DIMENSIONALITY REDUCTION IN R

Dimensionality Reduction

Alexandros Tantos
Assistant Professor
Aristotle University of Thessaloniki

Curse of Dimensionality

- **Dimensions**: Columns in the dataset that represent features of the row points
- **Dimensionality**: Number of features/columns characterizing the dataset

Curse of Dimensionality

The `iris` dataset:

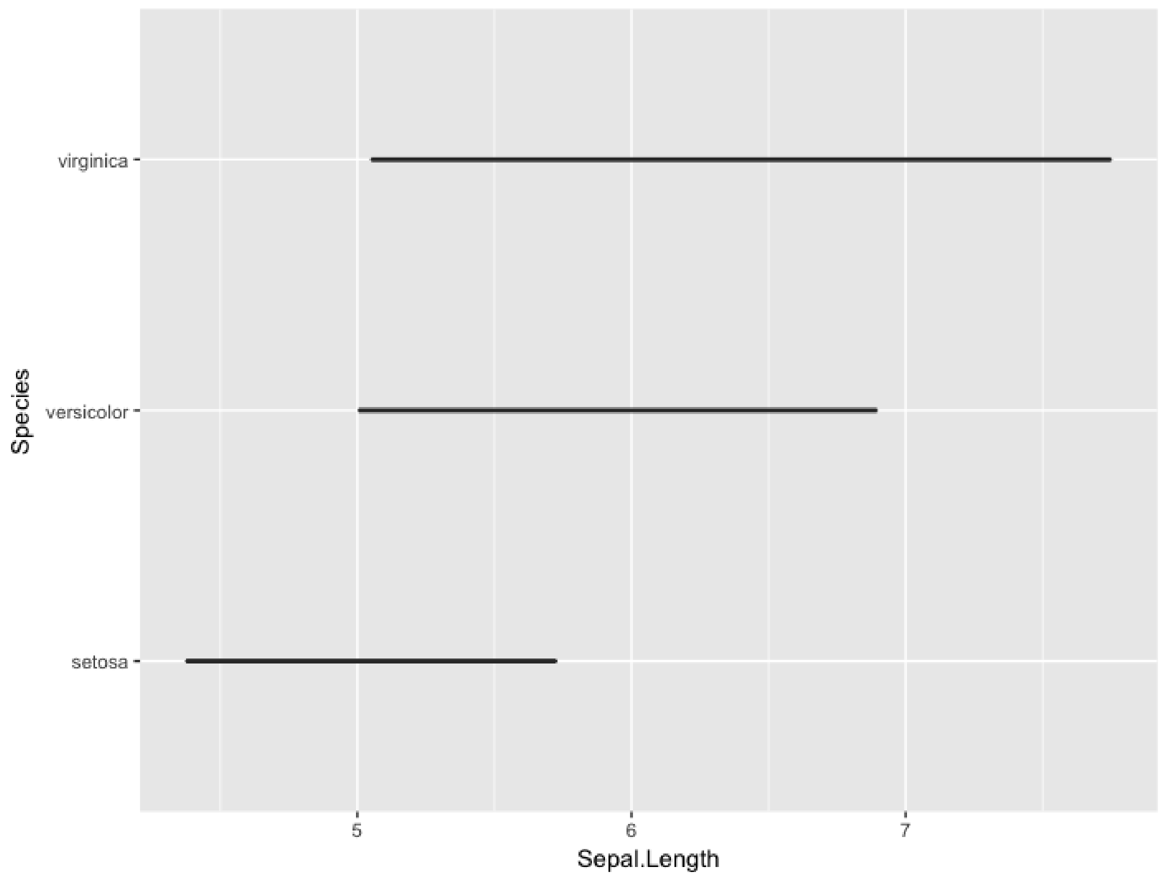
```
dim(iris)
```

```
[1] 150   5
```

5 columns: **4 features/dimensions** + 1 class

ID	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width
1	5.1	3.5	1.4	0.2
2	4.9	3.0	1.4	0.2
3	4.7	3.2	1.3	0.2
...	...	...	...	...

1 Dimension: Sepal.Length

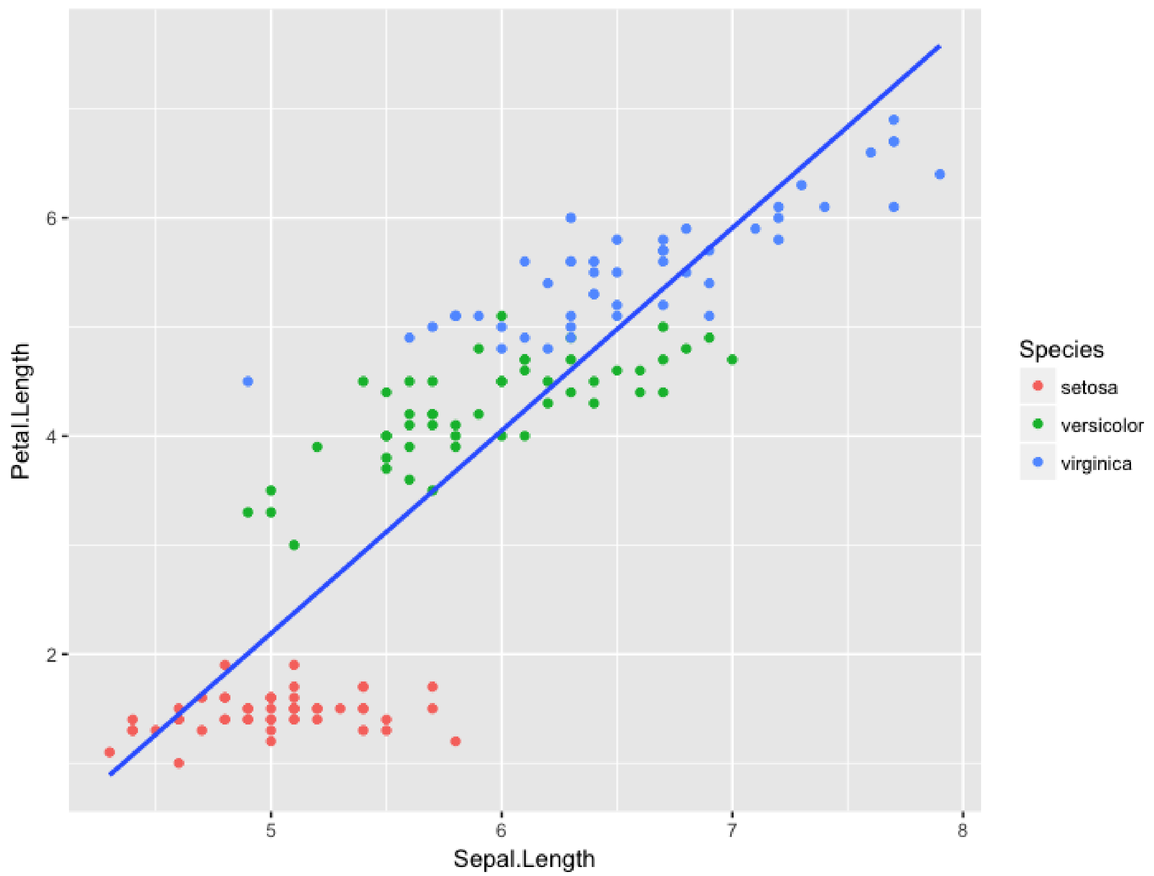



```
range(iris$Sepal.Length)

[1] 4.3 7.9
```

- Feature space filled within **4** units of measurement.
- Data density: 150/4 = **37.5** samples/interval.

2 Dimensions: Sepal.Length, Petal.Length

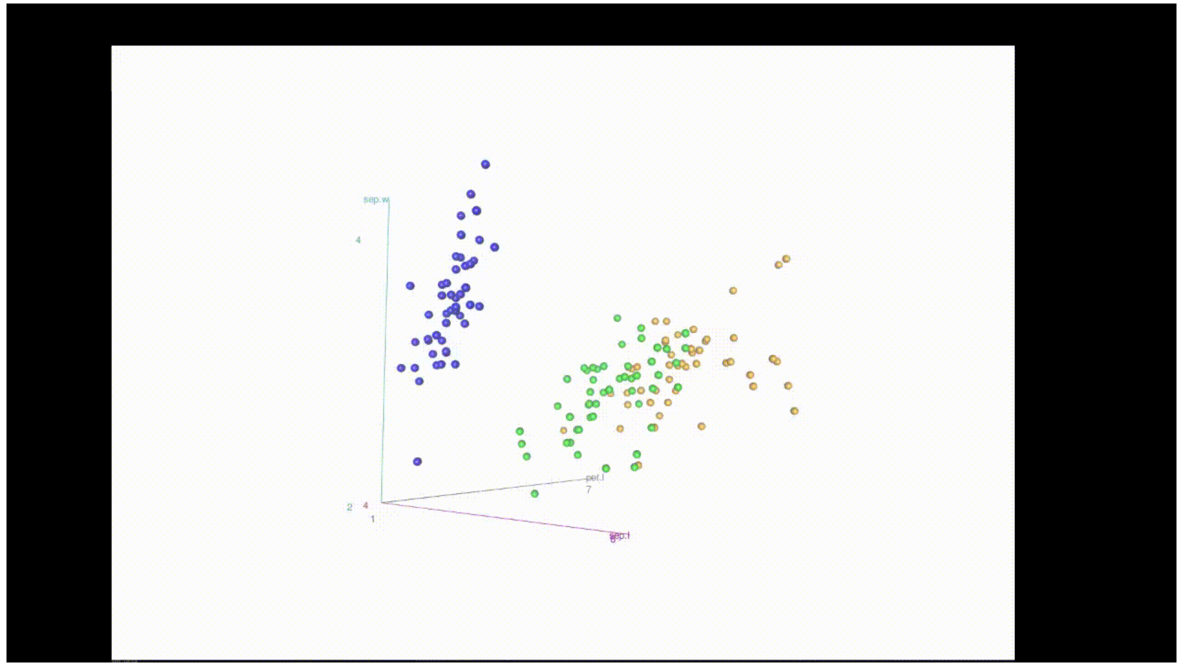



```
range(iris$Petal.Length)

[1] 1.0 6.9
```

- Feature space: filled within **24 [4*6]** possible combinations of unit measurements.
- Data density: 150/24 = **6.25** samples/interval

3 Dimensions: Sepal.Length, Petal.Length, Sepal.Width



```
range(iris$Sepal.Width)

[1] 2.0 4.4
```

- Feature space: filled within **72 [4*6*3]** possible combinations of unit measurements.
- Data density: 150/72 = **2.083333** samples/interval

What is this curse all about?

- **As the dimensionalities of the data grow, the feature space grows rapidly.**

Why even bother?

- **Big computational cost** to handle high-dimensional data.
- **Estimation accuracy** decreases.
- **Difficult interpretation** of the data.

The mtcars dataset

```
dim(mtcars)
```

```
[1] 32 11
```

- Most of the dimensions could probably be reduced due to a small set of latent dimensions, such as:
  - the size of the car or
  - the country of origin or
  - the construction year
- **Observed vs True Dimensionality**: observed features obscure the true or *intrinsic* dimensionality of the data.

Exploring correlation

How do we trace correlation patterns?

- **Correlation matrix** is a matrix of correlation coefficients.
- Smaller number of dimensions translates to less complex correlation matrix.

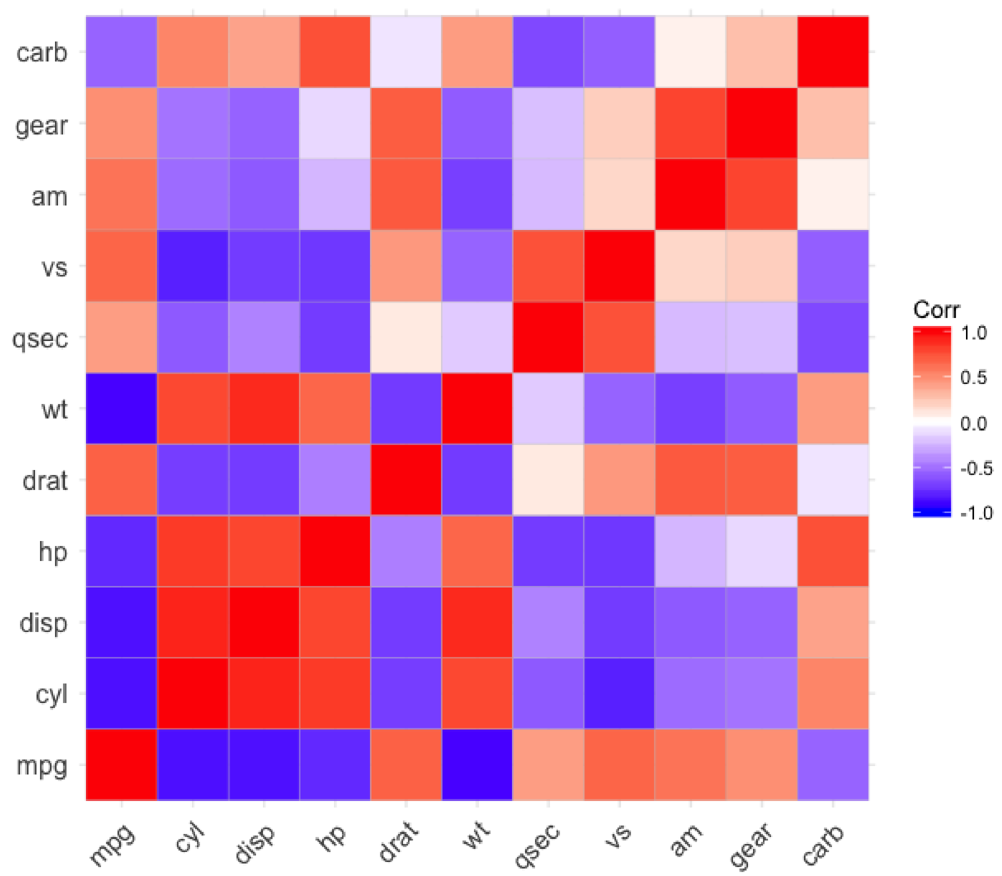
```
mtcars$cyl <- as.numeric(as.character(mtcars$cyl))

mtcars_correl <- cor(mtcars, use = "complete.obs")
```

	mpg	cyl	disp	hp	drat	wt	qsec	vs	am	gear	carb
mpg	1.0000000	-0.8521620	-0.8475514	-0.7761684	0.68117191	-0.8676594	0.41868403	0.6640389	0.59983243	0.4802848	-0.55092507
cyl	-0.8521620	1.0000000	0.9020329	0.8324475	-0.69993811	0.7824958	-0.59124207	-0.8108118	-0.52260705	-0.4926866	0.52698829
disp	-0.8475514	0.9020329	1.0000000	0.7909486	-0.71021393	0.8879799	-0.43369788	-0.7104159	-0.59122704	-0.5555692	0.39497686
hp	-0.7761684	0.8324475	0.7909486	1.0000000	-0.44875912	0.6587479	-0.70822339	-0.7230967	-0.24320426	-0.1257043	0.74981247
drat	0.6811719	-0.6999381	-0.7102139	-0.4487591	1.00000000	-0.7124406	0.09120476	0.4402785	0.71271113	0.6996101	-0.09078980
wt	-0.8676594	0.7824958	0.8879799	0.6587479	-0.71244065	1.0000000	-0.17471588	-0.5549157	-0.69249526	-0.5832870	0.42760594
qsec	0.4186840	-0.5912421	-0.4336979	-0.7082234	0.09120476	-0.1747159	1.00000000	0.7445354	-0.22986086	-0.2126822	-0.65624923
vs	0.6640389	-0.8108118	-0.7104159	-0.7230967	0.44027846	-0.5549157	0.74453544	1.0000000	0.16834512	0.2060233	-0.56960714

Visualising correlation patterns with ggcorrplot

```
library(ggcorrplot)
ggcorrplot(mtcars_correl)
```



How do we deal with the Curse of Dimensionality?

Two solutions:

- Feature Engineering: Requires domain knowledge
- **Remove redundancy**

Reduction methods we will explore

- **Principal Components Analysis [PCA]**
- Non-Negative Matrix Factorization [N-NMF]
- Exploratory Factor Analysis [EFA]

DIMENSIONALITY REDUCTION IN R

Let's practice!

DIMENSIONALITY REDUCTION IN R

Getting PCA to work with FactoMineR

Alexandros Tantos
Assistant Professor
Aristotle University of Thessaloniki

PCA: What does it do?

Conceptually:

1. Removes correlation.
2. Extracts new dimensions (=*principal components*).
3. Reveals the true dimensionality of the data.

Practically:

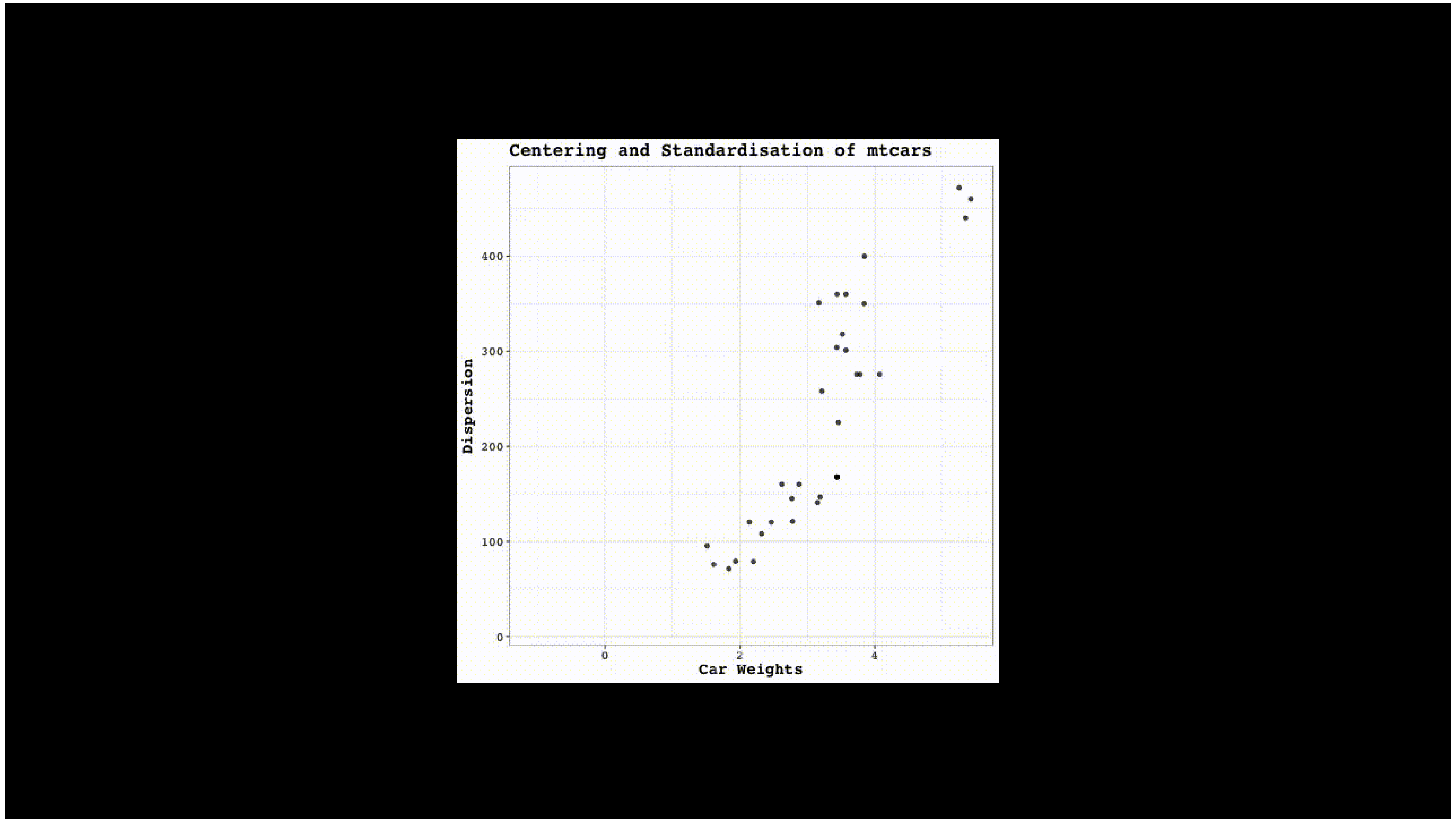
1. Decomposes the correlation matrix.
2. Changes the coordinate system.
3. Helps reduce the number of dimensions.

PCA: The five steps to perform

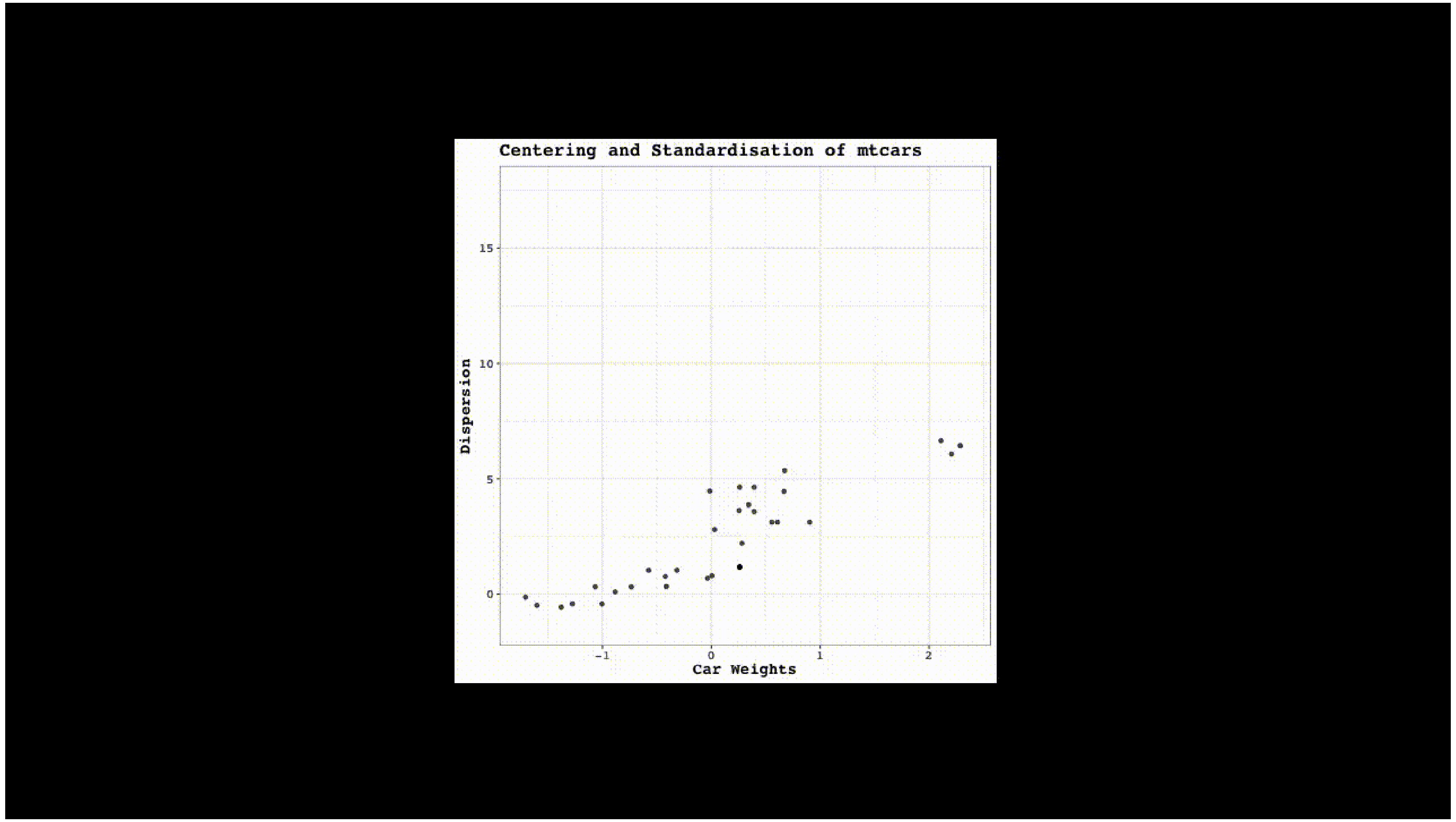
1. Pre-processing steps
2. Change of coordinate system
3. Explained variance

- Centering
- Standardisation
- Rotation
- Projection
- Reduction

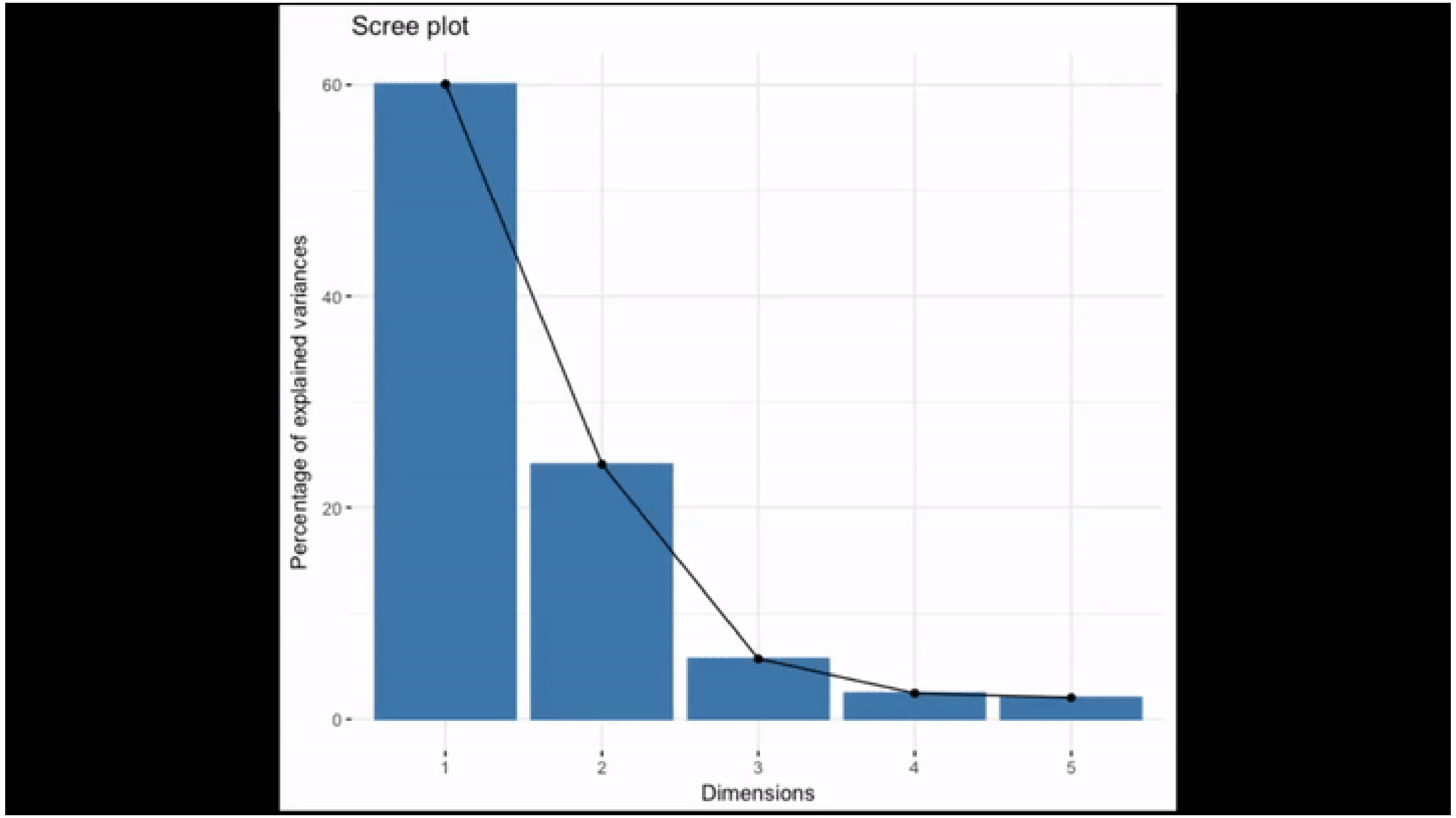
Pre-processing steps: Data Centering and Standardisation



Change of coordinate system: Rotation and Projection



Reduction: Screeplot and the explained variance



PCA with base R's prcomp()

```
mtcars_pca <- prcomp(mtcars)
```

```
Standard deviations (1, .., p=11):
 [1] 136.5330479  38.1480776   3.0710166   1.3066508   0.9064862   0.6635411   0.3085791   0.2859604   0.2506973   0.2106519   0.1984238

Rotation (n x k) = (11 x 11):
              PC1          PC2          PC3          PC4         PC5          PC6          PC7           PC8          PC9         PC10          PC11
mpg  -0.038118199  0.009184847  0.982070847  0.047634784 -0.08832843 -0.143790084 -0.039239174  2.271040e-02 -0.002790139  0.030630361 -0.0158569365
cyl   0.012035150 -0.003372487 -0.063483942 -0.227991962  0.23872590 -0.793818050  0.425011021 -1.890403e-01  0.042677206  0.131718534  0.1454453628
disp  0.899568146  0.435372320  0.031442656 -0.005086826 -0.01073597  0.007424138  0.000582398 -5.841464e-04  0.003532713 -0.005399132  0.0009420262
hp    0.434784387 -0.899307303  0.025093049  0.035715638  0.01655194  0.001653685 -0.002212538  4.748087e-06 -0.003734085  0.001862554 -0.0021526102
drat -0.002660077 -0.003900205  0.039724928 -0.057129357 -0.13332765  0.227229260  0.034847411 -9.385817e-01 -0.014131110  0.184102094 -0.0973818815
wt    0.006239405  0.004861023 -0.084910258  0.127962867 -0.24354296 -0.127142296 -0.186558915  1.561907e-01 -0.390600261  0.829886844 -0.0198581635
qsec -0.006671270  0.025011743 -0.071670457  0.886472188 -0.21416101 -0.189564973  0.254844548 -1.028515e-01 -0.095914479 -0.204240658  0.0110677880
vs   -0.002729474  0.002198425  0.004203328  0.177123945 -0.01688851  0.102619063 -0.080788938 -2.132903e-03  0.684043835  0.303060724  0.6256900918
am   -0.001962644 -0.005793760  0.054806391 -0.135658793 -0.06270200  0.205217266  0.200858874 -2.273255e-02 -0.572372433 -0.162808201  0.7331658036
gear -0.002604768 -0.011272462  0.048524372 -0.129913811 -0.27616440  0.334971103  0.801625551  2.174878e-01  0.156118559  0.203540645 -0.1909325849
carb  0.005766010 -0.027779208 -0.102897231 -0.268931427 -0.85520810 -0.283788381 -0.165474186  3.972219e-03  0.127583043 -0.239954748  0.0557957968
```

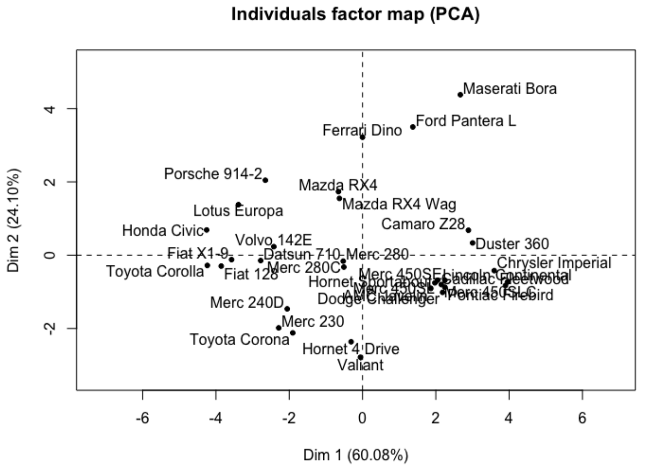
PCA with FactoMineR's PCA()

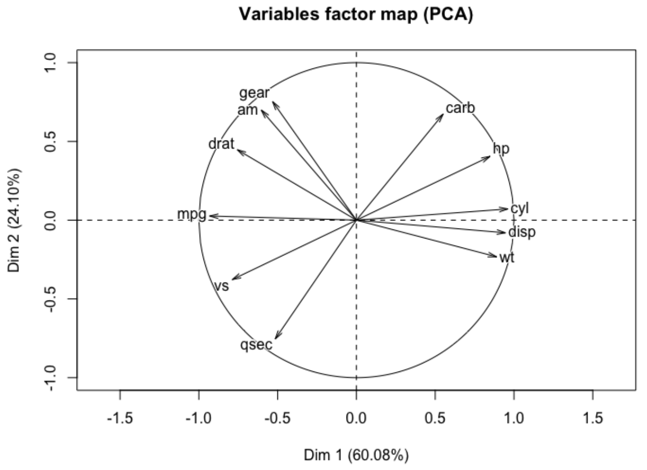
```
library(FactoMineR)

mtcars_pca <- PCA(mtcars)
```

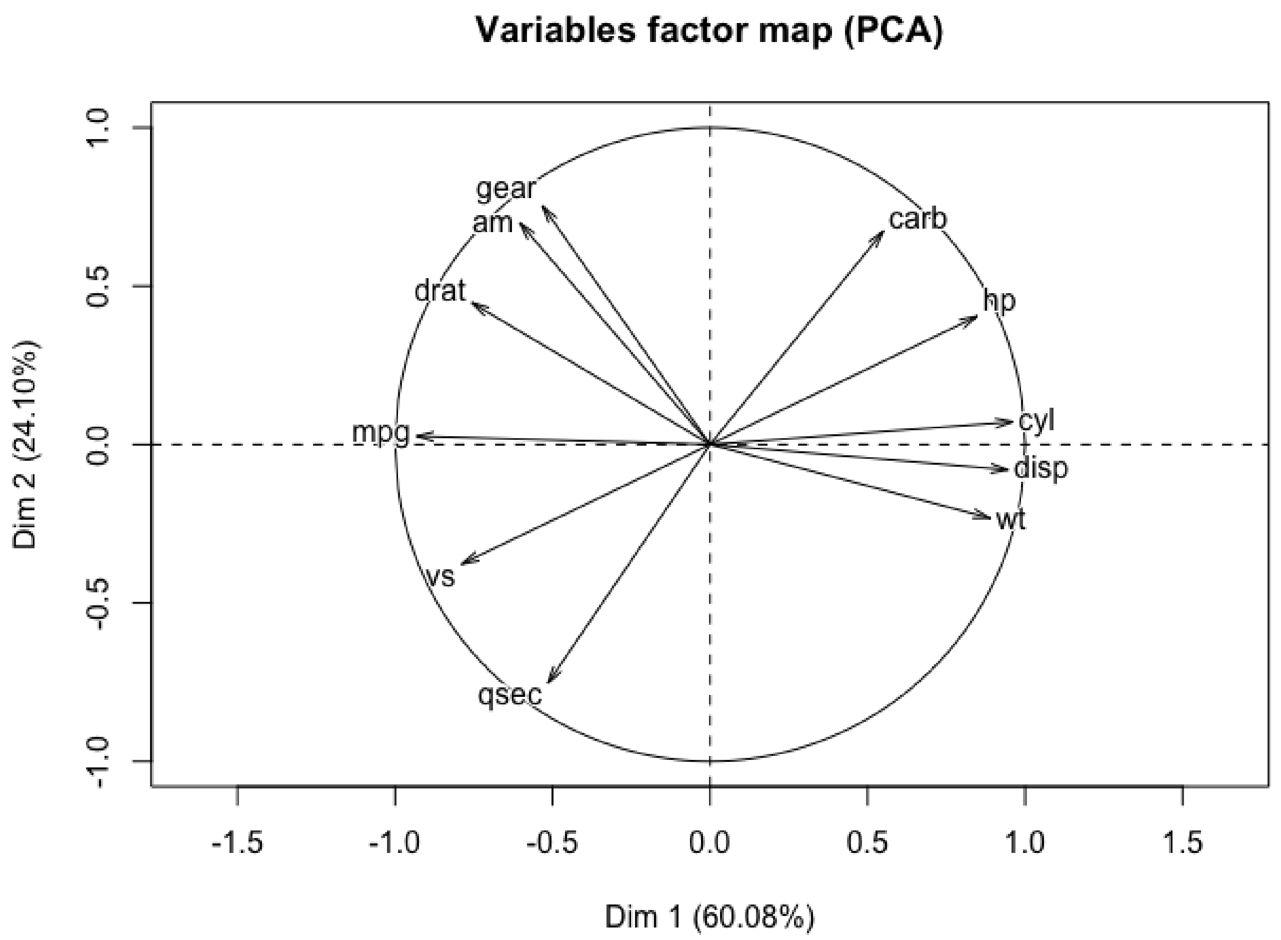
```
**Results for the Principal Component Analysis (PCA)**
The analysis was performed on 32 individuals, described by 11 variables
*The results are available in the following objects:

   name               description
1  "$eig"             "eigenvalues"
2  "$var"             "results for the variables"
3  "$var$coord"       "coord. for the variables"
4  "$var$cor"         "correlations variables - dimensions"
5  "$var$cos2"        "cos2 for the variables"
6  "$var$contrib"     "contributions of the variables"
7  "$ind"             "results for the individuals"
8  "$ind$coord"       "coord. for the individuals"
9  "$ind$cos2"        "cos2 for the individuals"
10 "$ind$contrib"     "contributions of the individuals"
11 "$call"            "summary statistics"
12 "$call$centre"     "mean of the variables"
13 "$call$ecart.type" "standard error of the variables"
14 "$call$row.w"      "weights for the individuals"
15 "$call$col.w"      "weights for the variables"
```





Variables' factor map



Digging into PCA()

```
mtcars_pca$eig
```

```
        eigenvalue percentage of variance cumulative percentage of variance
comp 1  6.60840025             60.0763659                          60.07637
comp 2  2.65046789             24.0951627                          84.17153
comp 3  0.62719727              5.7017934                          89.87332
comp 4  0.26959744              2.4508858                          92.32421
comp 5  0.22345110              2.0313737                          94.35558
comp 6  0.21159612              1.9236011                          96.27918
comp 7  0.13526199              1.2296544                          97.50884
comp 8  0.12290143              1.1172858                          98.62612
comp 9  0.07704665              0.7004241                          99.32655
comp 10 0.05203544              0.4730495                          99.79960
comp 11 0.02204441              0.2004037                         100.00000
```

```
mtcars_pca$var$cos2
```

```
        Dim.1        Dim.2       Dim.3        Dim.4        Dim.5
mpg  0.8685312 0.0006891117 0.031962249 1.369725e-04 0.0023634487
cyl  0.9239416 0.0050717032 0.019276287 1.811054e-06 0.0007642822
disp 0.8958370 0.0064482423 0.002370993 1.775235e-02 0.0346868281
hp   0.7199031 0.1640467049 0.012295659 1.234773e-03 0.0651697911
drat 0.5717921 0.1999959326 0.016295731 1.970035e-01 0.0013361275
wt   0.7916038 0.0542284172 0.073281663 1.630161e-02 0.0012578888
qsec 0.2655437 0.5690984542 0.101947952 1.249426e-03 0.0060588455
vs   0.6208539 0.1422249798 0.115330572 1.244460e-02 0.0803189801
am   0.3647715 0.4887450097 0.026555457 2.501834e-04 0.0018011675
gear 0.2829342 0.5665806069 0.052667265 1.888829e-02 0.0005219259
carb 0.3026882 0.4533387304 0.175213444 4.333912e-03 0.0291718181
```

Digging into PCA()

```
mtcars_pca$var$contrib
```

```
         Dim.1       Dim.2      Dim.3        Dim.4      Dim.5
mpg  13.142837  0.02599962  5.0960440 5.080631e-02  1.0577029
cyl  13.981320  0.19135124  3.0734010 6.717622e-04  0.3420355
disp 13.556034  0.24328694  0.3780299 6.584761e+00 15.5232297
hp   10.893757  6.18934888  1.9604134 4.580062e-01 29.1651238
drat  8.652504  7.54568403  2.5981826 7.307322e+01  0.5979507
wt   11.978751  2.04599412 11.6839894 6.046647e+00  0.5629370
qsec  4.018275 21.47162226 16.2545274 4.634414e-01  2.7114861
vs    9.394919  5.36603293 18.3882452 4.615993e+00 35.9447677
am    5.519816 18.43995209  4.2339880 9.279888e-02  0.8060678
gear  4.281433 21.37662593  8.3972408 7.006107e+00  0.2335750
carb  4.580356 17.10410194 27.9359384 1.607550e+00 13.0551238
```

```
dimdesc(mtcars_pca)
```

```
$Dim.1
$Dim.1$quanti
     correlation      p.value
cyl    0.9612188 2.471950e-18
disp   0.9464866 2.804047e-16
wt     0.8897212 9.780198e-12
hp     0.8484710 8.622043e-10
carb   0.5501711 1.105272e-03
qsec  -0.5153093 2.542578e-03
gear  -0.5319156 1.728737e-03
am    -0.6039632 2.520665e-04
drat  -0.7561693 5.575736e-07
vs    -0.7879428 8.658012e-08
mpg   -0.9319502 9.347042e-15


$Dim.2
$Dim.2$quanti
     correlation      p.value
gear   0.7527155 6.712704e-07
am     0.6991030 8.541542e-06
carb   0.6733043 2.411011e-05
drat   0.4472090 1.028069e-02
hp     0.4050268 2.147312e-02
vs    -0.3771273 3.335771e-02
qsec  -0.7543861 6.138696e-07


$Dim.3
$Dim.3$quanti
     correlation    p.value
carb    0.418585 0.01711089
```

DIMENSIONALITY REDUCTION IN R

Let's practice!

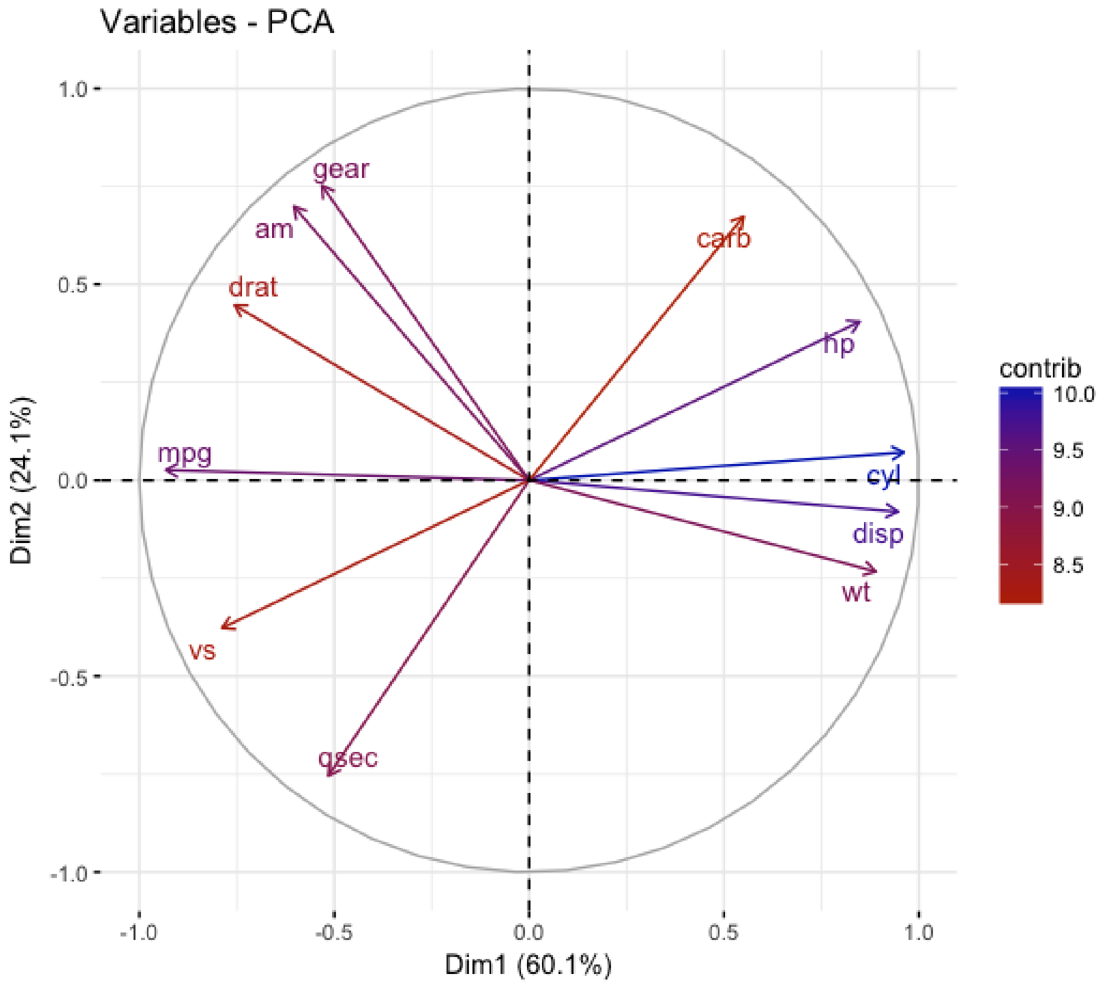
DIMENSIONALITY REDUCTION IN R

Interpreting and visualising PCA models with factoextra

Alexandros Tantos
Assistant Professor
Aristotle University of Thessaloniki

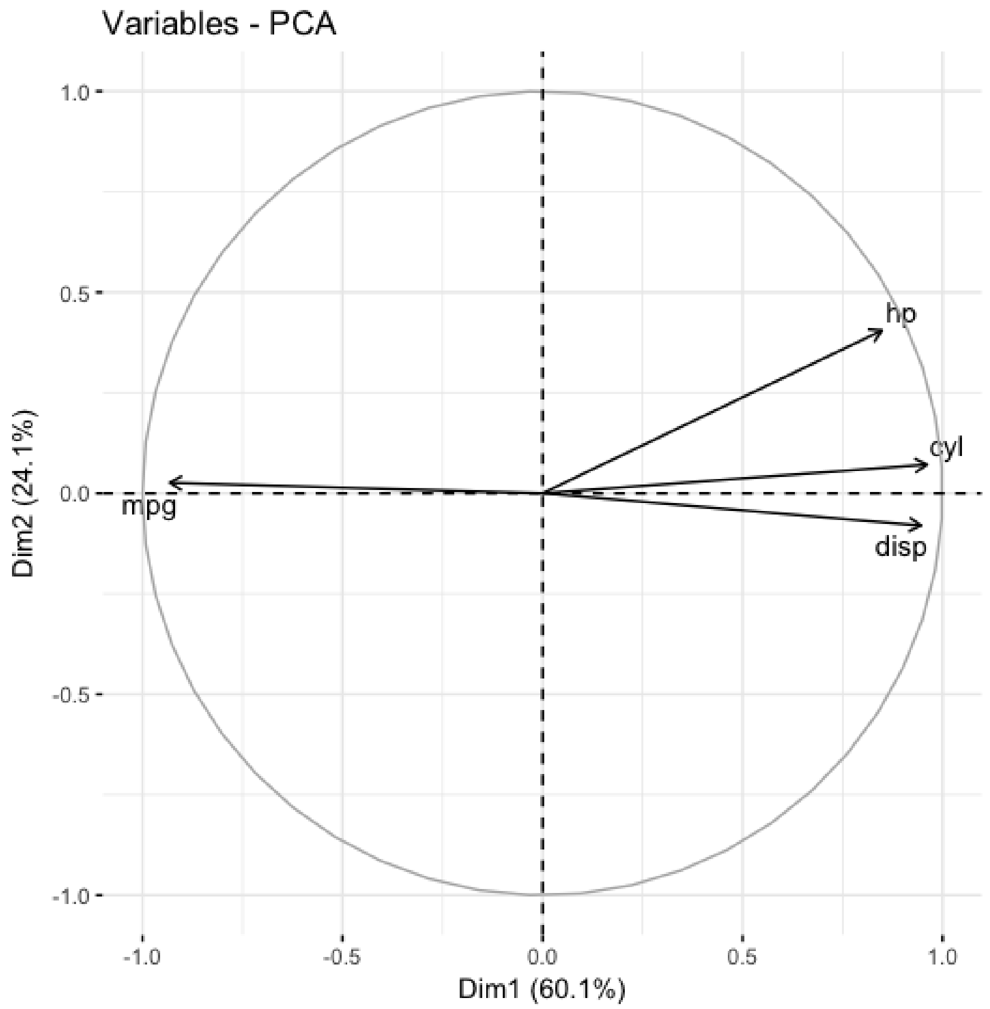
Plotting contributions of variables

```
fviz_pca_var(mtcars_pca,
 col.var = "contrib",
 gradient.cols = c("#bb2e00", "#002bbb
 repel = TRUE)
```



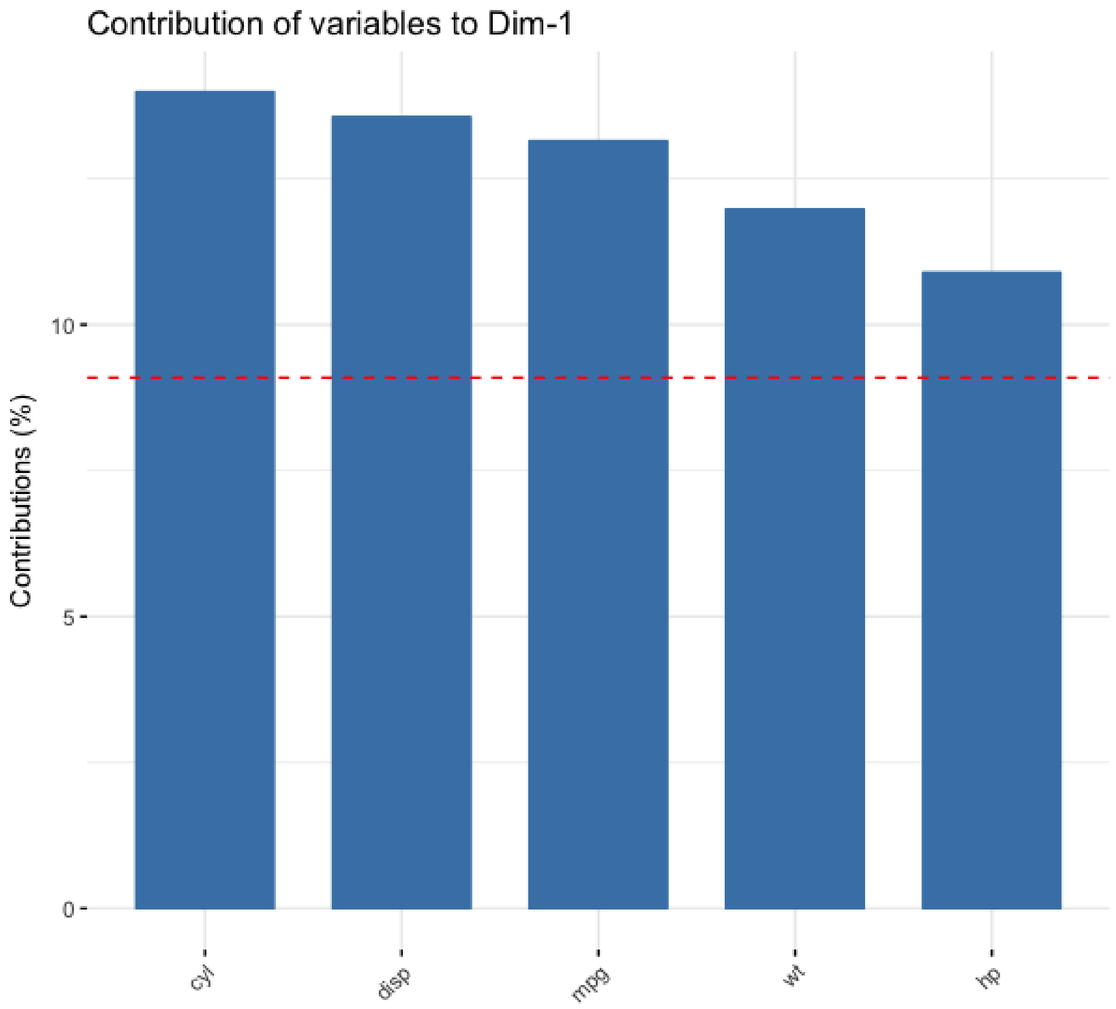
Plotting contributions of selected variables

```
fviz_pca_var(mtcars_pca,
 select.var = list(contrib = 4),
 repel = TRUE)
```



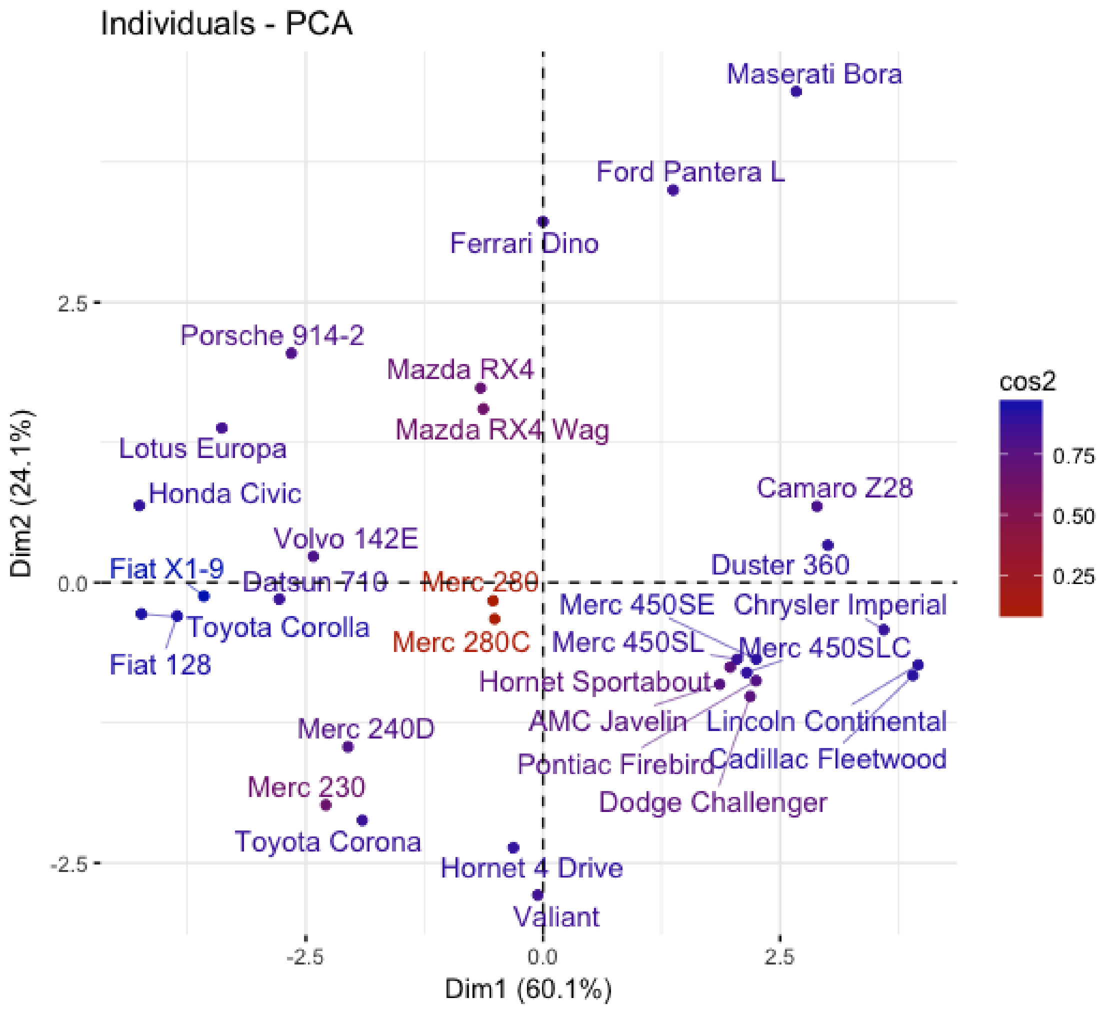
Barplotting the contributions of variables

```
fviz_contrib(mtcars_pca,
    choice = "var",
    axes = 1,
    top = 5)
```



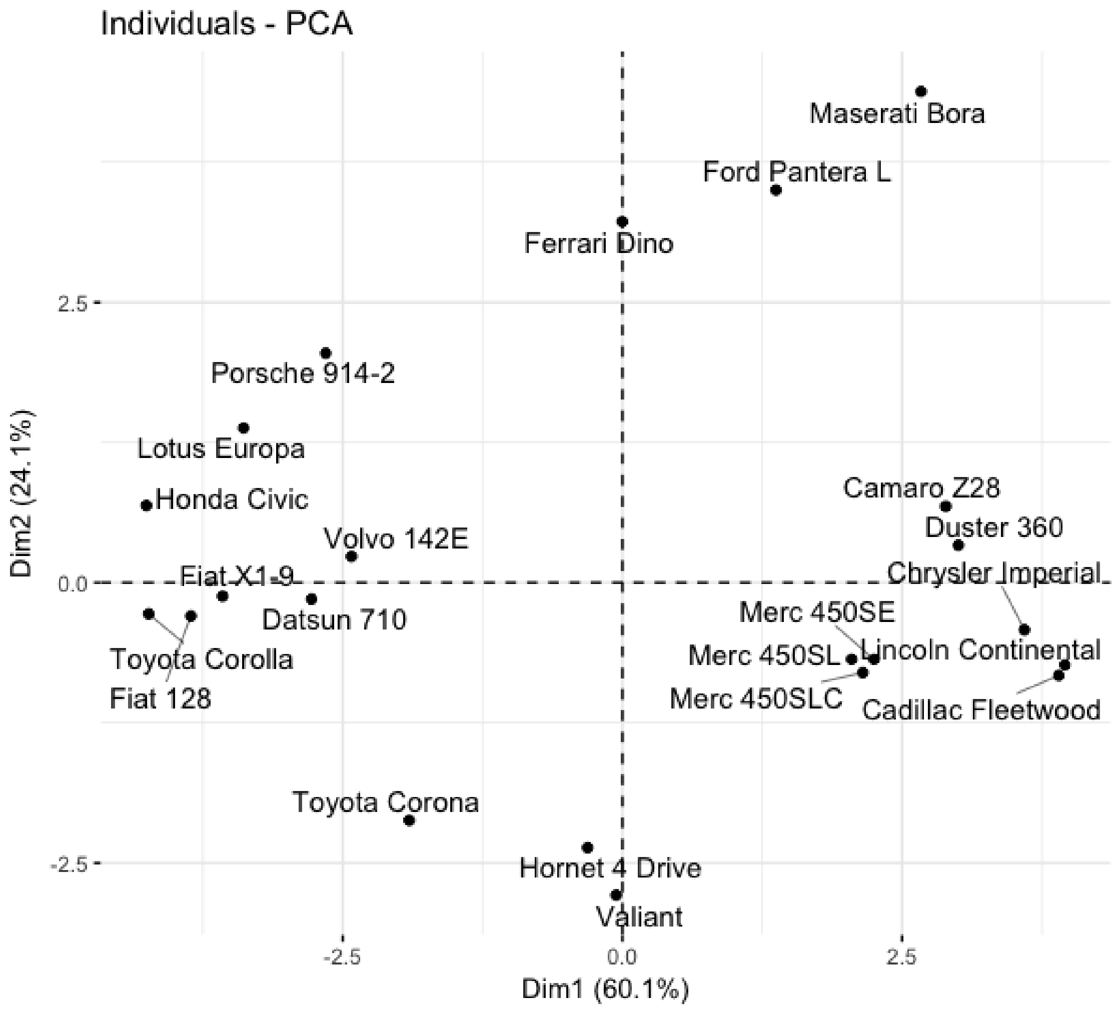
Plotting cos2 for individuals

```
fviz_pca_ind(mtcars_pca,
    col.ind="cos2",
    gradient.cols = c("#bb2e00", "#002
    repel = TRUE)
```



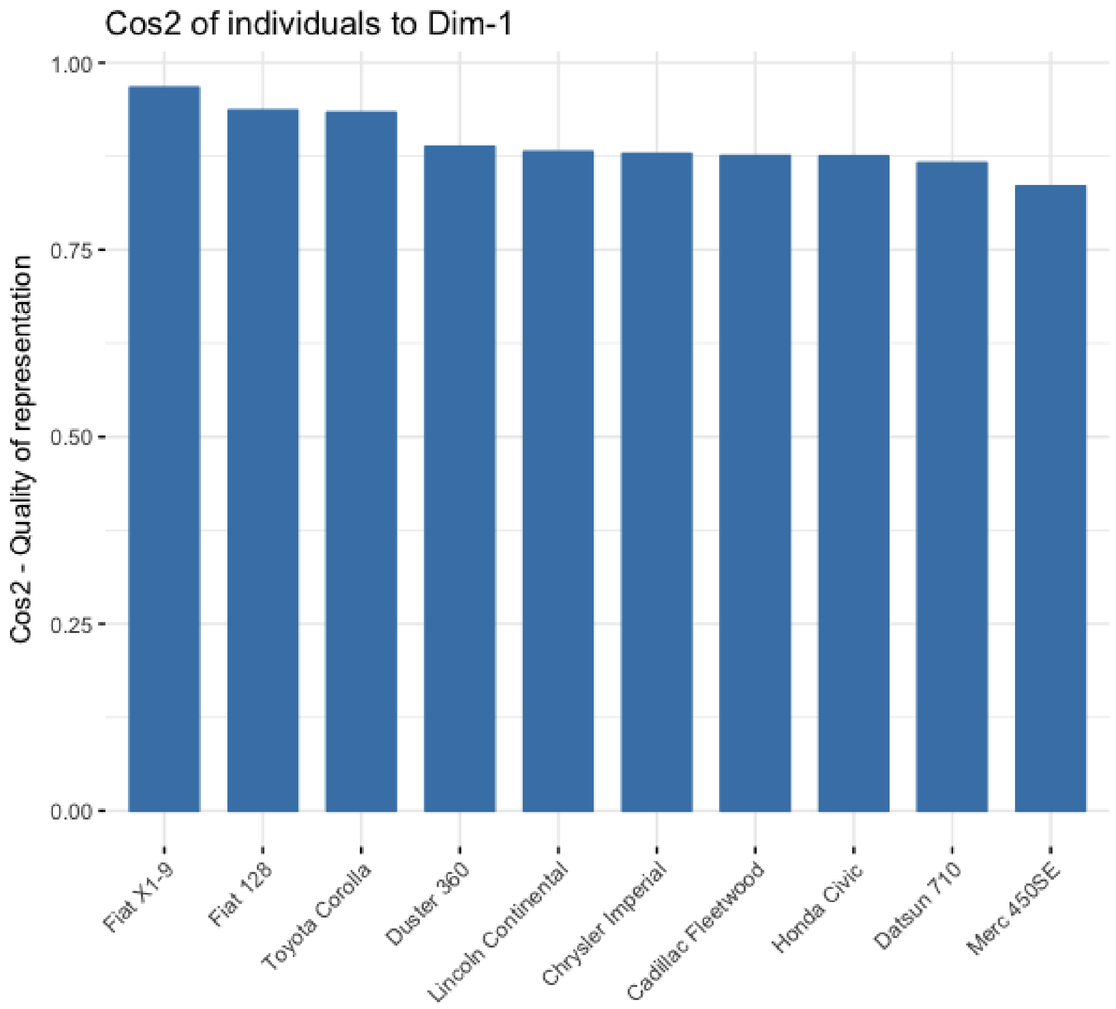
Plotting cos2 for selected individuals

```
fviz_pca_ind(mtcars_pca,
    select.ind = list(cos2 = 0.8),
    gradient.cols = c("#bb2e00", "#002
    repel = TRUE)
```



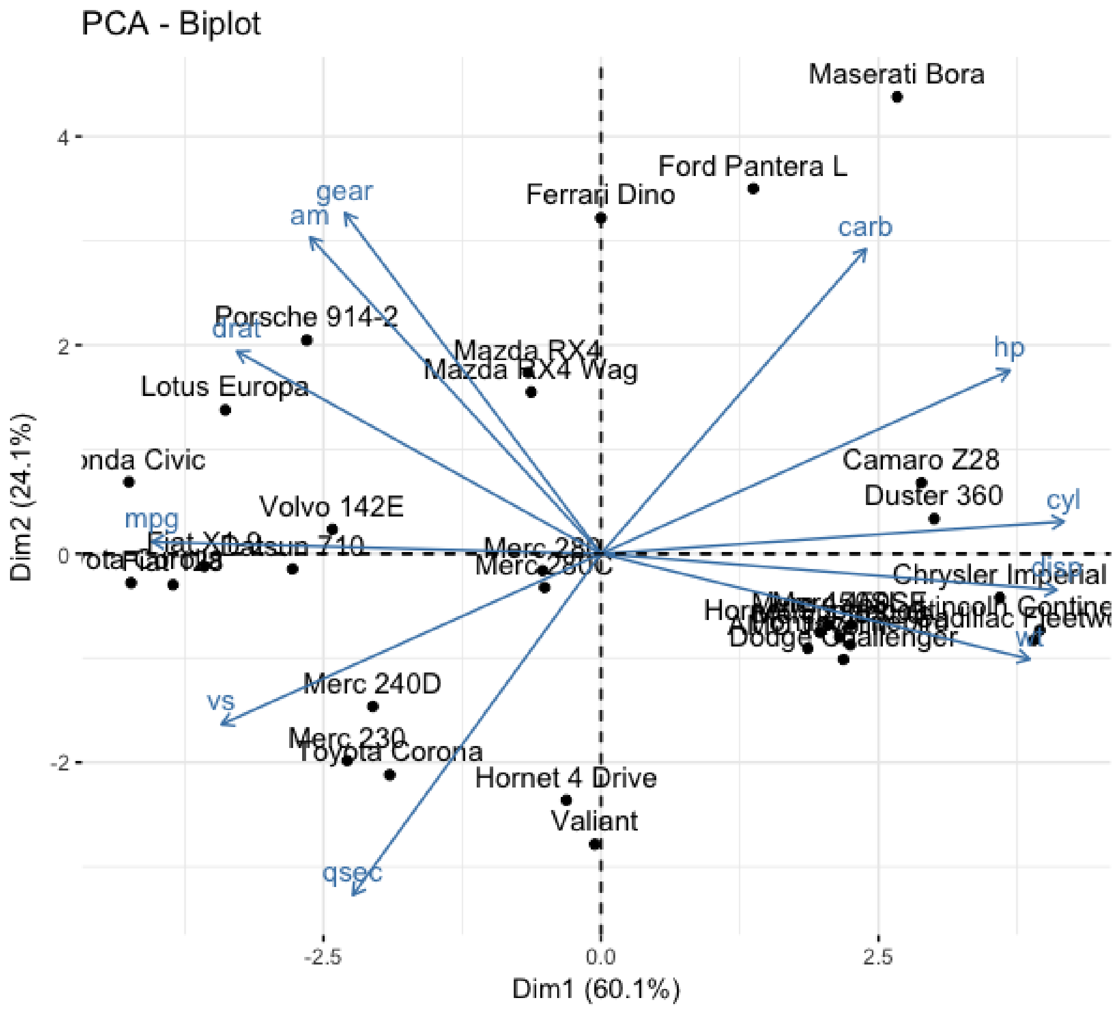
Barplotting cos2 for individuals

```
fviz_cos2(mtcars_pca,
    choice = "ind",
    axes = 1,
    top = 10)
```



Biplots

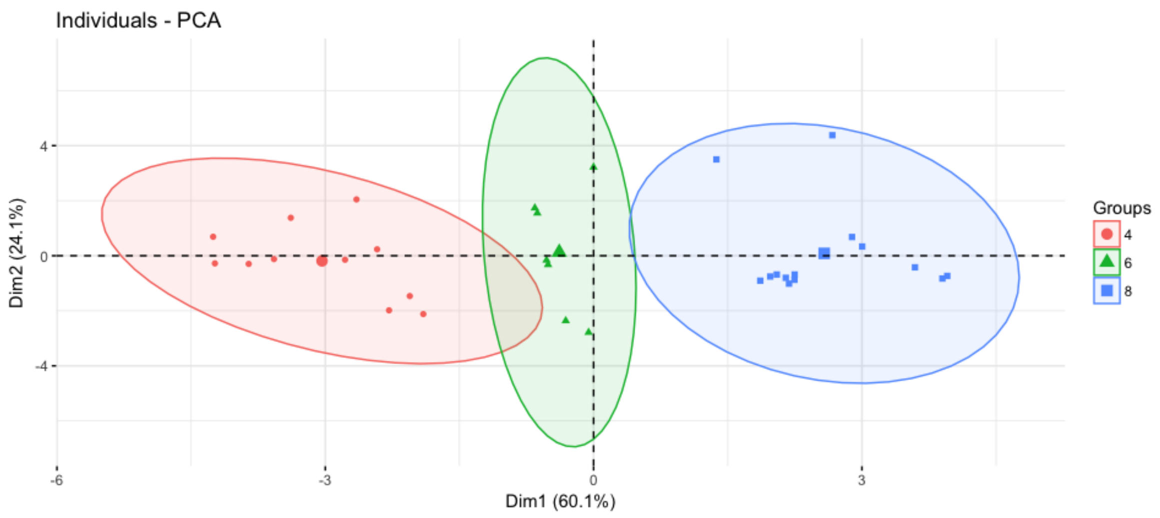
```
fviz_pca_biplot(mtcars_pca)
```



Adding ellipsoids

```
mtcars$cyl <- as.factor(mtcars$cyl)
```

```
fviz_pca_ind(mtcars_pca,
    label="var",
    habillage=mtcars$cyl,
    addEllipses=TRUE)
```



DIMENSIONALITY REDUCTION IN R

Let's practice!