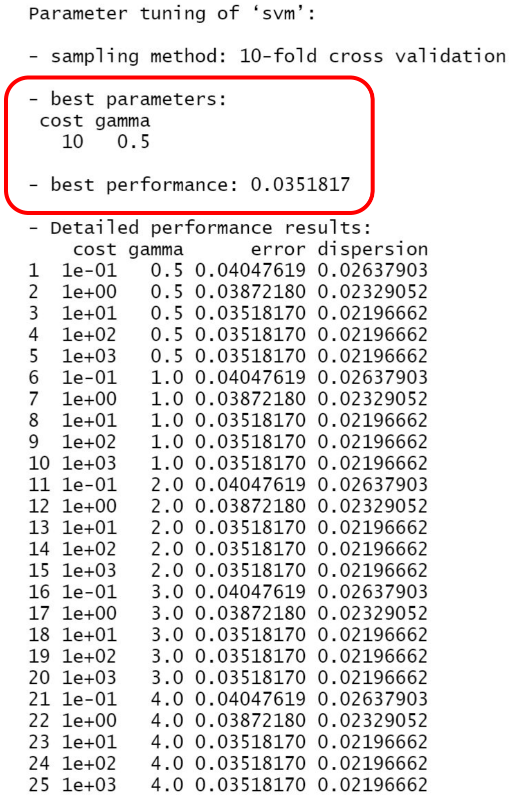
```
rfpred1   B   M
      B 109   3
      M   2  57
```

Misclassification error rate=5/171=0.0292

Support Vector Machine with PC

linear kernel

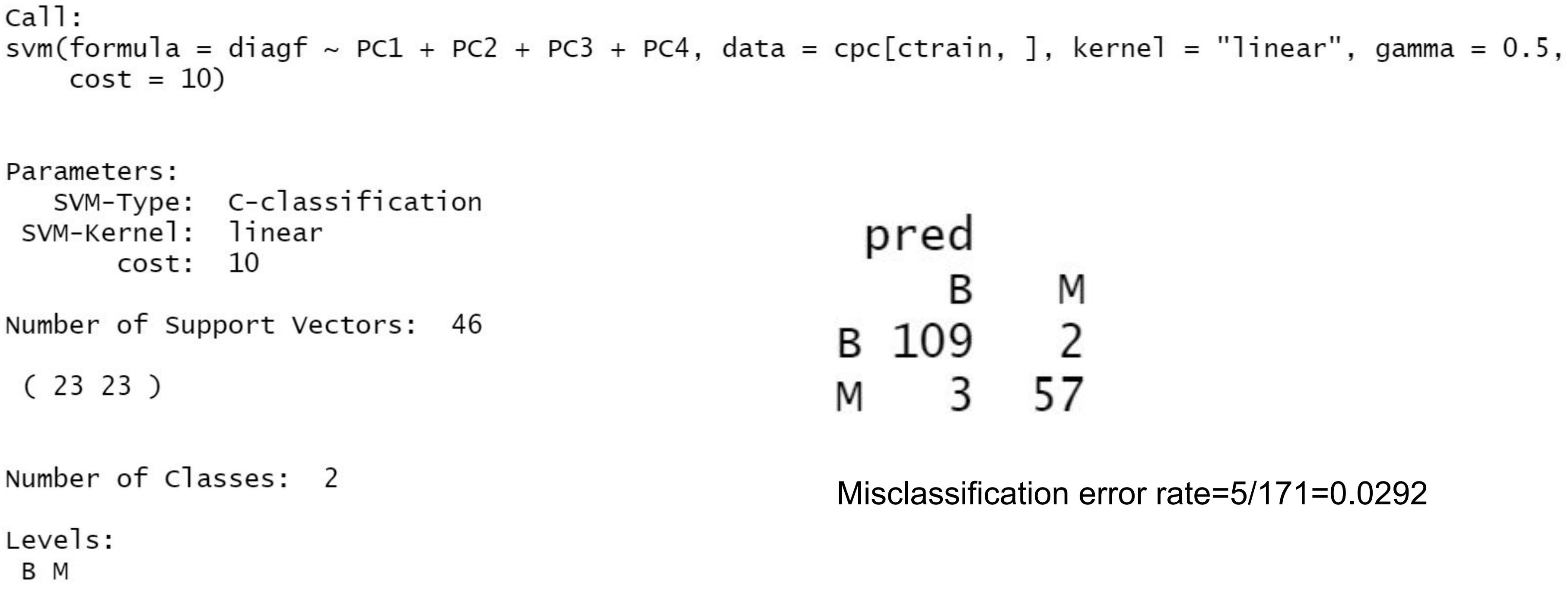




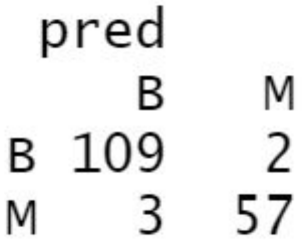
```
Parameter tuning of ‘svm’:

- sampling method: 10-fold cross validation

- best parameters:
 cost gamma
   10   0.5

- best performance: 0.0351817

- Detailed performance results:
    cost gamma      error dispersion
1  1e-01   0.5 0.04047619 0.02637903
2  1e+00   0.5 0.03872180 0.02329052
3  1e+01   0.5 0.03518170 0.02196662
4  1e+02   0.5 0.03518170 0.02196662
5  1e+03   0.5 0.03518170 0.02196662
6  1e-01   1.0 0.04047619 0.02637903
7  1e+00   1.0 0.03872180 0.02329052
8  1e+01   1.0 0.03518170 0.02196662
9  1e+02   1.0 0.03518170 0.02196662
10 1e+03   1.0 0.03518170 0.02196662
11 1e-01   2.0 0.04047619 0.02637903
12 1e+00   2.0 0.03872180 0.02329052
13 1e+01   2.0 0.03518170 0.02196662
14 1e+02   2.0 0.03518170 0.02196662
15 1e+03   2.0 0.03518170 0.02196662
16 1e-01   3.0 0.04047619 0.02637903
17 1e+00   3.0 0.03872180 0.02329052
18 1e+01   3.0 0.03518170 0.02196662
19 1e+02   3.0 0.03518170 0.02196662
20 1e+03   3.0 0.03518170 0.02196662
21 1e-01   4.0 0.04047619 0.02637903
22 1e+00   4.0 0.03872180 0.02329052
23 1e+01   4.0 0.03518170 0.02196662
24 1e+02   4.0 0.03518170 0.02196662
25 1e+03   4.0 0.03518170 0.02196662
```

Support Vector Machine with PC

linear kernel best model



```
Call:
svm(formula = diagf ~ PC1 + PC2 + PC3 + PC4, data = cpc[ctrain, ], kernel = "linear", gamma = 0.5,
    cost = 10)


Parameters:
   SVM-Type:  C-classification
 SVM-Kernel:  linear
       cost:  10

Number of Support Vectors:  46

 ( 23 23 )


Number of Classes:  2

Levels:
 B M
```

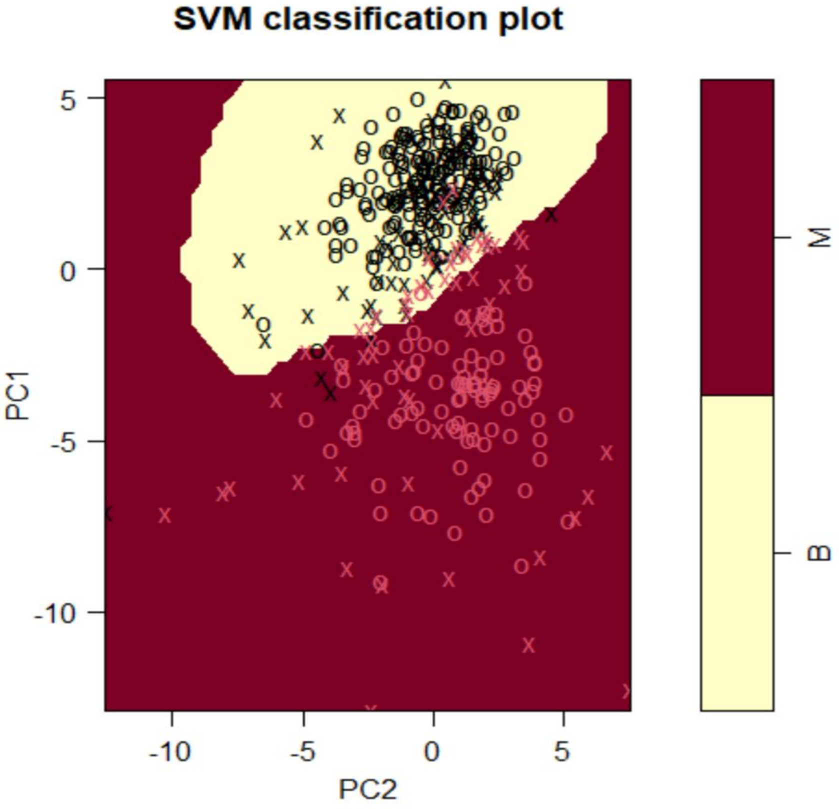


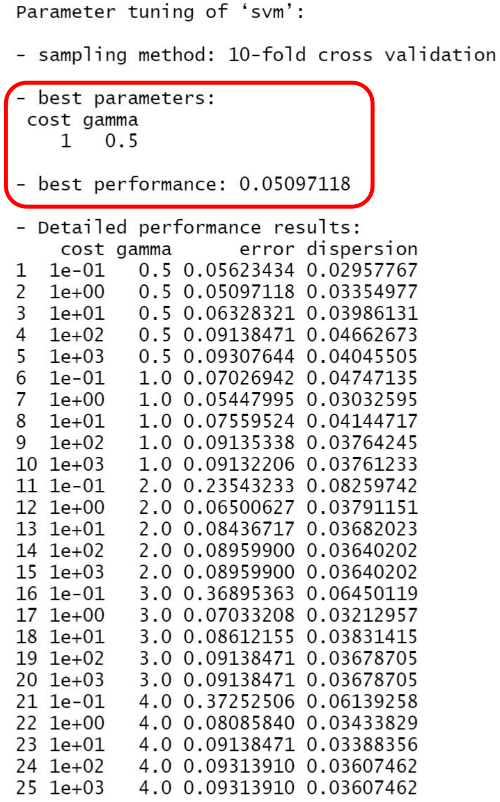
```
   pred
      B   M
B   109   2
M     3  57
```

Misclassification error rate=5/171=0.0292

Support Vector Machine with PC

radial kernel

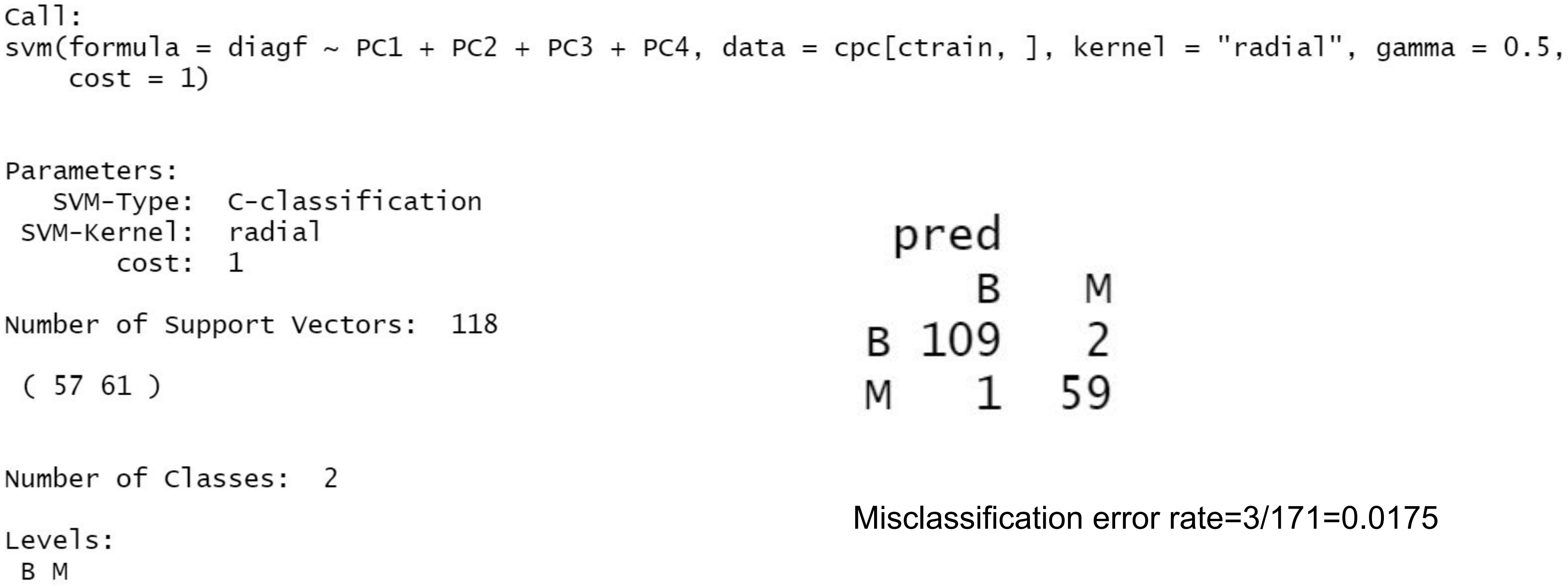




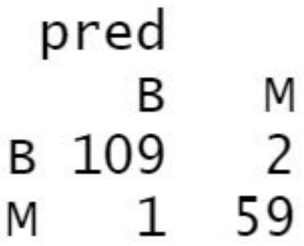
```
Parameter tuning of ‘svm’:

- sampling method: 10-fold cross validation

- best parameters:
 cost gamma
    1   0.5

- best performance: 0.05097118

- Detailed performance results:
    cost gamma      error dispersion
1  1e-01   0.5 0.05623434 0.02957767
2  1e+00   0.5 0.05097118 0.03354977
3  1e+01   0.5 0.06328321 0.03986131
4  1e+02   0.5 0.09138471 0.04662673
5  1e+03   0.5 0.09307644 0.04045505
6  1e-01   1.0 0.07026942 0.04747135
7  1e+00   1.0 0.05447995 0.03032595
8  1e+01   1.0 0.07559524 0.04144717
9  1e+02   1.0 0.09135338 0.03764245
10 1e+03   1.0 0.09132206 0.03761233
11 1e-01   2.0 0.23543233 0.08259742
12 1e+00   2.0 0.06500627 0.03791151
13 1e+01   2.0 0.08436717 0.03682023
14 1e+02   2.0 0.08959900 0.03640202
15 1e+03   2.0 0.08959900 0.03640202
16 1e-01   3.0 0.36895363 0.06450119
17 1e+00   3.0 0.07033208 0.03212957
18 1e+01   3.0 0.08612155 0.03831415
19 1e+02   3.0 0.09138471 0.03678705
20 1e+03   3.0 0.09138471 0.03678705
21 1e-01   4.0 0.37252506 0.06139258
22 1e+00   4.0 0.08085840 0.03433829
23 1e+01   4.0 0.09138471 0.03388356
24 1e+02   4.0 0.09313910 0.03607462
25 1e+03   4.0 0.09313910 0.03607462
```

Support Vector Machine with PC

radial kernel best model



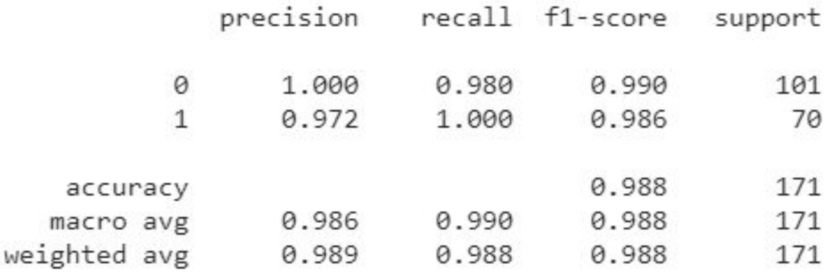
```
Call:
svm(formula = diagf ~ PC1 + PC2 + PC3 + PC4, data = cpc[ctrain, ], kernel = "radial", gamma = 0.5,
    cost = 1)


Parameters:
   SVM-Type:  C-classification
 SVM-Kernel:  radial
       cost:  1

Number of Support Vectors:  118

 ( 57 61 )


Number of Classes:  2

Levels:
 B M
```



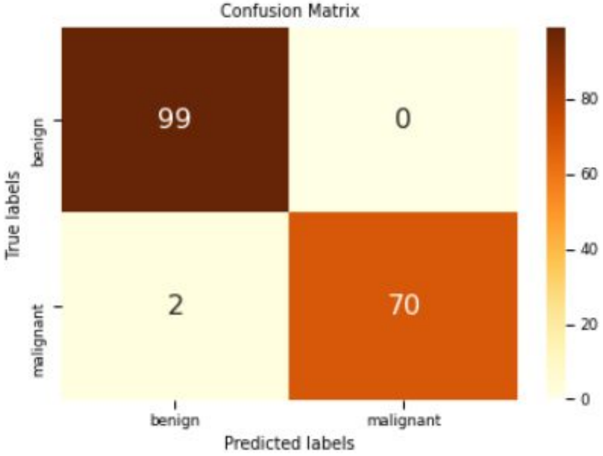
```
    pred
       B   M
 B   109   2
 M     1  59
```

Misclassification error rate=3/171=0.0175

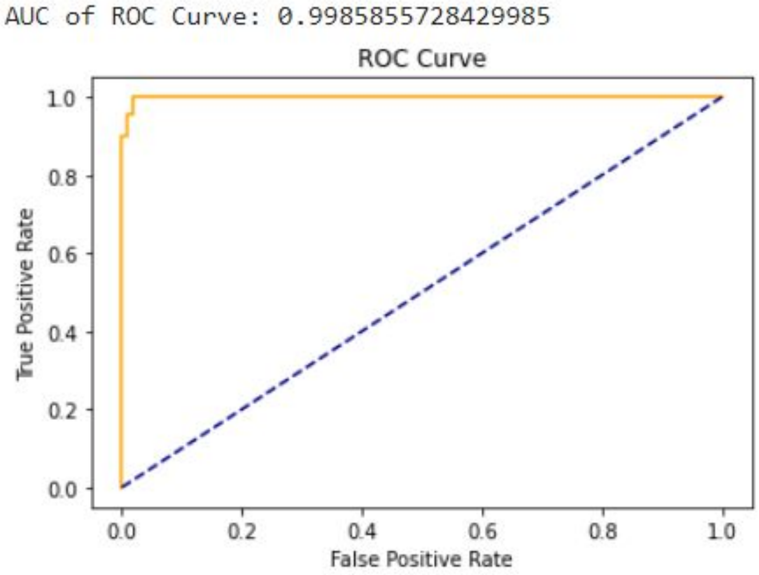
Neural Network



```
              precision    recall  f1-score   support

           0      1.000     0.980     0.990       101
           1      0.972     1.000     0.986        70

    accuracy                          0.988       171
   macro avg      0.986     0.990     0.988       171
weighted avg      0.989     0.988     0.988       171
```

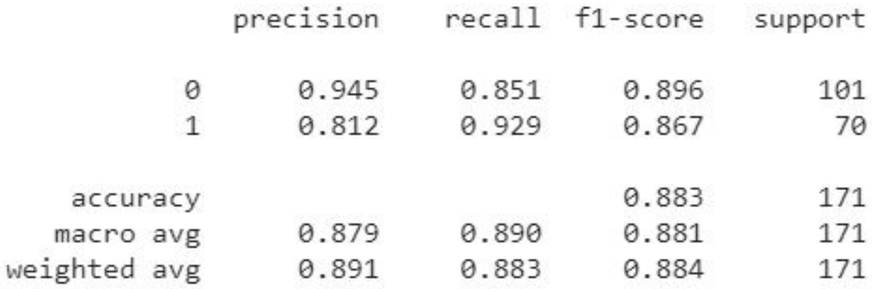


Neural Network Misclassification Error Rate: 1.169

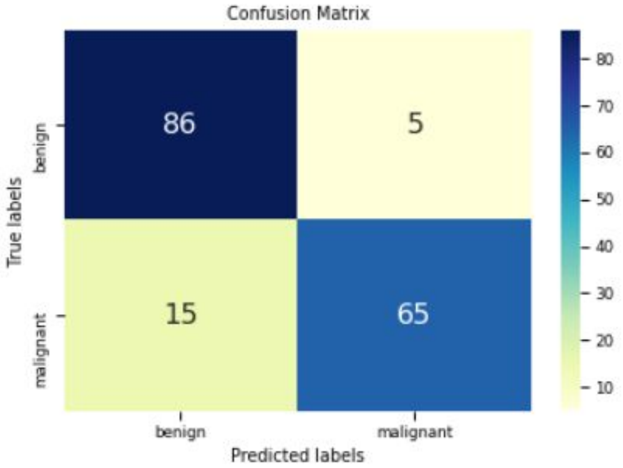


```
AUC of ROC Curve: 0.9985855728429985
```

Gaussian Mixture Model

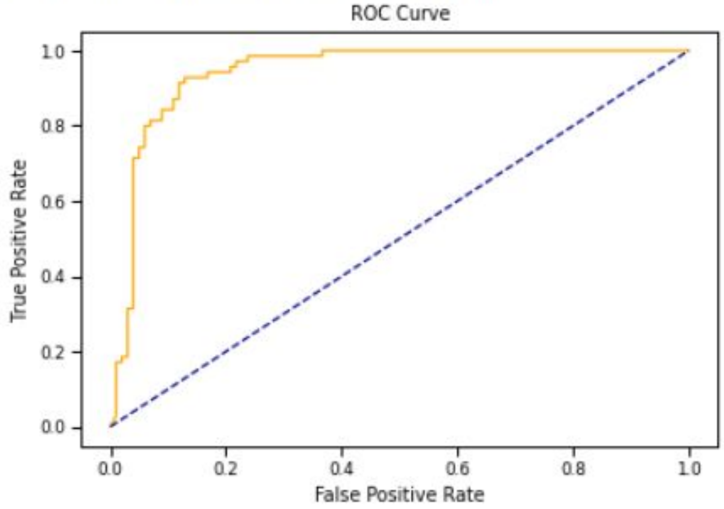


```
              precision    recall  f1-score   support

           0      0.945     0.851     0.896       101
           1      0.812     0.929     0.867        70

    accuracy                          0.883       171
   macro avg      0.879     0.890     0.881       171
weighted avg      0.891     0.883     0.884       171
```

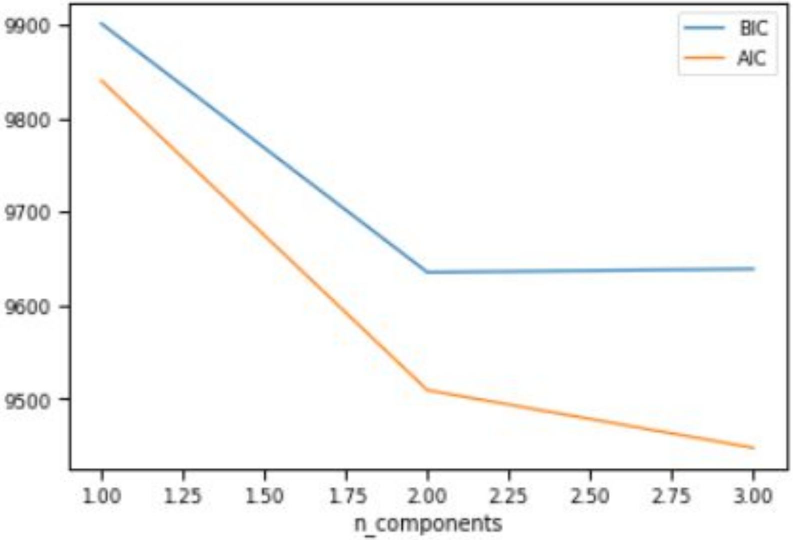


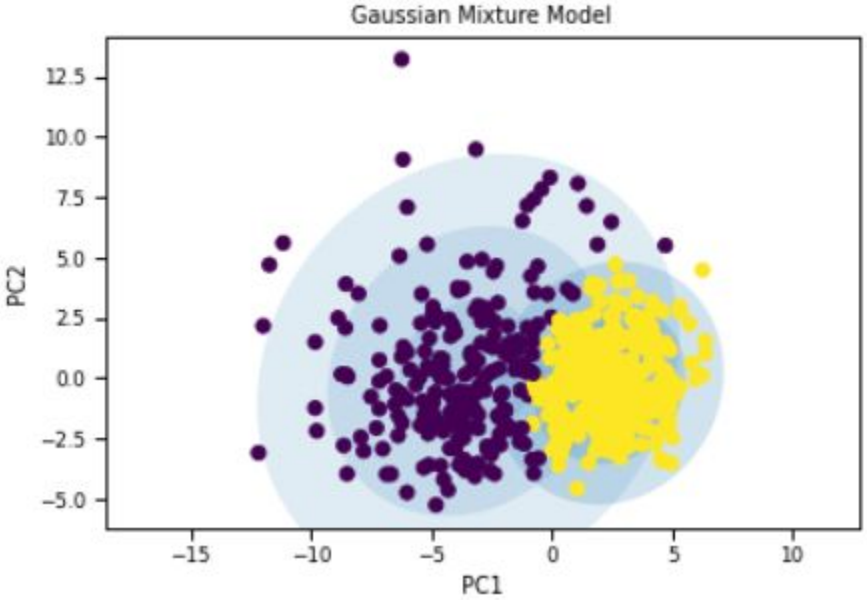
GMM misclassification error rate: 11.7

```
AUC of ROC Curve: 0.942998585572843
```



Gaussian Mixture Model

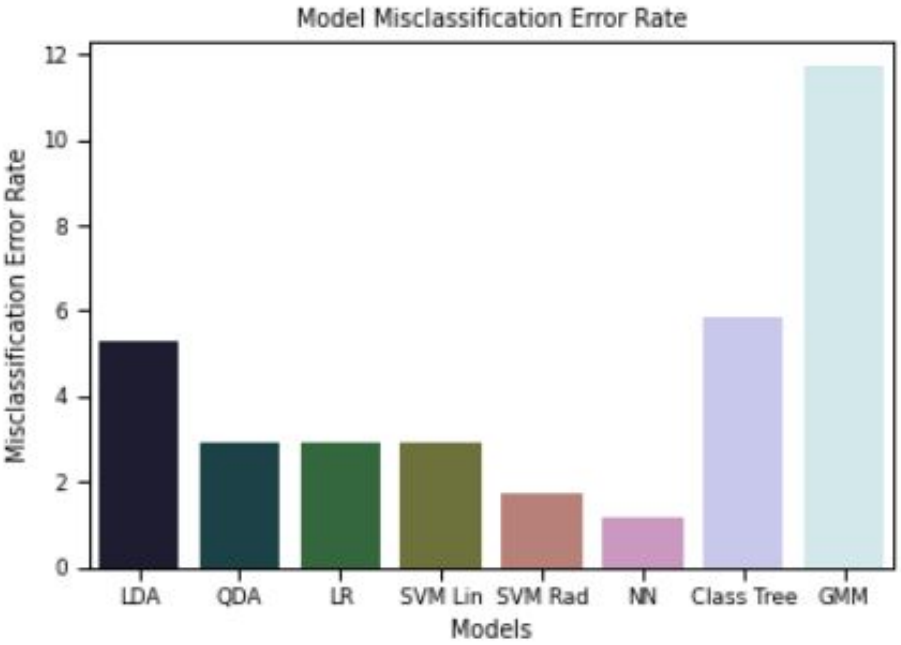


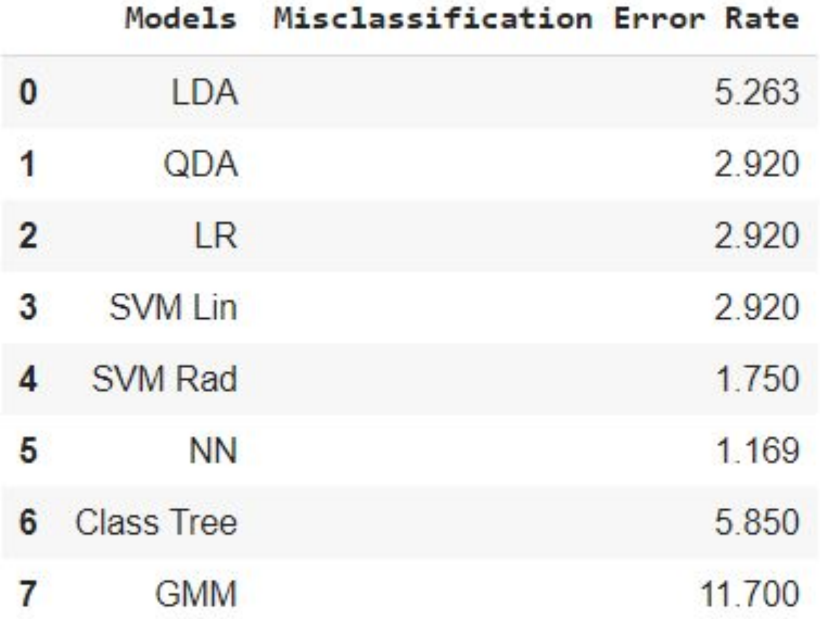


Chose two components because AIC and BIC remain relatively the same after two components.

Higher misclassification error rate than other models because pdfs for two gaussians overlap.

Classification Error Rate Comparison

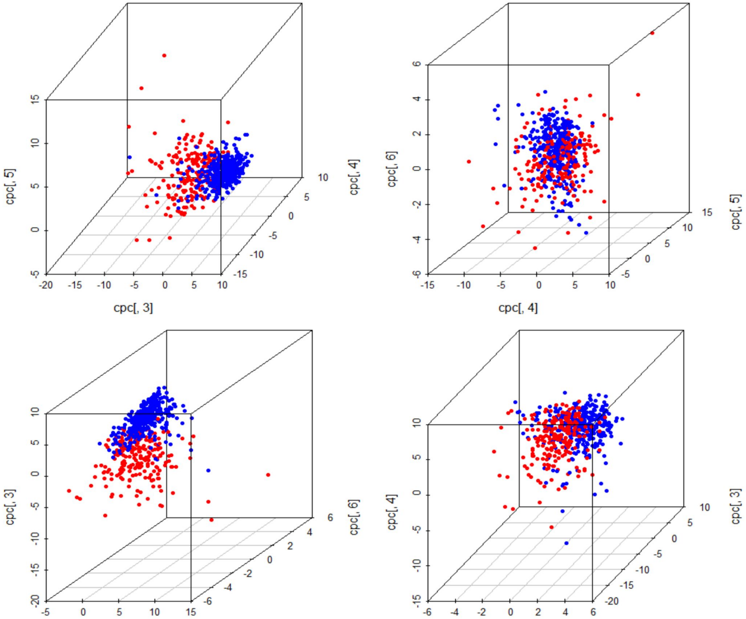




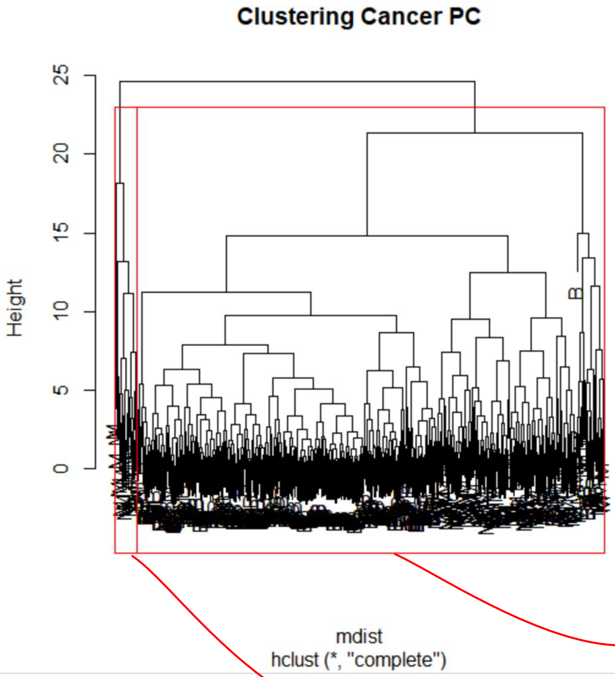
	Models	Misclassification Error Rate
0	LDA	5.263
1	QDA	2.920
2	LR	2.920
3	SVM Lin	2.920
4	SVM Rad	1.750
5	NN	1.169
6	Class Tree	5.850
7	GMM	11.700

Clustering Analysis

Cancer data

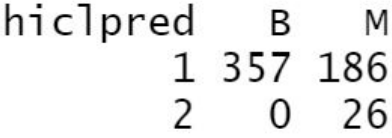


Hierarchical Method



The optimal number of clusters is 2

Compare with the original data

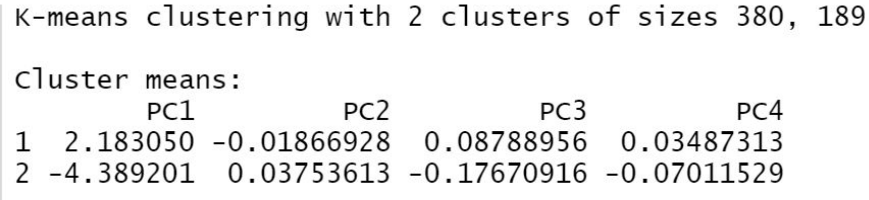


```
hiclpred   B   M
       1 357 186
       2   0  26
```

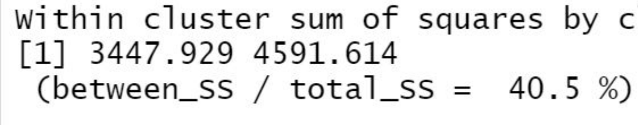
The size of the 2 clusters are 543 and 26

Clustering Analysis

K-Means Method



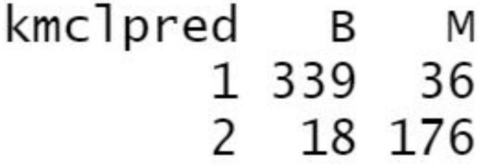
```
K-means clustering with 2 clusters of sizes 380, 189

Cluster means:
        PC1         PC2         PC3         PC4
1  2.183050 -0.01866928  0.08788956  0.03487313
2 -4.389201  0.03753613 -0.17670916 -0.07011529
```



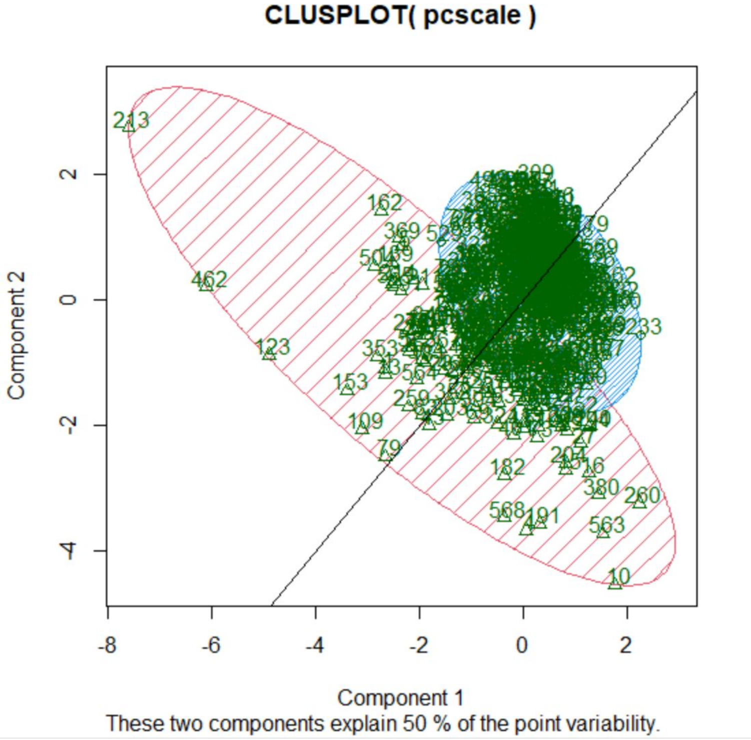
```
Within cluster sum of squares by cluster:
[1] 3447.929 4591.614
 (between_SS / total_SS =  40.5 %)
```

SS_total=8,039.543

Compare with the original data:



```
kmclpred   B   M
       1 339  36
       2  18 176
```



These two components explain 50 % of the point variability.

Q&A