


A dark green vertical bar runs down the left side of the page. A green arrow points to the right from this bar, containing the text 'Spring 2017'.

Spring 2017

Analysis over the Wisconsin Diagnostic Breast Cancer

Multivariate Analysis Project

Several thin, curved lines in shades of green and grey originate from the bottom left and sweep upwards and to the right.

Quentin Coviaux – Xavier Schmoor

Table des matières

1. A description of the problem and available data.....	1
2. The pre-process of data	2
3. The protocol of validation.....	4
4. The visualisation performed	4
5. The interpretation of the latent concepts.	5
6. The clustering performed	6
7. The interpretation of the found clusters.	7
8. Results obtained with the assignment of the test individuals.....	8
9. The prediction model with its best parameterization	10
10.The final model and its generalization error	11
11.Scientific and personal conclusions	12

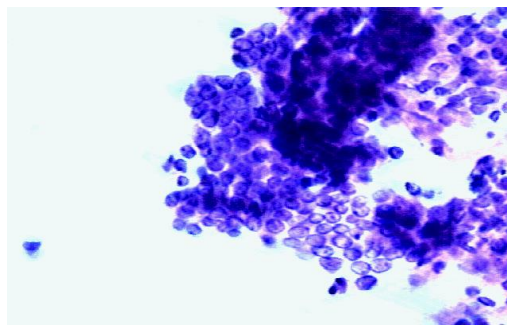
1. A description of the problem and available data

The *Wisconsin Diagnostic Breast Cancer* (WDBC) dataset contains features computed from digitized images of a fine needle aspirate (FNA) of breast mass. They describe characteristics of the cell nuclei present in the image. The aim is to predict the diagnosis: “B” for Benign or “M” for Malignant.

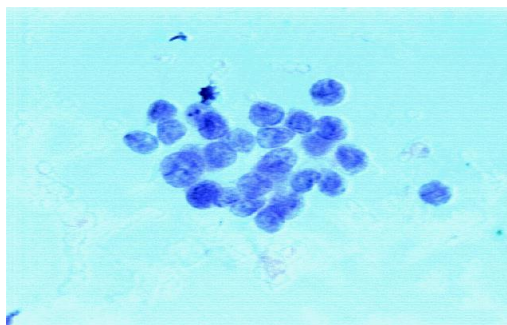
Source: <ftp://ftp.cs.wisc.edu/math-prog/cpo-dataset/machine-learn/cancer/WDBC/WDBC.doc>

Let's have a look some images taken:

The individual diagnosis for this image is Benign (ID 92_4934):



The individual diagnosis for this image is Malignant (ID 91_6799):



Source: ftp://ftp.cs.wisc.edu/math-prog/cpo-dataset/machine-learn/cancer/cancer_images/

The dataset has 569 instances and 32 attributes: the ID number, the Diagnosis (B or M) and 30 real-valued input features. The features are the following: the radius (mean of distances from center to points on the perimeter), the texture (standard deviation of gray-scale values), perimeter, area, smoothness (local variation in radius lengths), compactness ($\text{perimeter}^2 / \text{area} - 1.0$), concavity (severity of concave portions of the contour), concave points (number of concave portions of the contour), symmetry and fractal dimension ("coastline approximation" - 1). For each of these 10 features, their mean, standard error and mean of the three largest value were calculated (for instance, field 3 is Mean Radius, field 13 is Radius SE, field 23 is Largest Radius). Therefore, there are 3 different values for each of these 10 features, resulting in 30 attributes.

Out of the 569 individuals, 357 are from the benign class and 212 from the malignant class. There are no missing values.

In this report, we will go through : pre-processing, the protocol of validation, ...

2. The pre-process of data

The first task we have had to do is the pre-processing of the data. Even though in the description of the dataset, it was written that there were no missing values, we quickly verified this information by looking at the summary of the data and indeed, there were none.

There was no information on the outlier. We did a boxplot to detect univariate outliers, giving the following result.

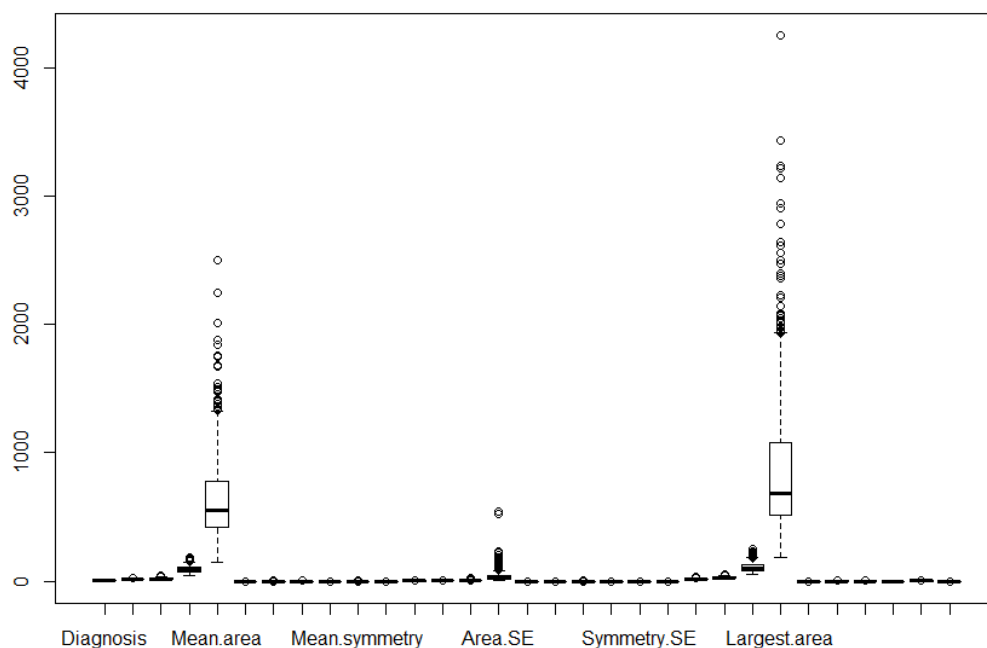


Figure 1, Boxplot

As we can see, we can decipher some outliers on some variables, such as Mean.Area, Area.SE and Largest.Area. Indeed, some values seem to be really outside their respective boxplot. To deal with this and detect multivariate outliers, we are going to use the Mahalanobis distance. As a reminder, this distance computes the distance between a point and the mean of the distribution. It is very useful as it is scale-invariant.

We computed this distance with the function Moutlier and we got several information. First of all, threshold that was computed was of 6.85. However, it excluded 83 individuals, which is almost 15% of the dataset. This is too much so we decided to have a look at the plot to have an idea to where we should cut.

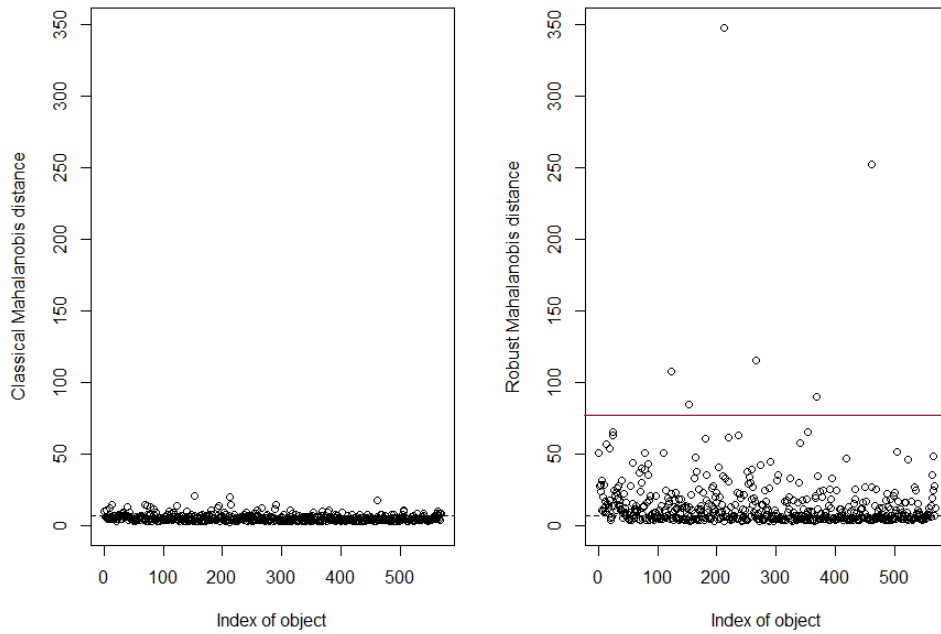


Figure 2, Mahalanobis distance

When computing the Mahalanobis distance, we get the classical distance and the robust one. It turns out that the robust distance is suited to detect outlier, and we can see why above. Indeed, we see that some values are way out the others. In order to make a “clean” cut, we used 75 as the new threshold, which seems to separate clear outliers from the rest. This means that if an individual has a Mahalanobis distance greater than 75, it will be removed. So after identifying 6 outliers and removing them (563 instances left), we get the following boxplot.

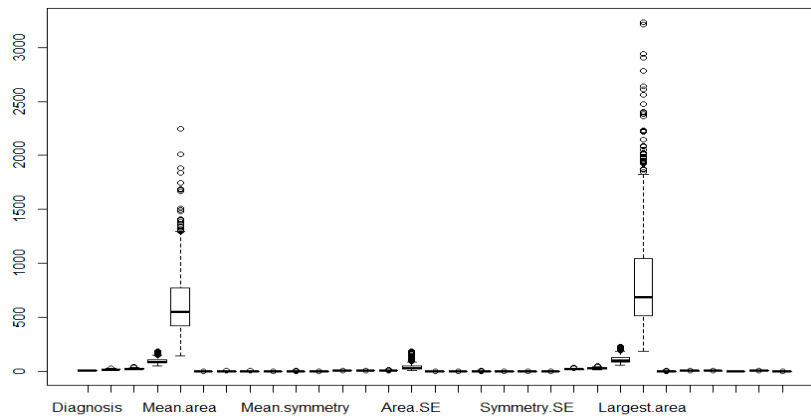


Figure 3, Boxplot without outliers

If we compare this one to the previous one, we see for instance that the outliers for Area.SE are now gone. Although we also see that for variables like the Largest.Area there are still some values that seem a bit off, we won’t consider them as outliers.

Another preprocessing aspect would be the feature selection. This is useful to only keep the relevant features and to get rid of the superfluous variables. We will discuss later about this.

3. The protocol of validation

In order to test our models and chose one, we will need to have data for the training of the model and data for testing it. We will split the dataset randomly into two parts: $\frac{2}{3}$ for the training sample (377 individuals) and $\frac{1}{3}$ for the test sample (186 instances).

Then, we will compute the training and test errors, over different choice of parameters and choose the best one according to what makes sense in the medical context of the diagnostic of breast cancer. We will also test the stability of the models.

4. The visualisation performed

To have a sense of what is the dataset we are working on, we decided to do a Principal Component Analysis. To do so, we placed the only categorical variable Diagnostic as supplementary.

In order to select the number of significant dimensions we are going to keep, we have two choices, either look directly at the eigenvalues or compute the screeplot. You can find both right below.

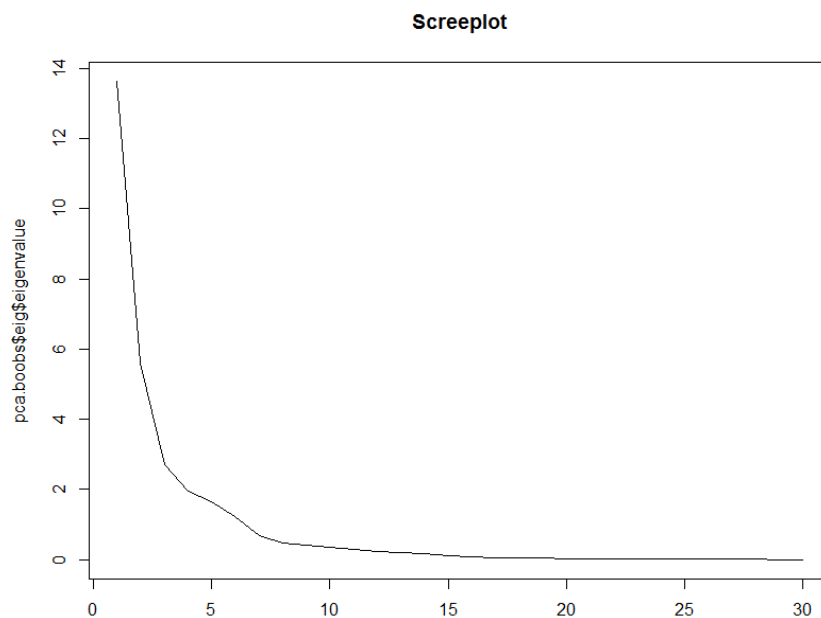


Figure 4, Screeplot

The screeplot doesn't really help us here because there are too many values. We cannot see exactly where the last elbow is. This is why we will determine the number of principal components with the cumulative percentage of variance kept by the eigenvalues.

	eigenvalue	percentage of variance	cumulative percentage of variance
comp 1	1.362577e+01	4.541922e+01	45.41922
comp 2	5.590598e+00	1.863533e+01	64.05455
comp 3	2.690433e+00	8.968109e+00	73.02265
comp 4	1.946031e+00	6.486769e+00	79.50942
comp 5	1.647113e+00	5.490375e+00	84.99980
comp 6	1.236008e+00	4.120027e+00	89.11983
comp 7	6.660001e-01	2.220000e+00	91.33983
comp 8	4.839030e-01	1.613010e+00	92.95284
comp 9	4.193505e-01	1.397835e+00	94.35067
comp 10	3.545719e-01	1.181906e+00	95.53258
comp 11	2.768443e-01	9.228144e-01	96.45539
comp 12	2.422829e-01	8.076097e-01	97.26300
comp 13	1.998097e-01	6.660324e-01	97.92903
comp 14	1.582819e-01	5.276063e-01	98.45664
comp 15	9.593278e-02	3.197759e-01	98.77642
comp 16	8.241372e-02	2.747124e-01	99.05113
comp 17	5.583851e-02	1.861284e-01	99.23726
comp 18	4.691747e-02	1.563916e-01	99.39365
comp 19	3.560132e-02	1.186711e-01	99.51232
comp 20	3.008188e-02	1.002729e-01	99.61259
comp 21	2.924338e-02	9.747794e-02	99.71007
comp 22	2.199382e-02	7.331273e-02	99.78338
comp 23	2.027997e-02	6.759991e-02	99.85098
comp 24	1.508373e-02	5.027911e-02	99.90126
comp 25	1.109159e-02	3.697197e-02	99.93823
comp 26	8.365228e-03	2.788409e-02	99.96612
comp 27	7.767686e-03	2.589229e-02	99.99201
comp 28	1.687197e-03	5.623989e-03	99.99764
comp 29	5.682749e-04	1.894250e-03	99.99953
comp 30	1.411866e-04	4.706221e-04	100.00000

Figure 5, Result of PCA

If we look at the eigenvalues, we see that the first 4 dimensions account for almost 80% of the total inertia, so we are going to keep 4 principal components.

Thanks to the PCA, we can also have a look to the most influencing variables in the significant dimensions. If we look those in the first dimension, we have the following:

Mean. concave. points	Mean. concavity	Largest. concave. points	Mean. compactness
6.635063e+00	6.503763e+00	6.261584e+00	5.630349e+00

Figure 6, Most influencing variables of the first dimension

From what we see here, we could assume that the mean concave and mean concavity are redundant. However, since we are not expert in this field, we cannot say with certainty they are, so we are not going to filter them.

5. The interpretation of the latent concepts.

We are now going to look at the latent concepts of the first two significant dimensions. It should give us a good enough view, even for people without medical background.

\$Dim.1		
\$Dim.1\$quant1		
	correlation	p.value
Mean.concave.points	0.9508302	9.484050e-288
Mean.concavity	0.9413753	6.527666e-267
Largest.concave.points	0.9236821	6.880944e-236
Mean.compactness	0.8758870	1.083371e-179
Largest.perimeter	0.8703705	9.471236e-175
Largest.concavity	0.8404323	2.180429e-151
Largest.radius	0.8382439	7.146231e-150
Mean.perimeter	0.8291452	8.333647e-144
Largest.area	0.8291304	8.519156e-144
Area.SE	0.8141405	1.512448e-134
Mean.area	0.8022565	8.624766e-128
Mean.radius	0.7965353	1.061808e-124

Figure 7, Latent concepts in the first dimension

For the first dimension, we see that the concavity, the compactness and the perimeter are the most correlated with the principal component. We can also see that the p-values are really low.

\$Dim.2		
\$Dim.2\$quant1		
	correlation	p.value
Mean.fractal.dimension	0.87149910	9.641120e-176
Fractal.dimension.SE	0.66618774	1.732456e-73
Largest.fractal.dimension	0.64362508	3.632234e-67
Compactness.SE	0.55427425	1.171713e-46
Smoothness.SE	0.52935772	5.573938e-42
Symmetry.SE	0.46207754	3.988619e-31
Mean.smoothness	0.45916493	1.043336e-30
Concavity.SE	0.44823897	3.544752e-29
Mean.symmetry	0.43961772	5.238842e-28
Largest.smoothness	0.41794646	3.275362e-25
Mean.compactness	0.36574372	2.921205e-19

Figure 8, Latent concepts in the second dimension

For the second dimension, the fractals variables are the most importants. As a reminder, the fractal dimension represents the coastline approximation. The coastline approximation is the phenomenon that states that because we use Euclidean distance to measure the length of the coastline of objects, such as the cells in our case, we can not have the perfect size of them, which is why it is an approximation. So this dimension is mainly centered on the size of the cells, which could mean that the presence or not of cancerous cells depend on the size of them.

6. The clustering performed

To synthesize the informations that we just saw, we decided to perform a clustering. To do so, we did a sequential clustering, which can be defined in three points: we first run a hierarchical algorithm, then we decide the number of clusters we want to keep and we calculate the centroids of said clusters. Finally the perform a k-means algorithm using the centroids found previously.

After computing the hierarchical algorithm, we get the following barplot.

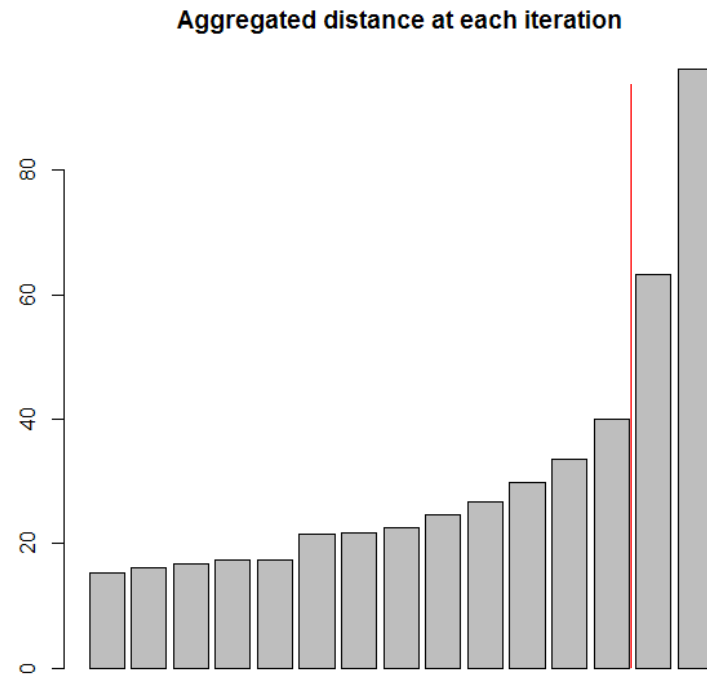


Figure 9, Barplot

We are going to cut where the line is. It is the last big jump so it seems a good place to cut. With this plot, we determine that we are going to keep 3 clusters.

7. The interpretation of the found clusters.

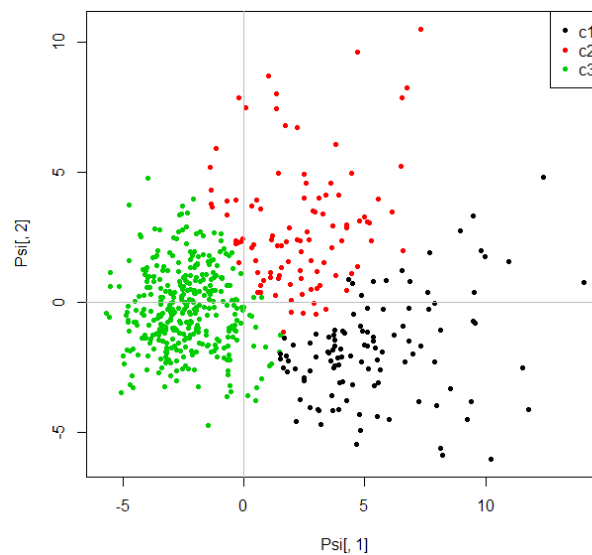


Figure 10, Clusters with consolidation operation

Finally, we find the clusters above. As we said before, it is a bit delicate to tell with precision what the clusters represent as we are not expert for this dataset. However, we can define the clusters as follow :

- Cluster 1 : the individuals in this group tends to have cells that are more concave and more smooth. The cells seem to be smaller. It could be perceived as Benign.
- Cluster 2 : the cells of this cluster appear to have a high length.
- Cluster 3 : here the cells don't seem to be large or to have asperity. The cluster seems to have more individuals in it, it could be perceived as Benign as well.

8. Results obtained with the assignment of the test individuals.

We will now focus on building a model able to predict the outcome of the Diagnosis variable (M for Malignant or B for Benign). One of the most used and popular classifier is the decision tree, because of its interpretability and easy implementation. We will use the CART (Classification and Regression Trees) algorithm and have a look at its predictions.

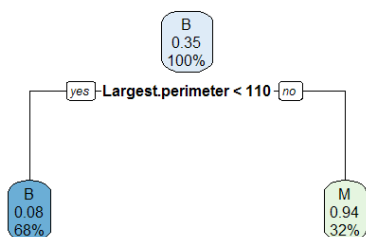
As said before, we split the dataset randomly into two parts: $\frac{2}{3}$ for the training sample (377 individuals) and $\frac{1}{3}$ for the test sample (186 instances).

We use the rpart function and set a low complex parameter (0.001) in order to build the maximal tree that we'll be able to prune later. We decide to do a 10-fold the cross. After pruning, the optimal tree obtained is the following:

The tree is very simple and has a good interpretability.

Let's also look at the confusion matrix over the training set:

Optimal Tree



```
dec.learn
pred_B.learn pred_M.learn
B          237          7
M          20          113
```

And over the test set:

```
dec.test
pred_B.test pred_M.test
B          102          10
M           9           65
```

Figure 11, First decision tree

The generalization is not that bad with a 10.2% error (89.8% accuracy) but there are more than 13% of Malignant cases that has not been detected. This is a lot, especially in a medical field. It is much worse to have cases where you don't detect the anomaly where there is one than the opposite.

To see how the model will behave if we change the learning data, and therefore to test the stability of our model, let's change the random seed and do again the sampling for the training and test sets. We proceed again to the same analysis over the new random sample, here is the optimal tree (after pruning). There is still only one used variable but this is not the same as the first time.

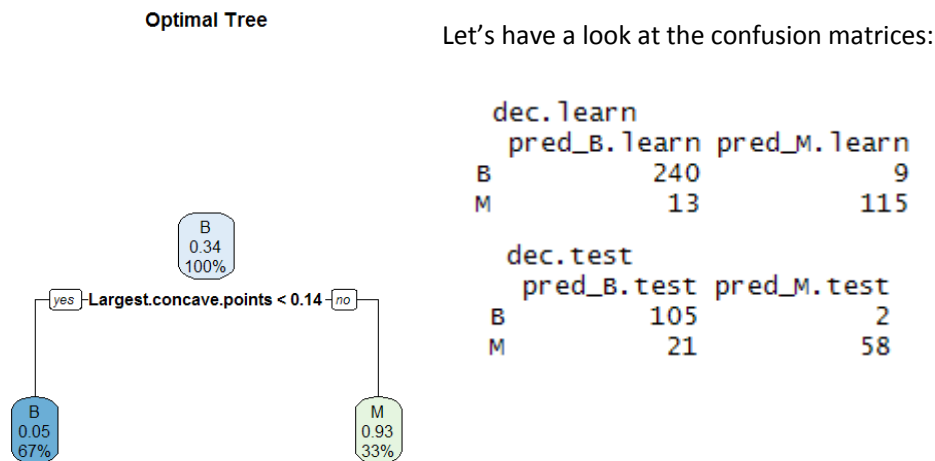


Figure 12, Second decision tree

The error of non detecting a Malignant is about 26%, which is huge and twice as big than the first model. Too many real Malignant are predicted as Benign, we need to handle this. We changed once more the random seed and obtained an optimal tree with 4 final leaves this time.

Over these 3 different sampling, we can see that the CART algorithm is pretty unstable, this is indeed the most important flaw of the decision trees algorithms. We saw that this can lead to a bad generalization. This high variability can also be due to our data but we already took care of the most extreme outliers. The CART algorithm might have a responsibility into this instability.

Fortunately, the Random forest provides a way out of this: by using Bootstrap Aggregating (also called Bagging) over decision trees, it decreases the variance without increasing the bias of the model. Therefore, the model is more stable and still has a good accuracy.

We wished to do some feature selection to improve the use of the Random forests, selecting the most important variables by looking at the importance of the variables, but there is a lot of variability in the different barplot obtained with the single tree rpart function so we decided not to attempt to do feature selection.

We set the number of trees to do at 100. This should allow us to have a good performance and stability of the model. For the number of features randomly sampled as candidates at each split, the dataset has 29 attributes and this is a classification problem, therefore we set its value at $\sqrt{29}$, which is approximately equal to 5.

The outcome of the random forest with these parameters is the following:

```

Type of random forest: classification
Number of trees: 100
No. of variables tried at each split: 5

OOB estimate of error rate: 3.98%
Confusion matrix:
  B  M class.error
B 244  5 0.02008032
M  10 118 0.07812500
Test set error rate: 1.55%
Confusion matrix:
  B  M class.error
B 129  1 0.007692308
M   2  62 0.031250000

```

Figure 13, Result of Random Forest

On the test set, the error is really low, way better than with a single tree. There are only 3% Malignant that were not detected. But the OOB errors aren't appropriate, we'll talk about this in the next part.

9. The prediction model with its best parameterization

We can see by looking at the OOB errors from our previous model that there are almost 4 times more false Benign than false Malignant (2% relative to 7.8%). We will try to balance this, or even try to have less false Benign because in real life this is more dangerous not to detect a Malignant than to detect a Benign as a Malignant.

One way to do this is to use more training data with the Malignant class than data with the Benign class.

In our training data, we have 143 Malignant and 226 Benign. We will use the 143 Malignant and 100 Benign out of the 226 to compute the random forest. The results are the following:

```

OOB estimate of error rate: 5.31%
Confusion matrix:
  B  M class.error
B 220 14 0.05982906
M   6 137 0.04195804
Test set error rate: 4.84%
Confusion matrix:
  B  M class.error
B 116  6 0.04918033
M   3  61 0.04687500

```

Figure 14, Result of Random Forest

It worked on the OOB errors: there are now about 4% false Benign, that's about twice less as with the previous model. But the false Malignant have increased, there are about 3 times higher than before (6% relative to 2%). On the test set, 1 more individual than before is false Benign and 5 more than before were predicted false Malign.

Because we don't have a lot of data and we are using less data than before, maybe the number of tree used is not sufficient. We will now try to optimize the number of trees. Below are the values of the OOB errors over 6 values of number of trees. The OOB is minimized when the number of trees is 200.

	ntrees	OOB
[1,]	50	0.04774536
[2,]	100	0.05305040
[3,]	200	0.03978780
[4,]	300	0.04244032
[5,]	400	0.04774536
[6,]	500	0.04509284

Figure 15, OOB errors for Random Forests

10.The final model and its generalization error

The model with its best parametrization is then the random forest with 5 features randomly sampled as candidates at each split, a training sample composed of more Malignant than Benign and 200 trees grown. It gives us the following result:

```

OOB estimate of error rate: 4.24%
Confusion matrix:
  B  M class.error
B 223 11 0.04700855
M  5 138 0.03496503
Test set error rate: 4.84%
Confusion matrix:
  B  M class.error
B 116 6 0.04918033
M  3 61 0.04687500

```

The model focuses on doing good prediction on the Malignant. Its OOB error is about 3.5% and test error about 4.7%. But it comes at a cost: there are more errors on the Benign data, about 4.7% for the OOB and 4.9% on the test set. The generalization error is 4.84%.

The three most important variables are the largest perimeter, the largest radius and the mean of concave points.

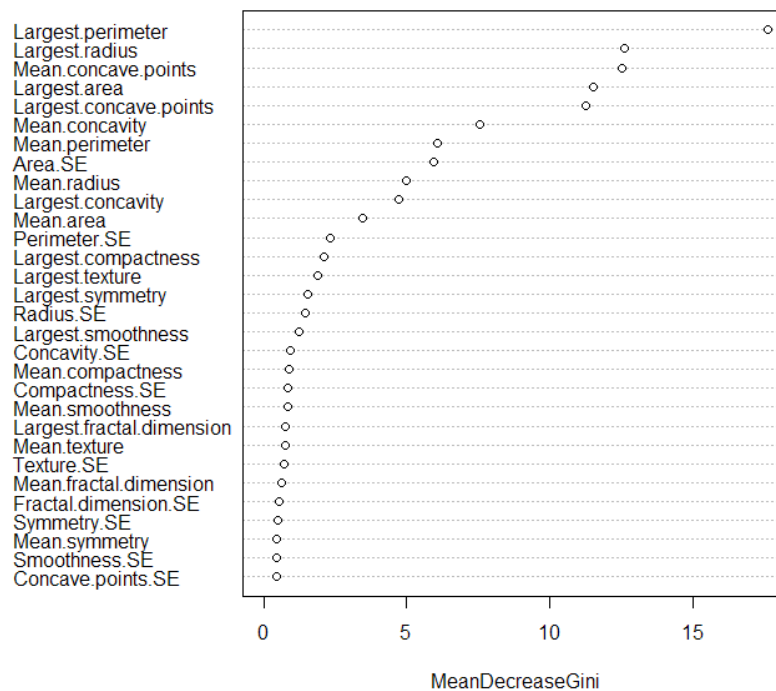


Figure 16, Importance of the variables for the final model

11. Scientific and personal conclusions

We've seen that the unstability of a model and the variance of a dataset can lead to bad generalization and therefore it is relevant to use methods improving the model stability. The random forests for example, which uses bagging, improved significantly our predictions of the diagnosis of breast cancer, compared to single tree models.

The accuracy of the random forest over our dataset is good but we noticed that too much Malignant cases were not detected, which is dangerous for the patient. Therefore, we gave priority to the Malignant data in the building of the model, which resulted into a decreasing of the not detected Malignant cases.

This project has been useful because we have been able to practice everything we have seen this semester, preprocessing, data visualisation, clustering, modelling. It also made us realize that it is quite difficult to analyze data without an expert, especially in the medical field.