Cannabis Genetics

Variety classification using data analysis methods

Marc Vernet Sancho

Jordi Puig Rabat

UPC

5/2020

Summary

Introduction

The objective of this project is to analyze the dataset developed in the research article *Broad-Scale Genetic Diversity of Cannabis for Forensic Applications*[1].

Cannabis is one of humanity's oldest cultivated plant and a very wide spread crop across the world. It can be used either for several industrial purposes such as animal food, fibre or oil (under the name of *hemp*) or for recreational or medicinal purposes (*marijuana*). The psychoactive and non-psychoactive varieties differences mainly in the proportion of two of its main components: THC and CBD. High THC:CBD varieties are prohibited, considered illicit drug sources and low THC:CBD varieties can be exploited under licensed control for industrial purposes. The cultivation and possession of *Cannabis* is under strict legal regulations, so the objective of the dataset of the research article is meant to be a summary of the most important genetic information necessary to determine if a specific plant is psychoactive or non-psychoactive.

The database consist of 1324 plants from 48 ascensions or families of *Cannabis* (30 fibre ascensions and 18 drug ones) each one with information numeric information of 13 microsatellite loci pairs.

Our main purpose is to use this dataset to learn to distinguish between psychoactive *Cannabis* varieties and non-psychoactive varieties based on their genotype data. Thus, with data from the 13 provided microsatellite we are going to train a model that classifies *Cannabis* plants into two groups: Fibers and Drugs. Before designing such model we are going to explore the data to check that this classification is correct, i.e. the two groups are different in a genetic sense and that this grouping criterion is good enough to train a classifier.

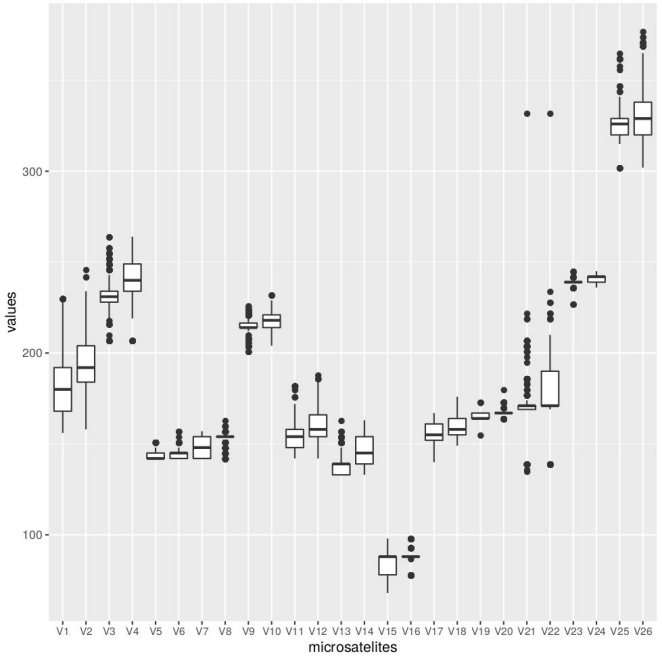
[1] Dufresnes C, Jan C, Bienert F, Goudet J, Fumagalli L (2017) Broad-Scale Genetic Diversity of Cannabis for Forensic Applications. PLoS ONE 12(1): e0170522.

Preprocessing

It's usual in genetic data to have a big proportion of missing data. In the case of the dataset used in this project, it has around 12% of plants in the dataset that have some missing value. The possible solutions for missing data can be either eliminating the plants with missing values or filling them with some approximate data.

In the box plot of each microsatellite numeric value, can be seen that in most of the cases the range of values for each microsatellite are very concentrated although there are some outliers. So a good way to approximate missing values could be with some statistic measure for each microsatellite. Because in some cases there are some extreme outliers, if we use the average the approximation could be biased and be incorrect. We will use the median of each microsatellite, because it will represent a more central element.

In the boxplot it can also be seen that there isn't any strange data that should be checked, because even the most extreme outliers are inside the range of possible values.



In addition to that the data we were given needed a recodification before we were able to analyze it. A microsatellite is a tract of repetitive DNA in which certain DNA motifs (ranging in length from one to six or more base pairs) are repeated[2]. We had the number of times each

[2] https://en.wikipedia.org/wiki/Microsatellite

DNA sequence was repeated –this is called an allele–, but this is not ordinal data so we recodified the data to a table counting the number of times each allele appeared on every microsatellite for every observation. Imagine we had a table like this, with 2 microsatellites and this observations:

ID	A201		A301	
A_01	204	206	246	249
A_02	164	228	207	249
A_03	206	206	234	246

We create a column for every allele that appears in each microsatellite and count them like this:

	A201				A301			
ID	164	204	206	228	207	234	246	249
A_01	0	1	1	0	0	0	1	1
A_02	1	0	0	1	1	0	0	1
A_03	0	0	2	0	0	1	1	0

Note that we have enlarged the number of columns of our database. Assuming that we had 13 Microsatellites and that each microsatellite had n_i factor levels, now we have the sum of n_i for i between 1 and 13 columns. Moreover our dataset is now very disperse considering we have rows with a lot of values equal to 0.

Used methods

Clustering

The first method used to explore the data is clustering analysis to check that Fiber and Drugs is a natural grouping of the data. One way to know about natural groups in the data is performing various k-means clustering iterating through k, then compute the Calinski-Harabasz pseudo F-statistic in order to compare the goodness for every different clusters by its *k* parameter.

Another clustering method will be hierarchical clustering, the distance matrix will be computed with the **manhattan** distance and we will use the Ward criterion.

Multidimensional Scaling

To visualize the data in a 2D plane it is useful to apply this method. This preserves the distances of the observations and gives us the data in 2 dimensions. To calculate the distance we have considered the **Manhattan** distance because we are considering it counts occurrences of the different alleles.

Principal Component Analysis

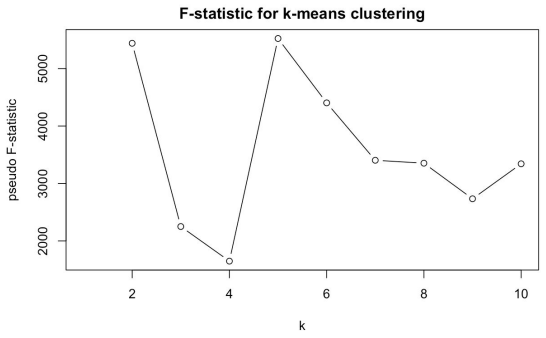
In order to reduce the number of dimensions of the data we can compute the principal components of the data matrix and choose a subset of the components to represent the data. Once we have a data matrix with few columns we can feed a model with this data and make predictions over the target variable. There are various models we can choose but we will only use Generalized Linear Models, this tool can be very useful to classificate in two classes. When the link of the model is the **logit** function it works as a classifier: when the response is over or equal 0.5 we classify this as non-drug, otherwise it is classified as drug. To compute the model accuracy we have considered two subsets of the data given, the training set which contains ⅔ of the observations and the test set which contains the other ⅓. We calculate the accuracy rate making predictions over the test set.

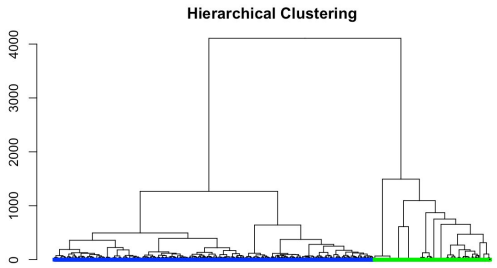
Cross Validation

There are several approaches to calculate the test error of a model like our classifier. One way to do it is the k-fold Cross Validation, we learned this method in another subject (AA1) and we think it is a good idea to apply it here because it gives us a reliable approximation of the goodness of our model. This method consists in training k times a model, each time you leave the i-th fold of the data and then you compute the accuracy with this subset.

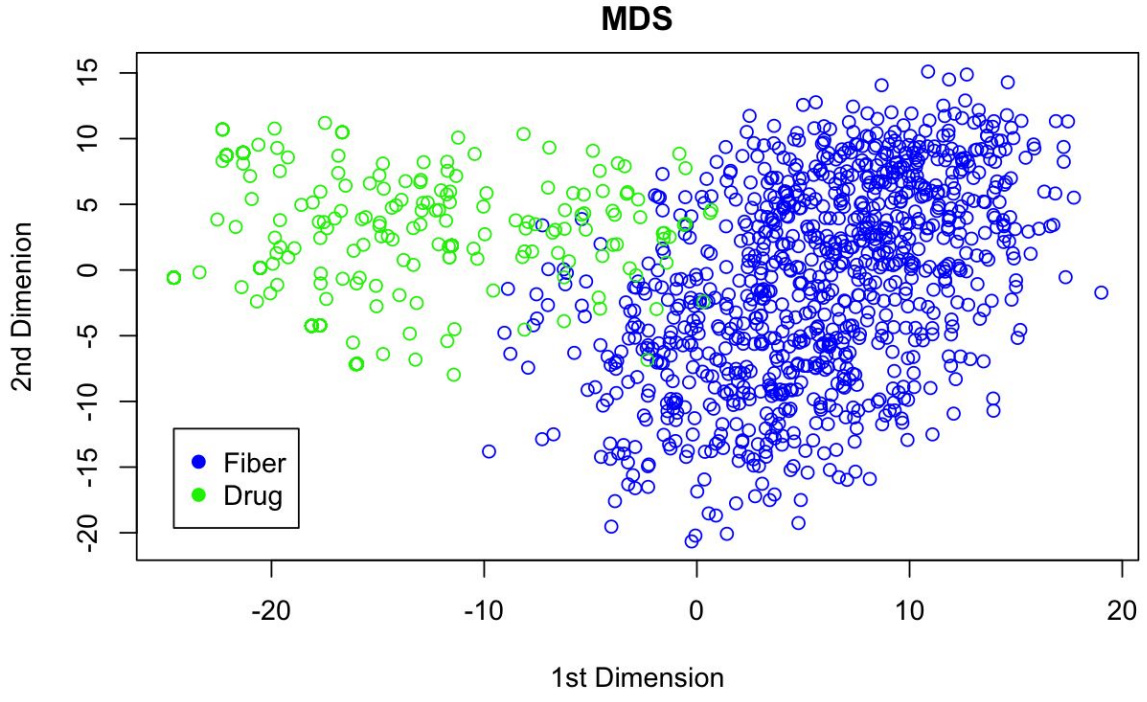
Results

Here we can see the results obtained with the methods explained before, both graphs show us that our grouping criterion is reasonable. In the first we can see that the value of the F-statistic is high when the cluster is made with two groups; the value when k = 5 is high as well, we think it is due to the sub varieties within each group. In the second graph we have plotted the dendrogram of the hierarchical clustering painting the two colors of each class. We see that this cluster is almost perfect when we cut the tree in two groups.





Another way to visualize the data is the MDS plot as explained before. Here it can be seen that the first dimension explains a lot more of the difference between classes than the second. We see that there is a region where the two point clouds intersect, this will not bring great problems as this intersectional region contains few points.

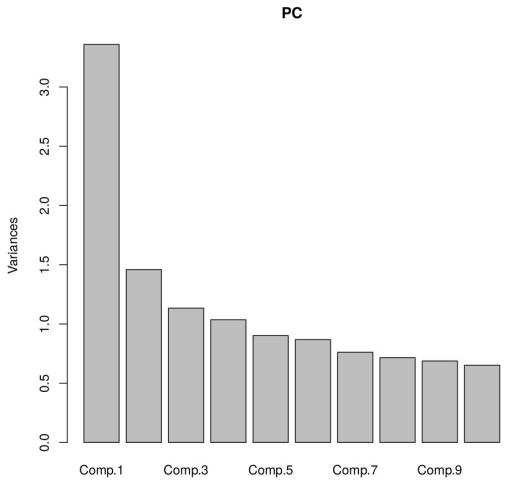


In order to know if the models developed are good or simple enough or how the performance relates among them is necessary to have some base model to use as comparison. In this case, we can look at the full model. This model uses all of the 174 variables in the dataset to determine the variety of Cannabis. What we can expect of this model is that it's going to have a very high train error (because of extreme overfitting).

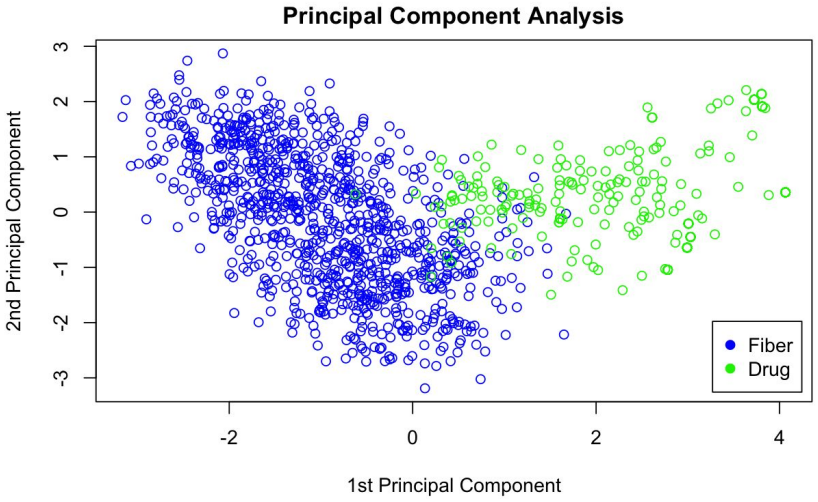
```
mod <- glm(dataTrain$variety ~., data=dataTrain[,2:174],
family=binomial(link=logit))

Train accuracy: 100%

Test accuracy: 98.41%
```

As predicted, the train accuracy is 100% and the test accuracy is also very high. It seems that although the apparent overfitting the model has a good response when generalized. So it seems to be a good model with very good performance, but it's too complex (needs 174 variables) and perhaps simpler models with similar performance can be found.

A way to design simpler models could be selecting less variables to be part of the model, but as there are a lot of variables, each one with small importance is difficult to choose which to keep. If we use the PCA , we can easily select the data that explains more variance.

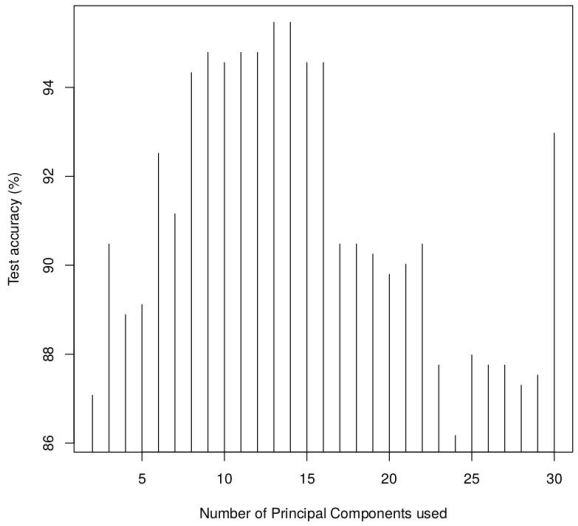


The scree-plot represent the amount variability that explains each component of the PCA. As it can be seen the first component has a big importance in comparison with the rest. If we consider the two first principal components about 19% of variance is explained (which is usually considered a good amount in genetic datasets). The separation of classes can easily be seen in the plot of these two components.



Different models can be calculated using a different number of PCA components. These models are going to have worse accuracy than the full model because they use less information, but the accuracy should be good enough considering that more than 20% of variance will be used.

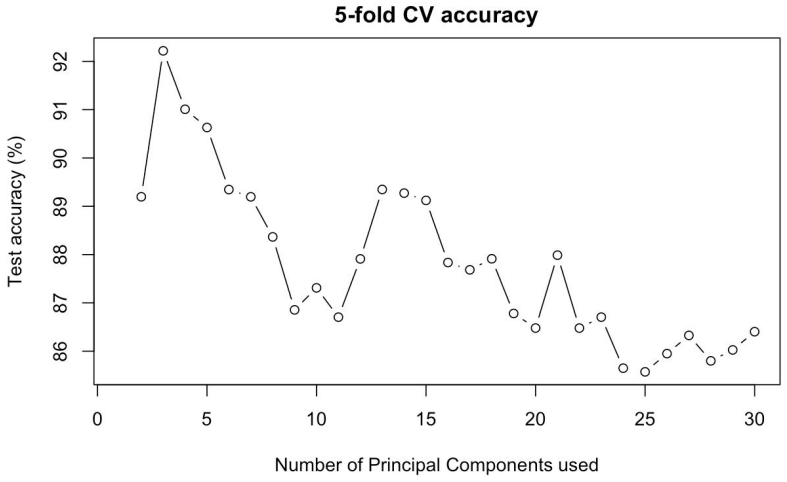
We have calculated the models using a different range of principal components, from 2 to 30. The test accuracy is represented in the histogram. This accuracy has been calculated splitting up the data in two subsets, one used for training and the other for testing the predictions.



In general the accuracy is good enough, being higher that 88% in most cases. It can be seen that the accuracy grows as the number of principal components used grows until a limit is reached at about 15 components used. It can be interpreted that from that moment, using

too much principal components makes the model overfitted, meaning that it will have worse performance on test data.

With the 5-Fold Cross Validation method we have obtained slightly different results, this could be due to the randomness of the folds and the partition in the first method. In this case we observe that the best accuracy is obtained when using 3 components, and then the model seems overfitted as before.



Conclusion and discussion

With the application of some methods learned in class, we have performed a complete exploration of the data and we have designed a classifier which was our main goal to achieve. Having a model that classifies in two groups shows it is possible to distinguish between varieties with genetic data. With clustering methods it is possible to explore the grouping nature in the data, it is important to make sure there are groups within your data when you want to design and train a classifier.

Our best model performance is when using 3 Principal Components and a Generalized Linear Model with the logit link to predict the target variable. We get a 92.2 % test accuracy which is fairly good.

It is not easy at all to work with genetic data, it requires the understanding of genetic concepts used in this report and the knowledge of how to deal with this kind of data. Our firsts approaches without recodifying the data into factors were good as well, but far from the rigorous ways we seek.

One method we left in this project is the Linear Discriminant Analysis because we couldn't make a training subset without high correlation variables (considered constants by R), when this occurs it is impossible to calculate the correlation matrix and therefore it is impossible to train a model with this method. Although we had obtained good results in terms of training accuracy, we have decided not including this method in our report as a possible way to design a classifier for this reason.

In the annexed files you can find all the tools we have used for recodify and plotting the data. There are two data files, the original one and the recodified. All files are briefly commented and should be enough to understand how we have done each part of the analysis.