pcadapt

karine Durand

9 avril 2019

# Using pcadapt to detect local adaptation

# A. Reading genotype data

library(pcadapt)  
library("pcadapt", lib.loc="~/R/x86\_64-pc-linux-gnu-library/3.3")  
setwd ("/home/kadurand/Applications/PCAdaptPackage/pcadapt-master/")  
  
filename <- read.pcadapt("/home/kadurand/Applications/PCAdaptPackage/pcadapt-master/pauca\_ordered.recode\_diplo.PCAdapt", type = "pcadapt")

## 41881 lines detected.  
## 19 columns detected.

## The bed file already exists. Returning..

# B. Choosing the number K of Principal Components

An important parameter of the pcadaptfunction is the parameter K. A way to choose K is to use a scree plot which consists of plotting the percentage of explained variance for each PC. The ideal pattern in a scree plot is a steep curve followed by a bend and an almost horizontal line. The recommended value of K corresponds to the largest value of K before the plateau is reached.

###### Une autre option pour choisir le nombre de PC est basée sur le «plot de partition» qui affiche la structure de la population. Le graphique affiche les projections des individus sur les composantes principales spécifiées. En utilisant le graphique de score, le choix de K peut être limité aux valeurs de K qui correspondent à un niveau pertinent de la structure de la population.

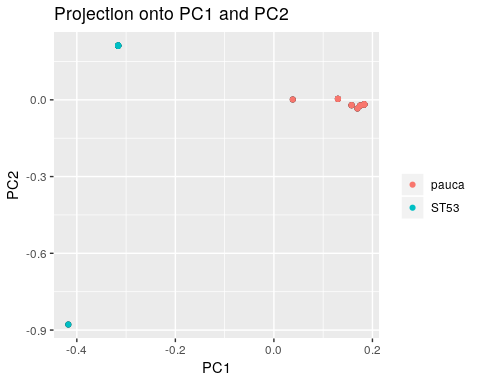
poplist.int <- c(rep(1, 13), rep(2, 6))  
poplist.names <- c(rep("pauca", 13),rep("ST53", 6))  
print(poplist.int)

## [1] 1 1 1 1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2

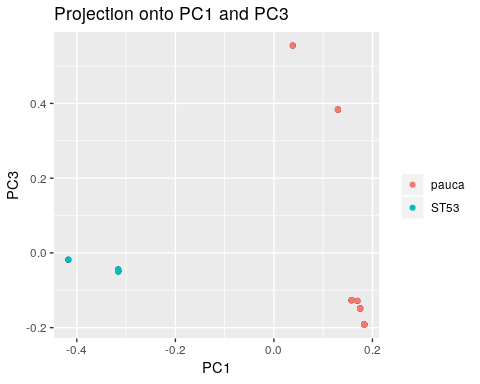
print(poplist.names)

## [1] "pauca" "pauca" "pauca" "pauca" "pauca" "pauca" "pauca" "pauca"  
## [9] "pauca" "pauca" "pauca" "pauca" "pauca" "ST53" "ST53" "ST53"   
## [17] "ST53" "ST53" "ST53"

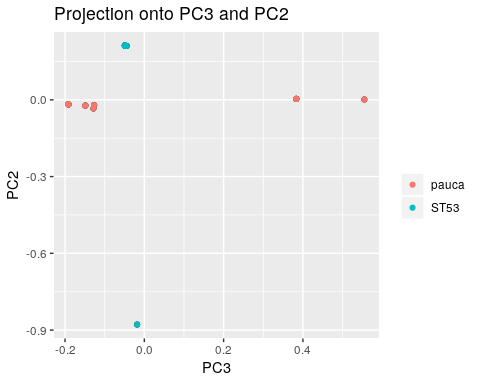
#plot(x, option = "scores", pop = poplist.int)  
#plot(x, option = "scores", pop = poplist.names)  
####Looking at population structure beyond K = 2 confirms the results of the scree plot. The third and the fourth principal components do not ascertain population structure anymore.  
par.default <- par(mfrow = c(3,1))  
plot(x, option = "scores", i = 1, j = 2, pop = poplist.names)



plot(x, option = "scores", i = 1, j = 3, pop = poplist.names)



plot(x, option = "scores", i = 3, j = 2, pop = poplist.names)



par(par.default)

# For the pauca data , it was found in section B that K=2 corresponds to the optimal choice of the number of PCs.

In addition to the number K of principal components to work with, the user can also set the parameter min.maf that corresponds to a threshold of minor allele frequency. By default, the parameter min.maf is set to 5%. P-values of SNPs with a minor allele frequency smaller than the threshold are not computed (NA is returned).

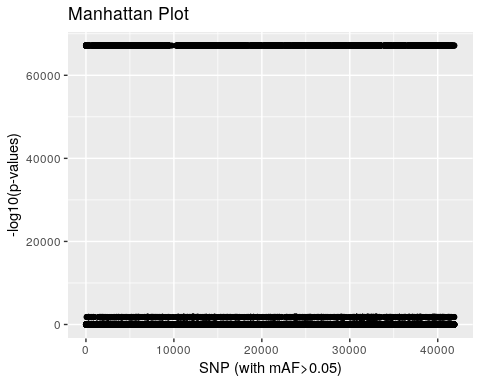
x <- pcadapt(filename, K = 2, min.maf=0.05)  
summary(x)

## Length Class Mode   
## scores 38 -none- numeric  
## singular.values 2 -none- numeric  
## loadings 83762 -none- numeric  
## zscores 83762 -none- numeric  
## af 41881 -none- numeric  
## maf 41881 -none- numeric  
## chi2.stat 41881 -none- numeric  
## stat 41881 -none- numeric  
## gif 1 -none- numeric  
## pvalues 41881 -none- numeric  
## pass 41881 -none- numeric

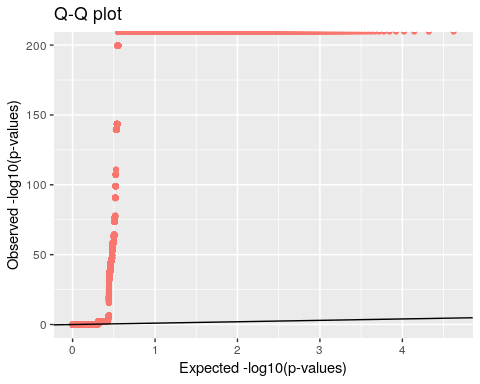
## D. Graphical tools

D.1. Manhattan Plot A Manhattan plot displays −log10−log10 of the p-values.

plot(x , option = "manhattan")

 #D.2. Q-Q Plot The user can also check the expected uniform distribution of the p-values using a Q-Q plot This plot confirms that most of the p-values follow the expected uniform distribution. However, the smallest p-values are smaller than expected confirming the presence of outliers.

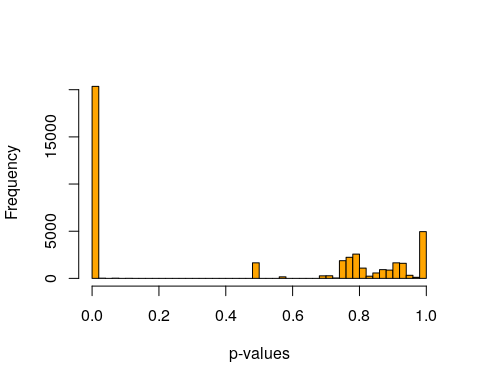
plot(x, option = "qqplot")



# D.3. Histograms of the test statistic and of the p-values

An histogram of p-values confirms that most of the p-values follow an uniform distribution. The excess of small p-values indicates the presence of outliers.

hist(x$pvalues, xlab = "p-values", main = NULL, breaks = 50, col = "orange")

 #The presence of outliers is also visible when plotting a histogram of the test statistic DjDj.

##plot(x, option = "stat.distribution")  
#Error in hist\_plot(x, 1) : Can't display the histogram as the values are too high.

# E. Choosing a cutoff for outlier detection

To provide a list of outliers and choose a cutoff for outlier detection, there are several methods that are listed below from the less conservative one to the more conservative one.

#source("https://bioconductor.org/biocLite.R")  
#biocLite("qvalue")  
library(qvalue)

For a given αα (real valued number between 00 and 11), SNPs with q-values less than αα will be considered as outliers with an expected false discovery rate bounded by αα. The false discovery rate is defined as the percentage of false discoveries among the list of candidate SNPs. Here is an example of how to provide a list of candidate SNPs for the geno3pops data, for an expected false discovery rate lower than 10%:

library(qvalue)  
qval <- qvalue(x$pvalues)$qvalues  
alpha <- 0.1  
outliers <- which(qval < alpha)  
length(outliers)

## [1] 20374

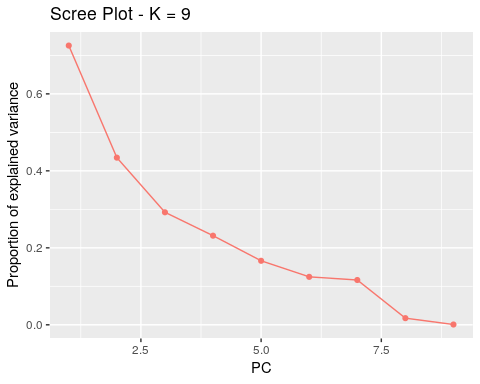
# F. Linkage Disequilibrium (LD) thinning

Linkage Disequilibrium can affect ascertainment of population structure (Abdellaoui et al. 2013). When working with RAD-seq data, it should not be an issue. However, users analyzing dense data such as whole genome sequence data or dense SNP data should account for LD in their genome scans.

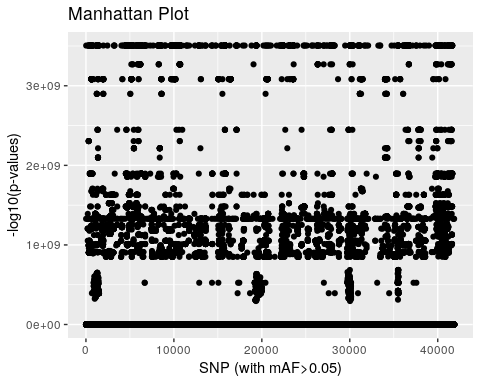
In pcadapt, there is an option to compute PCs after SNP thinning. SNP thinning make use of two parameters: window size (default 200 SNPs) and r2r2 threshold (default 0.1). The genome scan is then performed by looking at associations between all SNPs and PCs ascertained after SNP thinning.

We provide below an example of data analysis using simulated data where there is a LD. Genotype data are stored in a matrix (individuals in columns and SNPs in lines). An analysis without SNP thinning can be performed as follows

res <- pcadapt(input = filename, K = 9)  
plot(res, option = "screeplot")

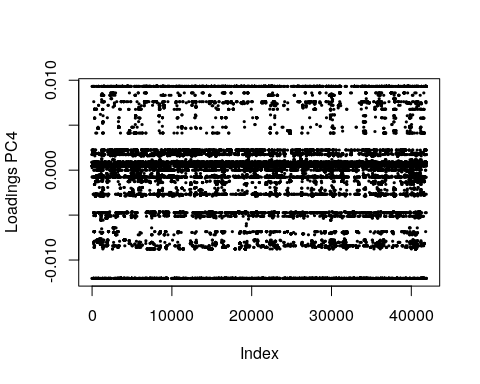
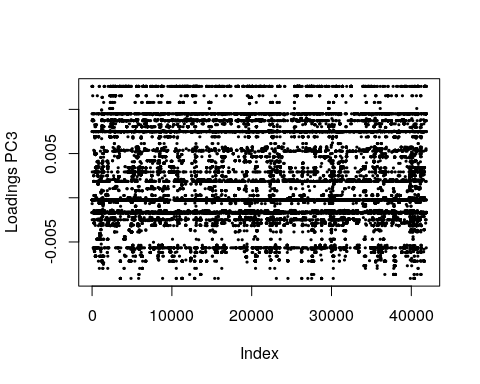
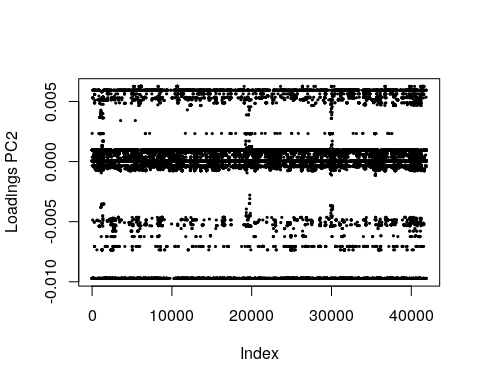
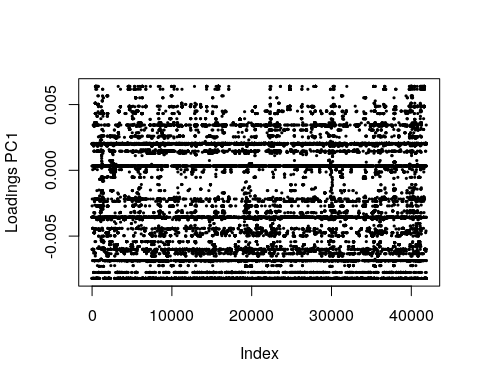


plot(res)



To evaluate if LD might be an issue for your dataset, we recommend to display the loadings (contributions of each SNP to the PC) and to evaluate if the loadings are clustered in a single or several genomic regions.

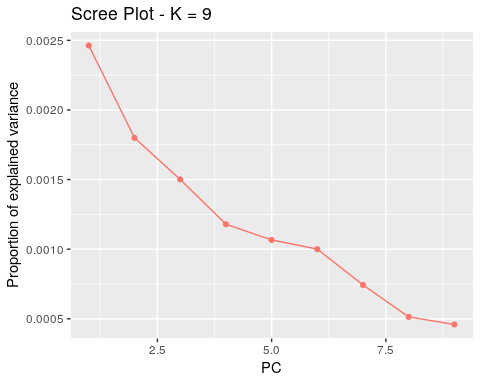
#par(mfrow = c(2, 2))  
for (i in 1:4)  
 plot(res$loadings[, i], pch = 19, cex = .3, ylab = paste0("Loadings PC", i))



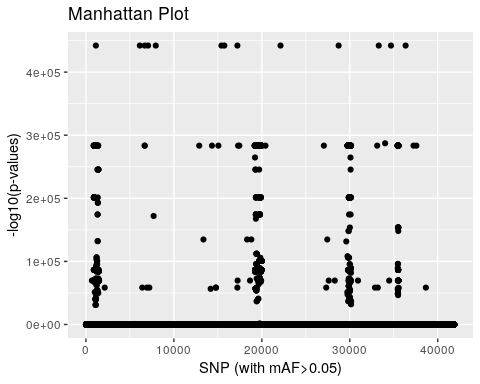
Here, the top-left figure shows that PC1 is determined by a single genomic region, which is likely to be a region of strong LD.

We should therefore thin SNPs in order to compute the PCS. size=windows size

res <- pcadapt(filename, K = 9, LD.clumping = list(size = 1000, thr = 0.1))  
plot(res, option = "screeplot")



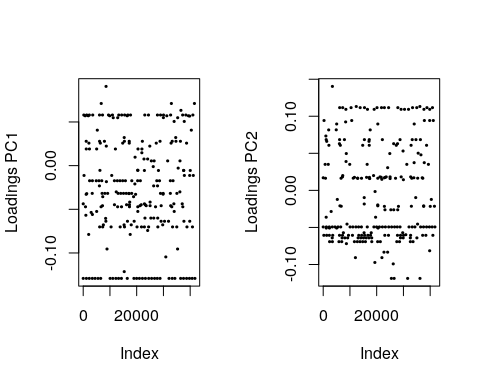
plot(res)



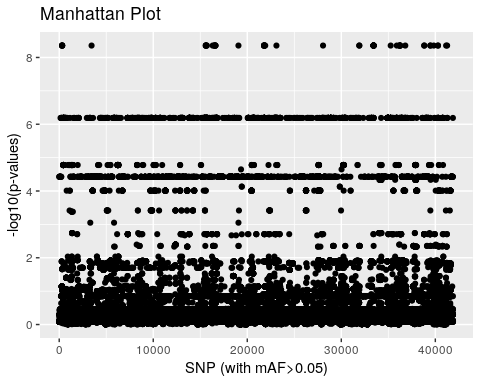
# After SNP thinning, we choose K=2K=2.

The distribution of the loadings is now evenly distributed, so we can have a look at the genome scan, which correctly identifies regions involved in adaptation.

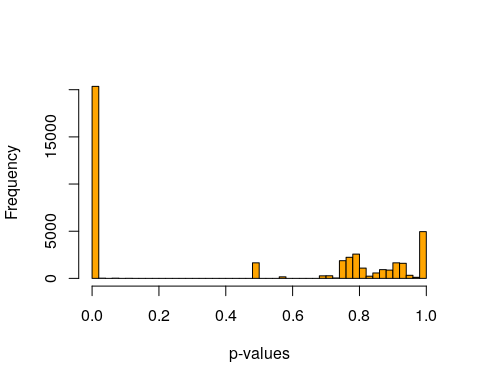
res <- pcadapt(filename, K = 2, LD.clumping = list(size = 1000, thr = 0.1))  
par(mfrow = c(1, 2))  
for (i in 1:2)  
 plot(res$loadings[, i], pch = 19, cex = .3, ylab = paste0("Loadings PC", i))

 #The distribution of the loadings is now evenly distributed, so we can have a look at the genome scan, which correctly identifies regions involved in adaptation.

plot(res)



hist(x$pvalues, xlab = "p-values", main = NULL, breaks = 50, col = "orange")

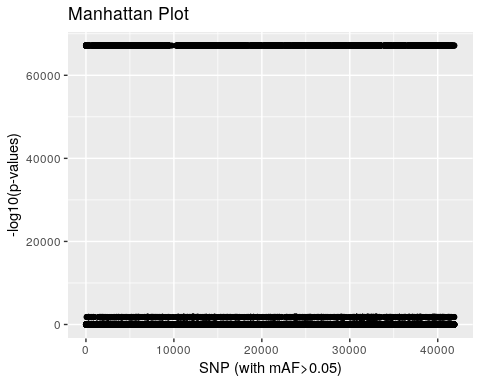
 A list of outliers can be obtained as for individual genotype data

padj <- p.adjust(res$pvalues, method = "BH")  
alpha <- 0.1  
outliers <- which(padj < alpha)  
length(outliers)

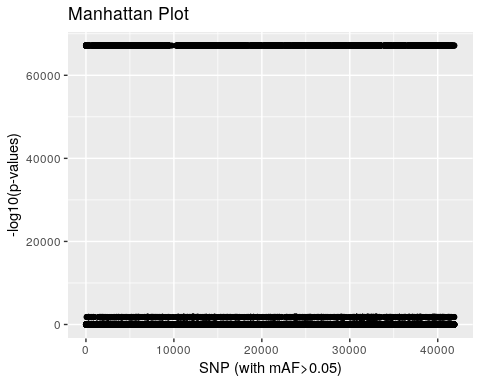
## [1] 3406

A Manhattan plot displays −log10 of the p-values.

K <- 2 # Index of the principal component the user is interested in  
plot(x,option="manhattan",K=1,threshold = 0.20)

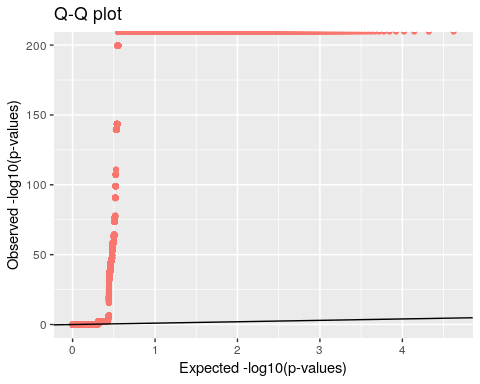


plot(x , option = "manhattan")



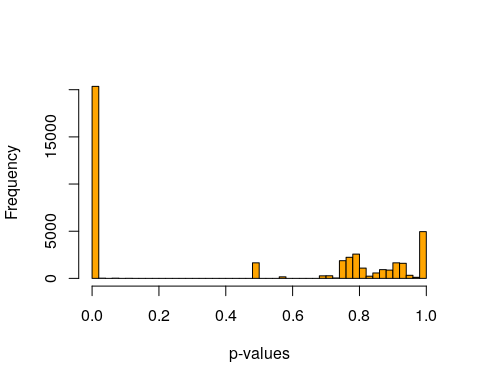
#### The user is also given the possibility to check the distribution of the p-values using a Q-Q plot

plot(x, option = "qqplot", threshold = 0.05)



###### A histogram of p-values confirms that most of the p-values follow the uniform distribution, and that the excess of small p-values indicates the presence of outliers.

hist(x$pvalues, xlab = "p-values", main = NULL, breaks = 50, col = "orange")



###### The presence of outliers is also visible when plotting a histogram of the test statistic Dj.

#plot(x, option = "stat.distribution")

###### 5. Additional features

5.1. Estimation of standard deviation

To compute p-values for each PC, we assume that the loadings of the neutral SNPs follow a normal distribution. The standard deviation of the Gaussian distribution is estimated after removing a proportion of genetic markers with the largest loadings (in absolute values).

To check the Gaussian assumption, we can display the histogram of loadings and superimpose their expected distribution for neutral SNPs. To evaluate if LD might be an issue for your dataset, we recommend to display the loadings (contributions of each SNP to the PC) and to evaluate if the loadings are clustered in a single or several genomic regions.

res <- pcadapt(input = filename, K = 7)  
par(mfrow = c(2, 2))  
for (i in 1:4)  
 plot(res$loadings[, i], pch = 19, cex = .3, ylab = paste0("Loadings PC", i))

