Chatzimoschou Angeliki

BAPT XXXX

**STATISTICS FOR BUSINESS ANALYTICS II, PROJECT 2 Professor: Karlis Dimitrios Assistant: Rui Miguel Forte**

*Summary*

This report analysis concerns the dataset Breast Cancer Wisconsin (FNA) and is consisted of 3 parts.

Our dataset refers to 569 patients from a study on breast cancer.  
Variables are computed from a digitized image of a breast mass. They describe characteristics of the cell nuclei present in the image. Ten real-valued features are computed for each cell nucleus:

a) radius (mean of distances from center to points on the perimeter) (data type: numeric)

b) texture (standard deviation of gray-scale values) (data type: numeric)

c) perimeter (data type: numeric)

d) area (data type: numeric)

e) smoothness (local variation in radius lengths) (data type: numeric)

f) compactness (perimeter^2 / area - 1.0) (data type: numeric)

g) concavity (severity of concave portions of the contour) (data type: numeric)

h) concave points (number of concave portions of the contour) (data type: numeric)

i) symmetry (data type: numeric)

j) fractal dimension ("coastline approximation" - 1) (data type: numeric)

There are also two more variables, namely the ID (not useful for this project) and the Diagnosis   
(M = malignant, B = benign, data type: factor) for the tumor.

In Part 1, we use SVM and Random Forests Classification methods in order to classify our dataset’s   
observations and create a rule to predict whether the tumor is malignant or benign.

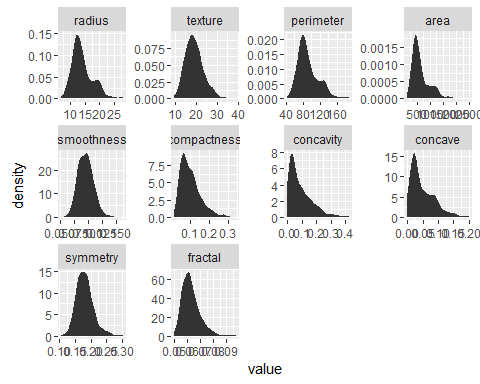
In Part 2, we use K-Means and Hierarchical Clustering so as to assign the patients into clusters created by common characteristics.

In Part 3, we apply the method of PCA to our initial dataset, in order to get a new dataset consisted of  
 independent principal components (the number of principal components is usually less than the number of variables of our original dataset) and then we re-run Parts 1 and 2, so as to spot any differences in the   
results.

Dataset Exploration

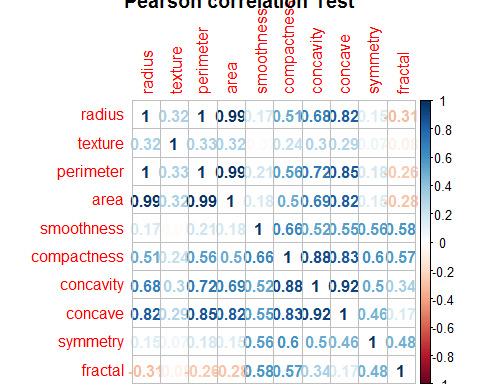
|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Min | Mean | Median | Max | S.D. | Skew | Kyrt | S.E. | NA | Near Zero Variance |
| Type | - | - | - | - | - | - | - | - | 0 | - |
| Radius | 6.98 | 14.13 | 13.37 | 28.11 | 3.52 | 0.94 | 0.81 | 0.15 | 0 | False |
| Texture | 9.71 | 19.29 | 18.84 | 39.28 | 4.30 | 0.65 | 0.73 | 0.18 | 0 | False |
| Perimeter | 43.79 | 91.97 | 86.24 | 188.50 | 24.30 | 0.99 | 0.94 | 1.02 | 0 | False |
| Area | 143.50 | 654.89 | 551.10 | 2501.00 | 351.91 | 1.64 | 3.59 | 14.75 | 0 | False |
| Smoothness | 0.05 | 0.10 | 0.10 | 0.16 | 0.01 | 0.45 | 0.82 | 0.00 | 0 | False |
| Compactness | 0.02 | 0.10 | 0.09 | 0.35 | 0.05 | 1.18 | 1.61 | 0.00 | 0 | False |
| Concavity | 0.00 | 0.09 | 0.06 | 0.43 | 0.08 | 1.39 | 1.95 | 0.00 | 0 | False |
| Concave | 0.00 | 0.05 | 0.03 | 0.20 | 0.04 | 1.17 | 1.03 | 0.00 | 0 | False |
| Symmetry | 0.11 | 0.18 | 0.18 | 0.30 | 0.03 | 0.72 | 1.25 | 0.00 | 0 | False |
| Fractal | 0.05 | 0.06 | 0.06 | 0.10 | 0.01 | 1.30 | 2.95 | 0.00 | 0 | False |

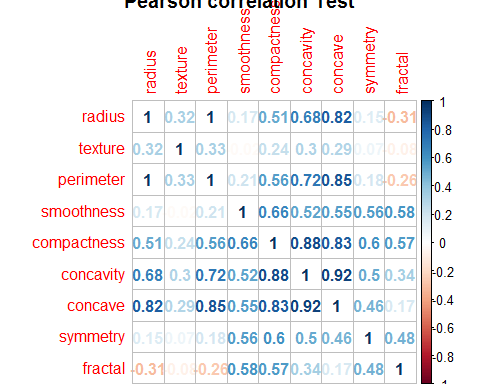
From our descriptive statistics table, we conclude that we have a dataset which contains no missing values   
(NAs), all variables have statistical information as none of them has near zero variance.   
Finally, we see that ‘Area’, ‘Perimeter’, ‘Texture’ and ‘Radius’ variables have bigger variance in contrast to   
the rest of our variables and this will create an issue to our analysis’ methods.

Histograms for each Variable  


Apparently, most of our variables are right skewed, meaning right asymmetric.

Pearson Correlation Test



Pearson correlation test creates a correlation matrix among dataset variables.  
As we see, the variables ‘area’ – ‘radius’ – ‘perimeter’ – ‘concave’ – ‘concavity’ are strongly positively   
correlated, which is logical as they are all metrics of the cell nucleus. As a first move, we remove ‘area’   
variable and recheck correlations.  
  
 Pearson Correlation Test without ‘area’ variable

As we see correlation remains. However, variable ‘perimeter’ contains the highest variance and is the most skewed, that’s why we decide to remove this one too. We don’t remove any of the rest of the variables in   
order not to lose more statistical information.

PART 1  
  
First, we split our dataset in train and test datasets, in analogy of 70% (382 observations)   
and 30% (187 observations)of our initial dataset (569 observations) respectively. Train subset will be used   
to train our classifiers and provide the raw information needed for our predictive models. Test subset will be used to evaluate the performance of our models. To split our dataset we will use function sample with   
replacement and given probabilities for each new dataset.

Method 1: SVM (radial kernel)

Given a set of training observations, each marked for belonging to one of two categories, an SVM classifier builds a model that assigns new observations into one category or the other, making it a [binary](https://en.wikipedia.org/wiki/Binary_classifier) [classifier](https://en.wikipedia.org/wiki/Linear_classifier).  
The observations of the separate categories are divided by a clear gap that is as wide as possible.   
Then, are mapped into that same space and predicted to belong to a category based on which side of the   
gap they fall on. Except from linear classification, SVMs provide also non-linear classification, implicitly mapping their inputs into high-dimensional feature spaces and thus more accurate.

In our project we will perform SVM with radial kernel (with response variable: ‘type’) in order to achieve   
higher accuracy and also because linearity assumptions such as variable independence or   
homoscedasticity are violated.  
We begin by using the tune() function that can carry out k-fold cross-validation to determine appropriate   
values for model meta parameters, such as: ‘cost’ and ‘gamma’. To do this, it receives an input called   
ranges, which is a list of values for all the parameters that we want to vary across the different runs of cross validation.

breast\_radial\_tune$best.parameters: cost gamma  
 1 0.5

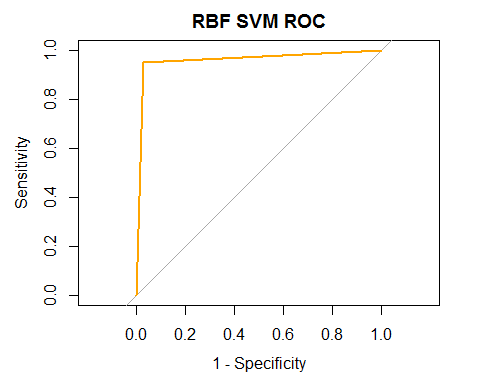
breast\_radial\_tune$best.performance: 0.04966262

After running svm.tune, the suggested best model has cost = 1(cost of constraints violation) and   
gamma = 0.5 (parameter needed for all kernels except linear)and achieves 95% training accuracy.   
Moreover, we train our svm model and continue to prediction.  
  
The performance on our test set is 96.25% and very close to what we saw in training.   
  
Confusion Matrix

actual  
predicted 0 1  
 0 104 4  
 1 3 76  
"Accuracy 0.962566844919786"

A confusion matrix, is a table that allows visualization of the performance of an algorithm.   
Each row of the matrix represents the instances in a predicted class while each column represents the instances in an actual class.   
From our svm confusion matrix, we conclude that 104 observations were correctly classified as benign tumors   
and 76 as malignant. Furthermore, 4 were misclassified as benign, while 3 as malignant. By the confusion matrix, we can receive the accuracy of our model. This one achieves an accuracy of 96.25% (close to overfitting).

**ROC Curve**  
  
This curve is a plot of the true positive rate on the y axis and the false positive rate on the x axis.   
The true positive rate (recall) or the sensitivity of a binary classifier. The false positive rate is: 1 - specificity.   
A random binary classifier will have a true positive rate equal to the false positive rate and thus, will be   
positioned on the diagonal line. Any curve above this line will perform better than a random classifier.



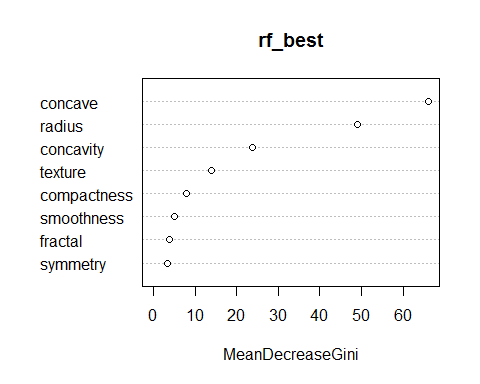
Area under the curve: 0.961  
  
We could probably, say that it is in the verge of overfitting.

*Method 2: Random Forests*  
  
Random forests is the general technique of random decision forests that are an [ensemble learning](https://en.wikipedia.org/wiki/Ensemble_learning) method   
for [classification](https://en.wikipedia.org/wiki/Statistical_classification), that operate by constructing a multitude of [decision trees](https://en.wikipedia.org/wiki/Decision_tree_learning) at training time and outputting   
the classes of the individual trees. Random decision forests correct for decision trees' habit of [overfitting](https://en.wikipedia.org/wiki/Overfitting) to   
their training set.

rf\_tune$best.parameters:ntree mtry  
 500 4

Like before, we tune our random forests model and the suggested best model has ntree= 500  
(No. of trees) and mtry= 4 (No. of variables randomly sampled as candidates at each split).   
and achieves 95% training accuracy. Moreover, we train our random forests model and continue to   
prediction.  
  
In Random Forests, after training our model, we can also find the most important variables for our classifier.  
  
 Overall  
radius 48.980757  
texture 13.894861  
smoothness 4.953918  
compactness 7.780576  
concavity 23.753589  
concave 66.178669  
symmetry 3.288280  
fractal 3.726648

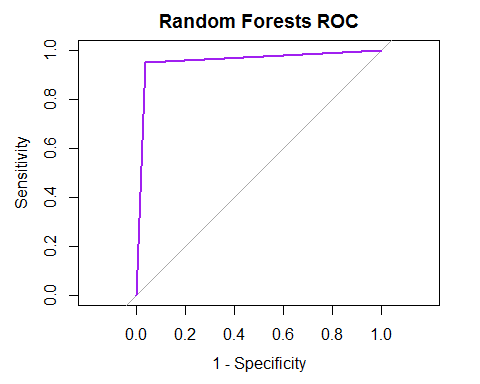
Random Forest Variance Importance Plot



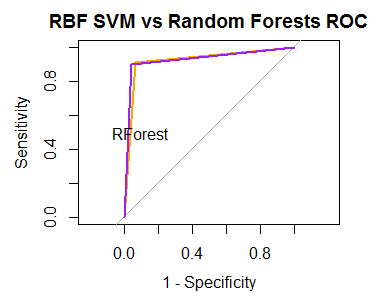
From the above plot we conclude that the most important features are: ‘concave’, ‘radius’ and ‘concavity’.  
  
Confusion Matrix  
 actual  
predicted 0 1  
 0 103 4  
 1 4 76

"Accuracy 0.957219251336898"

From our random forests confusion matrix, we conclude that 103 observations were correctly classified as   
benign tumors and 76 as malignant. Furthermore, 4 were misclassified as benign, while 4 as malignant.   
By the confusion matrix, we can receive the accuracy of our model.   
This one achieves an accuracy of 95.72% (close to overfitting).

  
Area under the curve: 0.9266

Also by observing the Roc plot we conclude that we almost have a perfect classifier.  
  
  
 SVM Radial (Basis Function) vs Random Forests ROC Plot



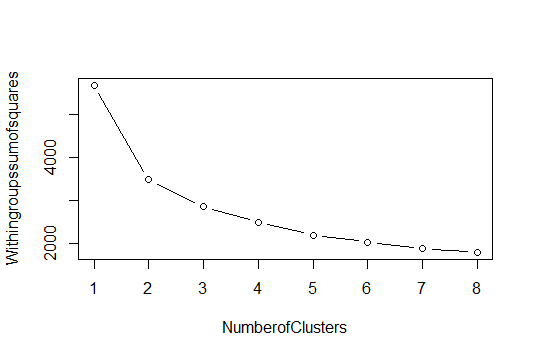
Comparing our two classifiers Roc plots we observe that both our classifiers offer great performance.  
  
PART 2  
In Part 2, in order to perform clustering, which is an Unsupervised learning method and there is no outcome to be predicted, the algorithm tries to find patterns in the data. That’s why we re-load our initial   
breast-cancer dataset, getting rid of the ‘type’ variable and keeping only the numerical variables.

As seen in our descriptives table, some of the variables have very large variance compared to the rest, and methods will load on the large variances, that’s why we will scale our variables.Function scale() will   
calculate the mean and standard deviation of the entire vector, then "scale" each element by those values   
by subtracting the mean and dividing by the standard deviation (standardization).

Method 1: K-Means   
  
In k means clustering, we have to specify the number of clusters we want the data to be grouped into,   
as it tries to cluster data based on their similarity. The algorithm aims to [partition](https://en.wikipedia.org/wiki/Partition_of_a_set) n observations into   
k clusters in which each observation belongs to the [cluster](https://en.wikipedia.org/wiki/Cluster_(statistics)) with the nearest mean and finds the centroid of   
each cluster.

As there are 2 types of tumors, we set the algorithm to group into 2 clusters, and as in the beginning it   
randomly assigns, we put nstart = 50. R will try 50 different random starting assignments and then will   
choose the one with the lowest within cluster variation.

No. of Clusters



|  |
| --- |
| **Results based on 26 criteria** |
| 12 proposed 2 as the best number of clusters |
| 7 proposed 3 as the best number of clusters |
| 2 proposed 5 as the best number of clusters |
| 1 proposed 6 as the best number of clusters |
| 1 proposed 7 as the best number of clusters |
| 2 proposed 8 as the best number of clusters |

As we observe from the plot and the results of the function we have built, which calculates based on the 26, the best number of clusters is 2 (The Hubert index, a graphical method of determining the number of clusters).  
  
 Cluster Means for each variable

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **radius** | **texture** | **perimeter** | **area** | **smoothness** | **compactness** | **concavity** | **concave** | **symmetry** | **fractal** |
| 1 | 1.1188158 | 0.5115176 | 1.1507004 | 1.110161 | 0.6584389 | 1.0778846 | 1.2046174 | 1.2798952 | 0.6198655 | 0.1782381 |
| **2** | -0.4726997 | -0.2161162 | -0.4861709 | -0.469043 | -0.2781905 | -0.4554062 | -0.5089508 | -0.5407557 | -0.2618932 | -0.0753056 |

In the first cluster variables have only positive values, while in second cluster variables take only negative ones.

Cluster Matrix   
 cluster  
 type 1 2  
 B 6 351  
 M 163 49

"Accuracy: 0.9"

From the confusion matrix above, we can assume that the first cluster is associated with malignant tumors (163M) and the second cluster with benign tumors(351B). Finally, the accuracy of K-Means clustering   
method is high: 90%.

Method 2: Hierarchical Clustering  
  
Hierarchical clustering is another method of cluster analysis which tries to build an hierarchy of clusters.

At first, we have to find the number of clusters in which we will group our data. That’s why, we will use   
silhouette values algorithm. This is a measure of how similar an object is to its own cluster (cohesion)   
compared to other clusters (separation). The silhouette values range from -1 to 1, where a high value   
indicates that the object is well matched to its own cluster and poorly matched to neighboring clusters.

We applied the silhouette values algorithm to both scaled and original dataset while creating silhouette   
values for 2,3 and 4 clusters. In both situations, clustering with 2 groups was the best possible   
choice: 0.39 and 0.74 respectively.  
  
  
 Silhouette values Plots

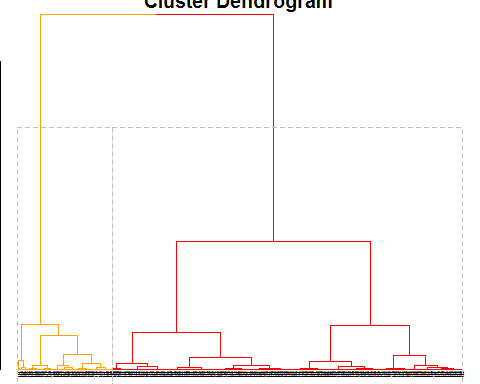
|  |
| --- |
| [scaled matrix] 2 clusters: plot(silhouette(hc\_sc[,1], dist(data\_sc))) |
|  |
| [scaled matrix] 3 clusters: plot(silhouette(hc\_sc[,2], dist(data\_sc))) |
|  |
| [scaled matrix] 4 clusters:plot(silhouette(hc\_sc[,3], dist(data\_sc))) |
|  |
| [original matrix] 2 clusters: plot(silhouette(hc\_sc2[,1], dist(data\_num))) |
|  |
| [original matrix] 3clusters: plot(silhouette(hc\_sc2[,2], dist(data\_num))) |
|  |
| [original matrix] 4 clusters: plot(silhouette(hc\_sc2[,3], dist(data\_num))) |
|  |

After running the hierarchical clustering model we obtain the characteristics table:

Characteristics of each cluster

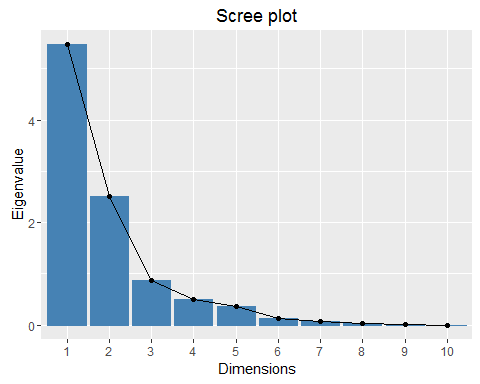
|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **radius** | **texture** | **perimeter** | **area** | **smoothness** | **compactness** | **Concavity** | **concave** | **symmetry** | **fractal** |
| **1** | 14.532167 | 19.41236 | 94.69663 | 687.4158 | 0.09610983 | 0.10572038 | 0.09152410 | 0.05136504 | 0.1805348 | 0.06204686 |
| **2** | 9.174628 | 17.78860 | 58.60349 | 257.0047 | 0.09942395 | 0.08746744 | 0.05546821 | 0.01899963 | 0.1888326 | 0.07198116 |

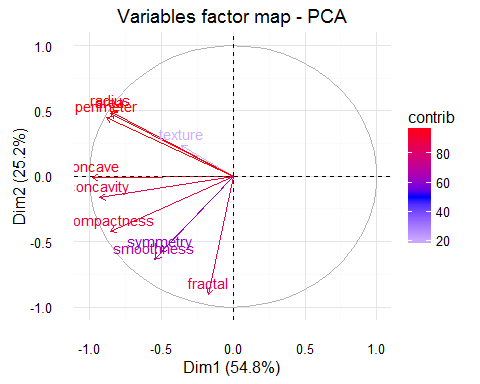
Cluster Matrix   
  
 cluster  
type 1 2  
 B 2 355  
 M 119 93  
"Accuracy: 0.83"  
As we see from both the characteristics table and the cluster matrix, the first cluster mainly correlates with malignant tumors and the second one with benign. Our model achieves good accuracy rate (83%).   
However, it is a little less accurate than the K-Means method (90%).

Hierarchical Dendrodiagram  


PART 3  
  
As mentioned in the beginning, now we will perform a dimension reduction method called Principal   
Components Analysis, in order to obtain variables that are not correlated but at the same time we will lose a portion of the statistical information contained in our initial dataset.  
  
Importance of components:  
 PC1 PC2 PC3 PC4 PC5 PC6  
Standard deviation 2.3406 1.5870 0.93841 0.7064 0.61036 0.35234  
Proportion of Variance 0.5479 0.2519 0.08806 0.0499 0.03725 0.01241  
Cumulative Proportion 0.5479 0.7997 0.88779 0.9377 0.97495 0.98736  
  
 PC7 PC8 PC9 PC10  
Standard deviation 0.28299 0.18679 0.10552 0.01680  
Proportion of Variance 0.00801 0.00349 0.00111 0.00003  
Cumulative Proportion 0.99537 0.99886 0.99997 1.00000

Principal Components Plot

  
  
  
  
  
 Biplot PCA



From the first plot we conclude that we can keep just the first three Principal Components (PC1, PC2, PC3),   
which cumulatively explain the 90% of the variance of our initial dataset and from the second plot we realize that most of our initial variables contribute to the creation of the principal components, except from the   
‘texture’, ‘symmetry’ and the ‘smoothness’ variables. Furthermore, all the variables are positively correlated as the vectors have the same orientation (left) and retain negative eigenvalues in the x’x and y’y axis.   
  
Now, we are ready to create out new\_data dataset, which is consisted of the three chosen principal   
components and our response variable (type). After that, we will re-run the processes of Part 1 and 2 with our new pca dataset and compare the results.  
  
PART 1 WITH PCA  
  
Method 1: SVM radial

breast\_radial\_tune\_pc$best.parameters: cost gamma  
 10 0.1

After running svm.tune, the suggested best model has cost = 10 and gamma = 0.1.  
Moreover, we train our svm model and continue to prediction.

Confusion Matrix   
 actual  
predicted B M  
 B 100 7  
 M 7 73

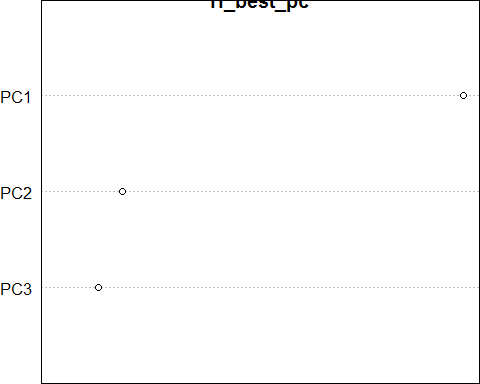
"Accuracy 0.925133689839572"  
  
We conclude that 100 observations were correctly classified as benign tumors and 73 as malignant.   
Furthermore, 7 were misclassified as benign, while 7 as malignant. By the confusion matrix, we can also   
receive the accuracy of our model. This one achieves an accuracy of 92.51% (close to overfitting).

The accuracy of our pca test set is 92.51%, a little less accurate than our initial classifier (96.25%).

Method 2: Random Forests

rf\_tune\_pc$best.parameters: ntree mtry  
 2000 3

Like before, we tune our random forests classifier and the suggested best model has ntree= 2000 and   
mtry= 3. Then, we train our random forests model and continue to prediction.  
  
In Random Forests, after training our model, we can also find the most important variables for our classifier.  
  
 Overall  
PC1 136.99890  
PC2 21.67152  
PC3 13.61160  
  
 Variable Importance Plot

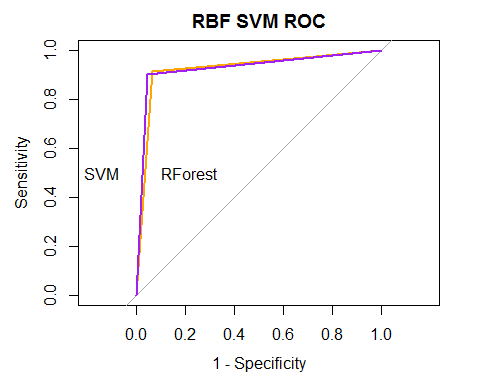


From the above plot we confirm that the most important feature is ‘PC1’, the one who gathers the majority of variance.

Confusion Matrix  
  
 actual  
predicted B M  
 B 102 8  
 M 5 72

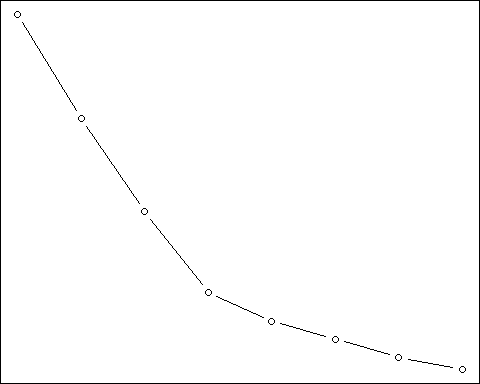
"Accuracy 0.93048128342246"

From our random forests confusion matrix, we conclude that 102 observations were correctly classified as   
benign tumors and 72 as malignant. Furthermore, 8 were misclassified as benign, while 5 as malignant.   
By the confusion matrix, we can receive the accuracy of our model: 93%.   
Again, a little less accurate than our initial random forest model.



Comparing the two Roc Plots, we conclude that after implementing the method of pca our random forest   
classifier performs a little bit better than the SVM radial classifier: 93% and 92.51% respectively.  
  
In a nutshell, we could say that both SVM and Random Forests are very sophisticated and analytical   
classifiers, which don’t get affected by variable correlation and thus the performances are similar before and after the PCA method. However, we observe that due to loss of statistical information both classifiers   
perform slightly worse than in our first attempt.   
  
  
  
  
  
PART 2 WITH PCA  
  
Again we act the same as we did in part 2 and receive the following results.  
  
  
**Results based on 26 criteria**    
6 proposed 2 as the best number of clusters   
3 proposed 3 as the best number of clusters   
9 proposed 4 as the best number of clusters   
2 proposed 5 as the best number of clusters   
1 proposed 6 as the best number of clusters   
1 proposed 7 as the best number of clusters   
1 proposed 8 as the best number of clusters   
According to the majority rule, the best number of clusters is 4

As we observe from the plot the best number of clusters is 2,which contradicts the results of the function,   
which calculates the number of clusters based on the 26 criteria and results into 4.  
  
 Cluster Means for each variable



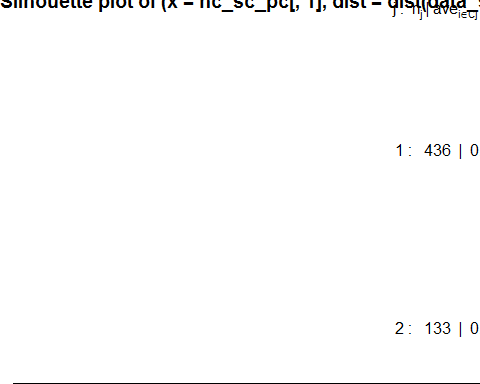
We will use two clusters in order to gain on accuracy cluster content and compare it with our previous try.   
  
  
Cluster Matrix

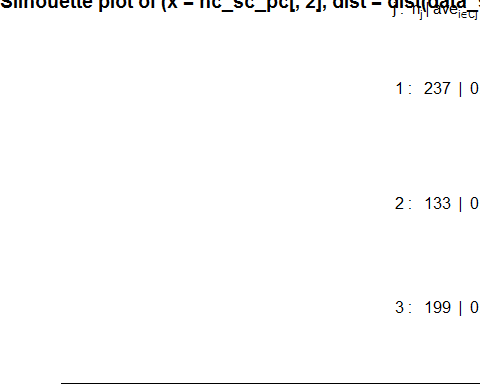
cluster  
type 1 2  
 B 2 355  
 M 165 47

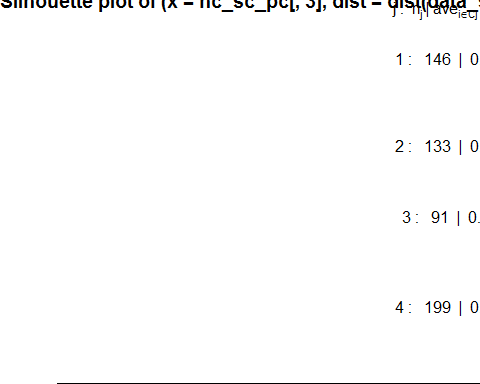
"Accuracy: 0.91"

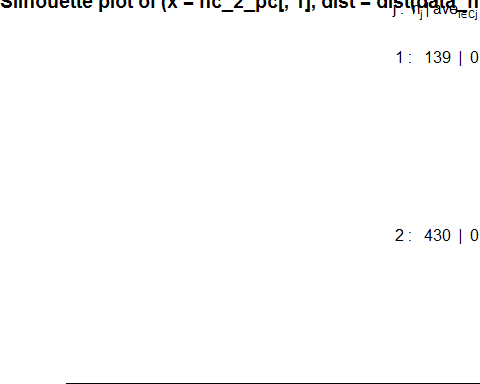
From the confusion matrix above, we can assume that the first cluster is associated with malignant tumors   
(165M) and the second cluster with benign tumors(355B). Finally, the accuracy of K-Means clustering   
method is high: 91%.  
  
Method 2: Hierarchical Clustering

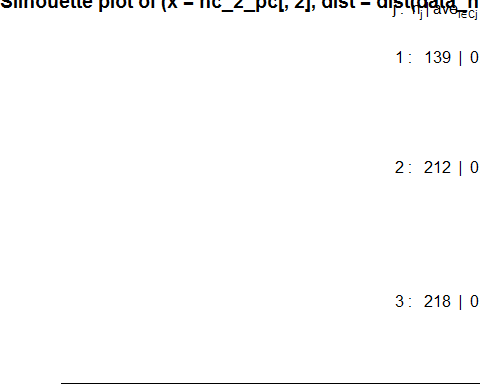
We act as in Part 2 and we get the following results.  
  
When using PCA the without scaling clustering with 2 groups was the best possible choice: 0.46.  
   
 Silhouette values Plots

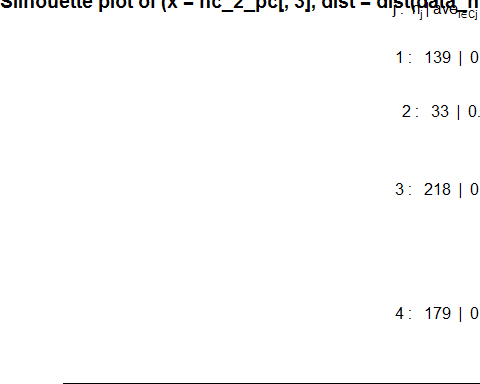










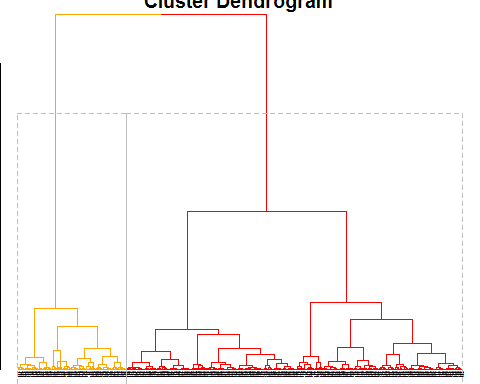


Cluster Matrix   
 cluster  
type 1 2  
 B 0 357  
 M 139 73

"Accuracy: 0.87"

As we see from both the characteristics table and the cluster matrix, the first cluster mainly correlates with malignant tumors and the second one with benign. Our model achieves good accuracy rate (87%). However, it is a little less accurate than the K-Means method (91%).

Hierarchical Dendrodiagram



In a nutshell, we could say that both K-Means and Hierarchical Clustering are very sophisticated and analytical are of similar performance before and after the PCA method. However, we observe that due to variable independency both methods perform slightly better than in our first attempt.