What is Principal-Component Analysis?

Principal-component analysis (PCA) is a multivariate analysis technique. The basic idea behind this technique is to find variables with strong correlations between them and extract a single variable that can then represent them at the same time.

```
data(mtcars)
print(mtcars)
```

```
                     mpg cyl  disp  hp drat    wt  qsec vs am gear carb
Mazda RX4           21.0   6 160.0 110 3.90 2.620 16.46  0  1    4    4
Mazda RX4 Wag       21.0   6 160.0 110 3.90 2.875 17.02  0  1    4    4
Datsun 710          22.8   4 108.0  93 3.85 2.320 18.61  1  1    4    1
Hornet 4 Drive      21.4   6 258.0 110 3.08 3.215 19.44  1  0    3    1
Hornet Sportabout   18.7   8 360.0 175 3.15 3.440 17.02  0  0    3    2
Valiant             18.1   6 225.0 105 2.76 3.460 20.22  1  0    3    1
Duster 360          14.3   8 360.0 245 3.21 3.570 15.84  0  0    3    4
Merc 240D           24.4   4 146.7  62 3.69 3.190 20.00  1  0    4    2
Merc 230            22.8   4 140.8  95 3.92 3.150 22.90  1  0    4    2
Merc 280            19.2   6 167.6 123 3.92 3.440 18.30  1  0    4    4
Merc 280C           17.8   6 167.6 123 3.92 3.440 18.90  1  0    4    4
Merc 450SE          16.4   8 275.8 180 3.07 4.070 17.40  0  0    3    3
Merc 450SL          17.3   8 275.8 180 3.07 3.730 17.60  0  0    3    3
Merc 450SLC         15.2   8 275.8 180 3.07 3.780 18.00  0  0    3    3
Cadillac Fleetwood  10.4   8 472.0 205 2.93 5.250 17.98  0  0    3    4
Lincoln Continental 10.4   8 460.0 215 3.00 5.424 17.82  0  0    3    4
Chrysler Imperial   14.7   8 440.0 230 3.23 5.345 17.42  0  0    3    4
Fiat 128            32.4   4  78.7  66 4.08 2.200 19.47  1  1    4    1
Honda Civic         30.4   4  75.7  52 4.93 1.615 18.52  1  1    4    2
Toyota Corolla      33.9   4  71.1  65 4.22 1.835 19.90  1  1    4    1
Toyota Corona       21.5   4 120.1  97 3.70 2.465 20.01  1  0    3    1
Dodge Challenger    15.5   8 318.0 150 2.76 3.520 16.87  0  0    3    2
AMC Javelin         15.2   8 304.0 150 3.15 3.435 17.30  0  0    3    2
Camaro Z28          13.3   8 350.0 245 3.73 3.840 15.41  0  0    3    4
Pontiac Firebird    19.2   8 400.0 175 3.08 3.845 17.05  0  0    3    2
Fiat X1-9           27.3   4  79.0  66 4.08 1.935 18.90  1  1    4    1
Porsche 914-2       26.0   4 120.3  91 4.43 2.140 16.70  0  1    5    2
Lotus Europa        30.4   4  95.1 113 3.77 1.513 16.90  1  1    5    2
Ford Pantera L      15.8   8 351.0 264 4.22 3.170 14.50  0  1    5    4
Ferrari Dino        19.7   6 145.0 175 3.62 2.770 15.50  0  1    5    6
Maserati Bora       15.0   8 301.0 335 3.54 3.570 14.60  0  1    5    8
Volvo 142E          21.4   4 121.0 109 4.11 2.780 18.60  1  1    4    2
```

	Active individuals
	Active variables
	Supplementary quantitative variables
	Supplementary qualitative variables
	Supplementary individuals

1. Compute the Principal Components

Because PCA works best with numerical data, we exclude the two categorical variables (vs and am) and two Supplementary quantitative variables (gear and carb). We are left with a matrix of 7 columns and 27 rows excluding bottom five Supplementary individuals , which you pass to the prcomp() function, assigning output to mtcars.pca. We also set two arguments, center and scale, to be TRUE. Then we can have a peek at your PCA object with summary().

```
#1. Load factoextra for visualization
library(factoextra)
data(mtcars)
mtcars.active <- mtcars[1:27, 1:7]
head(mtcars.active[, 1:7])
```

```
                   mpg cyl disp  hp drat    wt  qsec
Mazda RX4         21.0   6  160 110 3.90 2.620 16.46
Mazda RX4 Wag     21.0   6  160 110 3.90 2.875 17.02
Datsun 710        22.8   4  108  93 3.85 2.320 18.61
Hornet 4 Drive    21.4   6  258 110 3.08 3.215 19.44
Hornet Sportabout 18.7   8  360 175 3.15 3.440 17.02
Valiant           18.1   6  225 105 2.76 3.460 20.22
```

2. Show summury of components

```
#2. Compute PCA and View summary.
res.pca <- prcomp(mtcars.active[,c(1:7)], center = TRUE,scale. = TRUE)
summary(res.pca)
```

```
Importance of components:
                          PC1    PC2     PC3     PC4     PC5     PC6     PC7
Standard deviation     2.3289 0.9612 0.55448 0.39389 0.30025 0.24909 0.19420
Proportion of Variance 0.7748 0.1320 0.04392 0.02216 0.01288 0.00886 0.00539
Cumulative Proportion  0.7748 0.9068 0.95071 0.97287 0.98575 0.99461 1.00000
```

We obtain 9 principal components, PC1-9. Each of these explains a percentage of the total variation in the dataset. That is to say: PC1 explains 77% of the total variance, which means that nearly two-thirds of the information in the dataset (9 variables) can be encapsulated by just that one Principal Component. PC2 explains 13% of the variance. So, by knowing the position of a sample in relation to just PC1 and PC2, we can get a very accurate view on where it stands in relation to other samples, as just PC1 and PC2 can explain 90% of the variance.

```
#Let's call str() to have a look at PCA object.
str(res.pca)
```

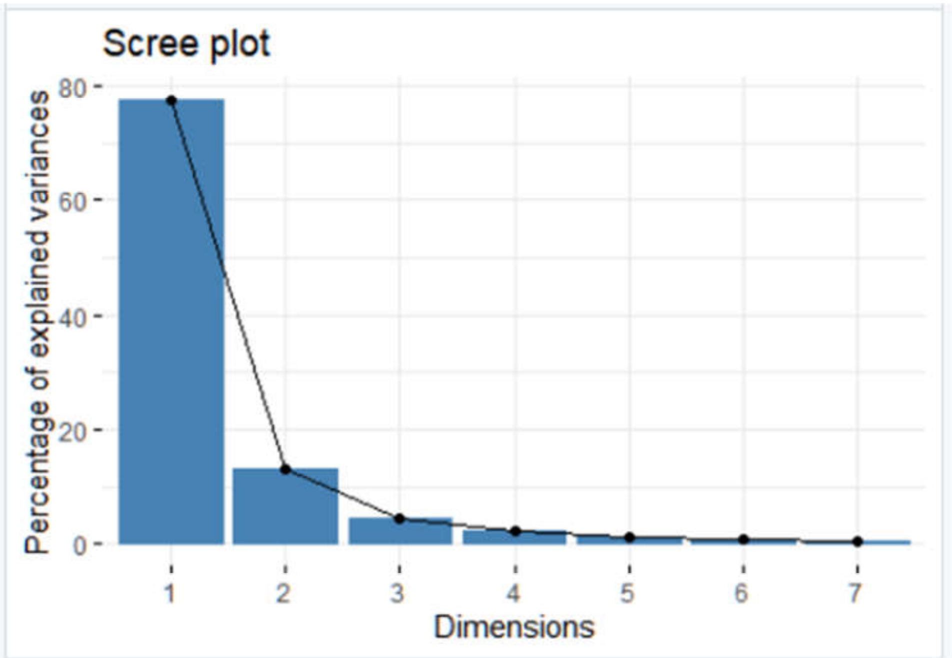
```
List of 5
 $ sdev    : num [1:7] 2.329 0.961 0.554 0.394 0.3 ...
 $ rotation: num [1:7, 1:7] -0.401 0.411 0.413 0.405 -0.352 ...
  ..- attr(*, "dimnames")=List of 2
  .. ..$ : chr [1:7] "mpg" "cyl" "disp" "hp" ...
  .. ..$ : chr [1:7] "PC1" "PC2" "PC3" "PC4" ...
 $ center  : Named num [1:7] 20.02 6.22 235.93 136.96 3.55 ...
  ..- attr(*, "names")= chr [1:7] "mpg" "cyl" "disp" "hp" ...
 $ scale   : Named num [1:7] 6.119 1.783 126.835 59.028 0.559 ...
  ..- attr(*, "names")= chr [1:7] "mpg" "cyl" "disp" "hp" ...
 $ x       : num [1:27, 1:7] -0.769 -0.756 -2.048 -0.184 1.646 ...
  ..- attr(*, "dimnames")=List of 2
  .. ..$ : chr [1:27] "Mazda RX4" "Mazda RX4 Wag" "Datsun 710" "Hornet 4 Drive" .
..
  .. ..$ : chr [1:7] "PC1" "PC2" "PC3" "PC4" ...
 - attr(*, "class")= chr "prcomp"
```

PCA object contains the following information:

- The center point (`$center`), scaling (`$scale`), standard deviation(`sdev`) of each principal component
- The relationship (correlation or anticorrelation, etc) between the initial variables and the principal components (`$rotation`)
- The values of each sample in terms of the principal components (`$x`)

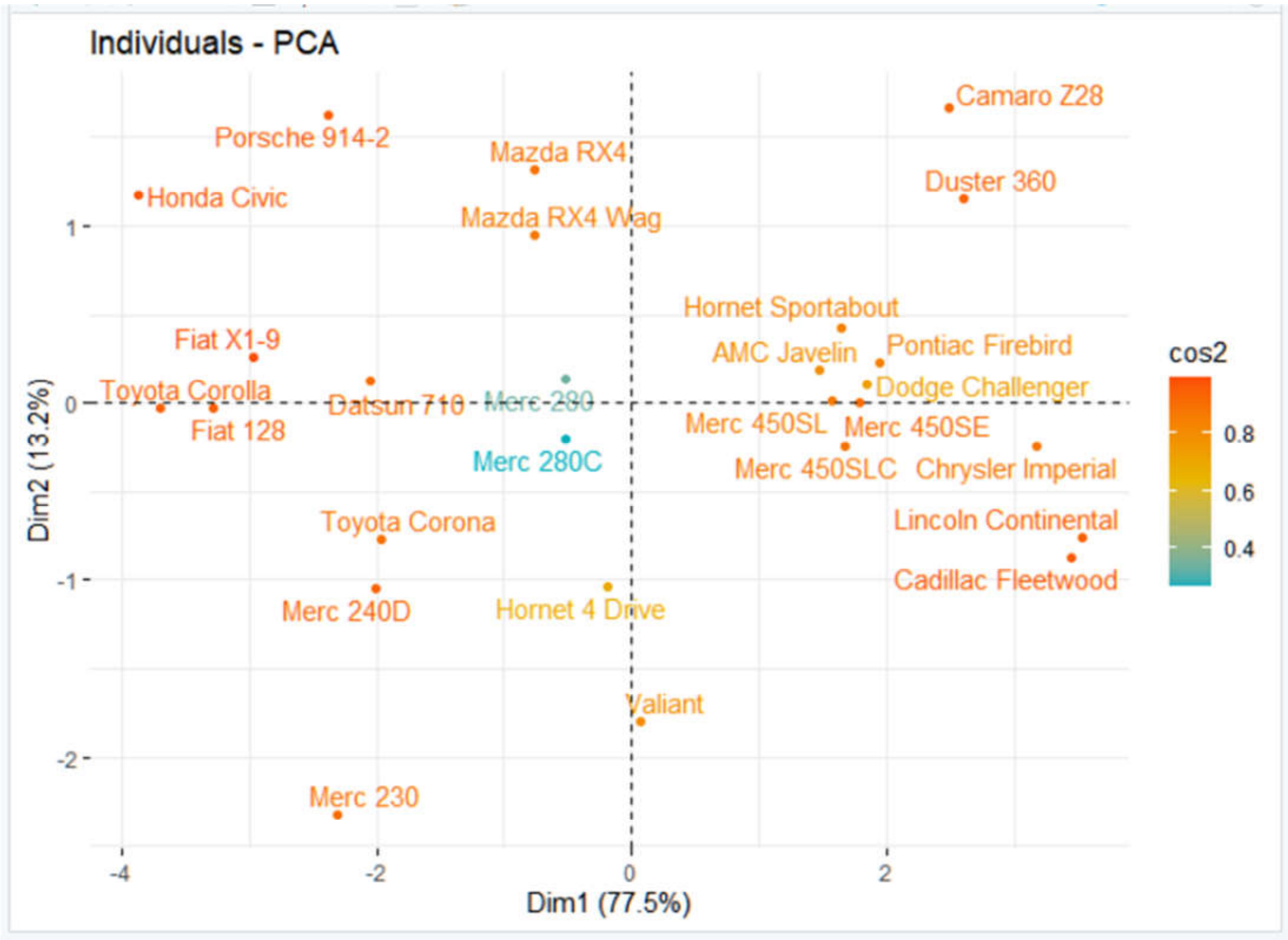
3. Visualize *eigenvalues* (*scree plot*). Show the percentage of variances explained by each principal component.

```
#3. Plotting PCA
fviz_eig(res.pca)
```



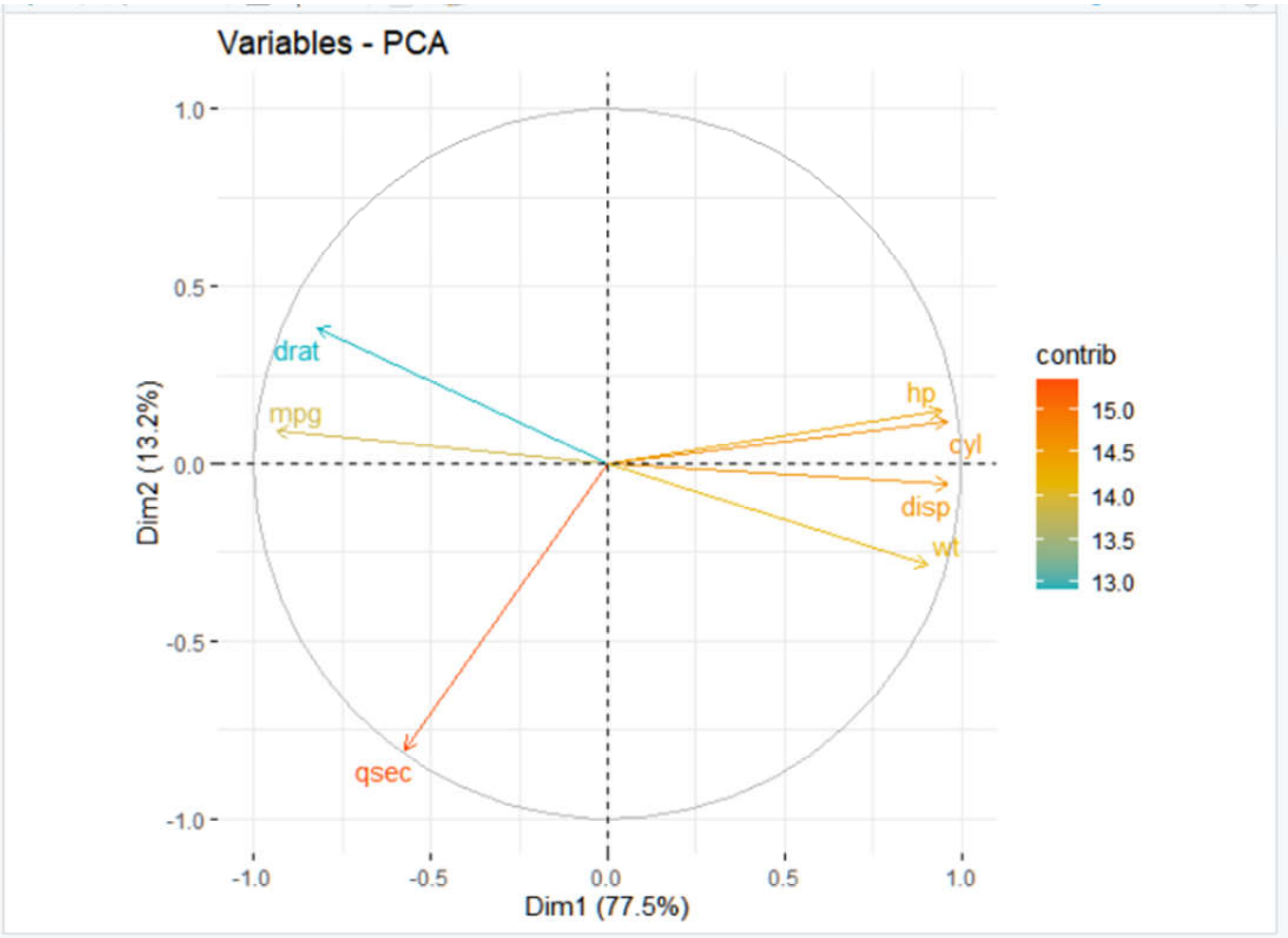
4. Graph of individuals. Individuals with a similar profile are grouped together.

```
#4.  Graph of individuals.
fviz_pca_ind(res.pca,
             col.ind = "cos2",
              gradient.cols = c("#00AFBB", "#E7B800", "#FC4E07"), repel = TRUE)
```



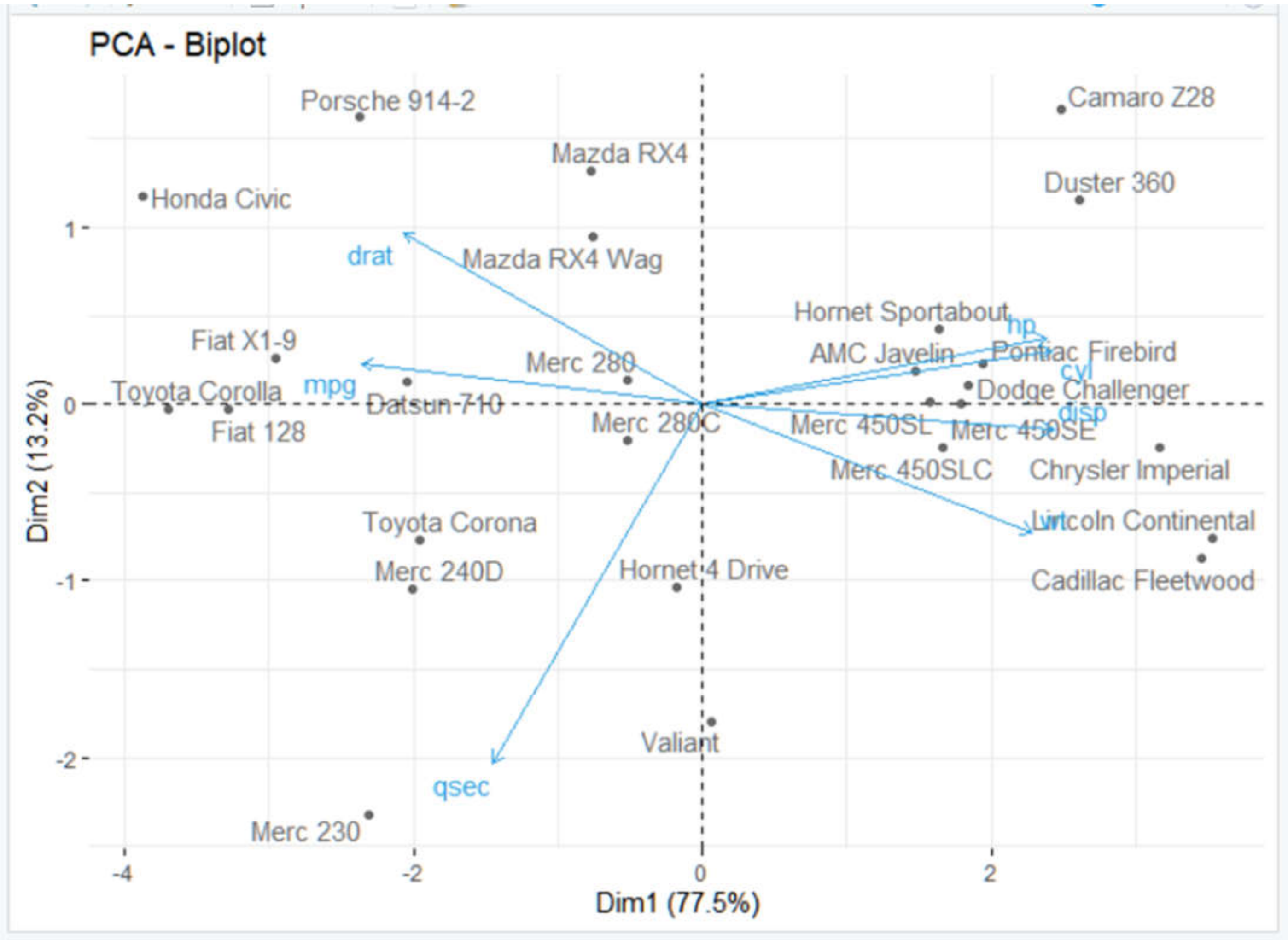
5. Graph of variables. Positive correlated variables point to the same side of the plot. Negative correlated variables point to opposite sides of the graph.

```
#5.  Graph of variables.
fviz_pca_var(res.pca,
             col.var = "contrib",
              gradient.cols = c("#00AFBB", "#E7B800", " #FC4E07"),repel = TRUE)
```



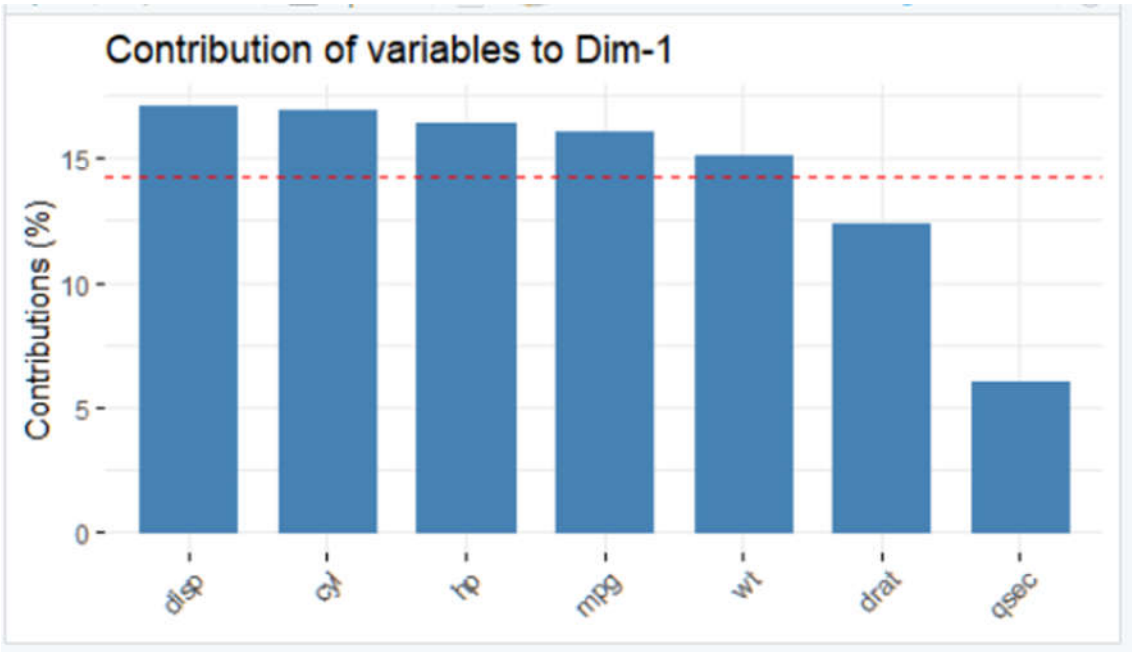
6. Biplot of individuals and variables

```
#6.  Biplot of individuals and variables
fviz_pca_biplot(res.pca, repel = TRUE, col.var = "#2E9FDF",
                                col.ind = "#696969" )
```



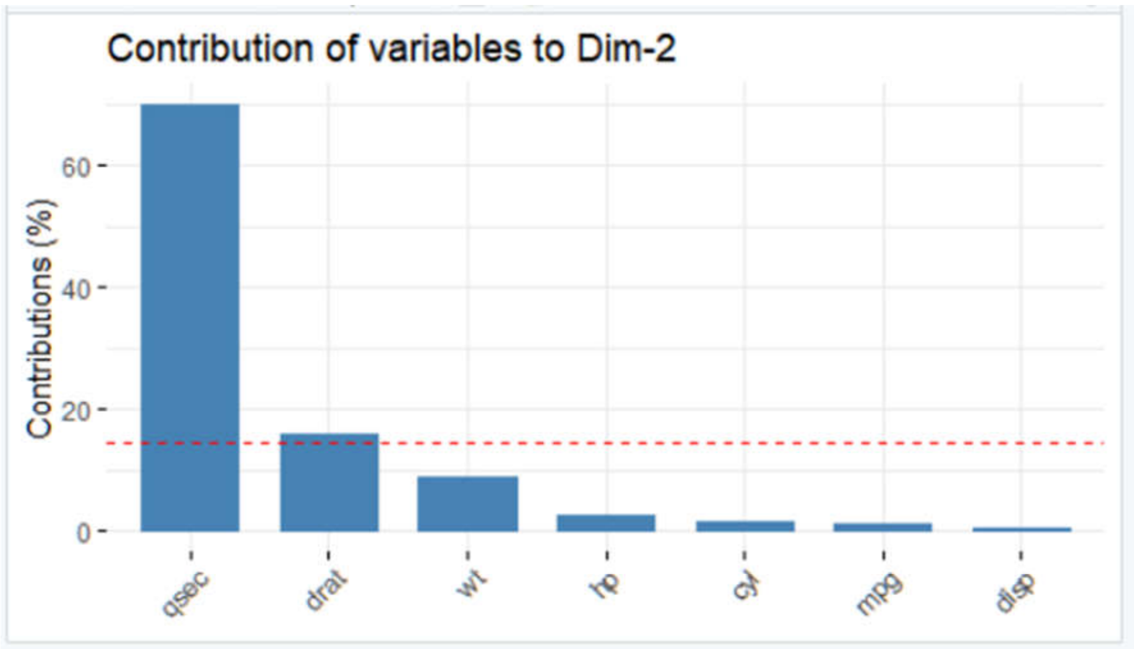
7. Contribution of variables to dimension one(Dim1)

```
#Contribution of variables to Dim-1
fviz_contrib(res.pca, choice = "var", axes = 1)
```



8. Contribution of variables to dimension two(Dim2)

```
#Contribution of variables to Dim-2
fviz_contrib(res.pca, choice = "var", axes = 2)
```



9. Access to PCA results

```
# Eigenvalues
eig.val <- get_eigenvalue(res.pca)
eig.val
```

```
      eigenvalue variance.percent cumulative.variance.percent
Dim.1 5.42363808       77.4805440                    77.48054
Dim.2 0.92385110       13.1978729                    90.67842
Dim.3 0.30745223        4.3921747                    95.07059
Dim.4 0.15515277        2.2164681                    97.28706
Dim.5 0.09014743        1.2878204                    98.57488
Dim.6 0.06204660        0.8863800                    99.46126
Dim.7 0.03771179        0.5387399                   100.00000
```

Results for Variables

```
# Results for Variables
res.var <- get_pca_var(res.pca)
res.var$coord          # Coordinates
res.var$contrib        # Contributions to the PCs
res.var$cos2           # Quality of representation
```

```
> res.var$coord          # Coordinates
          Dim.1       Dim.2       Dim.3         Dim.4        Dim.5       Dim.6       Dim.7
mpg  -0.9335653  0.09192380 -0.08241515  0.3302364350 -0.009108812 -0.01909104 -0.06090996
cyl   0.9579618  0.11983499 -0.13686937  0.0516067715 -0.075406903 -0.19844675  0.03853641
disp  0.9625280 -0.05740243  0.06792022  0.1968515031  0.073187667  0.08164891  0.12189368
hp    0.9441132  0.14864209  0.16689472  0.0483241098 -0.216735245  0.07847758 -0.05686668
drat -0.8194887  0.38434965  0.41602031 -0.0008928492 -0.009538592 -0.06782036  0.05430729
wt    0.9042815 -0.28630834  0.25611063  0.0413921167  0.139459612 -0.06113466 -0.09904576
qsec -0.5721069 -0.80371267  0.10389508  0.0251707535 -0.111837774 -0.03374456  0.04081032
> 
```

Contributions of variables to each dimension

```
> res.var$contrib          # Contributions to the PCs
         Dim.1      Dim.2     Dim.3        Dim.4       Dim.5      Dim.6     Dim.7
mpg  16.069365  0.9146480  2.209207 7.028950e+01  0.09203864  0.5874101  9.837832
cyl  16.920206  1.5544091  6.093052 1.716540e+00  6.30766860 63.4702198  3.937905
disp 17.081896  0.3566634  1.500447 2.497572e+01  5.94186075 10.7444165 39.399000
hp   16.434536  2.3915618  9.059570 1.505110e+00 52.10816038  9.9259749  8.575088
drat 12.382127 15.9900935 56.292615 5.138031e-04  0.10092882  7.4131403  7.820581
wt   15.077058  8.8729090 21.334259 1.104271e+00 21.57464019  6.0236122 26.013250
qsec  6.034811 69.9197153  3.510850 4.083503e-01 13.87470263  1.8352261  4.416344
> 
```

Quality of representation

```
> res.var$cos2          # Quality of representation
         Dim.1       Dim.2       Dim.3        Dim.4        Dim.5       Dim.6       Dim.7
mpg  0.8715442 0.008449985 0.006792257 1.090561e-01 8.297046e-05 0.000364468 0.003710023
cyl  0.9176907 0.014360425 0.018733224 2.663259e-03 5.686201e-03 0.039381114 0.001485055
disp 0.9264602 0.003295039 0.004613156 3.875051e-02 5.356435e-03 0.006666545 0.014858069
hp   0.8913497 0.022094470 0.027853849 2.335220e-03 4.697417e-02 0.006158730 0.003233819
drat 0.6715618 0.147724655 0.173072901 7.971797e-07 9.098474e-05 0.004599602 0.002949281
wt   0.8177251 0.081972468 0.065592656 1.713307e-03 1.944898e-02 0.003737447 0.009810062
qsec 0.3273063 0.645954060 0.010794188 6.335668e-04 1.250769e-02 0.001138695 0.001665482
> 
```

Results for Individuals

```
# Results for individuals
res.ind <- get_pca_ind(res.pca)
res.ind$coord          # Coordinates
res.ind$contrib        # Contributions to the PCs
res.ind$cos2           # Quality of representation

res.ind$coord          # Coordinates
                          Dim.1         Dim.2       Dim.3        Dim.4
Mazda RX4           -0.76916610  1.3175822244 -0.24874355 -0.379233387
Mazda RX4 Wag       -0.75587908  0.9503245754 -0.06580908 -0.330200785
Datsun 710          -2.04818889  0.1205521378 -0.10736211 -0.465566266
Hornet 4 Drive      -0.18437466 -1.0368468642 -0.64304549  0.245990722
Hornet Sportabout    1.64578560  0.4254478510 -0.50874341  0.487090451
Valiant              0.06726851 -1.7937771248 -0.84600854 -0.288555662
Duster 360           2.60780452  1.1583387722 -0.04196520 -0.003822454
Merc 240D           -2.01411930 -1.0505066560  0.08163844 -0.011046660
Merc 230            -2.30558045 -2.3187575837  0.91163080 -0.074746940
Merc 280            -0.51305984  0.1328516905  0.48829926 -0.409748605
Merc 280C           -0.51306299 -0.2012359742  0.59226651 -0.577713718
Merc 450SE           1.79305533 -0.0001766392 -0.28118323 -0.067864820
Merc 450SL           1.57166054  0.0109759098 -0.43657727  0.027718720
Merc 450SLC          1.66748171 -0.2448597549 -0.31585960 -0.238863860
Cadillac Fleetwood   3.45430165 -0.8739066670  0.60551198  0.082538679
Lincoln Continental  3.53172901 -0.7604690935  0.80053288  0.067655937
Chrysler Imperial    3.17351780 -0.2482874278  0.97903671  0.584071503
Fiat 128            -3.28089021 -0.0335172015 -0.15274460  0.698907395
Honda Civic         -3.87278710  1.1766949284  0.58238045  0.281329313
Toyota Corolla      -3.70625687 -0.0240674359 -0.13186013  0.850631618
Toyota Corona       -1.95943175 -0.7739629290 -0.01500634 -0.516200014
Dodge Challenger     1.84646035  0.1048284191 -1.10336973 -0.164799226
AMC Javelin          1.47616434  0.1874057416 -0.57494315 -0.254471745
Camaro Z28           2.48366919  1.6630141360  0.74540301 -0.171112509
Pontiac Firebird     1.93994058  0.2282249585 -0.38592610  0.757191704
Fiat X1-9           -2.96143096  0.2620649988 -0.21728355 -0.049179261
Porsche 914-2       -2.37461091  1.6220650081  0.28973105 -0.080000129
                          Dim.5        Dim.6        Dim.7
Mazda RX4            0.27441449 -0.111258605 -0.019850976
Mazda RX4 Wag        0.26290976 -0.220976080 -0.076517624
Datsun 710           0.02045689  0.451318422 -0.098741871
Hornet 4 Drive       0.09268826  0.139535603  0.120791467
Hornet Sportabout   -0.11313988  0.005041829  0.268043455
Valiant              0.05783152  0.099032984 -0.031532622
Duster 360          -0.61694244  0.472138279 -0.044177549
Merc 240D            0.55651569  0.113362426 -0.176950727
Merc 230            -0.55428561 -0.052465849  0.226904179
Merc 280             0.09174489 -0.365555225 -0.121545890
Merc 280C           -0.04038999 -0.398600091  0.028682761
Merc 450SE          -0.11596435 -0.304545970 -0.366832617
Merc 450SL          -0.32451838 -0.249334403 -0.213611924
Merc 450SLC         -0.38362782 -0.269007631 -0.079126524
Cadillac Fleetwood   0.40607331  0.141166505  0.192333312
Lincoln Continental  0.37454942  0.100250026  0.008815032
Chrysler Imperial    0.17434616  0.015763194 -0.281934845
Fiat 128            -0.02135879 -0.043927128 -0.313176865
Honda Civic          0.05447539 -0.292093036  0.443181220
Toyota Corolla      -0.30813614 -0.102696483 -0.110473248
Toyota Corona       -0.24747253  0.439755161  0.042071108
Dodge Challenger     0.22179328 -0.013759452  0.092666732
AMC Javelin          0.03509365 -0.251654401  0.333505582
Camaro Z28          -0.43584175  0.175473130  0.024102407
Pontiac Firebird     0.14616976  0.034458407  0.202939839
Fiat X1-9            0.01368755  0.133740953  0.010187828
Porsche 914-2        0.37892766  0.354837438 -0.059751641
```

```
  res.ind$contrib        # Contributions to the PCs
                          Dim.1        Dim.2        Dim.3        Dim.4
Mazda RX4             0.404004132 6.959687e+00  0.745353437 3.433127e+00
Mazda RX4 Wag         0.390166692 3.620580e+00  0.052171121 2.602753e+00
Datsun 710            2.864742215 5.826183e-02  0.138854733 5.174158e+00
Hornet 4 Drive        0.023213894 4.309863e+00  4.981291779 1.444489e+00
Hornet Sportabout     1.849660611 7.256495e-01  3.117856937 5.663644e+00
Valiant               0.003090076 1.289945e+01  8.622001313 1.987632e+00
Duster 360            4.644036275 5.379048e+00  0.021214715 3.487878e-04
Merc 240D             2.770230552 4.424171e+00  0.080287503 2.912988e-03
Merc 230              3.629995586 2.155485e+01 10.011440492 1.333717e-01
Merc 280              0.179755250 7.075684e-02  2.872306087 4.007852e+00
Merc 280C             0.179757457 1.623475e-01  4.225644329 7.967133e+00
Merc 450SE            2.195497338 1.250859e-07  0.952439576 1.099426e-01
Merc 450SL            1.686797833 4.829646e-04  2.296047306 1.834100e-02
Merc 450SLC           1.898749583 2.403638e-01  1.201840278 1.362002e+00
Cadillac Fleetwood    8.148282084 3.061712e+00  4.416762665 1.626267e-01
Lincoln Continental   8.517659608 2.318448e+00  7.719993611 1.092669e-01
Chrysler Imperial     6.877449545 2.471404e-01 11.546664410 8.143456e+00
Fiat 128              7.350703915 4.503703e-03  0.281054345 1.166047e+01
Honda Civic          10.242188866 5.550882e+00  4.085753541 1.889325e+00
Toyota Corolla        9.380296507 2.322169e-03  0.209452511 1.727268e+01
Toyota Corona         2.621837776 2.401456e+00  0.002712741 6.360815e+00
Dodge Challenger      2.328227986 4.405471e-02 14.665629855 6.483165e-01
AMC Javelin           1.488041184 1.407992e-01  3.982065396 1.545809e+00
Camaro Z28            4.212433274 1.108731e+01  6.693308906 6.989405e-01
Pontiac Firebird      2.569936487 2.088144e-01  1.794183651 1.368637e+01
Fiat X1-9             5.988923423 2.753292e-01  0.568738048 5.773520e-02
Porsche 914-2         3.850618150 1.054802e+01  1.011227011 1.527770e-01
                         Dim.5        Dim.6        Dim.7
Mazda RX4             3.09383366  0.738899657  0.038701056
Mazda RX4 Wag         2.83985598  2.914800088  0.575018773
Datsun 710            0.01719342 12.158622237  0.957550693
Hornet 4 Drive        0.35296557  1.162219913  1.432951812
Hornet Sportabout     0.52591349  0.001517381  7.056177353
Valiant               0.13740801  0.585434046  0.097651574
Duster 360           15.63767493 13.306280327  0.191673604
Merc 240D            12.72441190  0.767107985  3.075132094
Merc 230             12.62263725  0.164312899  5.056430725
Merc 280              0.34581726  7.976717950  1.450907180
Merc 280C             0.06702400  9.484032413  0.080798072
Merc 450SE            0.55249947  5.536355349 13.215845767
Merc 450SL            4.32674703  3.710929104  4.481362289
Merc 450SLC           6.04648635  4.319638899  0.614898231
Cadillac Fleetwood    6.77472682  1.189546900  3.633022559
Lincoln Continental   5.76369721  0.599911556  0.007631446
Chrysler Imperial     1.24884243  0.014832237  7.806503959
Fiat 128              0.01874288  0.115181803  9.632486712
Honda Civic           0.12192244  5.092847248 19.289535591
Toyota Corolla        3.90093009  0.629548777  1.198597362
Toyota Corona         2.51615217 11.543569798  0.173830899
Dodge Challenger      2.02106215  0.011301095  0.843347893
AMC Javelin           0.05059876  3.780309040 10.923587430
Camaro Z28            7.80441851  1.837974527  0.057053187
Pontiac Firebird      0.87780482  0.070877540  4.044768742
Fiat X1-9             0.00769723  1.067694652  0.010193476
Porsche 914-2         5.89923247  7.515832875  0.350637818
```

```
res.ind$cos2          # Quality of representation
                          Dim.1        Dim.2        Dim.3        Dim.4
Mazda RX4           0.225686659 6.622486e-01 2.360311e-02 5.486290e-02
Mazda RX4 Wag       0.333804474 5.276324e-01 2.530225e-03 6.370063e-02
Datsun 710          0.901828655 3.124164e-03 2.477913e-03 4.659583e-02
Hornet 4 Drive      0.020910177 6.612786e-01 2.543540e-01 3.722140e-02
Hornet Sportabout   0.780496900 5.215757e-02 7.457998e-02 6.836659e-02
Valiant             0.001121366 7.973718e-01 1.773672e-01 2.063398e-02
Duster 360          0.777247329 1.533488e-01 2.012740e-04 1.669912e-06
Merc 240D           0.734785375 1.998885e-01 1.207201e-03 2.210304e-05
Merc 230            0.447055734 4.521805e-01 6.989387e-02 4.698811e-04
Merc 280            0.311872831 2.091104e-02 2.824969e-01 1.989191e-01
Merc 280C           0.228978777 3.522614e-02 3.051323e-01 2.903215e-01
Merc 450SE          0.908339108 8.815254e-09 2.233775e-02 1.301218e-03
Merc 450SL          0.859256185 4.190697e-05 6.630212e-02 2.672708e-04
Merc 450SLC         0.862685023 1.860223e-02 3.095411e-02 1.770235e-02
Cadillac Fleetwood  0.897752850 5.746010e-02 2.758556e-02 5.125678e-04
Lincoln Continental 0.900763206 4.176371e-02 4.628010e-02 3.305580e-04
Chrysler Imperial   0.872522185 5.340772e-03 8.304100e-02 2.955471e-02
Fiat 128            0.946087956 9.873784e-05 2.050595e-03 4.293262e-02
Honda Civic         0.877817659 8.103728e-02 1.985046e-02 4.632197e-03
Toyota Corolla      0.941130170 3.968609e-05 1.191257e-03 4.957488e-02
Toyota Corona       0.773836771 1.207339e-01 4.538776e-05 5.370628e-02
Dodge Challenger    0.721881602 2.326720e-03 2.577675e-01 5.750377e-03
AMC Javelin         0.782347633 1.260945e-02 1.186807e-01 2.324928e-02
Camaro Z28          0.633297646 2.839306e-01 5.704304e-02 3.005965e-03
Pontiac Firebird    0.817862100 1.131956e-02 3.236768e-02 1.245991e-01
Fiat X1-9           0.984675723 7.710953e-03 5.300828e-03 2.715526e-04
Porsche 914-2       0.653144074 3.047618e-01 9.723310e-03 7.413195e-04
                           Dim.5        Dim.6        Dim.7
Mazda RX4           2.872630e-02 4.722075e-03 1.503244e-04
Mazda RX4 Wag       4.038322e-02 2.852844e-02 3.420664e-03
Datsun 710          8.996287e-05 4.378750e-02 2.095978e-03
Hornet 4 Drive      5.284510e-03 1.197637e-02 8.974862e-03
Hornet Sportabout   3.688554e-03 7.324887e-06 2.070309e-02
Valiant             8.288063e-04 2.430433e-03 2.464019e-04
Duster 360          4.350092e-02 2.547696e-02 2.230552e-04
Merc 240D           5.609769e-02 2.327707e-03 5.671459e-03
Merc 230            2.583856e-02 2.315019e-04 4.329982e-03
Merc 280            9.972528e-03 1.583243e-01 1.750339e-02
Merc 280C           1.419062e-03 1.382066e-01 7.156419e-04
Merc 450SE          3.799353e-03 2.620392e-02 3.801864e-02
Merc 450SL          3.663395e-02 2.162567e-02 1.587290e-02
Merc 450SLC         4.566151e-02 2.245222e-02 1.942559e-03
Cadillac Fleetwood  1.240638e-02 1.499338e-03 2.783206e-03
Lincoln Continental 1.013104e-02 7.257798e-04 5.611561e-06
Chrysler Imperial   2.633415e-03 2.152698e-05 6.886398e-03
Fiat 128            4.009605e-05 1.695953e-04 8.620401e-03
Honda Civic         1.736832e-04 4.993436e-03 1.149529e-02
Toyota Corolla      6.505249e-03 7.225864e-04 8.361668e-04
Toyota Corona       1.234364e-02 3.897723e-02 3.567443e-04
Dodge Challenger    1.041556e-02 4.008559e-05 1.818167e-03
AMC Javelin         4.421681e-04 2.273733e-02 3.993343e-02
Camaro Z28          1.950195e-02 3.161125e-03 5.964049e-05
Pontiac Firebird    4.643210e-03 2.580439e-04 8.950305e-03
Fiat X1-9           2.103495e-05 2.008255e-03 1.165342e-05
Porsche 914-2       1.663170e-02 1.458421e-02 4.135459e-04
```

Predict using PCA

In this section, we'll show how to predict the coordinates of supplementary individuals and variables using only the information provided by the previously performed PCA.

Supplementary individuals

1. Data: rows 28 to 32 and columns 1 to 7 [in mtcars data sets]. The new data must contain columns (variables) with the same names and in the same order as the active data used to compute PCA.

```
# Data for the supplementary individuals
ind.sup <- mtcars[28:32, 1:7]
ind.sup[, 1:7]
                mpg cyl  disp  hp drat    wt qsec
Lotus Europa   30.4   4  95.1 113 3.77 1.513 16.9
Ford Pantera L 15.8   8 351.0 264 4.22 3.170 14.5
Ferrari Dino   19.7   6 145.0 175 3.62 2.770 15.5
Maserati Bora  15.0   8 301.0 335 3.54 3.570 14.6
Volvo 142E     21.4   4 121.0 109 4.11 2.780 18.6
```

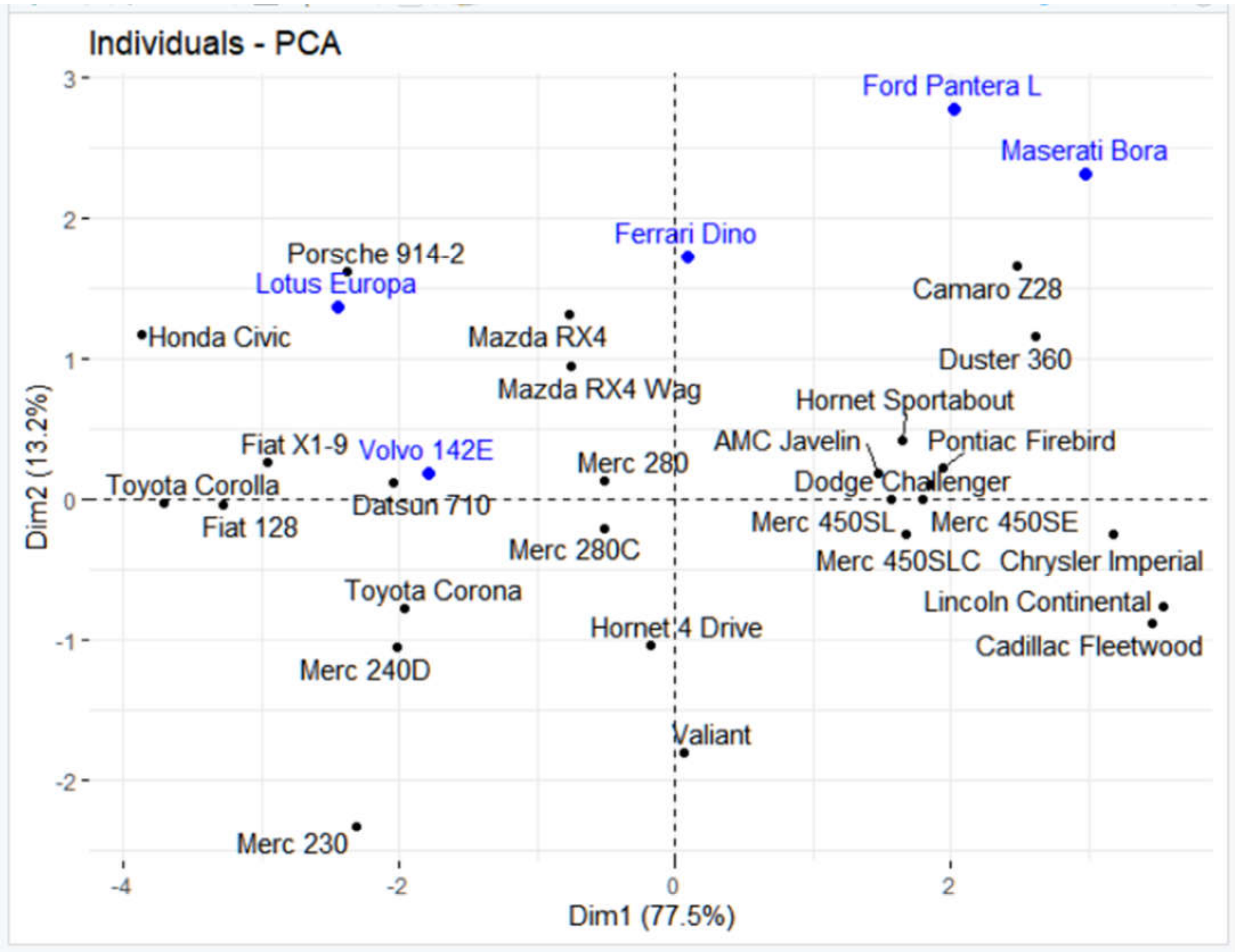
2. Predict the coordinates of new individuals' data. Use the R base function *predict*():

```
#2.  Predict the coordinates of new individuals data
ind.sup.coord <- predict(res.pca, newdata = ind.sup)
ind.sup.coord[, 1:5]

                       PC1       PC2        PC3          PC4         PC5
Lotus Europa   -2.45194747 1.3704319 -0.8815616  0.414130660 -0.26006507
Ford Pantera L  2.02429117 2.7749468  1.0253541  0.106364854 -0.80642423
Ferrari Dino    0.09480883 1.7291169 -0.3189611 -0.502660217 -0.23482367
Maserati Bora   2.96950866 2.3142637  0.6415471 -0.003976738 -1.56575666
Volvo 142E     -1.78805205 0.1889036  0.5809881 -0.526086523  0.05765481
```

3. Graph of individuals including the supplementary individuals:

```
# Plot of active individuals
p <- fviz_pca_ind(res.pca, repel = TRUE)
# Add supplementary individuals
fviz_add(p, ind.sup.coord, color ="blue")
```



The predicted coordinates of individuals can be manually calculated as follow:

1. Center and scale the new individuals data using the center and the scale of the PCA
2. Calculate the predicted coordinates by multiplying the scaled values with the eigenvectors (loadings) of the principal components.

The R code below can be used:

```
# Centering and scaling the supplementary individuals
ind.scaled <- scale(ind.sup,
                    center = res.pca$center,
                    scale = res.pca$scale)
# Coordinates of the individividuals
coord_func <- function(ind, loadings){
 r <- loadings*ind
     apply(r, 2, sum)
}
pca.loadings <- res.pca$rotation
ind.sup.coord <- t(apply(ind.scaled, 1, coord_func, pca.loadings ))
ind.sup.coord[, 1:5]
```

```
                       PC1       PC2        PC3          PC4         PC5
Lotus Europa   -2.45194747 1.3704319 -0.8815616  0.414130660 -0.26006507
Ford Pantera L  2.02429117 2.7749468  1.0253541  0.106364854 -0.80642423
Ferrari Dino    0.09480883 1.7291169 -0.3189611 -0.502660217 -0.23482367
Maserati Bora   2.96950866 2.3142637  0.6415471 -0.003976738 -1.56575666
Volvo 142E     -1.78805205 0.1889036  0.5809881 -0.526086523  0.05765481
```

Supplementary variables

Qualitative / categorical variables

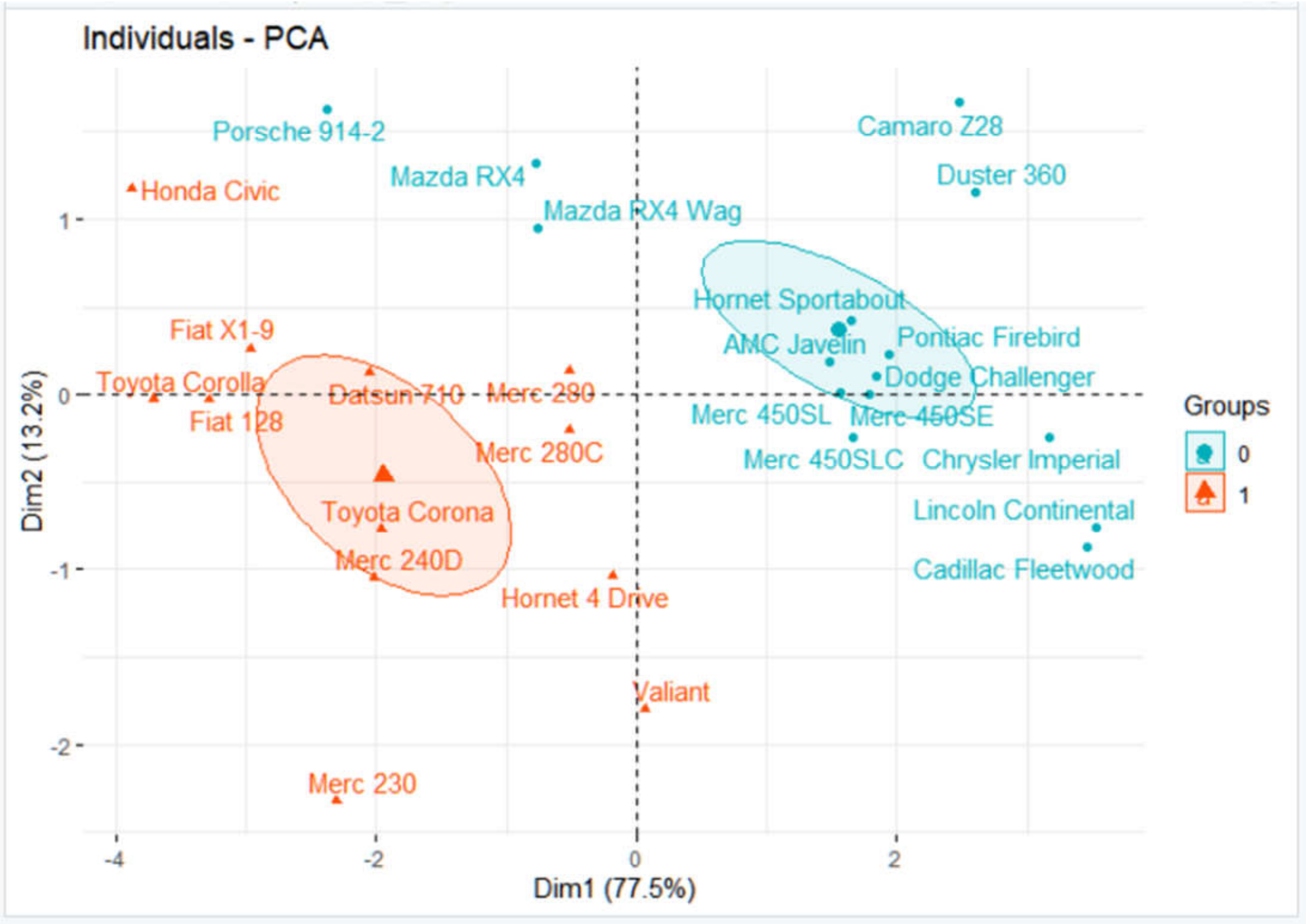
The data sets mtcars contain two *supplementary qualitative variable* at columns 8 and 9(vs and am)

Column 8: `vs`: Engine block: this denotes whether the vehicle's engine is shaped like a "V", or is a more common straight shape.

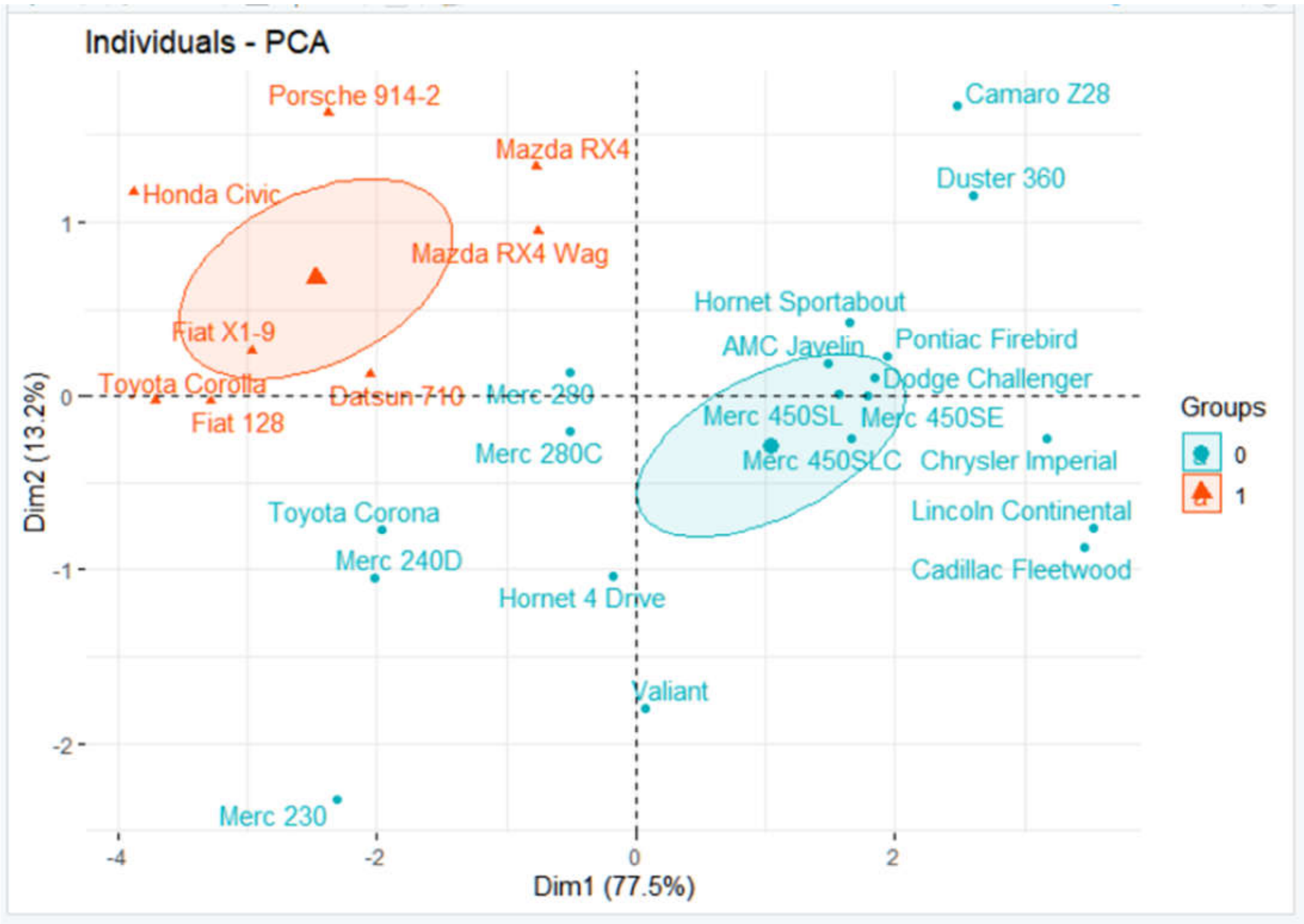
Column 9: `am`: Transmission: this denotes whether the car's transmission is automatic (0) or manual (1).

Qualitative / categorical variables can be used to color individuals by groups. The grouping variable should be of same length as the number of active individuals (here 27).

```
#For group mtcars$vs
groups <- as.factor(mtcars$vs[1:27])
fviz_pca_ind(res.pca,
          col.ind = groups,
          palette = c("#00AFBB",  "#FC4E07"),
          addEllipses = TRUE,
          ellipse.type = "confidence",
          legend.title = "Groups",
          repel = FALSE)
```



```
#For group mtcars$am
groups <- as.factor(mtcars$am[1:27])
fviz_pca_ind(res.pca,
           col.ind = groups,
           palette = c("#00AFBB",  "#FC4E07"),
           addEllipses = TRUE,
           ellipse.type = "confidence",
           legend.title = "Groups",
           repel = TRUE)
```



Calculate the coordinates for the levels of grouping variables. The coordinates for a given group is calculated as the mean coordinates of the individuals in the group.

```
#Calculate the coordinates for the levels of grouping variables
  library(magrittr) # for pipe  %>%
  library(dplyr)   # everything else
  # 1. Individual coordinates
  res.ind <- get_pca_ind(res.pca)
  # 2. Coordinate of groups for column vs
  coord.groups <- res.ind$coord  %>%
   as_tibble()  %>%
   select(Dim.1, Dim.2)  %>%
   mutate(competition = groups)  %>%
   group_by(competition)  %>%
   summarise(
     Dim.1 = mean(Dim.1),
     Dim.2 = mean(Dim.2)
   )
  coord.groups


# A tibble: 2 x 3
  competition Dim.1  Dim.2
  <fct>        <dbl>  <dbl>
1 0             1.55  0.369
2 1            -1.94 -0.462


  # 3. Coordinate of groups for column am
  coord.groups <- res.ind$coord  %>%
   as_tibble()  %>%
   select(Dim.1, Dim.2)  %>%
   mutate(competition = groups)  %>%
   group_by(competition)  %>%
   summarise(
     Dim.1 = mean(Dim.1),
     Dim.2 = mean(Dim.2)
  )
  coord.groups

# A tibble: 2 x 3
  competition Dim.1  Dim.2
  <fct>        <dbl>  <dbl>
1 0             1.04 -0.284
2 1            -2.47  0.674
```

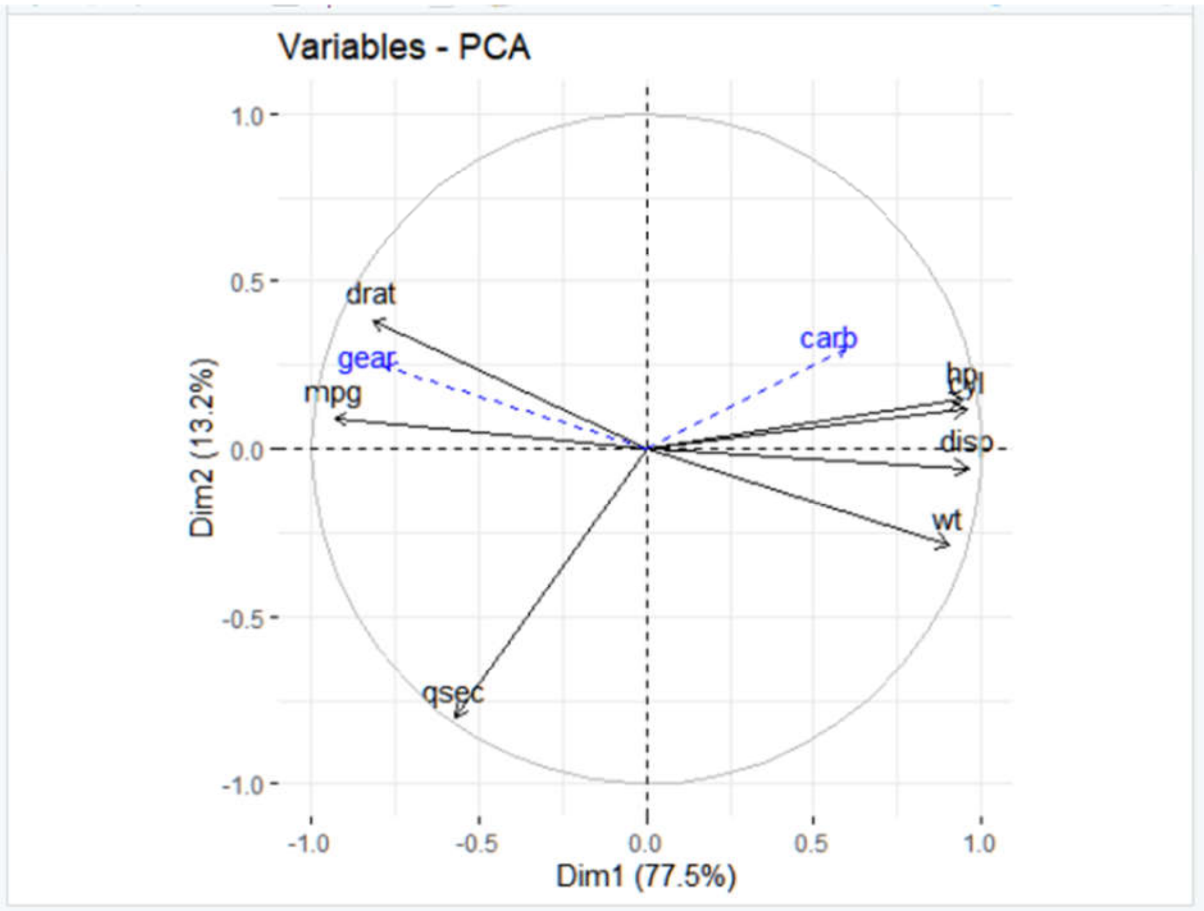
Quantitative variables

Data: columns 10:11(gear and carb). Should be of same length as the number of active individuals (here 27)

```
#Quantitative variables
quanti.sup <- mtcars[1:27, 10:11, drop = FALSE]
head(quanti.sup)
                  gear carb
Mazda RX4            4    4
Mazda RX4 Wag        4    4
Datsun 710           4    1
Hornet 4 Drive       3    1
Hornet Sportabout    3    2
Valiant              3    1
```

The coordinates of a given quantitative variable are calculated as the correlation between the quantitative variables and the principal components.

```
# Predict coordinates and compute cos2
quanti.coord <- cor(quanti.sup, res.pca$x)
quanti.cos2 <- quanti.coord^2
# Graph of variables including supplementary variables
p <- fviz_pca_var(res.pca)
fviz_add(p, quanti.coord, color ="blue", geom="arrow")
```



Theory behind PCA results:

PCA results for variables

Here we'll show how to calculate the PCA results for variables: coordinates cos2 and contributions:

- var.coord = loadings * the component standard deviations
- var.cos2 = var.coord^2
- var.contrib. The contribution of a variable to a given principal component is (in percentage) : (var.cos2 * 100) / (total cos2 of the component)

```
# Helper function
var_coord_func <- function(loadings, comp.sdev){
  loadings*comp.sdev
}
# Compute Coordinates
loadings <- res.pca$rotation
sdev <- res.pca$sdev
var.coord <- t(apply(loadings, 1, var_coord_func, sdev))
head(var.coord[, 1:5])
```

```
            PC1         PC2         PC3           PC4          PC5
mpg  -0.9335653  0.09192380 -0.08241515  0.3302364350 -0.009108812
cyl   0.9579618  0.11983499 -0.13686937  0.0516067715 -0.075406903
disp  0.9625280 -0.05740243  0.06792022  0.1968515031  0.073187667
hp    0.9441132  0.14864209  0.16689472  0.0483241098 -0.216735245
drat -0.8194887  0.38434965  0.41602031 -0.0008928492 -0.009538592
wt    0.9042815 -0.28630834  0.25611063  0.0413921167  0.139459612
```

```
# Compute Cos2
var.cos2 <- var.coord^2
head(var.cos2[, 1:5])
```

```
           PC1         PC2         PC3          PC4          PC5
mpg  0.8715442 0.008449985 0.006792257 1.090561e-01 8.297046e-05
cyl  0.9176907 0.014360425 0.018733224 2.663259e-03 5.686201e-03
disp 0.9264602 0.003295039 0.004613156 3.875051e-02 5.356435e-03
hp   0.8913497 0.022094470 0.027853849 2.335220e-03 4.697417e-02
drat 0.6715618 0.147724655 0.173072901 7.971797e-07 9.098474e-05
wt   0.8177251 0.081972468 0.065592656 1.713307e-03 1.944898e-02
```

```
# Compute contributions
comp.cos2 <- apply(var.cos2, 2, sum)
contrib <- function(var.cos2, comp.cos2){var.cos2*100/comp.cos2}
var.contrib <- t(apply(var.cos2,1, contrib, comp.cos2))
head(var.contrib[, 1:5])
```

```
          PC1        PC2       PC3          PC4         PC5
mpg  16.06936  0.9146480  2.209207 7.028950e+01  0.09203864
cyl  16.92021  1.5544091  6.093052 1.716540e+00  6.30766860
disp 17.08190  0.3566634  1.500447 2.497572e+01  5.94186075
hp   16.43454  2.3915618  9.059570 1.505110e+00 52.10816038
drat 12.38213 15.9900935 56.292615 5.138031e-04  0.10092882
wt   15.07706  8.8729090 21.334259 1.104271e+00 21.57464019
```

PCA results for individuals

- ind.coord = res.pca$x
- Cos2 of individuals. Two steps:
  - Calculate the square distance between each individual and the PCA center of gravity: d2 = [(var1_ind_i - mean_var1)/sd_var1]^2 + …+ [(var10_ind_i - mean_var10)/sd_var10]^2 + …+..
  - Calculate the cos2 as ind.coord^2/d2
- Contributions of individuals to the principal components: 100 * (1 / number_of_individuals)*(ind.coord^2 / comp_sdev^2). Note that the sum of all the contributions per column is 100

```
  # Coordinates of individuals
  ind.coord <- res.pca$x
  head(ind.coord[, 1:5])
                          PC1        PC2         PC3        PC4         PC5
Mazda RX4         -0.76916610  1.3175822 -0.24874355 -0.3792334  0.27441449
Mazda RX4 Wag     -0.75587908  0.9503246 -0.06580908 -0.3302008  0.26290976
Datsun 710        -2.04818889  0.1205521 -0.10736211 -0.4655663  0.02045689
Hornet 4 Drive    -0.18437466 -1.0368469 -0.64304549  0.2459907  0.09268826
Hornet Sportabout  1.64578560  0.4254479 -0.50874341  0.4870905 -0.11313988
Valiant            0.06726851 -1.7937771 -0.84600854 -0.2885557  0.05783152
```

```
  # Cos2 of individuals
  # 1. square of the distance between an individual and the
  # PCA center of gravity
  center <- res.pca$center
  scale<- res.pca$scale
  getdistance <- function(ind_row, center, scale){
   return(sum(((ind_row-center)/scale)^2))
  }
  d2 <- apply(mtcars.active,1,getdistance, center, scale)
  # 2. Compute the cos2. The sum of each row is 1
  cos2 <- function(ind.coord, d2){return(ind.coord^2/d2)}
  ind.cos2 <- apply(ind.coord, 2, cos2, d2)
  head(ind.cos2[, 1:5])


                          PC1         PC2         PC3        PC4          PC5
 Mazda RX4         0.225686659 0.662248632 0.023603113 0.05486290 2.872630e-02
 Mazda RX4 Wag     0.333804474 0.527632353 0.002530225 0.06370063 4.038322e-02
 Datsun 710        0.901828655 0.003124164 0.002477913 0.04659583 8.996287e-05
 Hornet 4 Drive    0.020910177 0.661278634 0.254354044 0.03722140 5.284510e-03
 Hornet Sportabout 0.780496900 0.052157569 0.074579980 0.06836659 3.688554e-03
 Valiant           0.001121366 0.797371774 0.177367234 0.02063398 8.288063e-04
```

```r
# Contributions of individuals
contrib <- function(ind.coord, comp.sdev, n.ind){
  100*(1/n.ind)*ind.coord^2/comp.sdev^2
}
ind.contrib <- t(apply(ind.coord, 1, contrib,
                       res.pca$sdev, nrow(ind.coord)))
head(ind.contrib[, 1:5])
```

```
                          PC1         PC2        PC3      PC4        PC5
Mazda RX4         0.404004132  6.95968701 0.74535344 3.433127 3.09383366
Mazda RX4 Wag     0.390166692  3.62058023 0.05217112 2.602753 2.83985598
Datsun 710        2.864742215  0.05826183 0.13885473 5.174158 0.01719342
Hornet 4 Drive    0.023213894  4.30986326 4.98129178 1.444489 0.35296557
Hornet Sportabout 1.849660611  0.72564954 3.11785694 5.663644 0.52591349
Valiant           0.003090076 12.89945071 8.62200131 1.987632 0.13740801
```

THE END