Principal Component Analysis exercices

Albert Xavier Lopez Barrantes

8 de octubre de 2018

PCA excercices

>Find and R package that performs truncated SVD.

With some research I found the package "irlba" which may require "Matrix". This library is speciallized on fast truncated singular value decomposition and principal component analysis which is what we were looking for.

```
library(irlba)
library(Matrix)
```

>Create a function (or write an R script) that performs PCA based on truncated SVD.

Since PCA based on SVD it's time-consuming to perform PCA in large datasets, the truncated SVD it's one of the solutions to do it faster in such situations. With the help of the library above to compute the PCA by truncated SVD, we created a function where "x" is the dataset we want to work on and "y" is the number of principal components.

```
f1<-function(x,y){

  library(irlba)

  pc<- prcomp_irlba(x, n=y, center=TRUE)
  summary(pc)

}
```

We are going to implement this function in the next part.

>Perform PCA analysis using this new R code and compare the results obtained using prcomp function on the data set musk.txt.

First we need to read the data from the file musk.txt and delete the last column so:

```
musk1<-read.table("C:/Users/Albert/Desktop/Data Science/Data visualisation and modelling/Visualization
                  header = TRUE, sep = "" )
musk<-musk1[1:166]
```

With our "musk" data stored, we are going to perform the PCA with normal SVD and compare it to the truncated SVD. So first we run "prcomp"" to get principal components:

```
pc<-prcomp(musk,center=TRUE)
summary(pc)
```

```
## Importance of components:
##                             PC1      PC2       PC3       PC4       PC5
## Standard deviation     673.2307 380.4653 278.94556 249.67115 221.53567
## Proportion of Variance   0.4065   0.1298   0.06979   0.05591   0.04402
## Cumulative Proportion    0.4065   0.5363   0.60611   0.66201   0.70603
##                             PC6       PC7       PC8       PC9      PC10
## Standard deviation     216.1523 175.15185 153.65192 134.8082 131.31809
```

```
## Proportion of Variance   0.0419   0.02751   0.02117   0.0163   0.01547
## Cumulative Proportion    0.7479   0.77545   0.79662   0.8129   0.82839
##                             PC11      PC12      PC13     PC14    PC15
## Standard deviation     119.07346 113.50449 110.87235 99.36725 95.5925
## Proportion of Variance   0.01272   0.01155   0.01102  0.00886  0.0082
## Cumulative Proportion    0.84110   0.85266   0.86368  0.87254  0.8807
##                             PC16     PC17     PC18    PC19     PC20
## Standard deviation     92.52041 87.16590 84.32162 82.4879 79.43184
## Proportion of Variance  0.00768  0.00681  0.00638  0.0061  0.00566
## Cumulative Proportion   0.88841  0.89522  0.90160  0.9077  0.91336
##                             PC21     PC22     PC23     PC24     PC25
## Standard deviation     76.80216 73.81026 72.29108 69.33194 65.36666
## Proportion of Variance  0.00529  0.00489  0.00469  0.00431  0.00383
## Cumulative Proportion   0.91865  0.92354  0.92823  0.93254  0.93637
##                             PC26    PC27     PC28     PC29     PC30
## Standard deviation     64.03125 61.6150 60.44529 58.53056 55.96865
## Proportion of Variance  0.00368  0.0034  0.00328  0.00307  0.00281
## Cumulative Proportion   0.94005  0.9435  0.94673  0.94980  0.95261
##                             PC31     PC32     PC33     PC34    PC35
## Standard deviation     54.31483 51.88019 49.80327 49.26944 48.3597
## Proportion of Variance  0.00265  0.00241  0.00222  0.00218  0.0021
## Cumulative Proportion   0.95526  0.95767  0.95989  0.96207  0.9642
##                             PC36    PC37     PC38     PC39     PC40    PC41
## Standard deviation     45.95658 44.7563 42.88435 41.79055 40.22553 39.4985
## Proportion of Variance  0.00189  0.0018  0.00165  0.00157  0.00145  0.0014
## Cumulative Proportion   0.96606  0.9679  0.96951  0.97107  0.97253  0.9739
##                             PC42     PC43     PC44     PC45     PC46
## Standard deviation     37.83523 37.09406 36.08196 34.37303 33.76904
## Proportion of Variance  0.00128  0.00123  0.00117  0.00106  0.00102
## Cumulative Proportion   0.97521  0.97644  0.97761  0.97867  0.97969
##                             PC47     PC48     PC49     PC50     PC51
## Standard deviation     32.67552 31.52300 31.30740 30.29428 29.47740
## Proportion of Variance  0.00096  0.00089  0.00088  0.00082  0.00078
## Cumulative Proportion   0.98065  0.98154  0.98242  0.98324  0.98402
##                             PC52     PC53     PC54     PC55     PC56
## Standard deviation     28.53048 28.14450 27.68169 27.33083 26.94086
## Proportion of Variance  0.00073  0.00071  0.00069  0.00067  0.00065
## Cumulative Proportion   0.98475  0.98546  0.98615  0.98682  0.98747
##                             PC57     PC58     PC59     PC60     PC61
## Standard deviation     25.74005 25.21935 23.82684 23.29859 23.15820
## Proportion of Variance  0.00059  0.00057  0.00051  0.00049  0.00048
## Cumulative Proportion   0.98807  0.98864  0.98915  0.98963  0.99011
##                             PC62     PC63     PC64     PC65     PC66
## Standard deviation     22.55993 22.20886 21.85598 21.31762 20.55301
## Proportion of Variance  0.00046  0.00044  0.00043  0.00041  0.00038
## Cumulative Proportion   0.99057  0.99101  0.99144  0.99185  0.99223
##                             PC67     PC68     PC69     PC70    PC71
## Standard deviation     20.29024 19.64887 19.24901 18.85577 18.4205
## Proportion of Variance  0.00037  0.00035  0.00033  0.00032  0.0003
## Cumulative Proportion   0.99260  0.99294  0.99328  0.99359  0.9939
##                             PC72     PC73     PC74     PC75     PC76
## Standard deviation     17.63659 17.54612 16.95395 16.75000 16.49247
## Proportion of Variance  0.00028  0.00028  0.00026  0.00025  0.00024
## Cumulative Proportion   0.99418  0.99445  0.99471  0.99496  0.99521
```

```
##                            PC77     PC78     PC79    PC80     PC81
## Standard deviation     15.85282 15.54531 15.27025 14.7901 14.65919
## Proportion of Variance  0.00023  0.00022  0.00021  0.0002  0.00019
## Cumulative Proportion   0.99543  0.99565  0.99586  0.9960  0.99625
##                            PC82     PC83     PC84     PC85     PC86
## Standard deviation     14.06402 13.67653 13.41860 13.16473 12.90247
## Proportion of Variance  0.00018  0.00017  0.00016  0.00016  0.00015
## Cumulative Proportion   0.99642  0.99659  0.99675  0.99691  0.99706
##                            PC87     PC88     PC89     PC90     PC91
## Standard deviation     12.34135 11.89893 11.76484 11.52159 11.27099
## Proportion of Variance  0.00014  0.00013  0.00012  0.00012  0.00011
## Cumulative Proportion   0.99719  0.99732  0.99745  0.99757  0.99768
##                            PC92    PC93    PC94    PC95     PC96     PC97
## Standard deviation     10.90049 10.7475 10.6927 10.3375 10.14542 10.12016
## Proportion of Variance  0.00011  0.0001  0.0001  0.0001  0.00009  0.00009
## Cumulative Proportion   0.99779  0.9979  0.9980  0.9981  0.99818  0.99827
##                            PC98    PC99   PC100   PC101   PC102   PC103
## Standard deviation     9.72447 9.45862 9.39353 9.13473 8.98237 8.90538
## Proportion of Variance 0.00008 0.00008 0.00008 0.00007 0.00007 0.00007
## Cumulative Proportion  0.99836 0.99844 0.99852 0.99859 0.99866 0.99873
##                           PC104   PC105   PC106   PC107   PC108   PC109
## Standard deviation     8.64455 8.26822 8.00874 7.91798 7.79931 7.59897
## Proportion of Variance 0.00007 0.00006 0.00006 0.00006 0.00005 0.00005
## Cumulative Proportion  0.99880 0.99886 0.99892 0.99898 0.99903 0.99908
##                           PC110   PC111   PC112   PC113   PC114   PC115
## Standard deviation     7.46120 7.19393 7.05771 6.98377 6.82384 6.58582
## Proportion of Variance 0.00005 0.00005 0.00004 0.00004 0.00004 0.00004
## Cumulative Proportion  0.99913 0.99918 0.99922 0.99927 0.99931 0.99935
##                           PC116   PC117   PC118   PC119   PC120   PC121
## Standard deviation     6.53568 6.28332 6.12328 5.88890 5.85458 5.62307
## Proportion of Variance 0.00004 0.00004 0.00003 0.00003 0.00003 0.00003
## Cumulative Proportion  0.99939 0.99942 0.99946 0.99949 0.99952 0.99955
##                           PC122   PC123   PC124   PC125   PC126   PC127
## Standard deviation     5.54017 5.42273 5.22421 5.11616 4.98581 4.90223
## Proportion of Variance 0.00003 0.00003 0.00002 0.00002 0.00002 0.00002
## Cumulative Proportion  0.99957 0.99960 0.99962 0.99965 0.99967 0.99969
##                           PC128   PC129   PC130   PC131   PC132   PC133
## Standard deviation     4.68316 4.65979 4.57225 4.34966 4.20285 4.12357
## Proportion of Variance 0.00002 0.00002 0.00002 0.00002 0.00002 0.00002
## Cumulative Proportion  0.99971 0.99973 0.99975 0.99977 0.99978 0.99980
##                           PC134   PC135   PC136   PC137   PC138   PC139
## Standard deviation     3.87673 3.81462 3.73832 3.66829 3.53225 3.36524
## Proportion of Variance 0.00001 0.00001 0.00001 0.00001 0.00001 0.00001
## Cumulative Proportion  0.99981 0.99982 0.99984 0.99985 0.99986 0.99987
##                           PC140   PC141   PC142   PC143   PC144   PC145
## Standard deviation     3.33477 3.31383 3.22377 3.18757 2.94172 2.85440
## Proportion of Variance 0.00001 0.00001 0.00001 0.00001 0.00001 0.00001
## Cumulative Proportion  0.99988 0.99989 0.99990 0.99991 0.99992 0.99992
##                           PC146   PC147   PC148   PC149   PC150 PC151 PC152
## Standard deviation     2.77978 2.67264 2.66809 2.52881 2.43990 2.301 2.245
## Proportion of Variance 0.00001 0.00001 0.00001 0.00001 0.00001 0.000 0.000
## Cumulative Proportion  0.99993 0.99994 0.99994 0.99995 0.99995 1.000 1.000
##                          PC153 PC154 PC155 PC156 PC157 PC158 PC159 PC160
## Standard deviation     2.217 2.157  2.13 1.985 1.908 1.774 1.769 1.595
```

```
## Proportion of Variance 0.000 0.000  0.00 0.000 0.000 0.000 0.000 0.000
## Cumulative Proportion  1.000 1.000  1.00 1.000 1.000 1.000 1.000 1.000
##                         PC161 PC162 PC163 PC164 PC165 PC166
## Standard deviation     1.515 1.433 1.389 1.333 1.185 1.047
## Proportion of Variance 0.000 0.000 0.000 0.000 0.000 0.000
## Cumulative Proportion  1.000 1.000 1.000 1.000 1.000 1.000
```

Next we want to compare results, this time using the function we created above based on truncated SVD. Using function "f1" we just have to specify the dataset and the number of principal components. In this case we just want the first two components.

```
f1(musk,2)
```

```
## Warning: package 'irlba' was built under R version 3.5.1
```

```
## Loading required package: Matrix
```

```
## Warning: package 'Matrix' was built under R version 3.5.1
```

```
## Importance of components:
##                              PC1      PC2
## Standard deviation     673.2307 380.4653
## Proportion of Variance   0.4065   0.1298
## Cumulative Proportion    0.4065   0.5363
```

Checking the importance of components we can see the same results as in the standard method. Now we should prove truncated SVD is faster than the standard SVD. To do so, we use the function "system.time()" which gives us "user time" representing the CPU time charged for the execution of user instructions of the calling process.

```
system.time(pc<-prcomp(musk,center=TRUE))    # standard SVD
```

```
##    user  system elapsed
##    0.05    0.00    0.05
```

```
system.time(f1(musk,2))                      # truncated SVD
```

```
##    user  system elapsed
##       0       0       0
```

As expected, the CPU time for the execution is smaller when using the truncated SVD with the function f1() we created using truncated SVD.

>Plot the molecules (observations/rows) in the first two axes and color each dot using the infromation given in the column mask.

To do that, we just get the dataset where we have all the variables including "musk":

```
df2 <- musk1[c(1:167)]
```

Using the full dataset and the PCA made above, we can create a biplot of the two principal components and colour them if they are musk or not. We will create the plot with the help of packages "ggplot2" and "ggfortify":

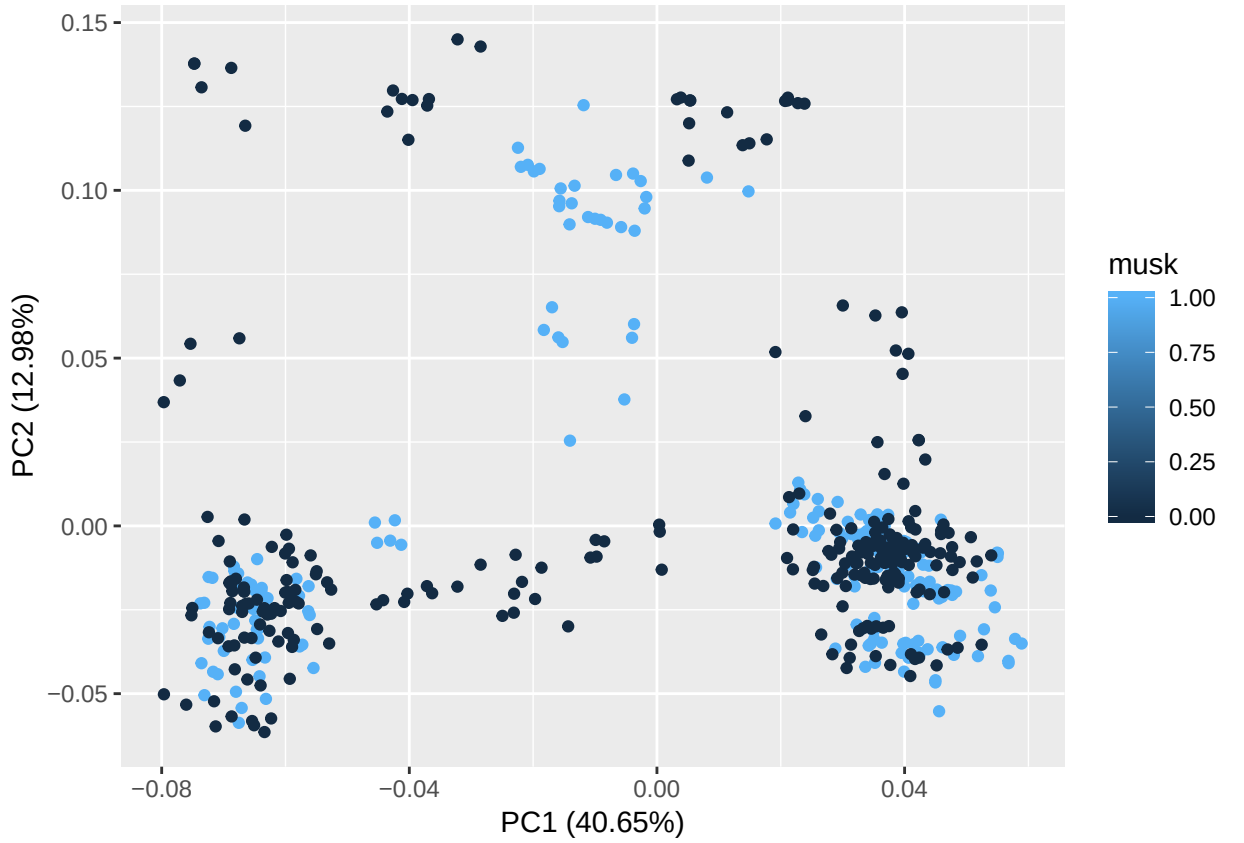
```
#install.packages("ggfortify")
#install.packages("ggplot2")
library(ggfortify)
```

```
## Warning: package 'ggfortify' was built under R version 3.5.1
```

```
## Loading required package: ggplot2
```

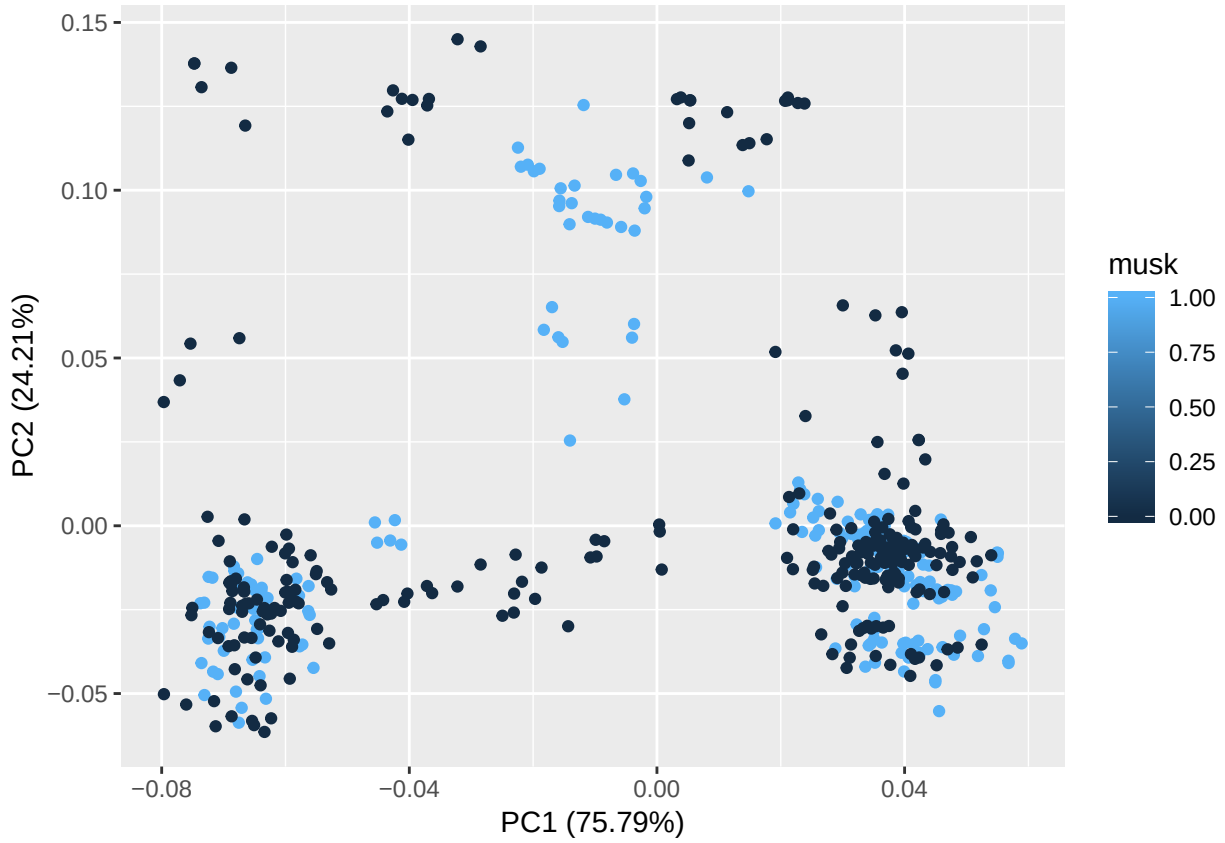
```
## Warning: package 'ggplot2' was built under R version 3.5.1
```

```
library(ggplot2)
autoplot(prcomp(df2), data = musk1, colour = 'musk')
```



We should get the same plot when using the truncated SVD:

```
autoplot(prcomp_irlba(df2, n=2, center=TRUE), data = musk1, colour = 'musk')
```



And as expected, we can appreciate a similar visualization on the first two principal components using two different methods of Single Value Decompostion. Notice the plot on the truncated SVD can has small variations between each execution due to the iterative method it's based on. When using large datasets, whe should use the second option to be more time efficient.

```
sessionInfo()
```

```
## R version 3.5.0 (2018-04-23)
## Platform: x86_64-w64-mingw32/x64 (64-bit)
## Running under: Windows 10 x64 (build 17134)
##
## Matrix products: default
##
## locale:
## [1] LC_COLLATE=Spanish_Spain.1252  LC_CTYPE=Spanish_Spain.1252
## [3] LC_MONETARY=Spanish_Spain.1252 LC_NUMERIC=C
## [5] LC_TIME=Spanish_Spain.1252
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base
##
## other attached packages:
## [1] ggfortify_0.4.5 ggplot2_3.0.0   irlba_2.3.2     Matrix_1.2-14
##
## loaded via a namespace (and not attached):
##  [1] Rcpp_0.12.16     compiler_3.5.0   pillar_1.2.2     plyr_1.8.4
##  [5] bindr_0.1.1      tools_3.5.0      digest_0.6.15    evaluate_0.10.1
```

```
##  [9] tibble_1.4.2     gtable_0.2.0     lattice_0.20-35  pkgconfig_2.0.1
## [13] rlang_0.2.0      yaml_2.1.19      bindrcpp_0.2.2   gridExtra_2.3
## [17] withr_2.1.2      stringr_1.3.1    dplyr_0.7.6      knitr_1.20
## [21] rprojroot_1.3-2  grid_3.5.0       tidyselect_0.2.4 glue_1.2.0
## [25] R6_2.2.2         rmarkdown_1.10   tidyr_0.8.1      purrr_0.2.4
## [29] magrittr_1.5     backports_1.1.2  scales_1.0.0     htmltools_0.3.6
## [33] assertthat_0.2.0 colorspace_1.3-2 labeling_0.3     stringi_1.1.7
## [37] lazyeval_0.2.1   munsell_0.5.0
```