title: "Dimensional Reduction Project"

author: "Rezart Abazi, Arman Shashaani"

date: "21/12/2019"

For our project for the second part of the Unsupervised Learning classes, we decided to perform Dimension Reduction through PCA (Principal Component Analysis) and Regression. We applied this on a dataset of 'Brest Cancer wisconsin' of 596 observation and 32 variables . [Kaggle](https://www.kaggle.com/uciml/breast-cancer-wisconsin-data/version/1#data.csv).

For the purposes of our project we also repurposed code from classes as well as taken some inspiration and suggestions in terms of techniques used, from similar projects based on this dataset.

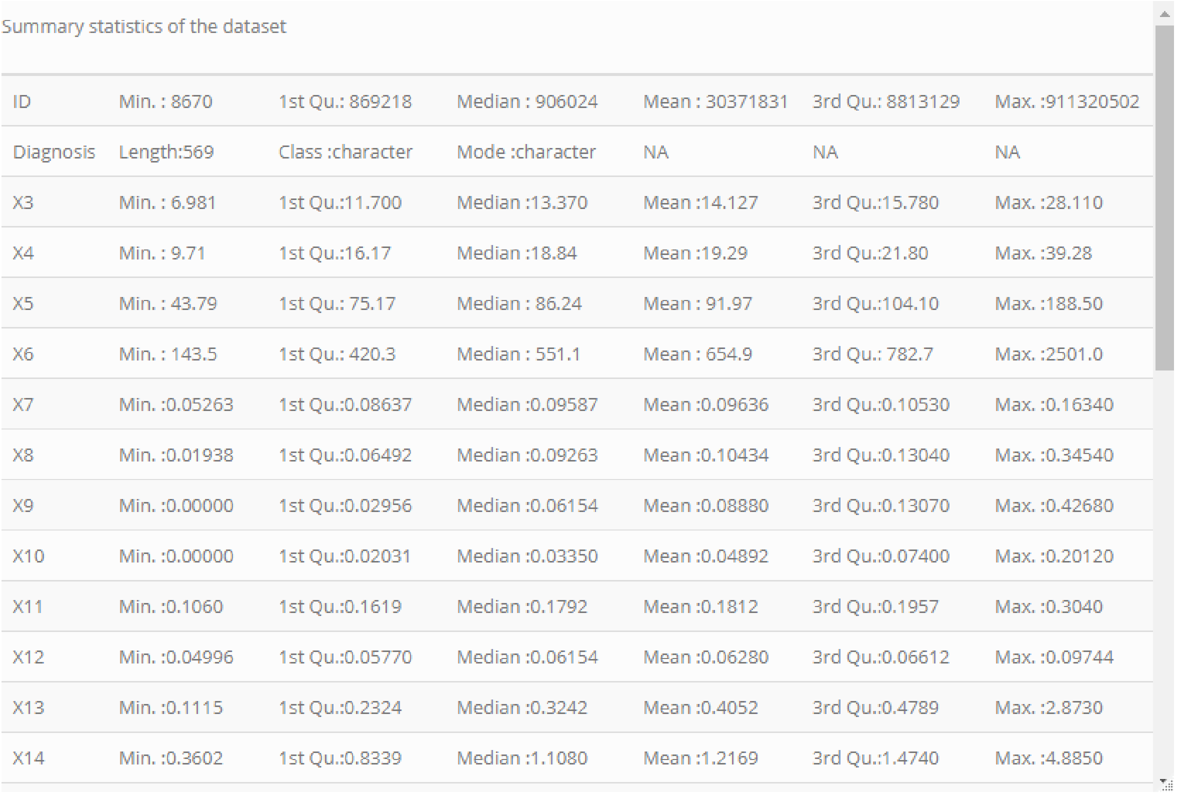
```
data<- read.csv("data.csv", stringsAsFactors=F)
```

1. Id : ID number
2. Diagnosis : The diagnosis of breast tissues (M = malignant, B = benign)
3. Radius_mean : mean of distances from center to points on the perimeter
4. Texture_mean : standard deviation of gray-scale values
5. Perimeter_mean : mean size of the core tumor
6. Area_mean :
7. Smoothness_mean : mean of local variation in radius lengths
8. Compactness_mean : mean of perimeter^2 / area - 1.0
9. Concavity_mean : mean of severity of concave portions of the contour
10. Concave points_mean : mean for number of concave portions of the contour
11. Symmetry_mean :
12. Fractal_dimension_mean : mean for "coastline approximation" - 1
13. Radius_se : standard error for the mean of distances from center to points on the perimeter
14. Texture_se : standard error for standard deviation of gray-scale values
15. Perimeter_se :
16. Area_se :
17. Smoothness_se : standard error for local variation in radius lengths
18. Compactness_se : standard error for perimeter^2 / area - 1.0
19. Concavity_se : standard error for severity of concave portions of the contour
20. Concave points_se : standard error for number of concave portions of the contour
21. Symmetry_se :
22. Fractal_dimension_se : standard error for "coastline approximation" - 1
23. Radius_worst : "worst" or largest mean value for mean of distances from center to points on the perimeter
24. Texture_worst : "worst" or largest mean value for standard deviation of gray-scale values

25. Perimeter_worst :
26. Area_worst :
27. Smoothness_worst : "worst" or largest mean value for local variation in radius lengths
28. Compactness_worst : "worst" or largest mean value for perimeter^2 / area - 1.0
29. Concavity_worst : "worst" or largest mean value for severity of concave portions of the contour
30. Concave points_worst : "worst" or largest mean value for number of concave portions of the contour
31. Symmetry_worst :
32. Fractal_dimension_worst : "worst" or largest mean value for "coastline approximation" – 1

We decided to divide this project in 2 parts. In the first part it's going to be a fast overlook of descriptive statistics of this data set, following PCA (Principal Component Analysis) as the main task of the project.

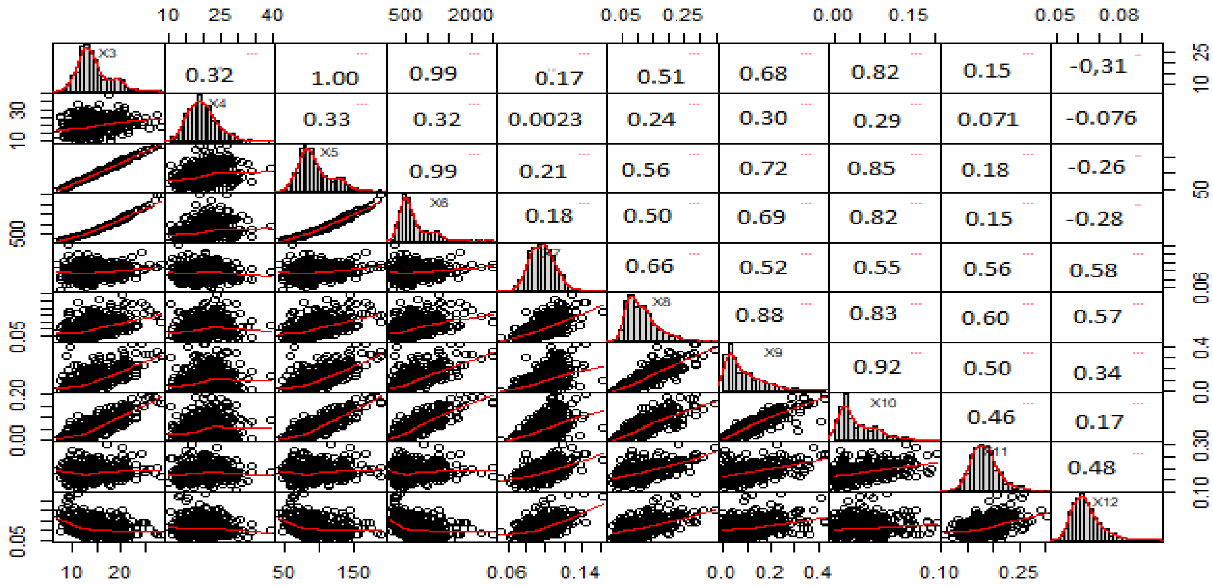
The second part of the project with be SVM (Support Vector Machine)



Summary statistics of the dataset

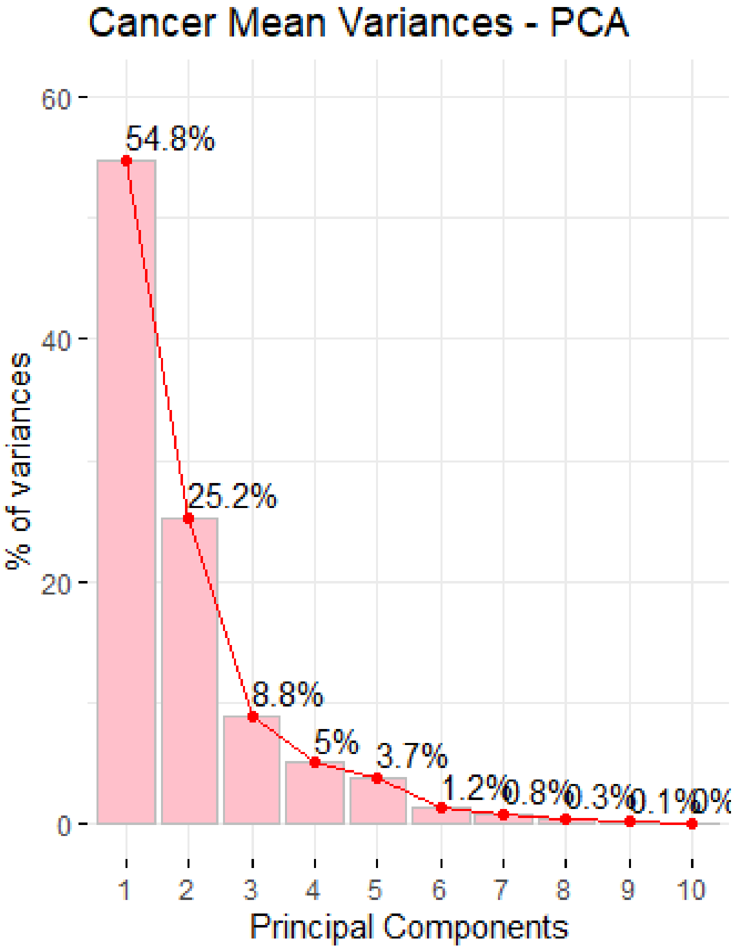
ID	Min. : 8670	1st Qu.: 869218	Median : 906024	Mean : 30371831	3rd Qu.: 8813129	Max. :911320502
Diagnosis	Length:569	Class :character	Mode :character	NA	NA	NA
X3	Min. : 6.981	1st Qu.:11.700	Median :13.370	Mean :14.127	3rd Qu.:15.780	Max. :28.110
X4	Min. : 9.71	1st Qu.:16.17	Median :18.84	Mean :19.29	3rd Qu.:21.80	Max. :39.28
X5	Min. : 43.79	1st Qu.: 75.17	Median : 86.24	Mean : 91.97	3rd Qu.:104.10	Max. :188.50
X6	Min. : 143.5	1st Qu.: 420.3	Median : 551.1	Mean : 654.9	3rd Qu.: 782.7	Max. :2501.0
X7	Min. :0.05263	1st Qu.:0.08637	Median :0.09587	Mean :0.09636	3rd Qu.:0.10530	Max. :0.16340
X8	Min. :0.01938	1st Qu.:0.06492	Median :0.09263	Mean :0.10434	3rd Qu.:0.13040	Max. :0.34540
X9	Min. :0.00000	1st Qu.:0.02956	Median :0.06154	Mean :0.08880	3rd Qu.:0.13070	Max. :0.42680
X10	Min. :0.00000	1st Qu.:0.02031	Median :0.03350	Mean :0.04892	3rd Qu.:0.07400	Max. :0.20120
X11	Min. :0.1060	1st Qu.:0.1619	Median :0.1792	Mean :0.1812	3rd Qu.:0.1957	Max. :0.3040
X12	Min. :0.04996	1st Qu.:0.05770	Median :0.06154	Mean :0.06280	3rd Qu.:0.06612	Max. :0.09744
X13	Min. :0.1115	1st Qu.:0.2324	Median :0.3242	Mean :0.4052	3rd Qu.:0.4789	Max. :2.8730
X14	Min. :0.3602	1st Qu.:0.8339	Median :1.1080	Mean :1.2169	3rd Qu.:1.4740	Max. :4.8850

For effect of space we decided to show the first 14 variables and their distribution on data and as we can see from the table the data has normal distribution and the difference between mean and median is very small.



From the correlation plot, among the whole variables, we can see a high correlation between variables (X3, X5) – (X3, X6) – (X5, X6) also we can see a high correlation between X10 and X3, X5, and X6, but not as much as the first pairs which were named.

On the other hand, the correlation for (X4, X7) – (X4, X11 and X12) – (X11, X12) – X11 and X12 with X7 and X8 is low correlated, that means close to zero.

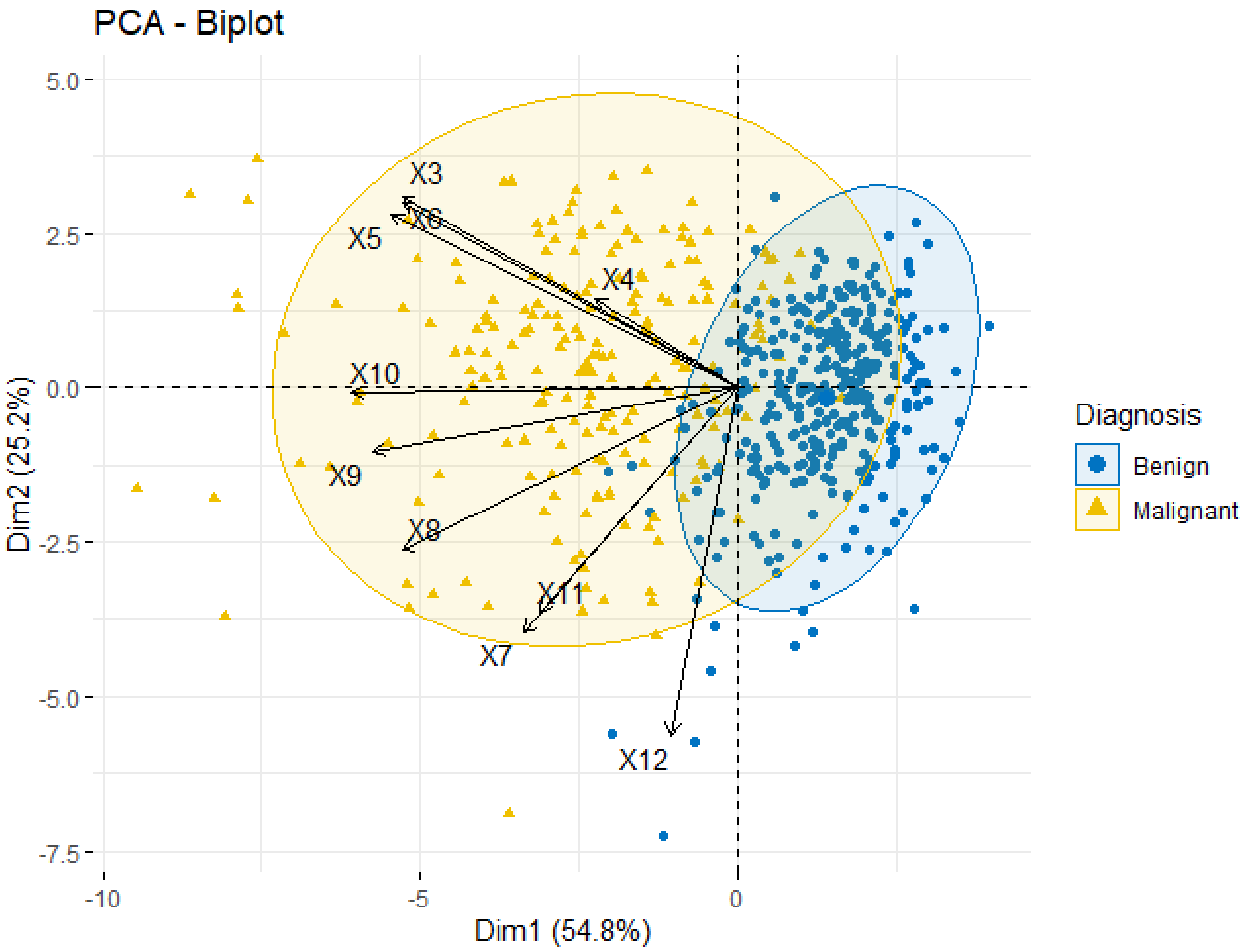


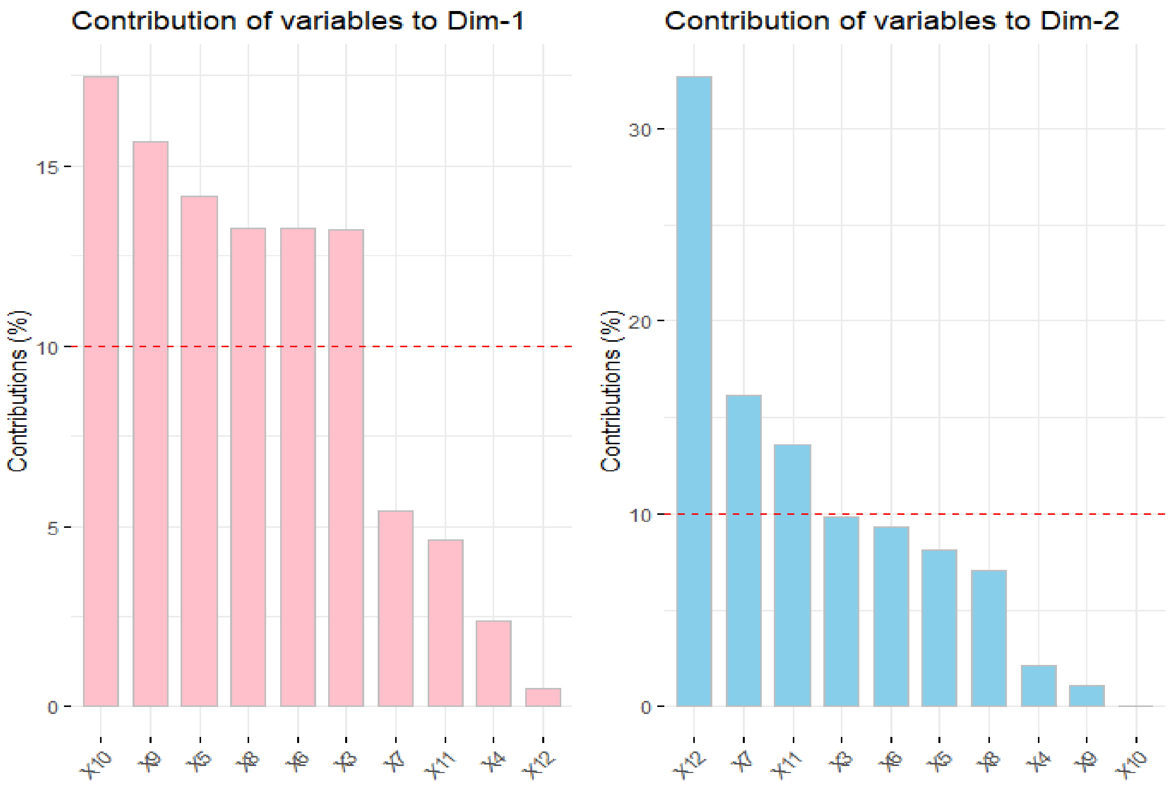
- It can be seen from the figure, we have an elbow after component 4, and for the rest of the components the difference between them is not as much significant as the first four ones. Hence, we can assume four factors instead of all 10.

As it can be seen on the Biplot, we have 54.8% variability for Dim1, and 25.2% variability for Dim2. Also, 12 axes are shown, which all of them are located on the negative side of the dim1, that says they have negative correlation with dim1. In addition to that, variables X3, X4, X5, X6 are high correlated, whereas other variables are not so correlated as you can see. However, X10 is more correlated to variables X3 to X6, compared to X12, which is so far from them.
While, variables are spread out for Dim2, half of them have positive correlation X3, X4, X5, X6 (the ones are highly correlated) and others, have negative correlation with Dim2.

For example, X12 is the most highly correlated with Dim2 in negative way. In the contrast, X3 is highly correlated with Dim2 in positive way. In other words, it means, that as Dim2 increases, X12 goes down faster than other variables.

Moreover, there are some points out of the circles, like a point which is placed below X12. This point has high value for X12 and low value for X3 in Dim2. Having said that, this point has low value in Dim2. This can be interpreted for other points likewise.





Based on two charts above, the one which corresponds to Dim2, shows a better representation of variables on the principle components compared to Dim1. On the other hand, for Dim2 variables are placed nearly to the margin of the correlation area. While, the variables to Dim2, are not represented by principle components as good as to Dim1. Which can be interpreted that the variables, are mostly located near the center of the correlation area.

According to what was mentioned above and the percentages, we can say that the variables to Dim1 are more effective and important to describe and interpret the components, because they are more close to the circle of correlation.

```
library(gridExtra)

p1 <- fviz_contrib(mean_pca, choice="var", axes=1, fill="pink", color="grey", top=10)

p2 <- fviz_contrib(mean_pca, choice="var", axes=2, fill="skyblue", color="grey", top=10)

grid.arrange(p1,p2,ncol=2)
```

SVM (Support Vector Machine)

A computer is not able to create true random numbers, but pseudo random numbers that’s why to optimize this process inside the computer there is actually a list of random numbers

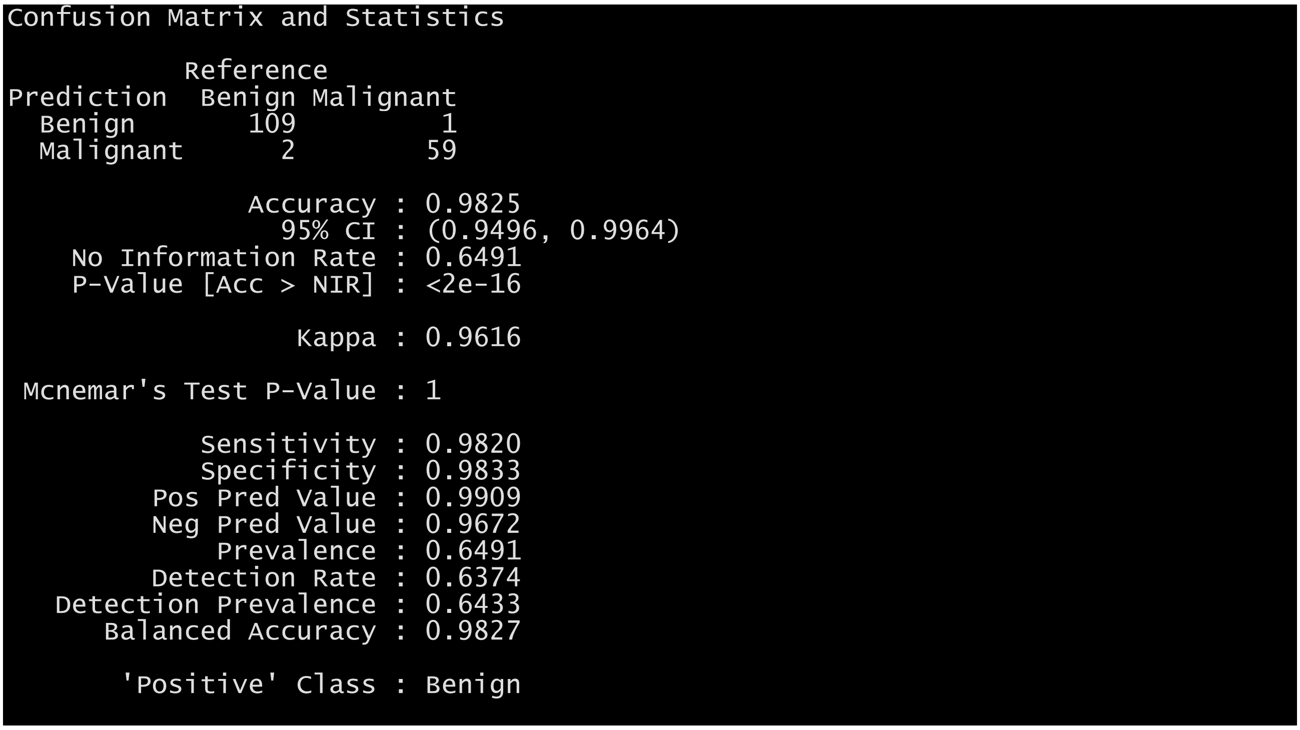
The random component is where in this list we start to generate numbers forcing the seed to be the same will generate exactly the same list of random numbers

That the position of the seed is defined by us and then we will be able to compare performances removing the random effects

We start the list of random numbers at the position (218)

```
nrows <- NROW(data)
set.seed(218)

index <- sample(1:nrows, 0.7 * nrows)
dividelearn_svm <- svm(diagnosis~., data=train)
pre_svm <- predict(learn_svm, test[,-1])
cm_svm <- confusionMatrix(pre_svm, test$diagnosis)
cm_svm
```



```
Confusion Matrix and Statistics

           Reference
Prediction  Benign Malignant
  Benign       109         1
  Malignant      2        59

               Accuracy : 0.9825
                 95% CI : (0.9496, 0.9964)
    No Information Rate : 0.6491
    P-Value [Acc > NIR] : <2e-16

                  Kappa : 0.9616

 Mcnemar's Test P-Value : 1

            Sensitivity : 0.9820
            Specificity : 0.9833
         Pos Pred Value : 0.9909
         Neg Pred Value : 0.9672
             Prevalence : 0.6491
         Detection Rate : 0.6374
   Detection Prevalence : 0.6433
      Balanced Accuracy : 0.9827

       'Positive' Class : Benign
```

Support Vector Machine is an algorithm which can be used for both classification or regression challenges. However, it is mostly used in classification problems. In this algorithm, we plot each data item as a point in n-dimensional space (where n is number of features you have) with the value of each feature being the value of a particular coordinate. Then, we perform classification by finding the hyper-plane that differentiate the two classes very well. As we can see in this case the classification ratio achieves an accuracy of 98% which is the highest between all the other techniques used.