University Data Analysis

Introduction:

This document presents an analysis of university data, focusing on student distribution across different specialties, the popularity of specialties among public and private universities.

Methodology:

We have used several statistical modeling methods such as correlations and PCA analysis.

Data Preparation:

The dataset UniversityData.csv is loaded and preprocessed to separate specialties into individual columns, we have created a wider version for further pca use that dummifies the “Domaine” column.

Installing Packages

```r
# Load necessary libraries
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```r
library(tidyr)
library(ggplot2)
library(corrplot)
```

```
## corrplot 0.92 loaded
```

Data Importing

Reading the csv file

```
my_data <- read.csv("C:\\Users\\shily\\data mining\\UniversityData.csv")
```

Data Transformation

```
my_data_long <- my_data %>%
  separate_rows(Domaine, sep = ",\\s*")

head(my_data_long)
```

```
## # A tibble: 6 x 6
##   Nom          Adresse                   Statut Téléphone Domaine NombreEtudiants
##   <chr>        <chr>                     <chr>      <int> <chr>             <int>
## 1 University 1 100 Rue, Ville 0, Tunis~ Publi~    1.23e9 Chimie             4968
## 2 University 1 100 Rue, Ville 0, Tunis~ Publi~    1.23e9 Physiq~            4968
## 3 University 1 100 Rue, Ville 0, Tunis~ Publi~    1.23e9 Cyber ~            4968
## 4 University 2 101 Rue, Ville 1, Tunis~ Publi~    1.23e9 Réseaux            9150
## 5 University 2 101 Rue, Ville 1, Tunis~ Publi~    1.23e9 Cyber ~            9150
## 6 University 3 102 Rue, Ville 2, Tunis~ Publi~    1.23e9 TIC                2040
```

```
# Create dummy variables for each specialty
my_data_wide <- my_data_long %>%
  mutate(Indicator = 1) %>%
  pivot_wider(names_from = Domaine , values_from = Indicator, values_fill = list(Indicator = 0))

head(my_data_wide)
```

```
## # A tibble: 6 x 13
##   Nom          Adresse          Statut Téléphone NombreEtudiants Chimie Physique
##   <chr>        <chr>            <chr>      <int>           <int>  <dbl>    <dbl>
## 1 University 1 100 Rue, Ville ~ Publi~    1.23e9            4968      1        1
## 2 University 2 101 Rue, Ville ~ Publi~    1.23e9            9150      0        0
## 3 University 3 102 Rue, Ville ~ Publi~    1.23e9            2040      0        0
## 4 University 4 103 Rue, Ville ~ Publi~    1.23e9            7250      0        0
## 5 University 5 104 Rue, Ville ~ Publi~    1.23e9            9356      0        0
## 6 University 6 105 Rue, Ville ~ Privée    1.23e9            9798      0        0
## # i 6 more variables: `Cyber Security` <dbl>, Réseaux <dbl>, TIC <dbl>,
## #   `Data Science` <dbl>, `Génie Logiciels` <dbl>, Business <dbl>
```

Descriptive Statistics

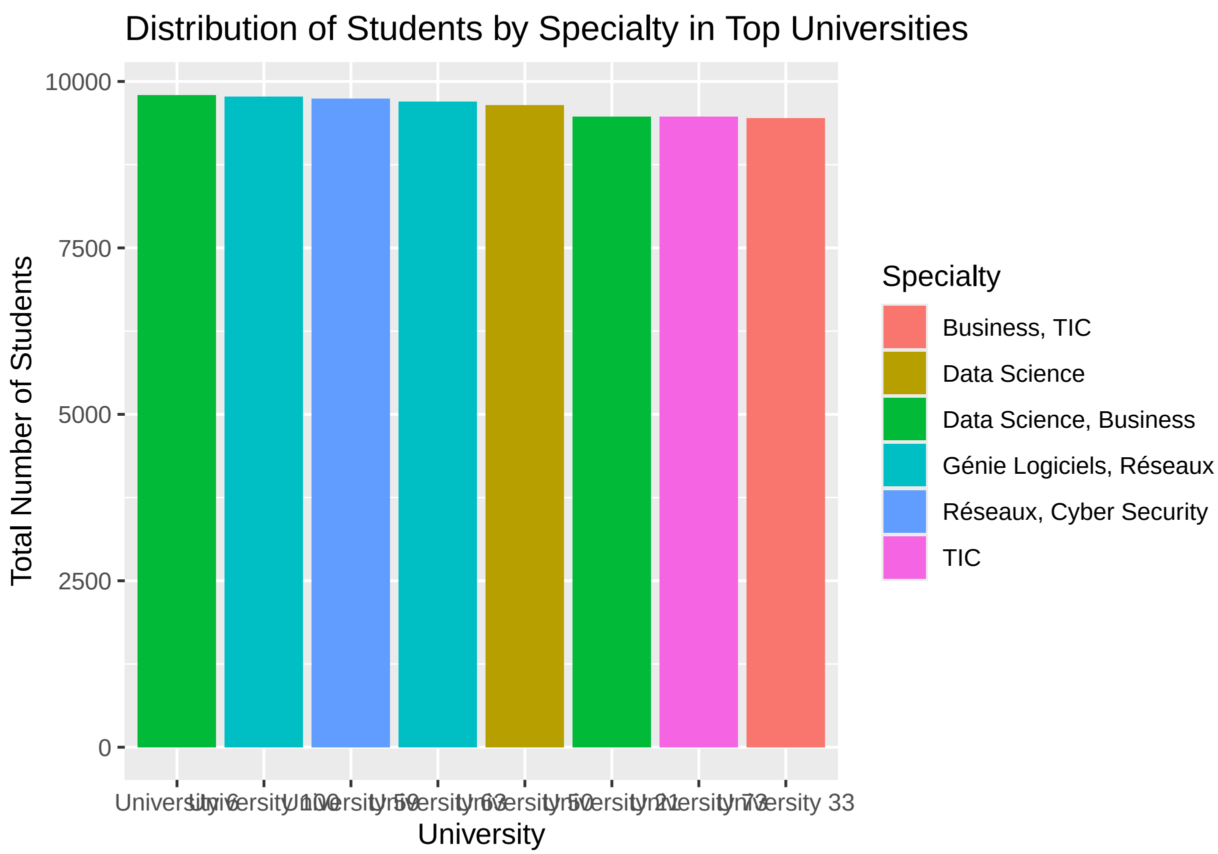
We analyze the distribution of students across the top universities and their specialties. **Top Universities**

```
#university stats
university_popularity <- my_data %>%
  group_by(Nom, Domaine) %>%
```

```r
  summarise(TotalStudents = sum(NombreEtudiants), .groups = 'drop')

top_universities <- university_popularity %>%
  group_by(Nom) %>%
  summarise(TotalStudents = sum(TotalStudents), .groups = 'drop') %>%
  top_n(8, TotalStudents)

top_universities_with_specialties <- top_universities %>%
  inner_join(university_popularity, by = "Nom")

print(top_universities_with_specialties)
```

```
## # A tibble: 8 x 4
##   Nom            TotalStudents.x Domaine                   TotalStudents.y
##   <chr>                    <int> <chr>                               <int>
## 1 University 100            9769 Génie Logiciels, Réseaux             9769
## 2 University 21             9473 Data Science, Business               9473
## 3 University 33             9448 Business, TIC                        9448
## 4 University 50             9641 Data Science                         9641
## 5 University 59             9742 Réseaux, Cyber Security              9742
## 6 University 6              9798 Data Science, Business               9798
## 7 University 63             9696 Génie Logiciels, Réseaux             9696
## 8 University 73             9472 TIC                                  9472
```

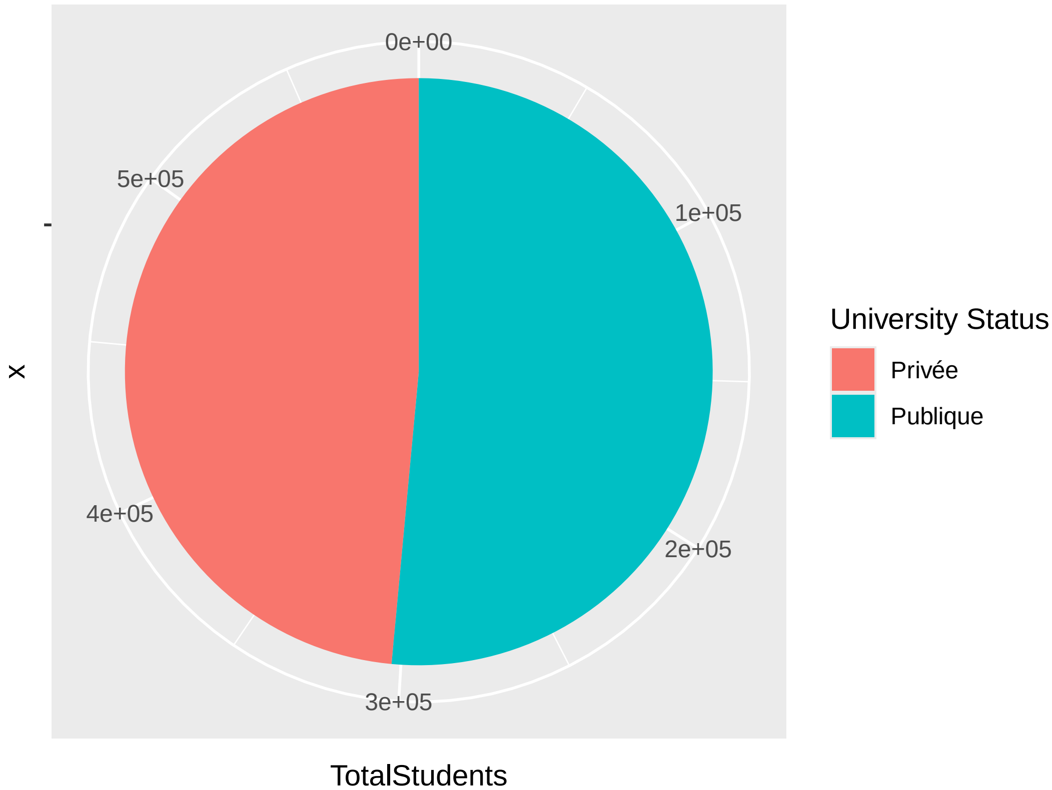
```r
ggplot(top_universities_with_specialties, aes(x = reorder(Nom, -TotalStudents.y), y = TotalStudents.y, f
  geom_bar(stat = "identity") +
  labs(title = "Distribution of Students by Specialty in Top Universities",
       x = "University",
       y = "Total Number of Students",
       fill = "Specialty")
```



Universities By Status

```
#statut university stat
status_summary <- my_data %>%
  group_by(Statut) %>%
  summarise(TotalStudents = sum(NombreEtudiants), .groups = 'drop')

# Create pie chart
ggplot(status_summary, aes(x = "", y = TotalStudents, fill = Statut)) +
  geom_bar(stat = "identity", width = 1) +
  coord_polar(theta = "y") +
  labs(title = "Distribution of Students in Public vs Private Universities", fill = "University Status")
```

Distribution of Students in Public vs Private Universities



Speciality Statistics

```r
#speciality stats

specialty_counts <- my_data_wide[,6:13] %>%
  summarise(across(everything(), sum)) %>%
  pivot_longer(cols = everything(), names_to = "Specialty", values_to = "NumberOfUniversities")

head(specialty_counts)
```

```
## # A tibble: 6 x 2
##   Specialty      NumberOfUniversities
##   <chr>                         <dbl>
## 1 Chimie                           13
## 2 Physique                         18
## 3 Cyber Security                   16
## 4 Réseaux                          27
## 5 TIC                              29
## 6 Data Science                     24
```

```r
my_data_public <- my_data_wide %>%
  filter(Statut=="Publique")

my_data_priv <- my_data_wide %>%
  filter(Statut=="Privée")
```

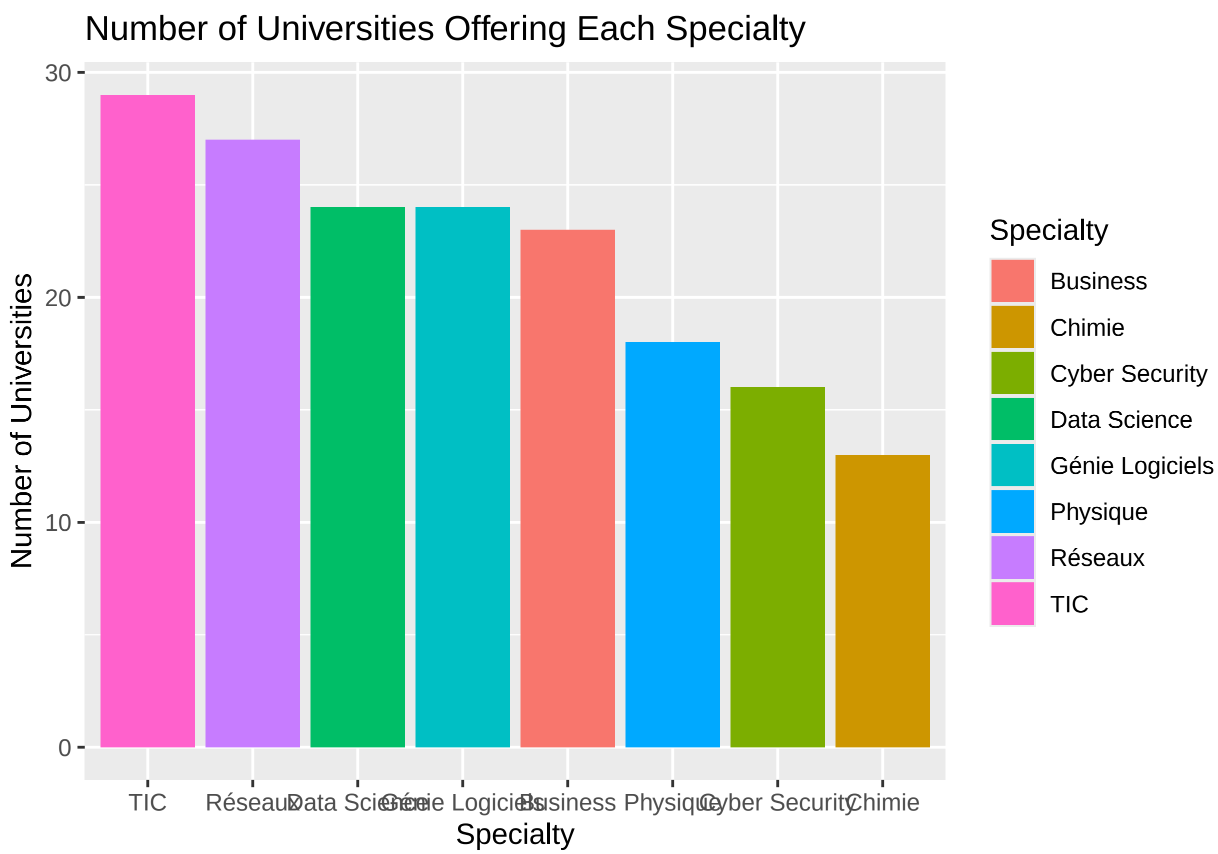
```r
specialty_counts_public <-
  my_data_public[,6:13] %>%
  summarise(across(everything(), sum)) %>%
  pivot_longer(cols = everything(), names_to = "Specialty", values_to = "NumberOfUniversities")

head(specialty_counts_public)
```

```
## # A tibble: 6 x 2
##   Specialty      NumberOfUniversities
##   <chr>                         <dbl>
## 1 Chimie                            5
## 2 Physique                          6
## 3 Cyber Security                   10
## 4 Réseaux                          16
## 5 TIC                              12
## 6 Data Science                     14
```

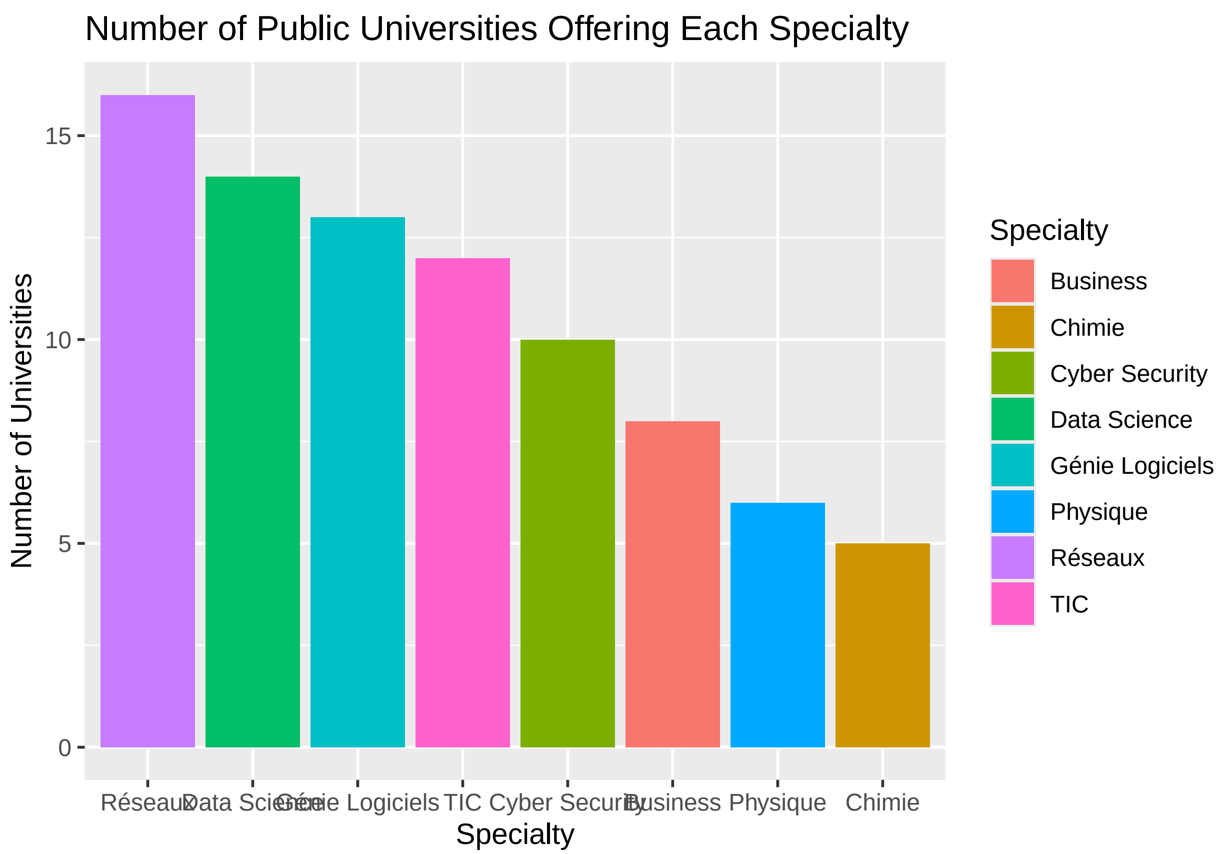
```r
specialty_counts_private <-
  my_data_priv[,6:13] %>%
  summarise_each(funs(sum), everything()) %>%
  pivot_longer(cols = everything(), names_to = "Specialty", values_to = "NumberOfUniversities")

head(specialty_counts_private)
```

```
## # A tibble: 6 x 2
##   Specialty      NumberOfUniversities
##   <chr>                         <dbl>
## 1 Chimie                            8
## 2 Physique                         12
## 3 Cyber Security                    6
## 4 Réseaux                          11
## 5 TIC                              17
## 6 Data Science                     10
```

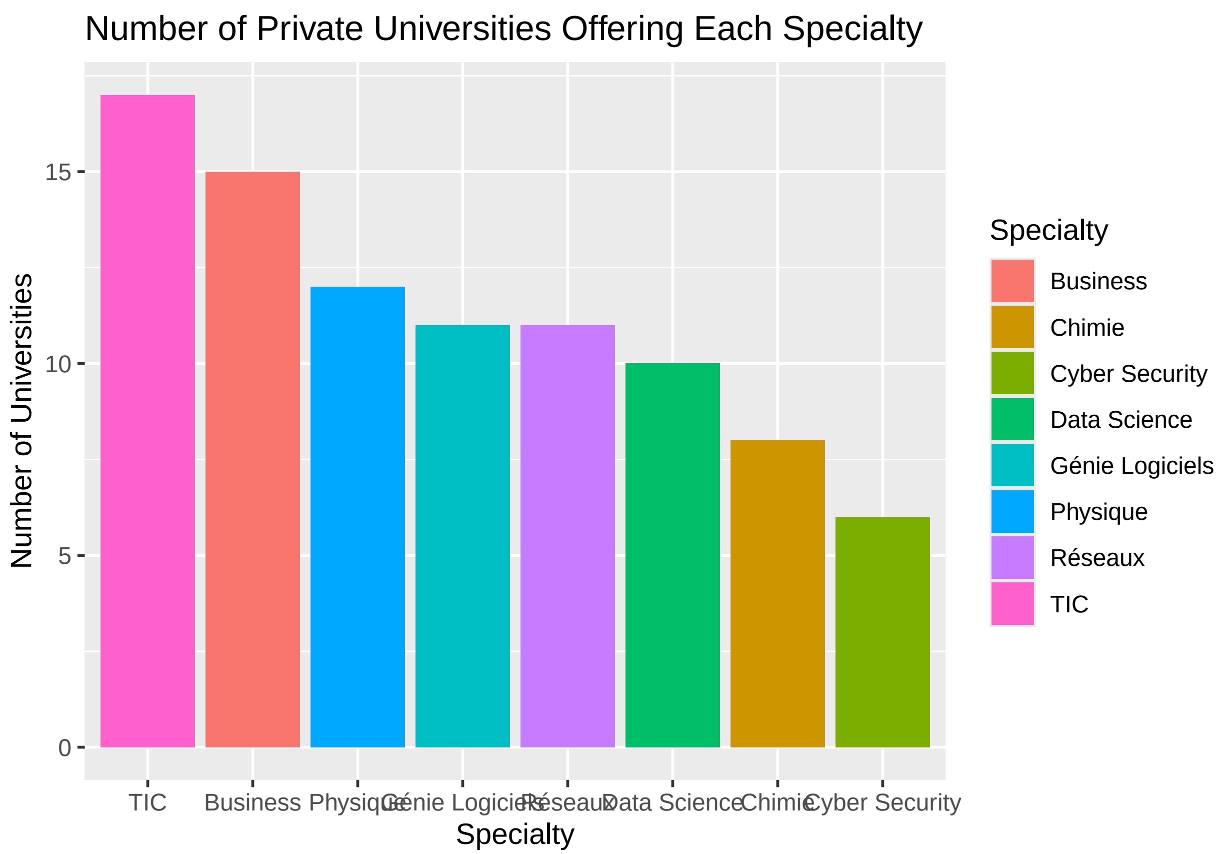
```r
# Create a bar plot
ggplot(specialty_counts, aes(x = reorder(Specialty, -NumberOfUniversities), y = NumberOfUniversities, fi
  geom_bar(stat = "identity") +
  labs(title = "Number of Universities Offering Each Specialty", x = "Specialty", y = "Number of Univers
```




```r
ggplot(specialty_counts_public, aes(x = reorder(Specialty, -NumberOfUniversities), y = NumberOfUniversi
  geom_bar(stat = "identity") +
  labs(title = "Number of Public Universities Offering Each Specialty", x = "Specialty", y = "Number of
```




```
ggplot(specialty_counts_private, aes(x = reorder(Specialty, -NumberOfUniversities), y = NumberOfUnivers
  geom_bar(stat = "identity") +
  labs(title = "Number of Private Universities Offering Each Specialty", x = "Specialty", y = "Number o
```



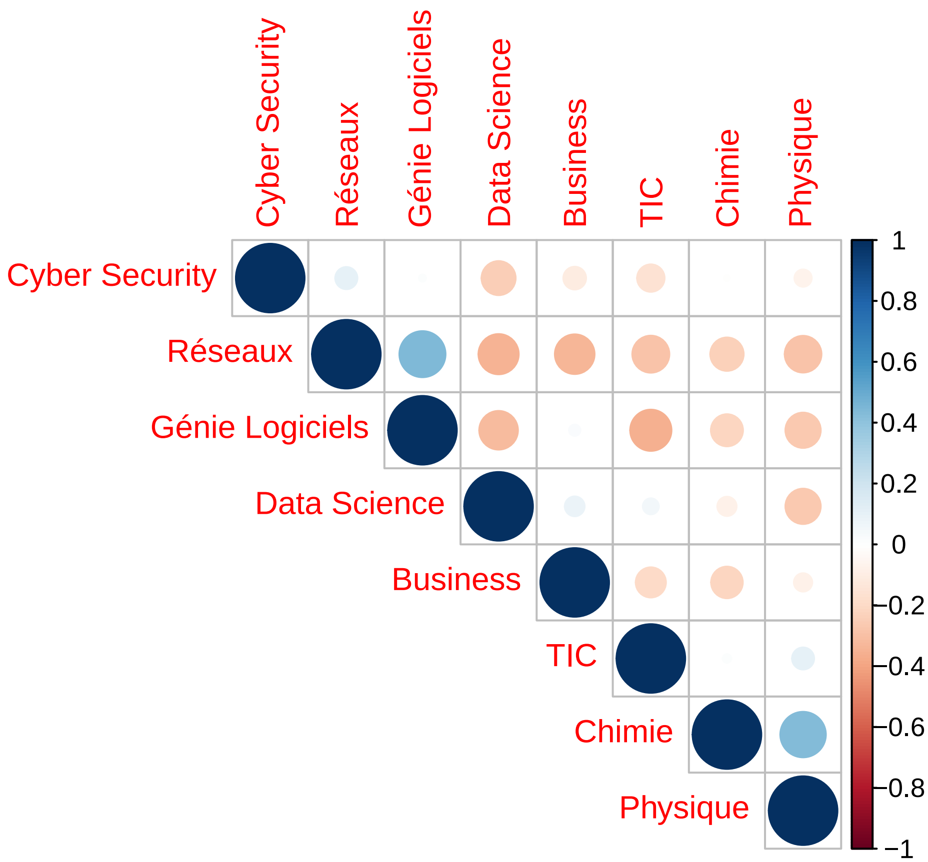
Modeling

The Principal Component Analysis is used to describe a dataset and to cluster variables as well as individuals based on common criteria. The objective of this PCA is to identify groupings of variables and individuals that provide better insights into the specialties of Tunisian universities. To perform this PCA, we began by extracting the portion of the database on which the PCA would be conducted.

Correlation Analysis:

We will calculate and visualize the correlation matrix to examine relationships between different specialities.

```
#correlation
cor_matrix <- cor(my_data_wide[,6:13])
library(RColorBrewer)
corrplot(cor_matrix,type="upper",order="hclust")
```



PCA Analaysis:

Eigenvalues measure the amount of variance explained by each principal axis. The eigenvalues are large for the first axes and small for the subsequent axes. In other words, the first axes correspond to the directions carrying the maximum amount of variation contained in the dataset. We start with the criterion of the cumulative inertia rate and the Kaiser criterion:

```
pca_data <- my_data_wide[,6:13]
library(FactoMineR)
res.pca1 <- PCA (pca_data,graph=FALSE)
library(factoextra)
```

```
## Welcome! Want to learn more? See two factoextra-related books at https://goo.gl/ve3WBa
```

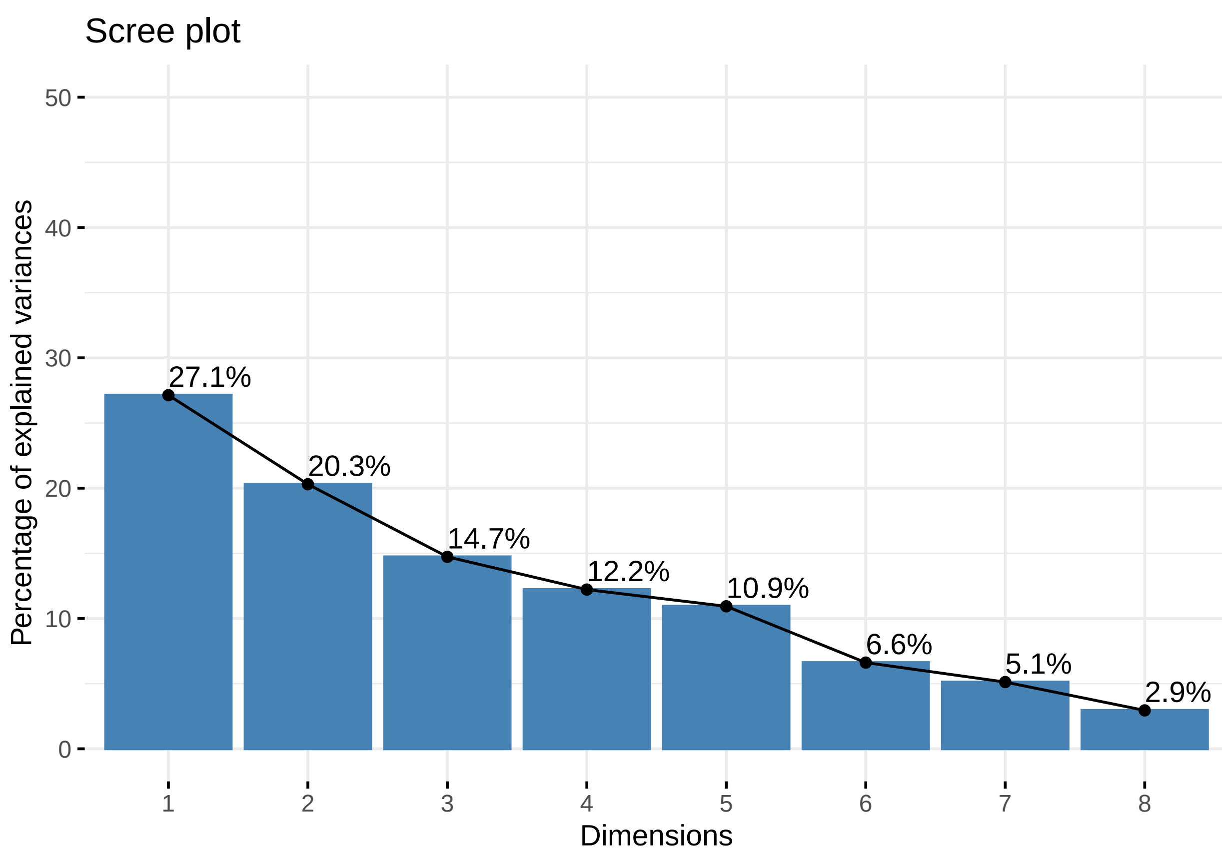
```
eig.val1 <- get_eigenvalue(res.pca1)
eig.val1
```

```
##       eigenvalue variance.percent cumulative.variance.percent
## Dim.1  2.1708725        27.135906                    27.13591
## Dim.2  1.6241190        20.301488                    47.43739
## Dim.3  1.1783007        14.728759                    62.16615
## Dim.4  0.9773616        12.217020                    74.38317
## Dim.5  0.8748045        10.935057                    85.31823
## Dim.6  0.5291794         6.614743                    91.93297
```

```
## Dim.7  0.4096649          5.120811                      97.05378
## Dim.8  0.2356974          2.946217                     100.00000
```

According to the previous table, we can notice that the first 3 dimensions can explain 62% of the data variance. And they have Eigenvalues that are superior than 1. According to the Kaiser Criterion, we can consider these 3 axes as our Principal Components.

```
fviz_eig(res.pca1, addlabels = TRUE, ylim = c(0, 50))
```



According to the Elbow Criterion and from the Scree plot of the Eigenvalues, we can notice a knee bend starting from the 3rd axis. Therefore, we can retain 3 principal components.

Numbers of Axis chosen According to both the Elbow and Kaiser Criterions, the optimal number of principal components to retain is 3.

Now we will try to explain the 3 Principal Components so we can classify our data based on these 3 Clusters.

Variables Analysis

After choosing the number of axes to retain, we begin the study of the variables and individuals in order to produce and interpret the maps of variables and individuals. To interpret the axes, we start by extracting the variables, in the first instance.

```
var1 <- get_pca_var(res.pca1)
var1
```

```
## Principal Component Analysis Results for variables
##  ===================================================
##   Name       Description
## 1 "$coord"   "Coordinates for the variables"
## 2 "$cor"     "Correlations between variables and dimensions"
## 3 "$cos2"    "Cos2 for the variables"
## 4 "$contrib" "contributions of the variables"
```

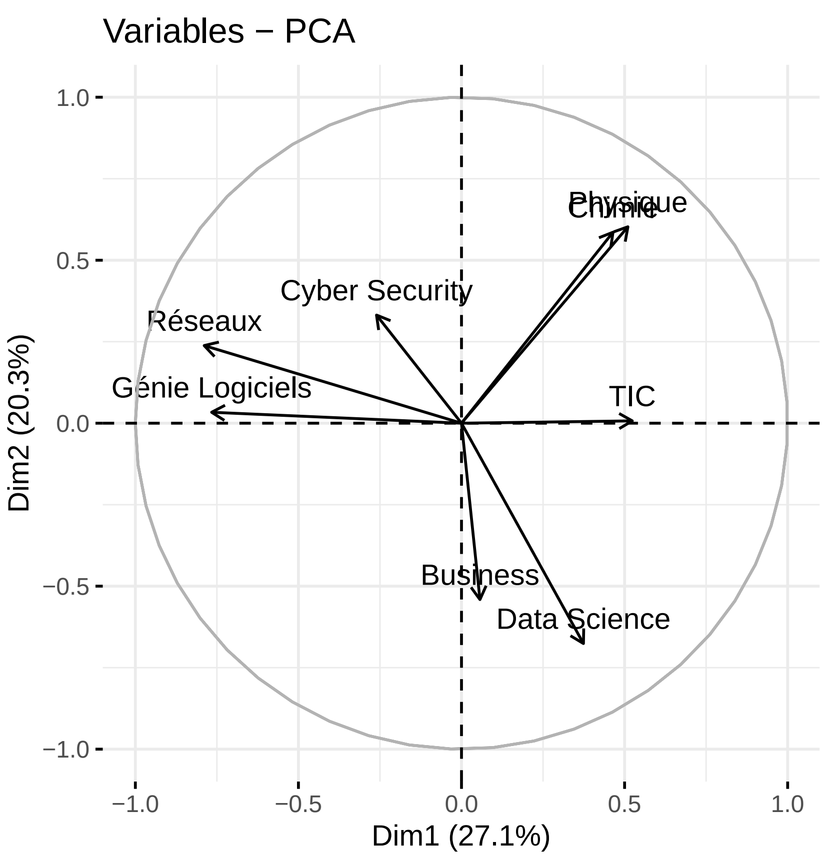
```r
var1$coord
```

```
##                        Dim.1        Dim.2       Dim.3       Dim.4       Dim.5
## Chimie            0.46454739  0.585924327  0.17328972 -0.09046237  0.45642505
## Physique          0.51021351  0.602612398  0.33045409 -0.20061694 -0.12307817
## Cyber Security   -0.26059489  0.331742890  0.06657303  0.88548456 -0.05990112
## Réseaux          -0.78905475  0.238190579 -0.30503501 -0.15561887  0.09091852
## TIC               0.52368533  0.007140046 -0.57058388 -0.06478247 -0.55007180
## Data Science      0.37385080 -0.675957143 -0.21493679  0.08859363  0.50677153
## Génie Logiciels -0.76617595  0.033532734  0.17592555 -0.32884710 -0.02796320
## Business          0.05654183 -0.541099253  0.73408415  0.02109171 -0.28160228
```

The correlation between a variable and a principal component is used as the coordinates of the variable on the principal component. The representation of variables is done through these correlations.

For a clearer interpretation, here is the correlation graph of the variables:

```r
fviz_pca_var(res.pca1, col.var = "black",axes = 1:2)
```



The correlation plot can be interepreted as the following: 1.Positively correlated variables are grouped together. 2.Negatively correlated variables are positioned on opposite sides of the origin of the graph (opposite quadrants). 3.The distance between the variables and the origin measures the quality of representation of the variables. Variables that are far from the origin are well represented by the PCA.

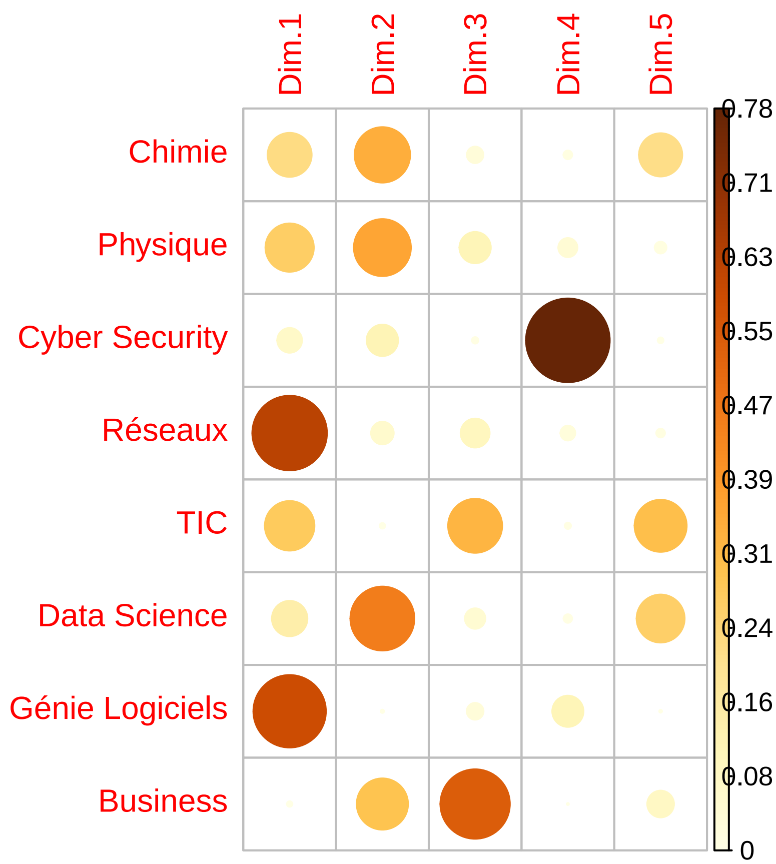
We can see from the previous plot that Specialities like Physics and Chemistry are positively correlated, also "Cyber Security" , "Réseaux" and "Genie Logiciels" which is explained by the big similarity amongst them as Physics and Chemistry are basically the same fundamental science , whether CyberSec , Networks , and GL are all IT Subjects.

Variable contribution to Principal Components

```
head(var1$cos2)
```

```
##                    Dim.1        Dim.2       Dim.3       Dim.4       Dim.5
## Chimie         0.21580428 3.433073e-01 0.030029325 0.008183441 0.208323825
## Physique       0.26031782 3.631417e-01 0.109199907 0.040247158 0.015148235
## Cyber Security 0.06790969 1.100533e-01 0.004431969 0.784082914 0.003588145
## Réseaux        0.62260741 5.673475e-02 0.093046357 0.024217234 0.008266176
## TIC            0.27424632 5.098026e-05 0.325565966 0.004196769 0.302578983
## Data Science   0.13976442 4.569181e-01 0.046197823 0.007848831 0.256817381
```

```
corrplot(var1$cos2, is.corr=FALSE)
```

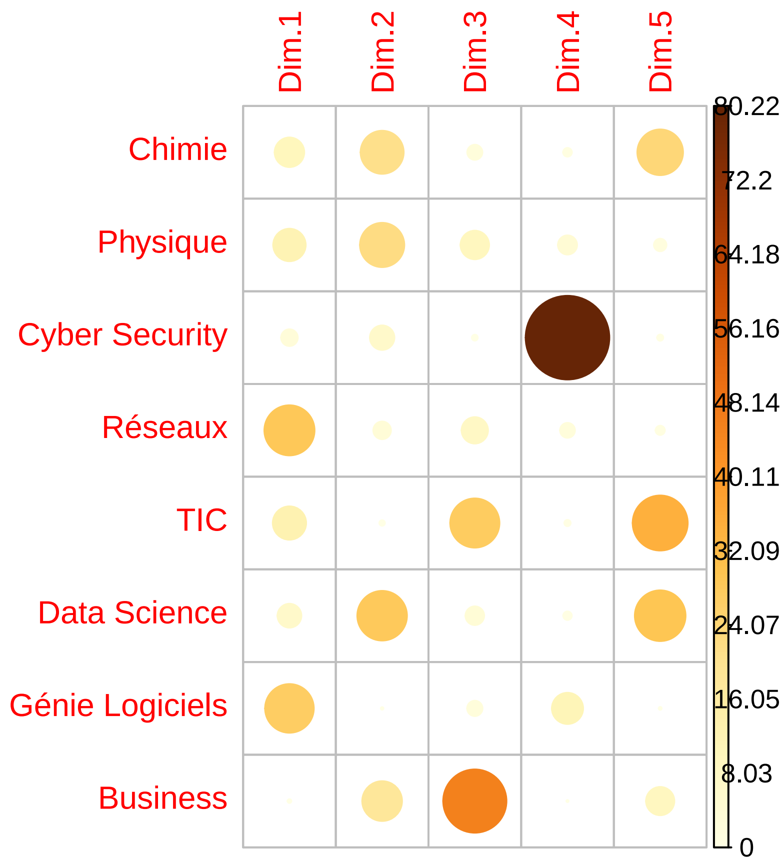



```
var1$contri
```

```
##                    Dim.1        Dim.2      Dim.3       Dim.4       Dim.5
```

```
## Chimie             9.940901 21.138064226  2.5485282  0.83729917 23.81375701
## Physique          11.991392 22.359303815  9.2675756  4.11793928  1.73161369
## Cyber Security     3.128221  6.776187258  0.3761322 80.22444293  0.41016529
## Réseaux           28.680054  3.493263241  7.8966565  2.47781716  0.94491697
## TIC               12.633000  0.003138949 27.6301262  0.42939773 34.58818207
## Data Science       6.438168 28.133286963  3.9207160  0.80306315 29.35711612
## Génie Logiciels 27.040998  0.069234103  2.6266468 11.06452414  0.08938462
## Business           0.147267 18.027521444 45.7336184  0.04551644  9.06486423
```

```
corrplot(var1$contrib, is.corr=FALSE)
```

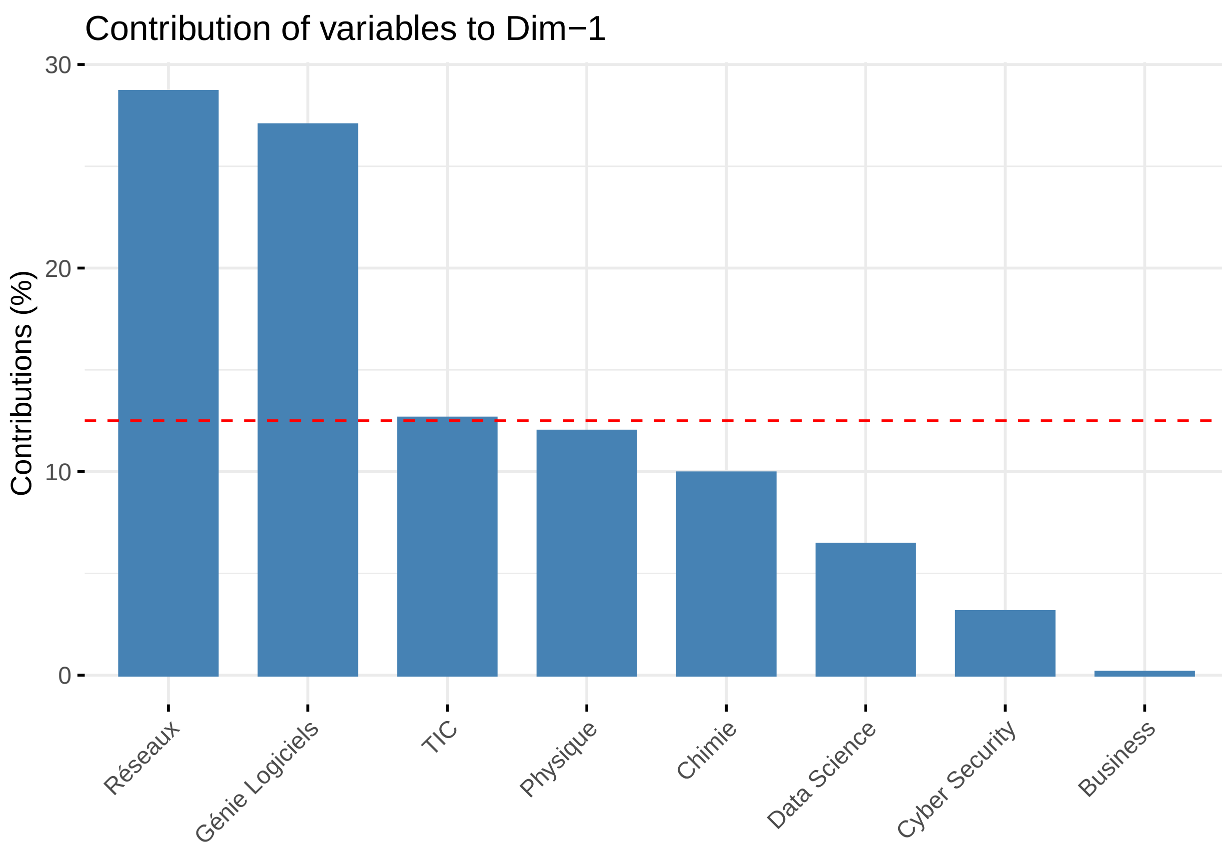


A high cos2 value indicates a good representation of the variable on the principal axes being considered. In this case, the variable is positioned close to the circumference of the correlation circle. A low cos2 value suggests that the variable is not perfectly represented by the principal axes. In this case, the variable is close to the center of the circle.

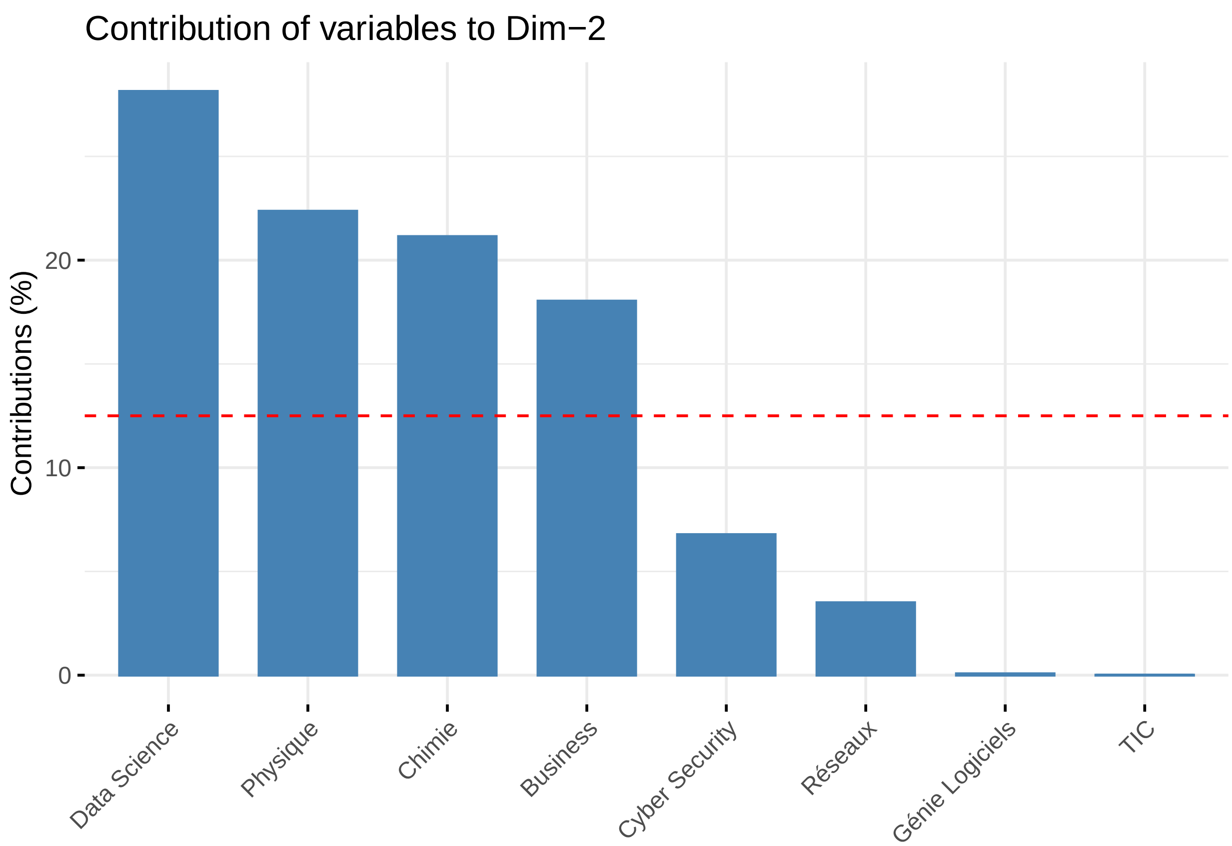
As we can see from the cos2 correlation matrix, Subjects like Réseau and Génie Logiciels have a high cos2 value on the first component which means that this axis is from IT specialities. Subjects like Chemistry, Physics, Buisness and Data Science have a high cos2 value for Dim2, which can be interpreted as Fundamental sciences.

As for the third Dimension, Buisness has a high cos2 which indicates that this is for Buisness only specialities, meaning universities that are Buisness schools.

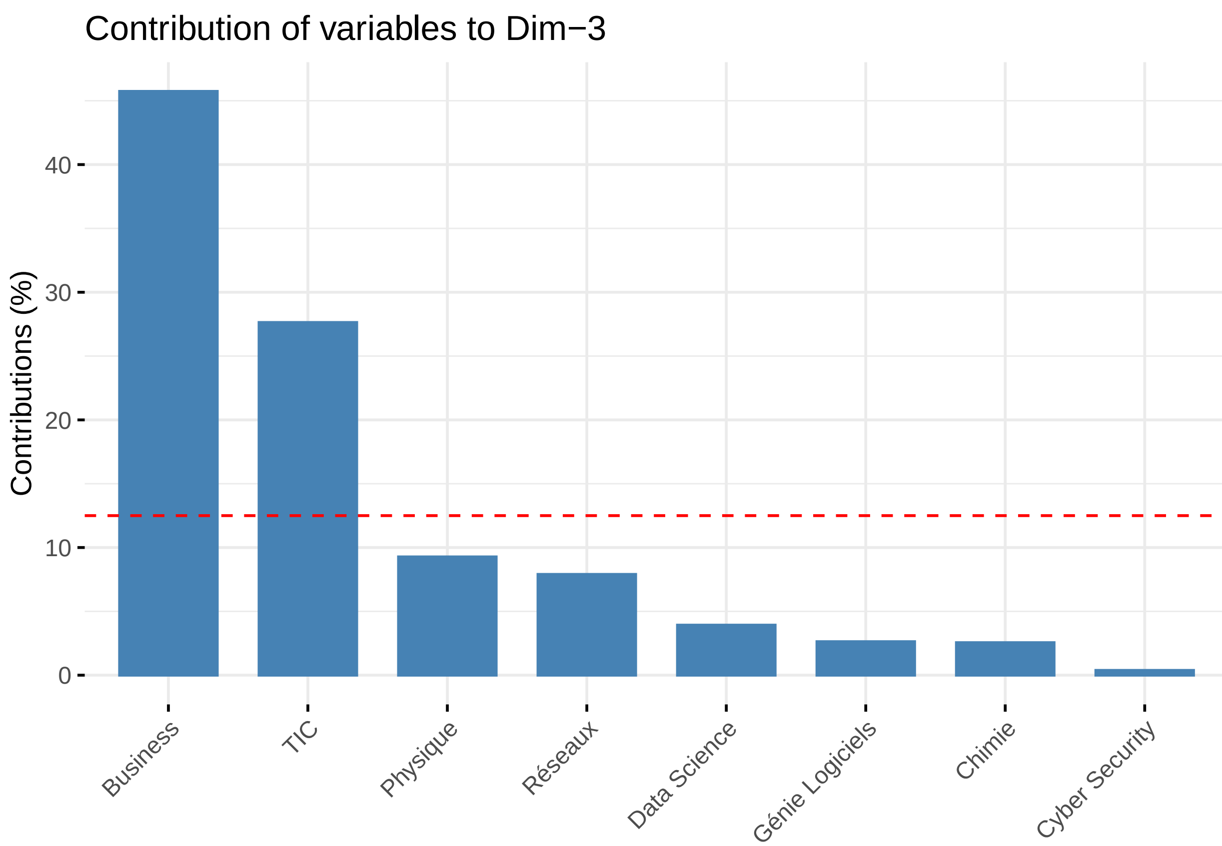
```
fviz_contrib(res.pca1, choice = "var", axes = 1, top = 10)
```



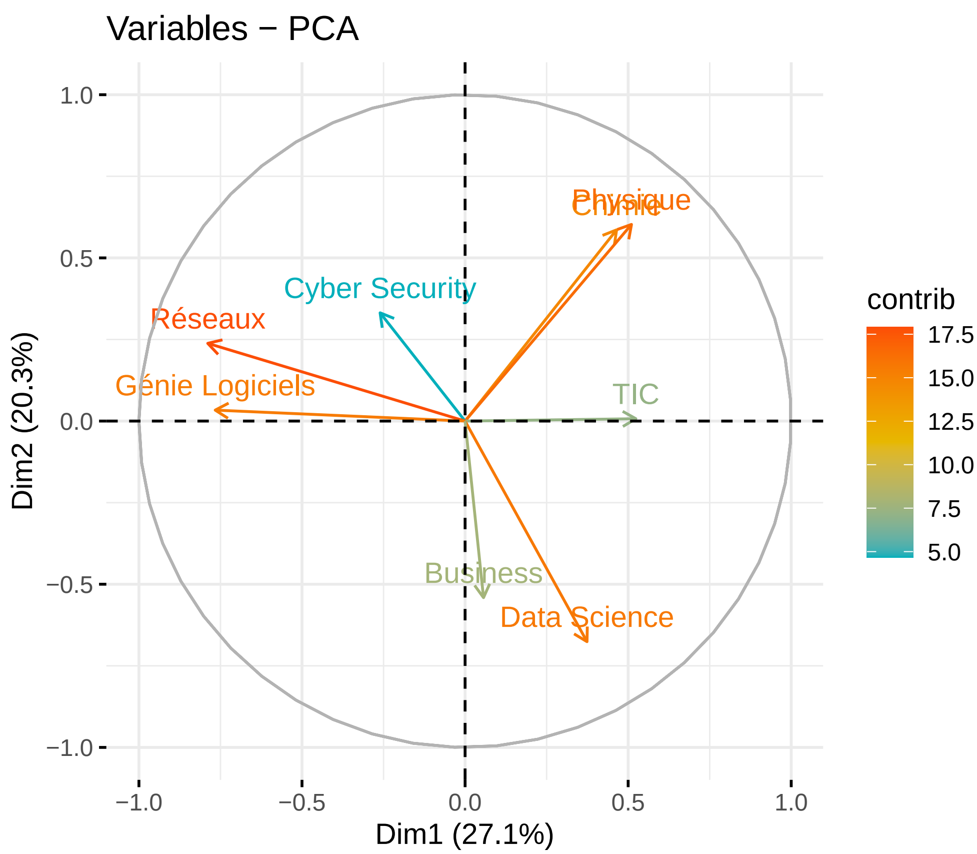

```r
fviz_contrib(res.pca1, choice = "var", axes = 2, top = 10)
```




```r
fviz_contrib(res.pca1, choice = "var", axes = 3, top = 10)
```



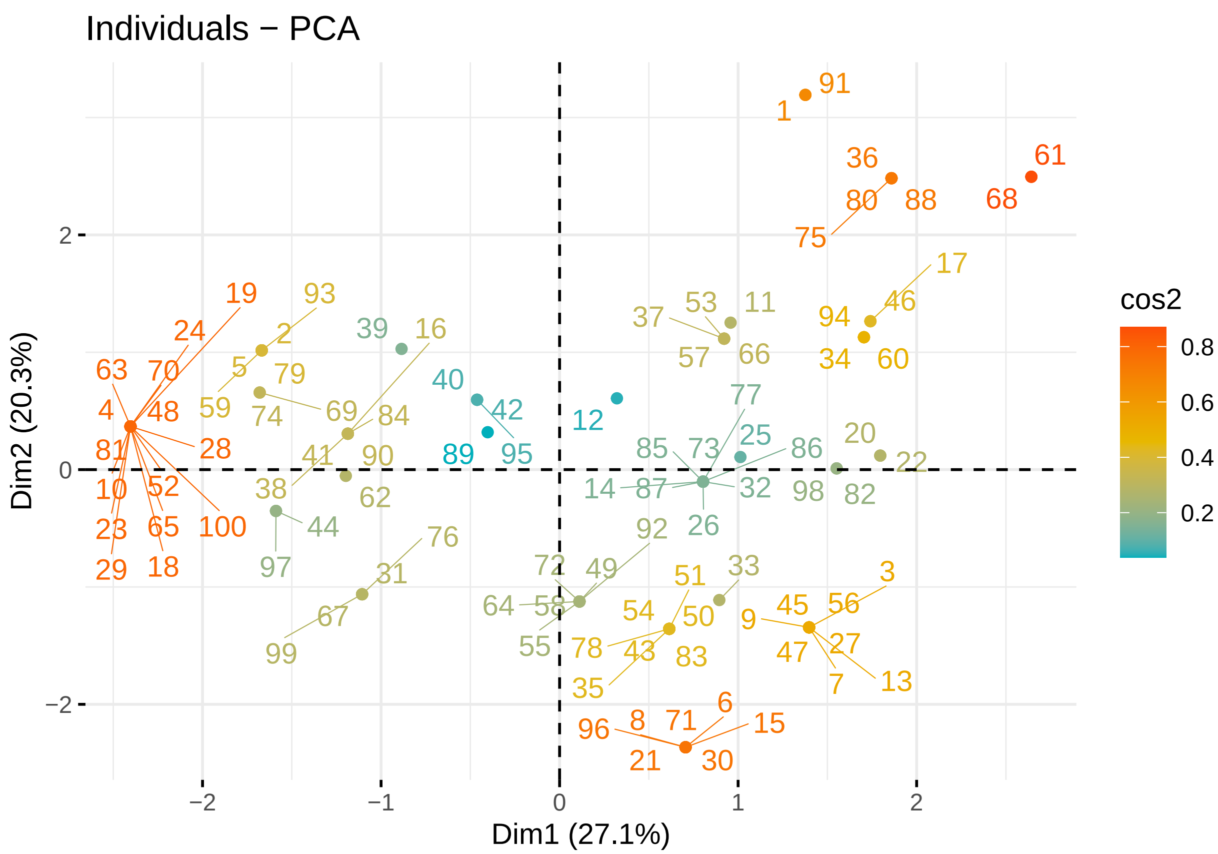

```
fviz_pca_var(res.pca1, col.var = "contrib",
             gradient.cols = c("#00AFBB", "#E7B800", "#FC4E07")
)
```



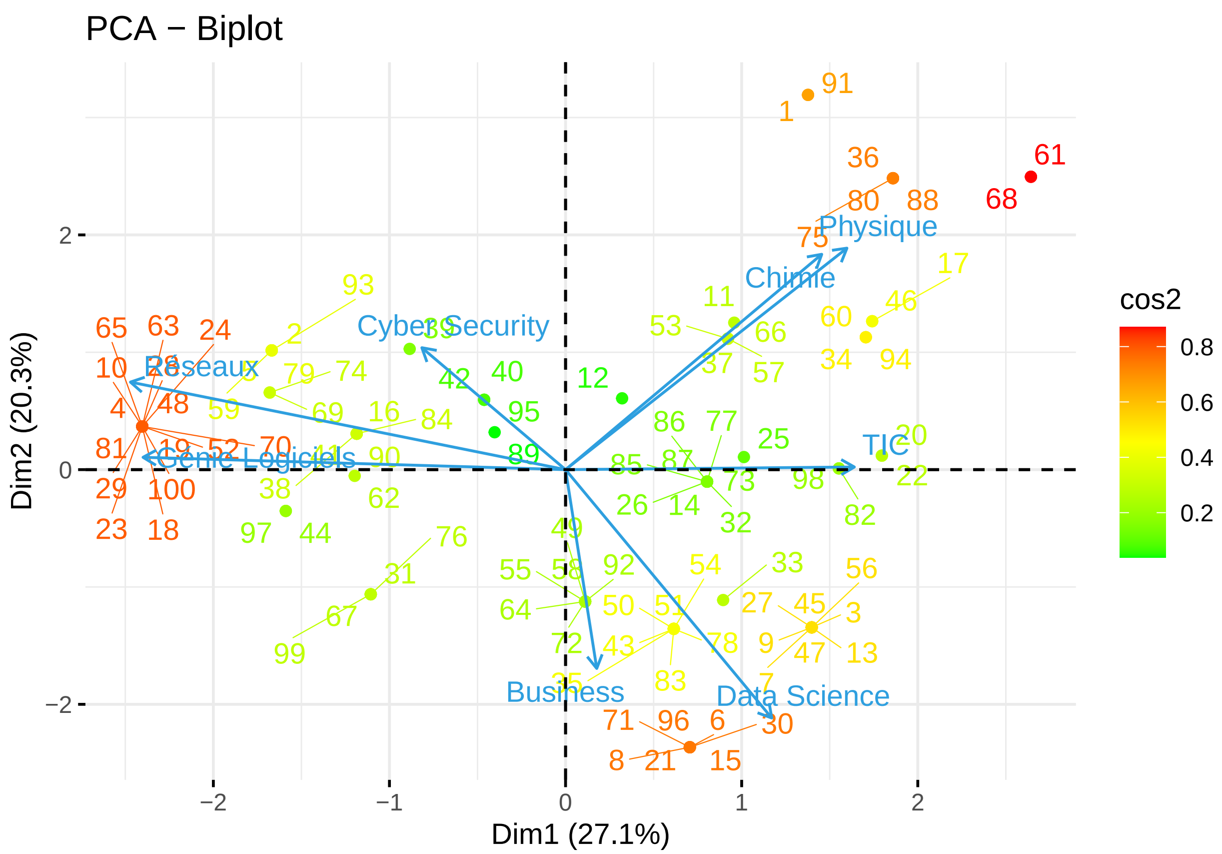

```r
ind1 <- get_pca_ind(res.pca1)
ind1
```

```
## Principal Component Analysis Results for individuals
## ===================================================
##   Name       Description
## 1 "$coord"   "Coordinates for the individuals"
## 2 "$cos2"    "Cos2 for the individuals"
## 3 "$contrib" "contributions of the individuals"
```

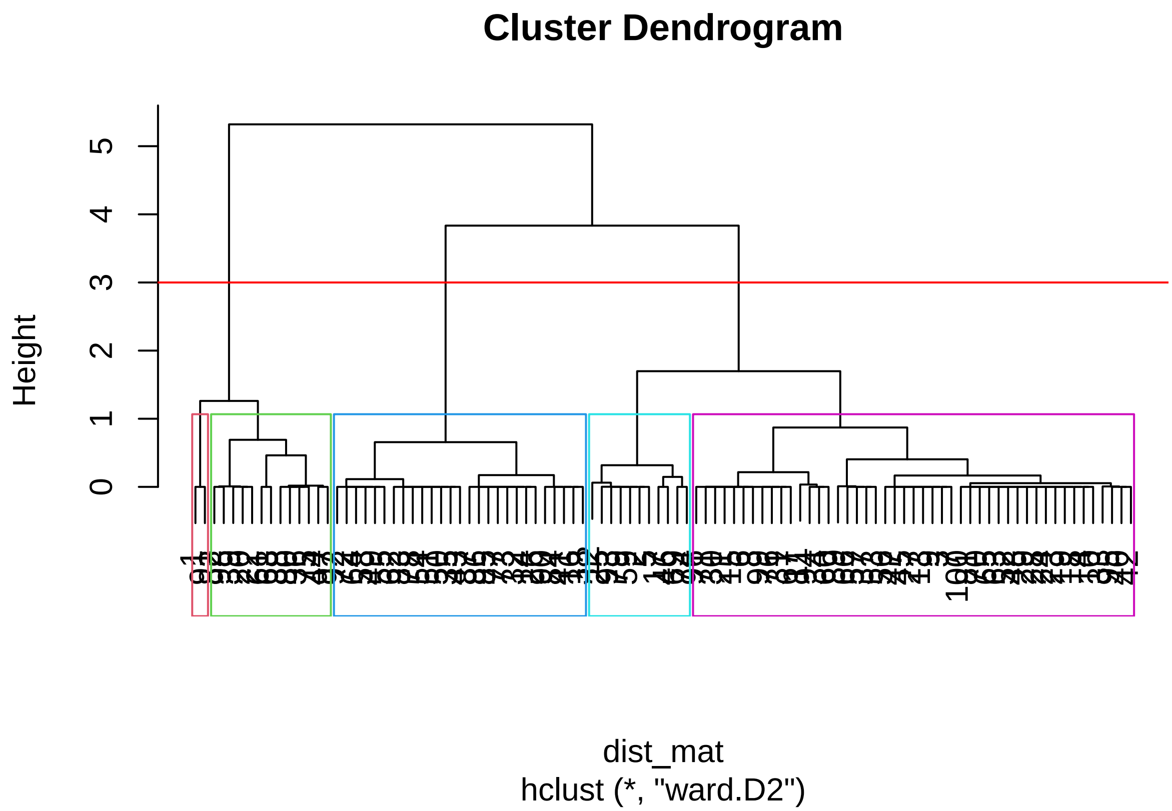
```r
fviz_pca_ind (res.pca1, col.ind = "cos2",
              gradient.cols = c("#00AFBB", "#E7B800", "#FC4E07"),
              repel = TRUE
)
```




```r
fviz_pca_biplot (res.pca1, col.ind = "cos2",
                 gradient.cols = c("green", "yellow", "red"),
                 col.var = "#2E9FDF",
                 repel = TRUE
)
```




```r
dist_mat <- dist(res.pca1$ind$dist, method = 'euclidean')
hclust_avg <- hclust(dist_mat, method = 'ward.D2')
plot(hclust_avg)
cut_avg <- cutree(hclust_avg, k = 5)
plot(hclust_avg)
rect.hclust(hclust_avg , k = 5, border = 2:6)
abline(h = 3, col = 'red')
```



Cluster Dendrogram
Height
0
1
2
3
4
5
dist_mat
hclust (*, "ward.D2")