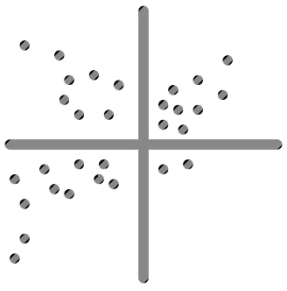
Montogue



Tutorial ST6
Principal Component Analysis with R

Lucas Monteiro Nogueira

• Summary •	
Problem 1	**Preparing data**
Problem 2	**Basic PCA and visualization**
Problem 3	**More visualization**

PROBLEMS

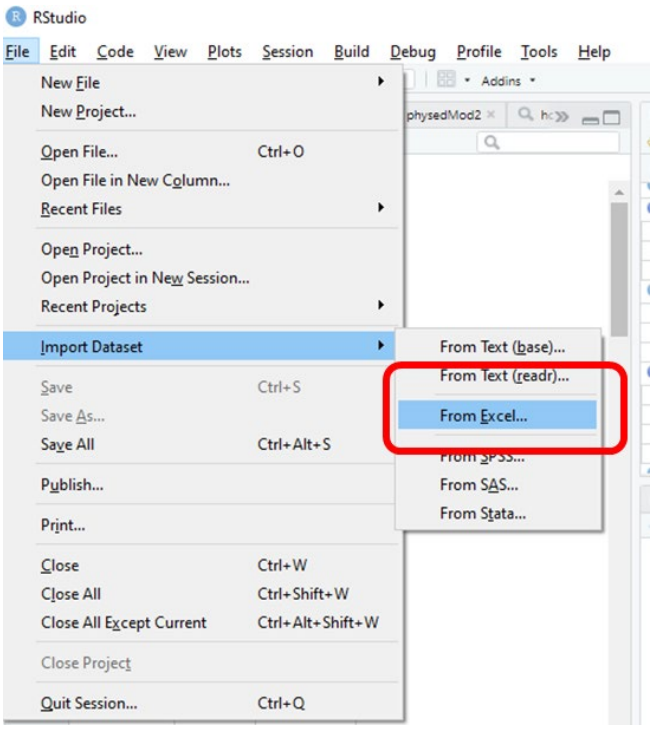
▶ Packages used

stats – *R's main statistical package*	**factoextra** – *Multivariate data analysis*
tibble – *Simple data frames*	**FactoMineR** – *Multivariate data analysis*
corrplot – *Visualization of a correlation matrix*	

▶ Problem 1 – Preparing data

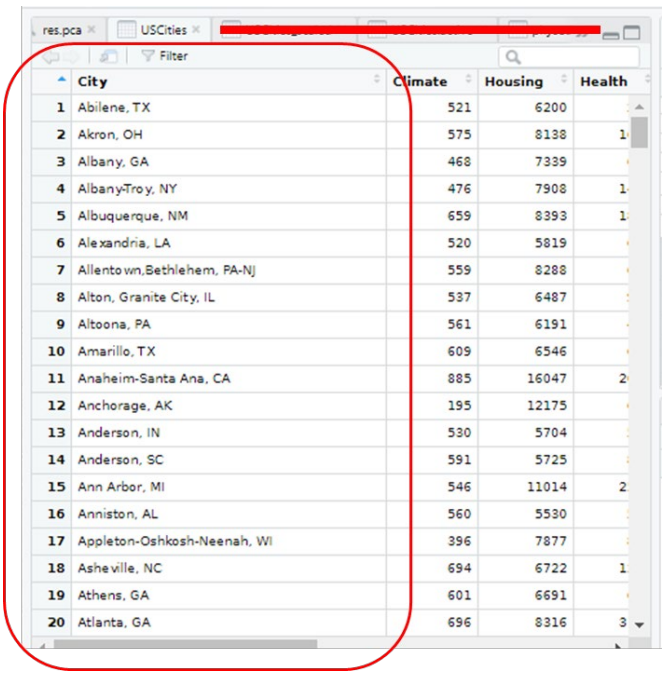
We will be working with a simple dataset loosely based on a tutorial for the MATLAB® Statistics and Machine Learning toolbox. The dataset, *USCities*, describes 329 American cities according to their scores in 9 attributes, namely *Climate*, *Housing*, *Health*, *Crime*, *Transportation*, *Education*, *Arts*, *Recreation*, and *Economics*. Needless to say, the greater the score, the better the city performs in that particular attribute. Download the dataset in our Google Drive folder.

To begin loading the dataset in RStudio, head to *File* → *Import Dataset* → *From Excel*, and browse to the folder in which you've saved *USCities.xlsx*.



Upon loading the data, you can access it in the Environment pane. At this point, note that one of the columns in *USCities* contains the city names, but we need to have these be assigned as row names instead. This can be easily done with the *column_to_rownames* command in the *tibble* package:

```
> USCities.active <- column_to_rownames(USCities, var =
"City")
```



	City	Climate	Housing	Health
1	Abilene, TX	521	6200	
2	Akron, OH	575	8138	1
3	Albany, GA	468	7339	
4	Albany-Troy, NY	476	7908	1
5	Albuquerque, NM	659	8393	1
6	Alexandria, LA	520	5819	
7	Allentown,Bethlehem, PA-NJ	559	8288	
8	Alton, Granite City, IL	537	6487	
9	Altoona, PA	561	6191	
10	Amarillo, TX	609	6546	
11	Anaheim-Santa Ana, CA	885	16047	2
12	Anchorage, AK	195	12175	
13	Anderson, IN	530	5704	
14	Anderson, SC	591	5725	
15	Ann Arbor, MI	546	11014	2
16	Anniston, AL	560	5530	
17	Appleton-Oshkosh-Neenah, WI	396	7877	
18	Asheville, NC	694	6722	1
19	Athens, GA	601	6691	
20	Atlanta, GA	696	8316	3

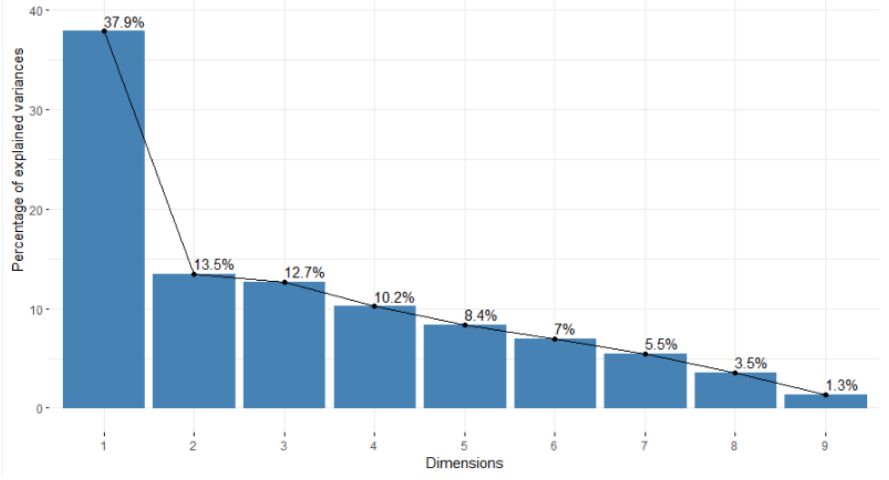
▶ Problem 2 – Basic PCA and visualization

An important first step would be to scale the data, especially when variables are measured in different units/scales. This can be done via the *scale* command; however, we needn't use it because the command *PCA* (*Facto*MineR package), which we will use in the sequel, automatically scales the data for us.

```
> pcaModel <- PCA(USCities.active, graph = FALSE)
```

As an initial visualization step, we may plot a scree plot of the dimensions using the *fviz_screeplot* or *fviz_eig* commands (*factoextra* package):

```
> fviz_eig(pcaModel, addlabels = TRUE, ylim = c(0,50))
```



As can be seen, the first dimension alone explains nearly 38% of the variance associated with the dataset. The second and third dimensions are quite close to one another, explaining 13.5 and 12.7% of variance respectively. Retaining dimensions 1 to 4 only, we'd end up with a component set that explains 74.3% of the variances, which is a reasonably high value.

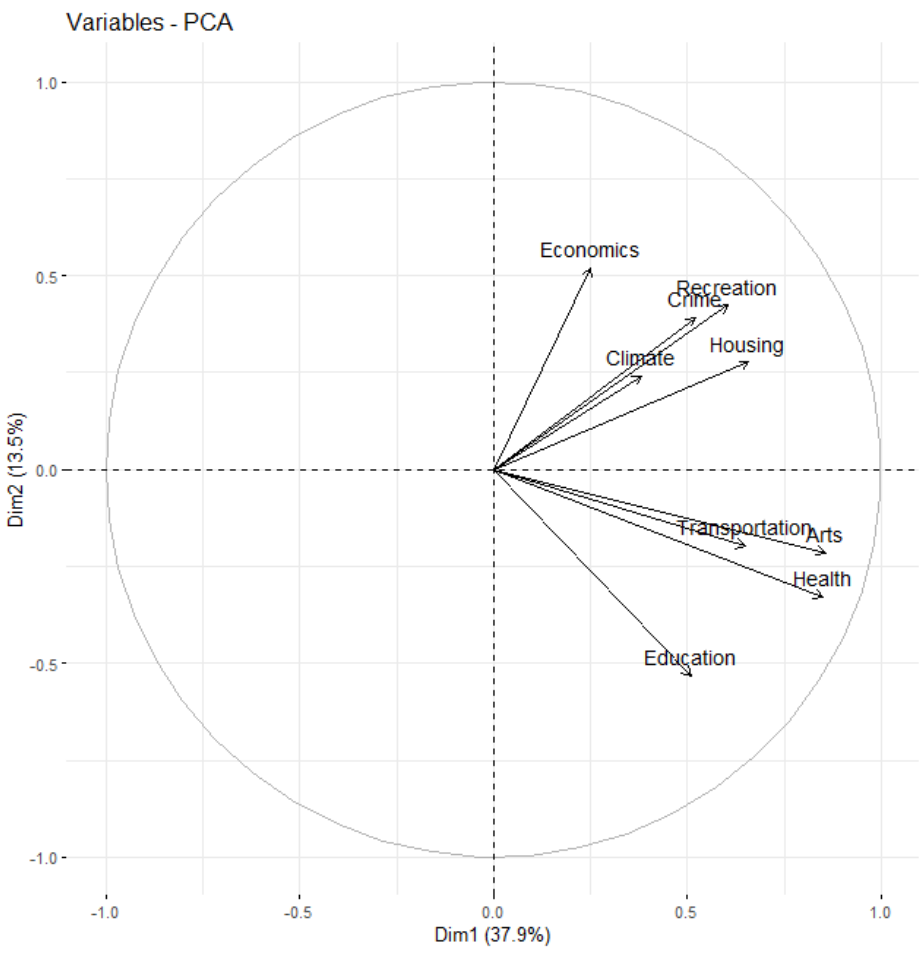
Now, we can extract variable results from a PCA object using the *get_pca_var* command (*factoextra* package):

```
> var <- get_pca_var(pcaModel)
> head(var$coord)
```

```
             Dim.1      Dim.2       Dim.3       Dim.4     Dim.5
Climate   0.3810724  0.2400122 -0.737149161  0.13177960 0.3203926
Housing   0.6581945  0.2761390 -0.222411268  0.49117367 -0.2026487
Health    0.8496280 -0.3299526 -0.007825952  0.01410853 0.0896045
Crime     0.5193207  0.3915182  0.197766213 -0.51729689 0.4547377
Transp.   0.6482791 -0.1978893  0.156388461 -0.29067992 -0.3509421
Education 0.5082331 -0.5325932  0.245414266  0.32187540 0.1812382
```

To plot the correlation circle, we may use *fviz_pca_var* command (*factoextra* package):

```
> fviz_pca_var(pcaModel,col.var="black")
```



As can be seen, this is a biplot that describes how each variable contributes to the two principal components of *pcaModel*. For instance, the first principal dimension (component), which is represented in the horizontal axis, has positive components for all 9 variables, which explains why all nine vectors appear on the right-hand side of the plot. The largest (i.e., most rightward) vectors occur for *Arts* and *Health*, which together explain most of the variance in this particular component.

In turn, the second principal component has positive coefficients for 5 variables (including *Economics* and *Housing*) and negative ones for the other 4 (including *Education* and *Health*). This indicates that the second component distinguishes among cities that have high values for the former set of variables and low for the second, and vice versa.

The quality of representation of the variables on the factor map can be accessed with *cos2*.

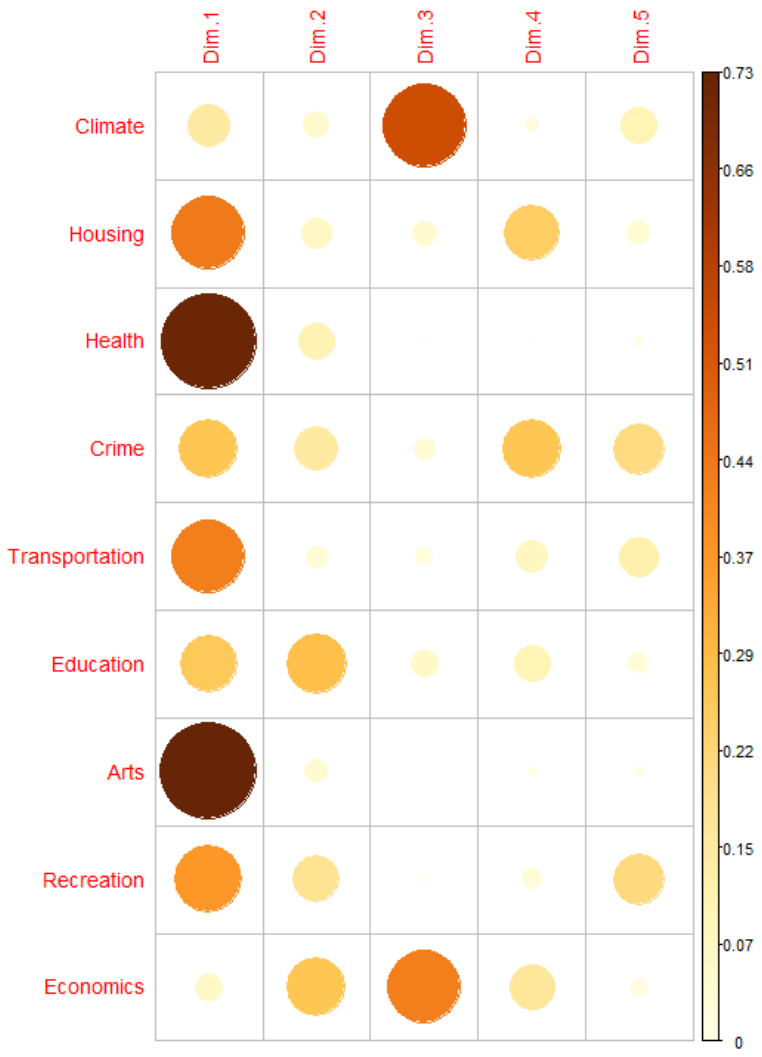
```
> head(var$cos2)
```

```
                Dim.1      Dim.2         Dim.3        Dim.4
Climate        0.1452161 0.05760587 0.54338888512 0.0173658621
Housing        0.4332200 0.07625274 0.04946677205 0.2412515769
Health         0.7218678 0.10886872 0.00006124553 0.0001990507
Crime          0.2696940 0.15328650 0.03911147486 0.2675960760
Transportation 0.4202658 0.03916016 0.02445735063 0.0844948133
Education      0.2583009 0.28365554 0.06022816176 0.1036037726

                Dim.5
Climate        0.102651388
Housing        0.041066488
Health         0.008028966
Crime          0.206786338
Transportation 0.123160337
Education      0.032847288
```

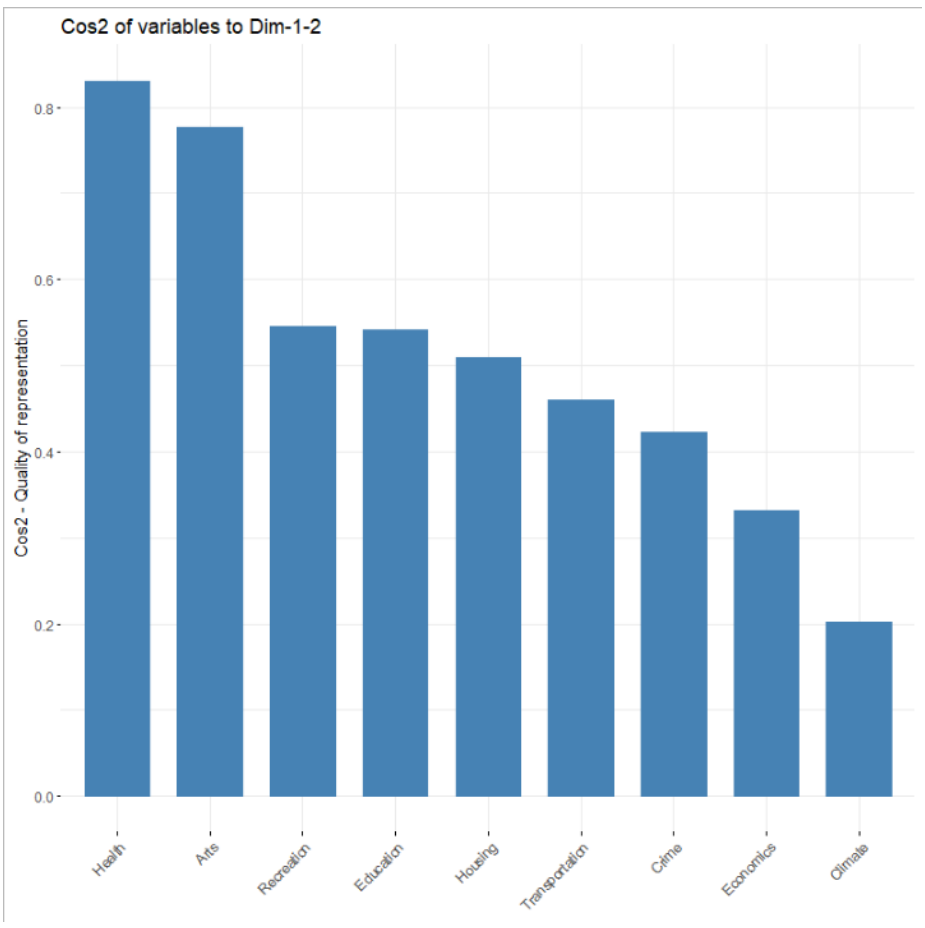
The *cos2* of all variables can be visualized with *corrplot* in the eponymous R package:

```
> corrplot(var$cos2, is.corr=FALSE)
```



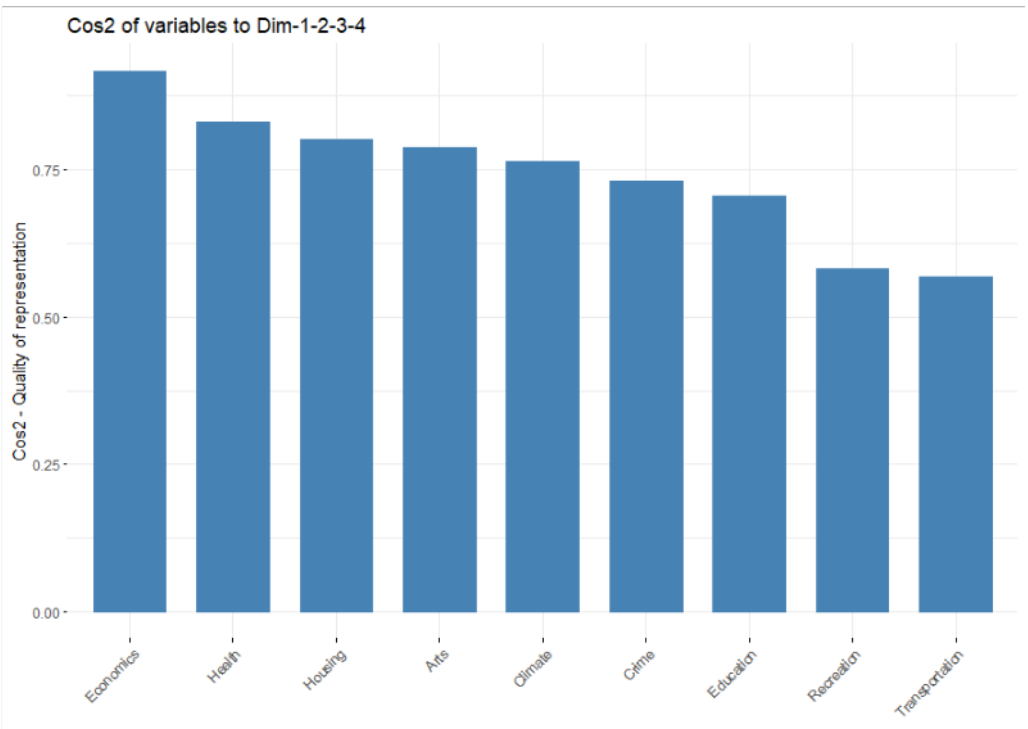
We may also use the *fviz_cos2* command (*factoextra* package) to obtain bar plots of the variables. The argument *axes = 1:2* is used to account for contributions to dimensions 1 and 2 at the same time.

```
> fviz_cos2(pcaModel, choice = "var", axes=1:2)
```



As can be seen, *Health* and *Arts* account for most of the quality of representation in dimensions 1 and 2 of the PCA model. On the other hand, *Economics* and *Climate* are not well-represented by this pair of dimensions. The quality of representation can be improved if we account for dimensions 1 to 4, as mentioned above:

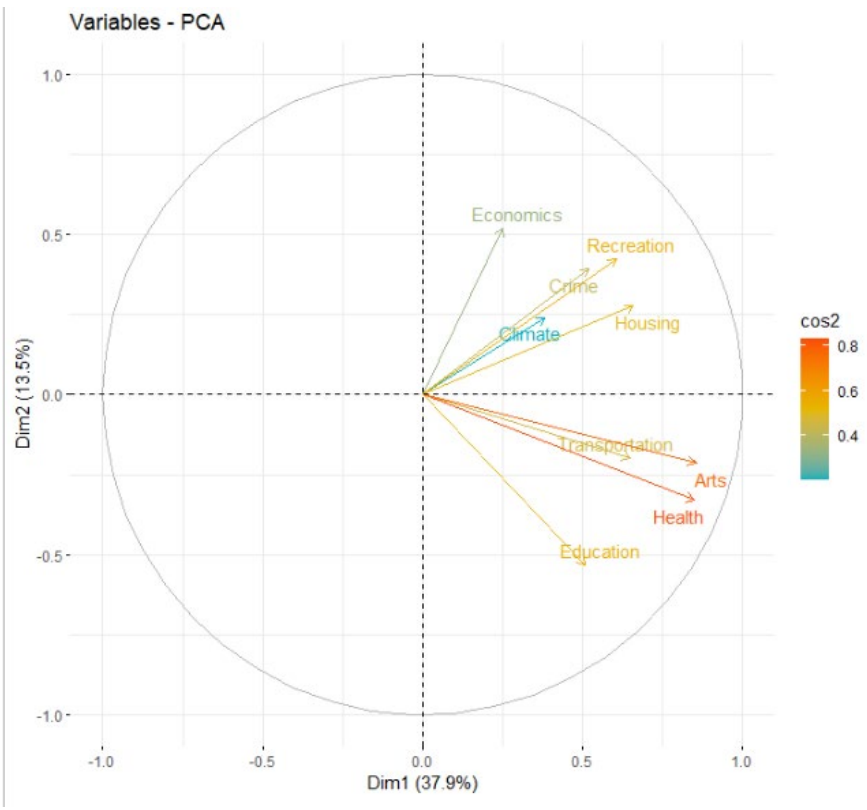
```
> fviz_cos2(pcaModel, choice = "var", axes=1:4)
```



Clearly, a 4-component set provides a decent quality of representation for most variables.

We may color variables in the circle plot according to quality of representation; all we have to do is complement the *fviz_pca_var* command with the arguments *col.var* = "*cos2*" and a list of colors:

```
> fviz_pca_var(pcaModel, col.var="cos2", gradient.cols =
c("#00AFBB","#E7B800","#FC4E07"), repel=TRUE)
```



Note that we've added an argument *repel* to prevent text overlapping. Clearly, the factors associated with the best qualities of representation (i.e., *Arts* and *Health*) have reddish vectors, whereas the worst ones (namely, *Climate* and *Economics*) have greenish vectors.

▶ Problem 3 – More visualization

The contribution of variables to each dimension can be extracted with the *contrib* option:

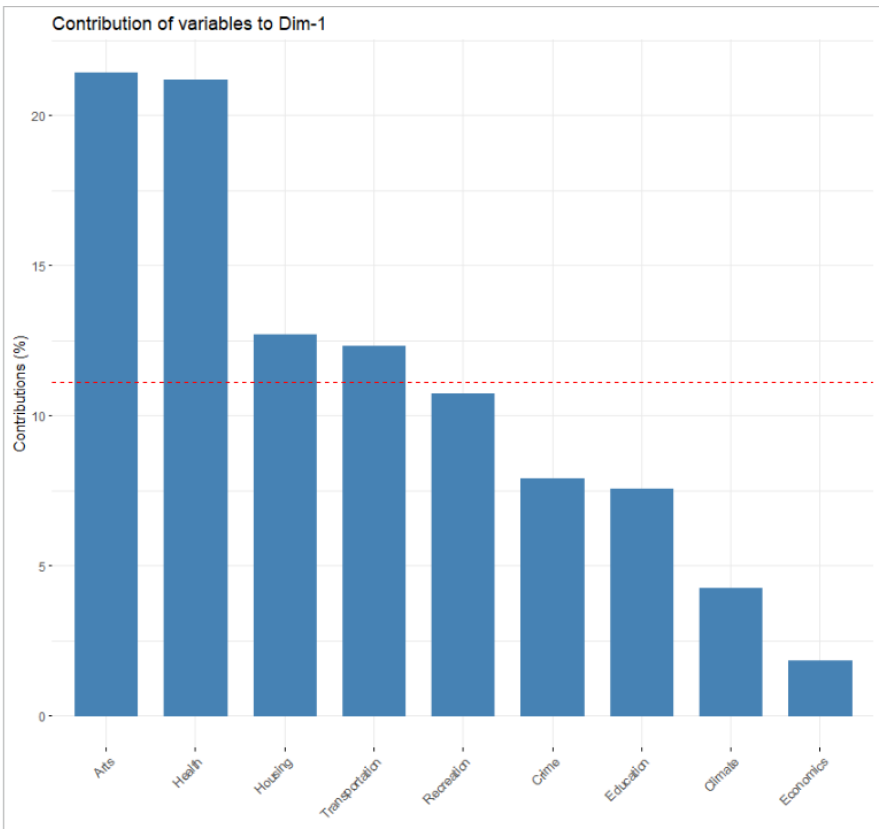
```
> head(var$contrib,9)
```

	Dim.1	Dim.2	Dim.3	Dim.4	Dim.5
Climate	4.260672	4.745222	47.603925700	1.88571246	13.627167
Housing	12.710766	6.281239	4.333567737	26.19686260	5.451654
Health	21.179752	8.967946	0.005365453	0.02161439	1.065860
Crime	7.912878	12.626813	3.426385402	29.05754120	27.451280
Transp.	12.330687	3.225777	2.142601613	9.17506547	16.349769

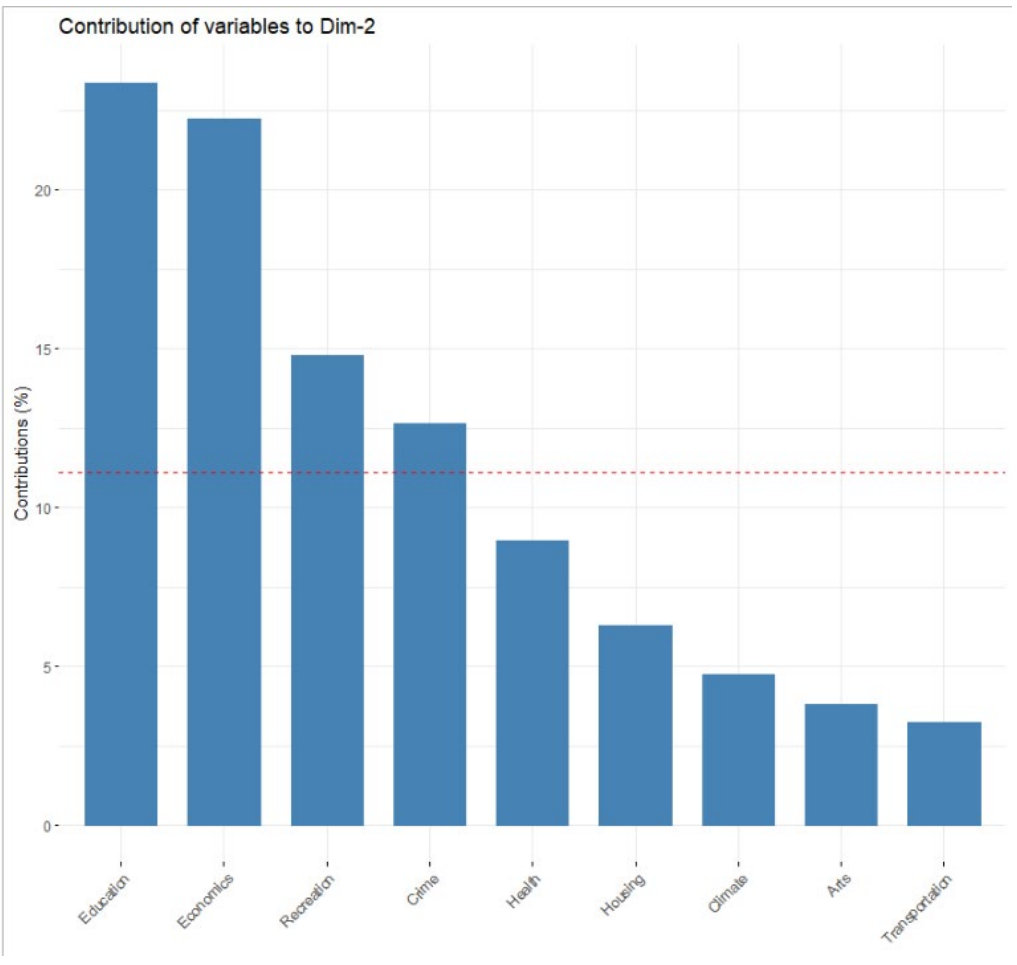
```
Education    7.578604 23.365825  5.276326064 11.25005619  4.360540
Arts        21.441946  3.794311  0.070141804  1.02172454  1.104551
Recreation 10.751048 14.782075  0.258599099  3.60243497 28.041322
Economics    1.833648 22.210793 36.883087127 17.78898819  2.547856
```

The contributions are expressed as percentages; clearly, *Arts* and *Health* account for most of the first principal component, while *Education* and *Economics* dominate the second. Another way to proceed is to produce bar plots for each dimension using the *fviz_contrib* command (*factoextra* package):

```
> fviz_contrib(pcaModel, choice="var", axes=1, top=9)
```



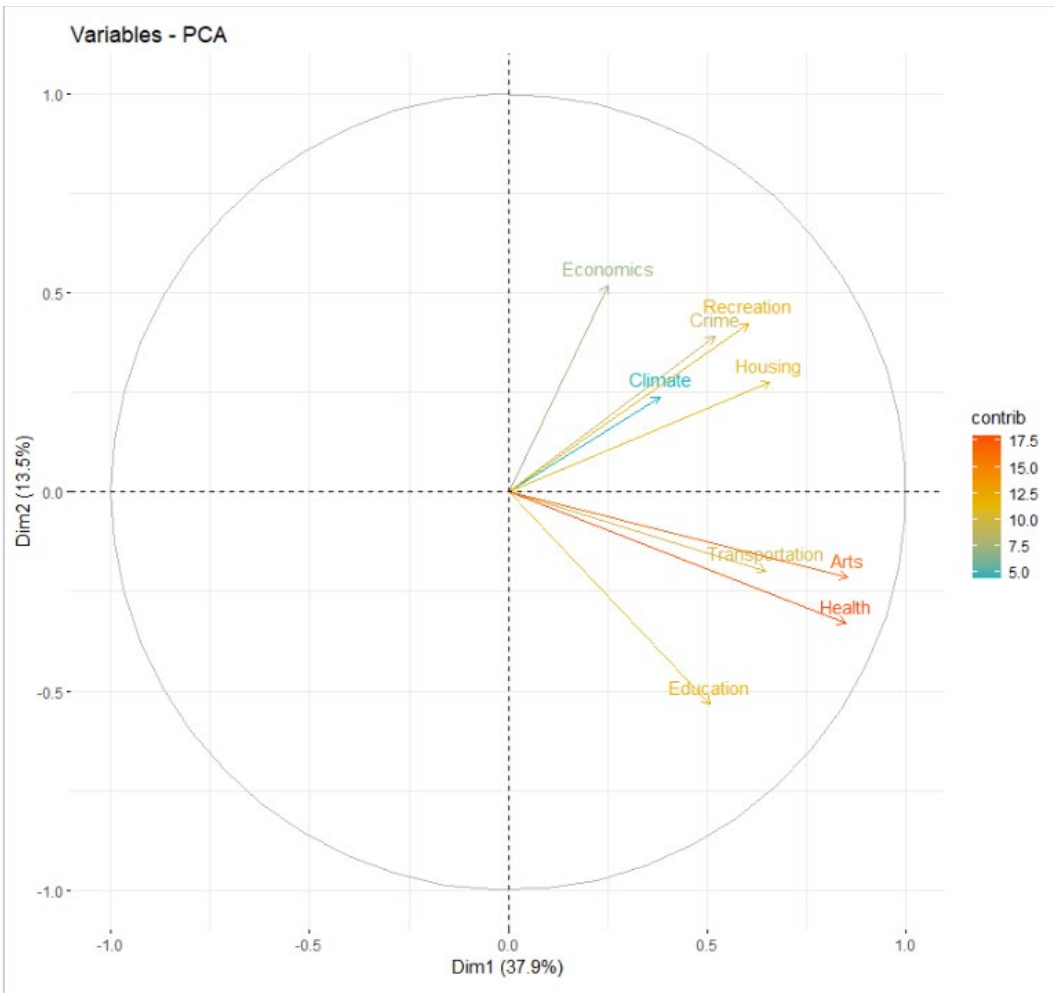

```
> fviz_contrib(pcaModel, choice="var", axes=2, top=9)
```



The bar plots only reinforce the findings conveyed by the *var$contrib* table: *Arts* and *Health* are particularly expressive in dimension 1, while *Education* and *Economics* seem to be particularly important in dimension 2. Note that both plots also include a red dashed line, which demarcates the average contribution expected for any variable.

Importantly, we can also highlight vectors in the correlation circle plot according to variables' contribution levels; all we have to do is add the *col.var = "contrib"* option to the usual command:

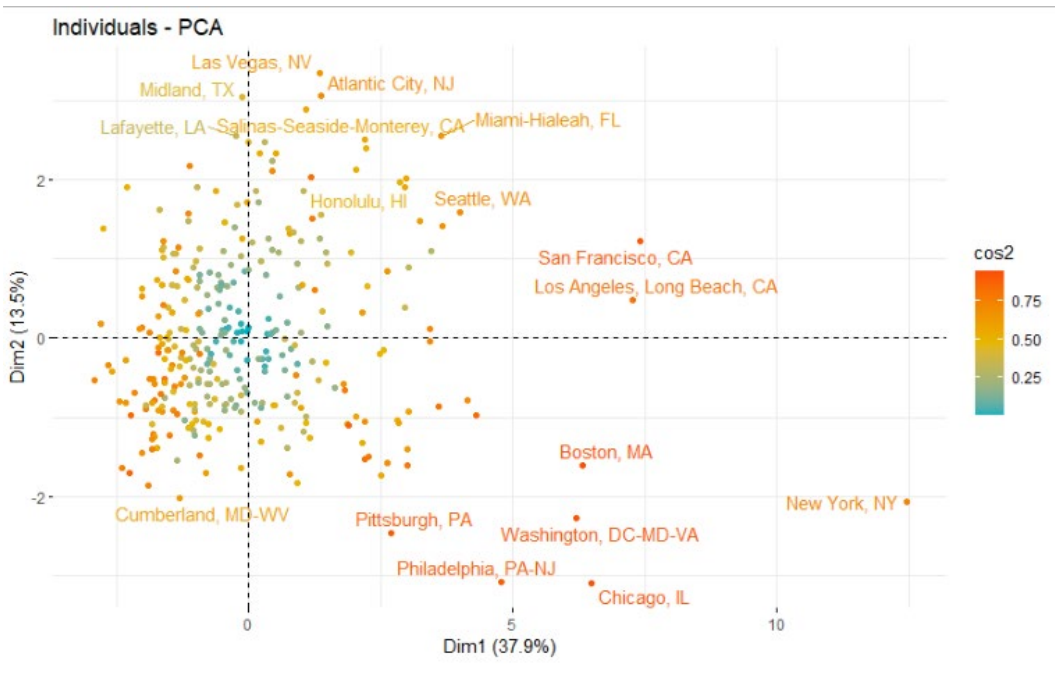
```
> fviz_pca_var(pcaModel, col.var="contrib",
gradient.cols=c("#00AFBB","#E7B800","#FC4E07"))
```



Most of the techniques used heretofore can be used in individual-based visualization. For example, we can plot *cos2* contributions for the first two dimensions using *fviz_pca_ind* (*factoextra* package):

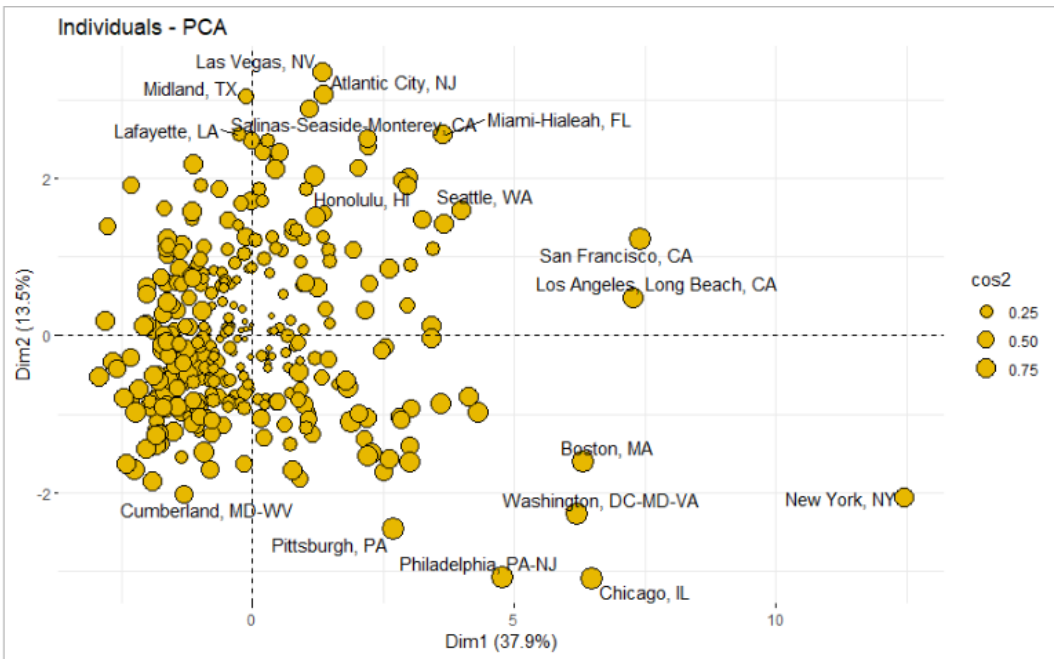
```
> fviz_pca_ind(pcaModel,col.ind="cos2",gradient.cols =
c("#00AFBB","#E7B800","#FC4E07"),repel=TRUE)
```

Note that use of *repel* is particularly important here because we have over 300 city names, and some names are quite lengthy. The ensuing plot is shown below.



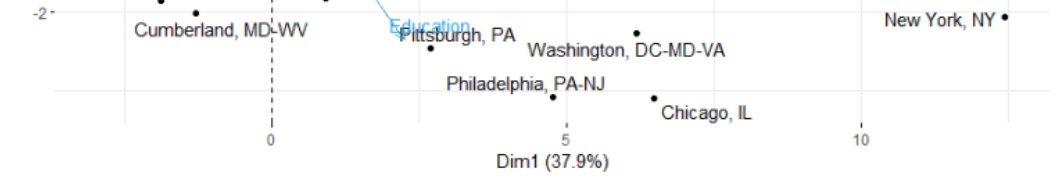
As can be seen, one particularly important outlier is New York City. Individuals with similar contribution profiles are grouped close together, as in the case of Los Angeles/San Francisco and Philadelphia/Pittsburgh. We can also scale points' sizes according to their *cos2* scores, as follows:

```
> fviz_pca_ind(pcaModel, pointsize="cos2", pointshape=21, fill
= "#E7B800", repel=TRUE)
```



To make a simple biplot of individuals and variables, we may use the *fuiz_pca_biplot* command:

```
> fviz_pca_biplot(pcaModel, repel=TRUE, col.var = "#2E9FDF",
ind.var = "#696969")
```



REFERENCES

- KASSAMBARA, A. (2017). *A Practical Guide to Cluster Analysis in R*. STHDA.