Multiple Correspondence Analysis Essentials:

Alboukadel Kassambara 訳:藤本一男

2017/4/16

Contents

多重対応分析（MCA）の要点：質的多変量変数のカテゴリー間の連関調査への応用 ー Rソフトウェアとデータマイニング・ツール

Multiple Correspondence Analysis Essentials: Interpretation and application to investigate the associations between categories of multiple qualitative variables - R software and data mining Tools

私の以前の論文で示したように、シンプル対応分析（CA）は、二つのカテゴリカル変数によって形成される分割表の分析に用いられた。

As described in my previous article, the simple correspondence analysis (CA) is used to analyse the contingency table formed by two categorical variables.

CAについてさらに学びたい人は以下を参照されたい：「Rによる対応分析：解析、可視化そして解釈への根本的なガイド」

To learn more about CA, read this article: Correspondence Analysis in R: The Ultimate Guide for the Analysis, the Visualization and the Interpretation.

多重対応分析（MCA）は、シンプル対応分析CAの拡張であり、それは、カテゴリカル変数が二つ以上のデータ表の解析に用いられる。

Multiple Correspondence Analysis (MCA) is an extension of simple CA to analyse a data table containing more than two categorical variables.

MCAは、一般に調査データの解析に用いられる。

MCA is generally used to analyse a data from survey.

目的は以下のようになる：

質問に対する答えが似たプロファイルを有する個体のグループを識別する。 変数カテゴリー間の連関を式べtする。

The objectives are to identify:

```
A group of individuals with similar profile in their answers to the questions
The associations between variable categories
```

MCAを計算するためのRのfunctionとしては、いくつものパッケージがある。それには以下のものがある。： MCA()[FactoMineR パッケージ] dudi.mca()[ade4 パッケージ] （? jmca()[ca パッケージは?]）

There are several R functions from different packages to compute MCA, including:

```
MCA() [in FactoMineR package]
dudi.mca() [in ade4 package]
```

これらのパッケージには、分析のレザルトを表示する標準functionsも備わっている。factoextra パッケージを使うことで、より簡単に美しいグラフを生成することができる。

These packages provide also some standard functions to visualize the results of the analysis. It's also possible to use the package factoextra to generate easily beautiful graphs.

本論では、FactoMineRパッケージをもちいて、多重対応分析を実行し解釈する方法を記述する。

This article describes how to perform and interpret multiple correspondence analysis using FactoMineR package.

「参考」http://www.sthda.com/english/wiki/fviz-mca-quick-multiple-correspondence-analysis-data-visualization-r-software-and-data

1 必要なパッケージ Required packages

FactoMineR(for computing MCA) and factoextra (for MCA visualization) packages are used.

These packages can be installed as follow :

```
#install.packages("FactoMineR")
# install.packages("devtools")
#devtools::install_github("kassambara/factoextra")
```

Note that, for factoextra a version >= 1.0.2 is required for this tutorial. If it's already installed on your computer, you should re-install it to have the most updated version.

> 現在のversionでは、まだ、font指定のハンドリングが完全ではない。そのために、冗長な指定が必要になっている部分がある。

2 Load FactoMineR and factoextra

```
library("FactoMineR")
library("factoextra")
```

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

```
## Welcome! Related Books: `Practical Guide To Cluster Analysis in R` at https://goo.gl/13EFCZ
```

3 日本語環境構築に必要なパッケージ

```
library(ggpubr)#  fviz_ 系でfont指定をするために必要
```

```
## Loading required package: magrittr
```

```
library(lattice)# .RprofileでHookしてあるlatticeよう設定を呼びたす。plotecllipse で使う
library(car)# Recode function を使う
library(tidyverse)
```

```
## Loading tidyverse: tibble
## Loading tidyverse: tidyr
## Loading tidyverse: readr
## Loading tidyverse: purrr
## Loading tidyverse: dplyr
```

```
## Conflicts with tidy packages ----------------------------------------------
```

```
## filter(): dplyr, stats
## lag():    dplyr, stats
## recode(): dplyr, car
## some():   purrr, car
```

```
my_theme <- theme_minimal(base_family = "HiraKakuProN-W3")
# fviz_contrib(res.mca, "var", ggtheme = my_theme)
```

4 データの書式

データセットとして　FactoMineRで提供される posion(食中毒)を使う。

We'll use the data sets poison [in FactoMineR]

```
data(poison)
head(poison[, 1:7])
```

```
##   Age Time   Sick Sex   Nausea Vomiting Abdominals
## 1   9   22 Sick_y   F Nausea_y  Vomit_n     Abdo_y
## 2   5    0 Sick_n   F Nausea_n  Vomit_n     Abdo_n
## 3   6   16 Sick_y   F Nausea_n  Vomit_y     Abdo_y
## 4   9    0 Sick_n   F Nausea_n  Vomit_n     Abdo_n
```

```
## 5   7   14 Sick_y   M Nausea_n  Vomit_y     Abdo_y
## 6  72    9 Sick_y   M Nausea_n  Vomit_n     Abdo_y
```

4.1 データの日本語化

poisonは、変数もカテゴリーも英語である。これを日本語に変換しておく。- まず、変数名(列名)を日本語にする。- Recode をもちいて、カテゴリーを日本語にする。- summaryで変更結果を確認しておく。

```
colnames(poison) <- c("年齢","時刻","発症","性別","吐き気","嘔吐","腹痛","発熱","下痢",
                      "ポテト","魚","マヨ","ズッキーニ","チーズ","アイスクリーム")
poison$性別 <- Recode(poison$性別,'"F" = "女性";"M" = "男性";',as.factor.result=TRUE)
poison$発症 <- Recode(poison$発症,'"Sick_n" = "発症_n";"Sick_y" = "発症_y";',as.factor.result=TRUE)
poison$吐き気 <- Recode(poison$吐き気,'"Nausea_n" = "吐き気_n";"Nausea_y" = "吐き気_y";',as.factor.result=TRUE)
poison$嘔吐 <- Recode(poison$嘔吐,'"Vomit_n" = "嘔吐_n";"Vomit_y" = "嘔吐_y";',as.factor.result=TRUE)
poison$発熱 <- Recode(poison$発熱,'"Fever_n" = "発熱_n";"Fever_y" = "発熱_y";',as.factor.result=TRUE)
poison$下痢 <- Recode(poison$下痢,'"Diarrhea_n" = "下痢_n";"Diarrhea_y" = "下痢_y";',as.factor.result=TRUE)
poison$ポテト <- Recode(poison$ポテト,'"Potato_n" = "ポテト_n";"Potato_y" = "ポテト_y";',as.factor.result=TRUE)
poison$魚 <- Recode(poison$魚,'"Fish_n" = "魚_n";"Fish_y" = "魚_y";',as.factor.result=TRUE)
poison$マヨ <- Recode(poison$マヨ,'"Mayo_n" = "マヨ_n";"Mayo_y" = "マヨ_y";',as.factor.result=TRUE)
poison$ズッキーニ <- Recode(poison$ズッキーニ,'"Courg_n" = "ズッキ_n";"Courg_y" = "ズッキ_y";',as.factor.result=TRUE)
poison$チーズ <- Recode(poison$チーズ,'"Cheese_n" = "チーズ_n";"Cheese_y" = "チーズ_y";',as.factor.result=TRUE)
poison$アイスクリーム <- Recode(poison$アイスクリーム,'"Icecream_n" = "アイス_n";"Icecream_y" = "アイス_y";',as.factor.result=TRUE)

summary(poison)
```

```
##       年齢             時刻           発症       性別          吐き気
##  Min.   : 4.00   Min.   : 0.00   発症_n:17   女性:28   吐き気_n:43
##  1st Qu.: 6.00   1st Qu.: 0.00   発症_y:38   男性:27   吐き気_y:12
##  Median : 8.00   Median :12.00
##  Mean   :16.93   Mean   :10.16
##  3rd Qu.:10.00   3rd Qu.:16.50
##  Max.   :88.00   Max.   :22.00
##      嘔吐         腹痛         発熱         下痢          ポテト       魚
##  嘔吐_n:33   Abdo_n:18   発熱_n:20   下痢_n:20   ポテト_n: 3   魚_n: 1
##  嘔吐_y:22   Abdo_y:37   発熱_y:35   下痢_y:35   ポテト_y:52   魚_y:54
##
##
##
##
##      マヨ         ズッキーニ       チーズ     アイスクリーム
##  マヨ_n:10   ズッキ_n: 5   チーズ_n: 7   アイス_n: 4
##  マヨ_y:45   ズッキ_y:50   チーズ_y:48   アイス_y:51
##
##
##
##
```

```
# 当初、これで日本語データを読み込んでいた。
#write_csv(poison,"poison.csv")
#poison <- read.csv("poison.csv")# factor で読み込む
poison <- read.csv("poison_01.csv")# factor で読み込む
# http://419kfj.sakura.ne.jp/db/wp-content/uploads/2017/05/poison.csv
```

An image of the data is shown below:

```
tbl_df(poison)
```

```
## # A tibble: 55 x 15
##     年齢  時刻   発症   性別   吐き気   嘔吐   腹痛   発熱   下痢   ポテト
##  * <int> <int> <fctr> <fctr>  <fctr> <fctr> <fctr> <fctr> <fctr>  <fctr>
##  1     9    22 発症_y   女性 吐き気_y 嘔吐_n Abdo_y 発熱_y 下痢_y ポテト_y
##  2     5     0 発症_n   女性 吐き気_n 嘔吐_n Abdo_n 発熱_n 下痢_n ポテト_y
##  3     6    16 発症_y   女性 吐き気_n 嘔吐_y Abdo_y 発熱_y 下痢_y ポテト_y
##  4     9     0 発症_n   女性 吐き気_n 嘔吐_n Abdo_n 発熱_n 下痢_n ポテト_y
##  5     7    14 発症_y   男性 吐き気_n 嘔吐_y Abdo_y 発熱_y 下痢_y ポテト_y
##  6    72     9 発症_y   男性 吐き気_n 嘔吐_n Abdo_y 発熱_y 下痢_y ポテト_y
##  7     5    16 発症_y   女性 吐き気_n 嘔吐_y Abdo_y 発熱_y 下痢_y ポテト_y
##  8    10     8 発症_y   女性 吐き気_y 嘔吐_y Abdo_y 発熱_y 下痢_y ポテト_y
##  9     5    20 発症_y   男性 吐き気_y 嘔吐_n Abdo_y 発熱_y 下痢_y ポテト_y
## 10    11    12 発症_y   男性 吐き気_n 嘔吐_y Abdo_n 発熱_y 下痢_y ポテト_y
## # ... with 45 more rows, and 5 more variables: 魚 <fctr>, マヨ <fctr>,
## #   ズッキーニ <fctr>, チーズ <fctr>, アイスクリーム <fctr>
```

このデータは、食中毒にかかった小学校児童に対して実施された調査からとられている。そこでは、症状となにを食べたかが聞かれている。

This data is a result from a survey carried out on children of primary school who suffered from food poisoning. They were asked about their symptoms and about what they ate.

データは、55行(児童、個体)と15列(変数)からなっている。

The data contains 55 rows (children, individuals) and 15 columns (variables).

Only some of these individuals (children) and variables will be used to perform the multiple correspondence analysis (MCA). The coordinates of the remaining individuals and variables on the factor map will be predicted after the MCA.

In MCA terminology, our data contains :

- Active individuals (rows 1:55): Individuals that are used during the correspondence analysis.
- Active variables (columns 5:15) : Variables that are used for the MCA.
- Supplementary variables : They don't participate to the MCA. The coordinates of these variables will be predicted.
- Supplementary continuous variables : Columns 1 and 2 corresponding to the columns age and time, respectively.
- Supplementary qualitative variables : Columns 3 and 4 corresponding to the columns Sick and Sex, respectively. This factor variables will be used to color individuals by groups.

Subset only active individuals and variables for multiple correspondence analysis:

```
poison.active <- poison[1:55, 5:15]
head(poison.active[, 1:6])
```

```
##     吐き気   嘔吐   腹痛   発熱   下痢   ポテト
## 1 吐き気_y 嘔吐_n Abdo_y 発熱_y 下痢_y ポテト_y
## 2 吐き気_n 嘔吐_n Abdo_n 発熱_n 下痢_n ポテト_y
## 3 吐き気_n 嘔吐_y Abdo_y 発熱_y 下痢_y ポテト_y
## 4 吐き気_n 嘔吐_n Abdo_n 発熱_n 下痢_n ポテト_y
## 5 吐き気_n 嘔吐_y Abdo_y 発熱_y 下痢_y ポテト_y
## 6 吐き気_n 嘔吐_n Abdo_y 発熱_y 下痢_y ポテト_y
```

5　探索的データ解析 EDA:Exploratory data analysis

The function summary() can be used to compute the frequency of variable categories. As the data table contains a large number of variables, we'll display only the results for the first 4 variables.

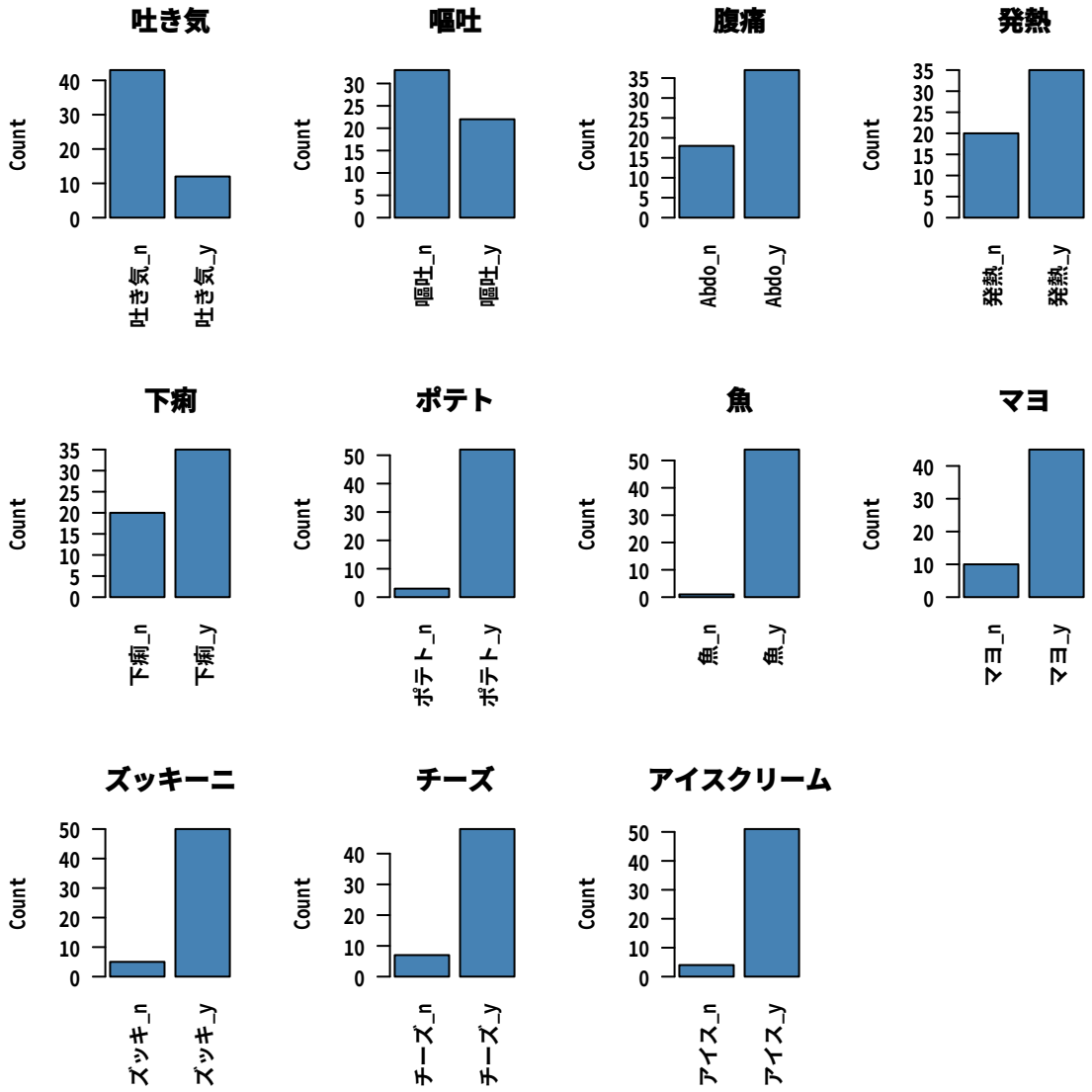
Statistical summaries:

```
# Summary of the 4 first variables
summary(poison.active)[, 1:4]
```

```
##       吐き気       嘔吐        腹痛        発熱
##  吐き気_n:43   嘔吐_n:33   Abdo_n:18   発熱_n:20
##  吐き気_y:12   嘔吐_y:22   Abdo_y:37   発熱_y:35
```

It's also possible to plot the frequency of variable categories:

```
oldpar <- par(mfrow=c(3,4))
for (i in 1:ncol(poison.active)) {
  plot(poison.active[,i], main=colnames(poison.active)[i],
       ylab = "Count", col="steelblue", las = 2)
}
par(oldpar)
```



The graphs above can be used to identify variable categories with a very low frequency. These types of variables can distort the analysis.

6 Multiple Correspondence Analysis (MCA)

The function MCA() [in FactoMineR package] can be used. A simplified format is :

```
MCA(X, ncp = 5, graph = TRUE)
```

- X : a data frame with n rows (individuals) and p columns (categorical variables)
- ncp : number of dimensions kept in the final results.
- graph : a logical value. If TRUE a graph is displayed.

In the R code below, the MCA is performed only on the active individuals/variables :

```
res.mca <- MCA(poison.active, graph = FALSE)
```

The output of the function MCA() is a list including :

```
print(res.mca)
```

```
## **Results of the Multiple Correspondence Analysis (MCA)**
## The analysis was performed on 55 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. of the categories"
## 4  "$var$cos2"       "cos2 for the categories"
## 5  "$var$contrib"    "contributions of the categories"
## 6  "$var$v.test"     "v-test for the categories"
## 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"           "intermediate results"
## 12 "$call$marge.col" "weights of columns"
## 13 "$call$marge.li"  "weights of rows"
```

The object that is created using the function MCA() contains results as lists. These values are described in the next sections.

7 Summary of MCA outputs

The function summary.MCA() [in FactoMineR] is used to print a summary of multiple correspondence analysis results:

```
summary(object, nb.dec = 3, nbelements = 10,
        ncp = TRUE, file ="", ...)
```

- object: an object of class MCA
- nb.dec: number of decimal printed
- nbelements: number of row/column variables to be written. To have all the elements, use nbelements = Inf.
- ncp: Number of dimensions to be printed
- file: an optional file name for exporting the summaries.

Print the summary of the MCA for the dimensions 1 and 2:

```
summary(res.mca, nb.dec = 2, ncp = 2)
```

```
##
## Call:
## MCA(X = poison.active, graph = FALSE)
##
##
## Eigenvalues
##                        Dim.1  Dim.2  Dim.3  Dim.4  Dim.5  Dim.6  Dim.7
## Variance               0.34   0.13   0.11   0.10   0.08   0.07   0.06
## % of var.             33.52  12.91  10.73   9.59   7.88   7.11   6.02
## Cumulative % of var.  33.52  46.44  57.17  66.76  74.64  81.75  87.77
##                        Dim.8  Dim.9 Dim.10 Dim.11
## Variance               0.06   0.04   0.01   0.01
## % of var.              5.58   4.12   1.30   1.23
```

```
## Cumulative % of var.  93.35  97.47  98.77 100.00
##
## Individuals (the 10 first)
##          Dim.1   ctr  cos2   Dim.2   ctr  cos2
## 1      | -0.45  1.11  0.35 | -0.26  0.98  0.12 |
## 2      |  0.84  3.79  0.56 | -0.03  0.01  0.00 |
## 3      | -0.45  1.09  0.55 |  0.14  0.26  0.05 |
## 4      |  0.88  4.20  0.75 | -0.09  0.10  0.01 |
## 5      | -0.45  1.09  0.55 |  0.14  0.26  0.05 |
## 6      | -0.36  0.70  0.02 | -0.44  2.68  0.04 |
## 7      | -0.45  1.09  0.55 |  0.14  0.26  0.05 |
## 8      | -0.64  2.23  0.62 | -0.01  0.00  0.00 |
## 9      | -0.45  1.11  0.35 | -0.26  0.98  0.12 |
## 10     | -0.14  0.11  0.04 |  0.12  0.21  0.03 |
##
## Categories (the 10 first)
##            Dim.1   ctr  cos2 v.test   Dim.2   ctr  cos2 v.test
## 吐き気_n |  0.27  1.52  0.26   3.72 |  0.12  0.81  0.05   1.69 |
## 吐き気_y | -0.96  5.43  0.26  -3.72 | -0.43  2.91  0.05  -1.69 |
## 嘔吐_n   |  0.48  3.73  0.34   4.31 | -0.41  7.07  0.25  -3.68 |
## 嘔吐_y   | -0.72  5.60  0.34  -4.31 |  0.61 10.61  0.25   3.68 |
## Abdo_n   |  1.32 15.42  0.85   6.76 | -0.04  0.03  0.00  -0.18 |
## Abdo_y   | -0.64  7.50  0.85  -6.76 |  0.02  0.01  0.00   0.18 |
## 発熱_n   |  1.17 13.54  0.78   6.51 | -0.17  0.78  0.02  -0.97 |
## 発熱_y   | -0.67  7.74  0.78  -6.51 |  0.10  0.45  0.02   0.97 |
## 下痢_n   |  1.18 13.80  0.80   6.57 |  0.00  0.00  0.00  -0.02 |
## 下痢_y   | -0.68  7.88  0.80  -6.57 |  0.00  0.00  0.00   0.02 |
##
## Categorical variables (eta2)
##              Dim.1 Dim.2
## 吐き気     |  0.26  0.05 |
## 嘔吐       |  0.34  0.25 |
## 腹痛       |  0.85  0.00 |
## 発熱       |  0.78  0.02 |
## 下痢       |  0.80  0.00 |
## ポテト     |  0.03  0.40 |
## 魚         |  0.01  0.03 |
## マヨ       |  0.38  0.03 |
## ズッキーニ |  0.02  0.45 |
## チーズ     |  0.19  0.05 |
```

The result of the function summary() contains 4 tables:

- Table 1 - Eigenvalues: table 1 contains the variances and the percentage of variances retained by each dimension.
- Table 2 contains the coordinates, the contribution and the cos2 (quality of representation [in 0-1]) of the first 10 active individuals on the dimensions 1 and 2.
- Table 3 contains the coordinates, the contribution and the cos2 (quality of representation [in 0-1]) of the first 10 active variable categories on the dimensions 1 and 2. This table contains also a column called v.test. The value of the v.test is generally comprised between 2 and -2. For a given variable category, if the absolute value of the v.test is superior to 2, this means that the coordinate is significantly different from 0.
- Table 4 - categorical variables (eta2): contains the squared correlation between each variable and the dimensions.
    - For exporting the summary to a file, use the code: summary(res.mca, file =“myfile.txt”)
    - For displaying the summary of more than 10 elements, use the argument nbelements in the func-

tion summary()

8　MCA出力の解釈

MCA results is interpreted as the results from a simple correspondence analysis (CA).

I recommend to read the interpretation of simple CA which has been comprehensively described in my previous post: Correspondence Analysis in R: The Ultimate Guide for the Analysis, the Visualization and the Interpretation.

9　固有値/分散とスクリープロット

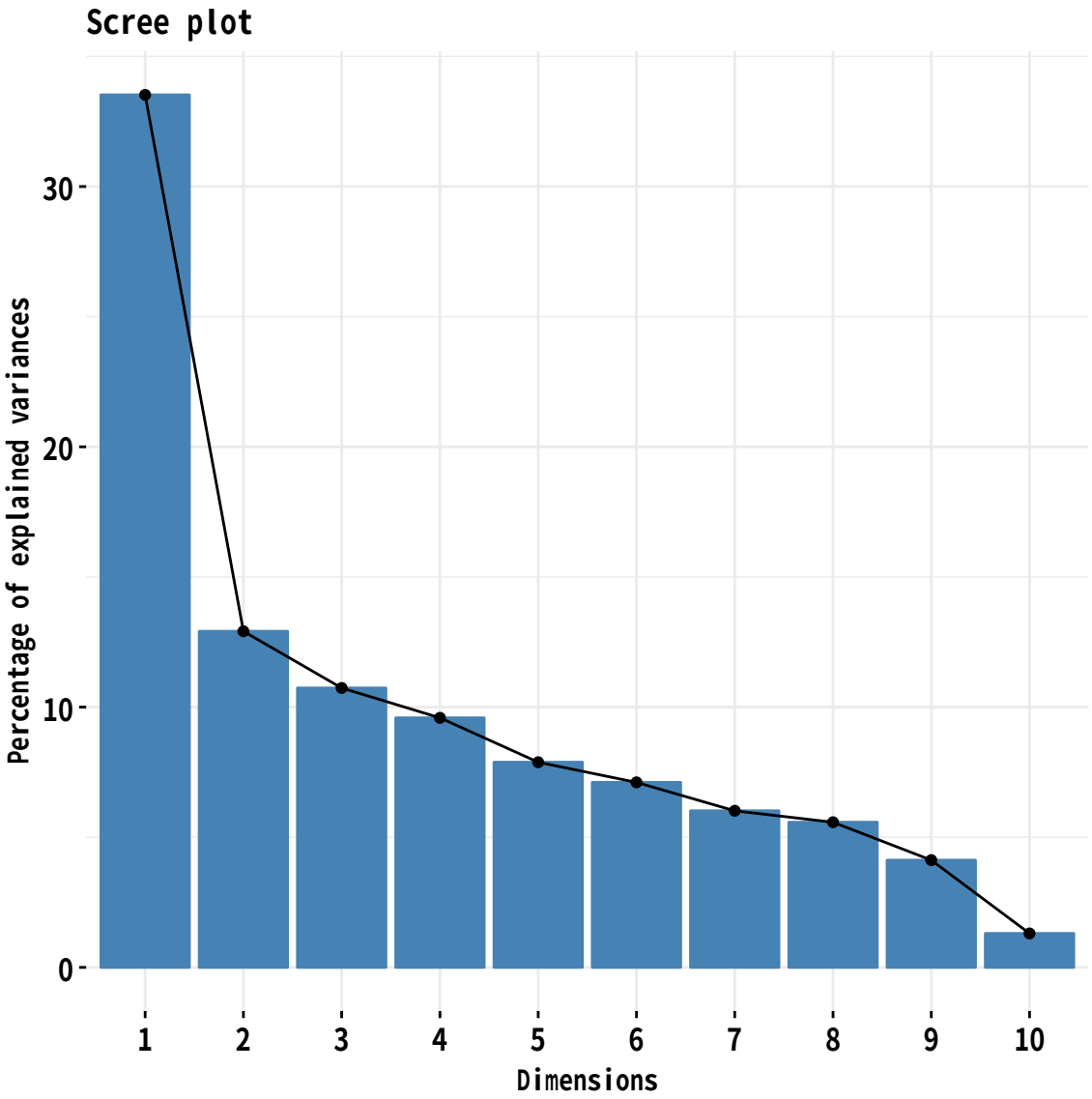
The proportion of variances retained by the different dimensions (axes) can be extracted using the function get_eigenvalue() [in factoextra] as follow :

```
eigenvalues <- get_eigenvalue(res.mca)
head(round(eigenvalues, 2))
```

```
##       eigenvalue variance.percent cumulative.variance.percent
## Dim.1       0.34            33.52                        33.52
## Dim.2       0.13            12.91                        46.44
## Dim.3       0.11            10.73                        57.17
## Dim.4       0.10             9.59                        66.76
## Dim.5       0.08             7.88                        74.64
## Dim.6       0.07             7.11                        81.75
```

The function fviz_screeplot() [in factoextra package] can be used to draw the scree plot (the percentages of inertia explained by the MCA dimensions):

```
fviz_screeplot(res.mca)
```



Read more about eigenvalues and screeplot: Eigenvalues data visualization

10 MCA scatter plot: Biplot of individuals and variable categories

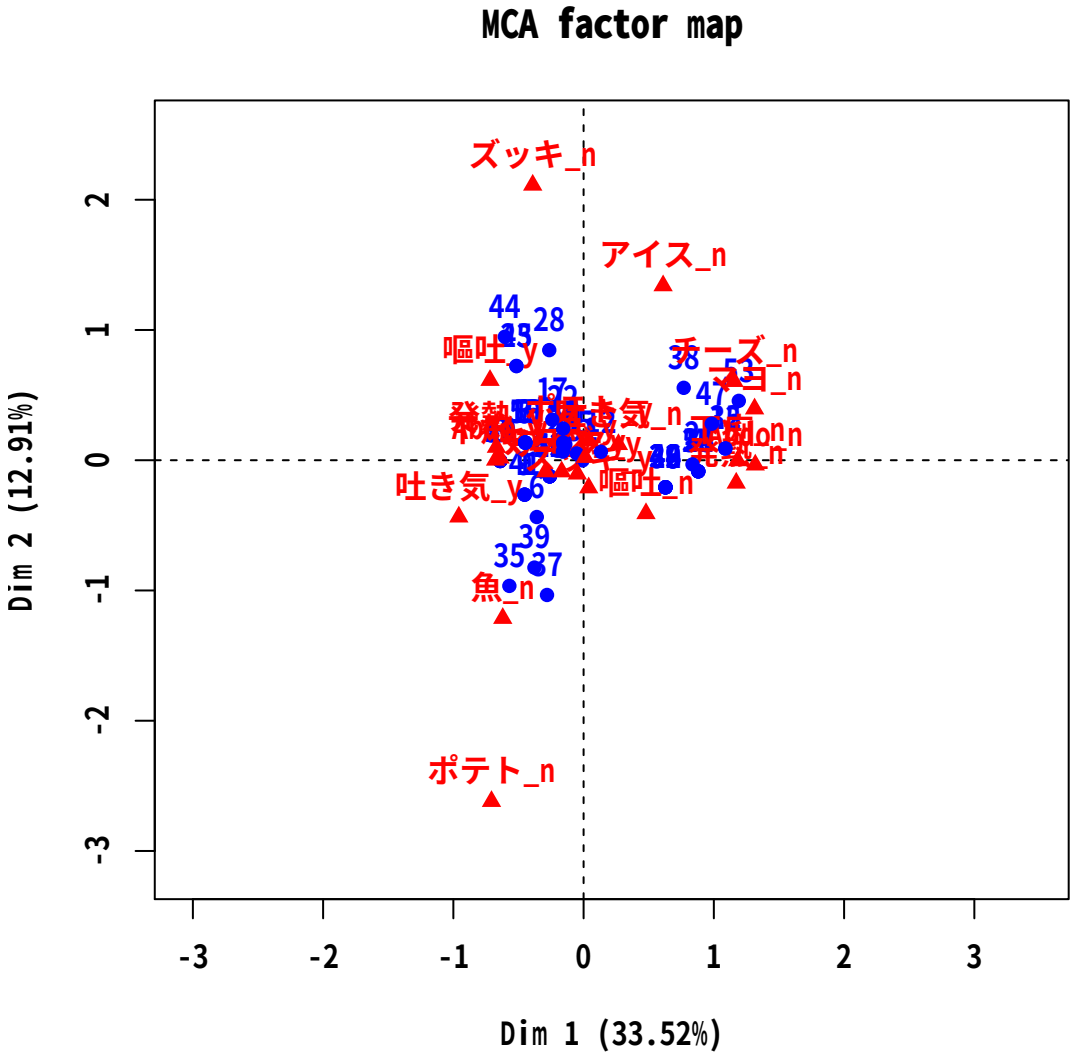
The function plot.MCA() [in FactoMineR package] can be used. A simplified format is :

```
plot(x, axes = c(1,2), choix=c("ind", "var"))
```

- x : An object of class MCA
- axes : A numeric vector of length 2 specifying the component to plot
- choix : The graph to be plotted. Possible values are "ind" for the individuals and "var" for the variables

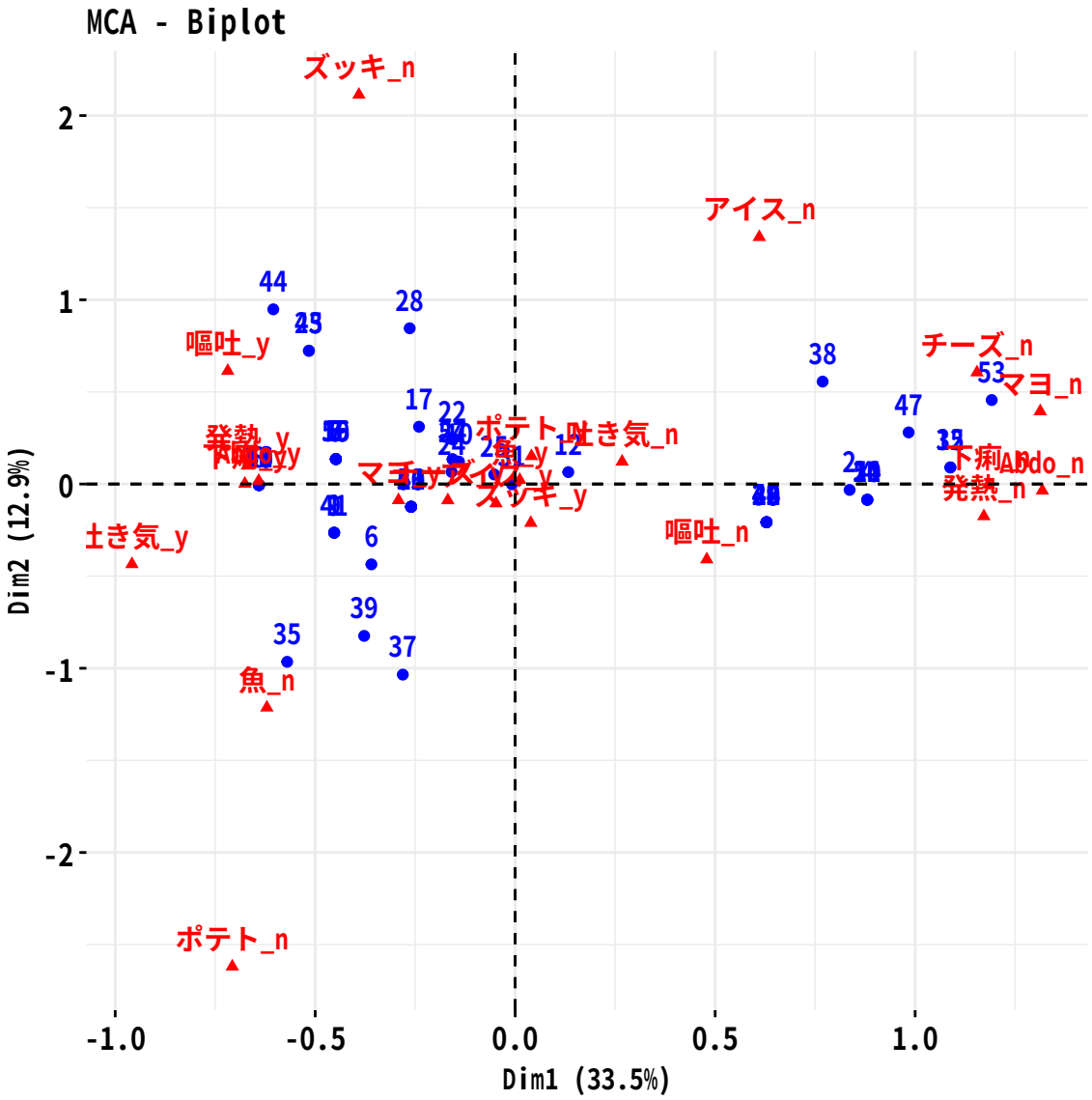
FactoMineR base graph for MCA:

```
plot(res.mca)
```



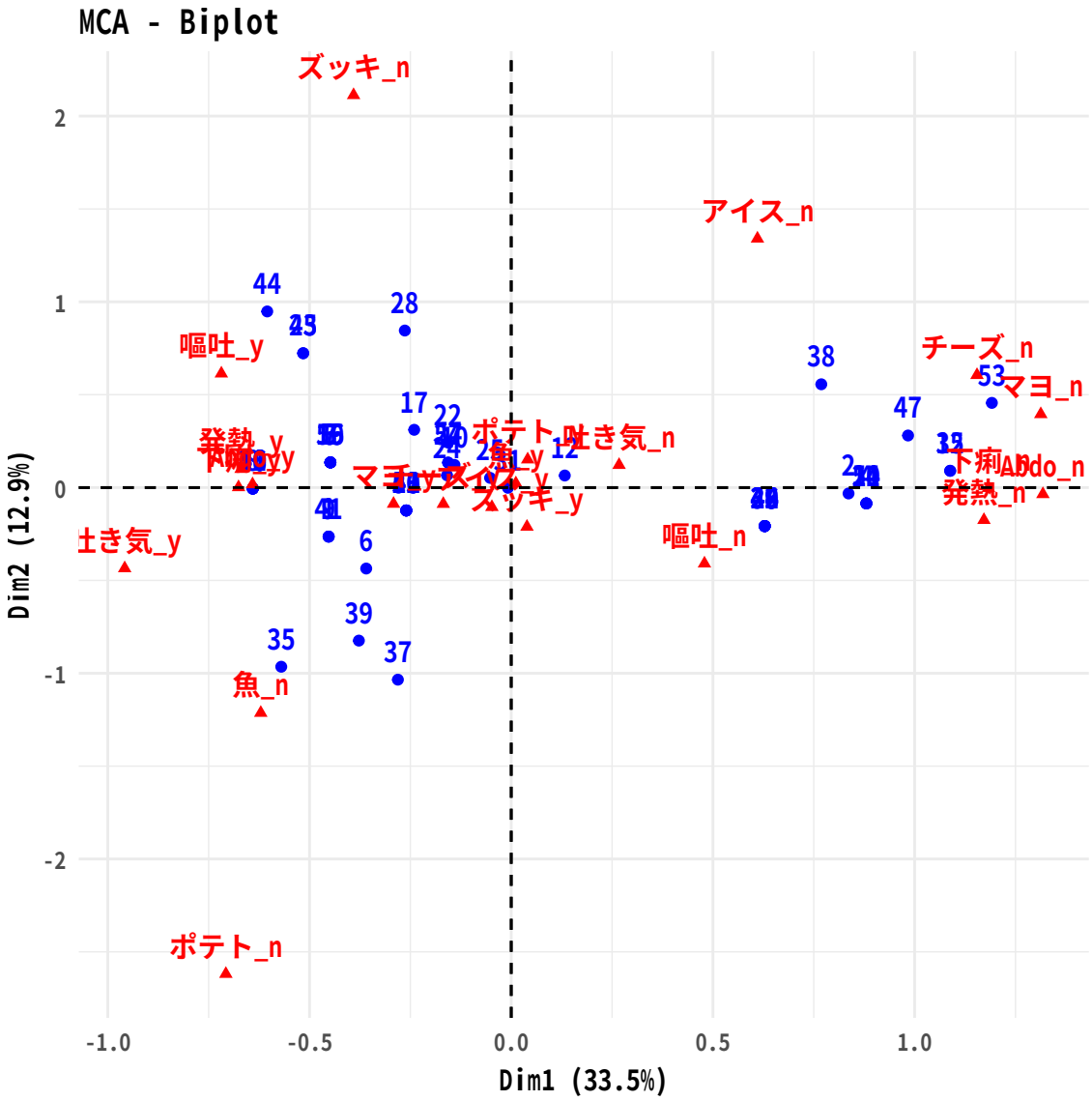
It's also possible to use the function fviz_mca_biplot()[in factoextra package] to draw a nice looking plot:

```
fviz_mca_biplot(res.mca,font.family = "sans")
```




```
#fviz_mca_biplot(res.mca,ggtheme = my_theme)
```

```
# Change the theme
fviz_mca_biplot(res.mca,font.family = "sans") +
  theme_minimal()
```



Read more about fviz_mca_biplot(): fviz_mca_biplot

The graph above shows a global pattern within the data. Rows (individuals) are represented by blue points and columns (variable categories) by red triangles.

The distance between any row points or column points gives a measure of their similarity (or dissimilarity).

Row points with similar profile are closed on the factor map. The same holds true for column points.

11 Variable categories

The function get_mca_var()[in factoextra] is used to extract the results for variable categories. This function returns a list containing the coordinates, the cos2 and the contribution of variable categories:

```
var <- get_mca_var(res.mca)
var
```

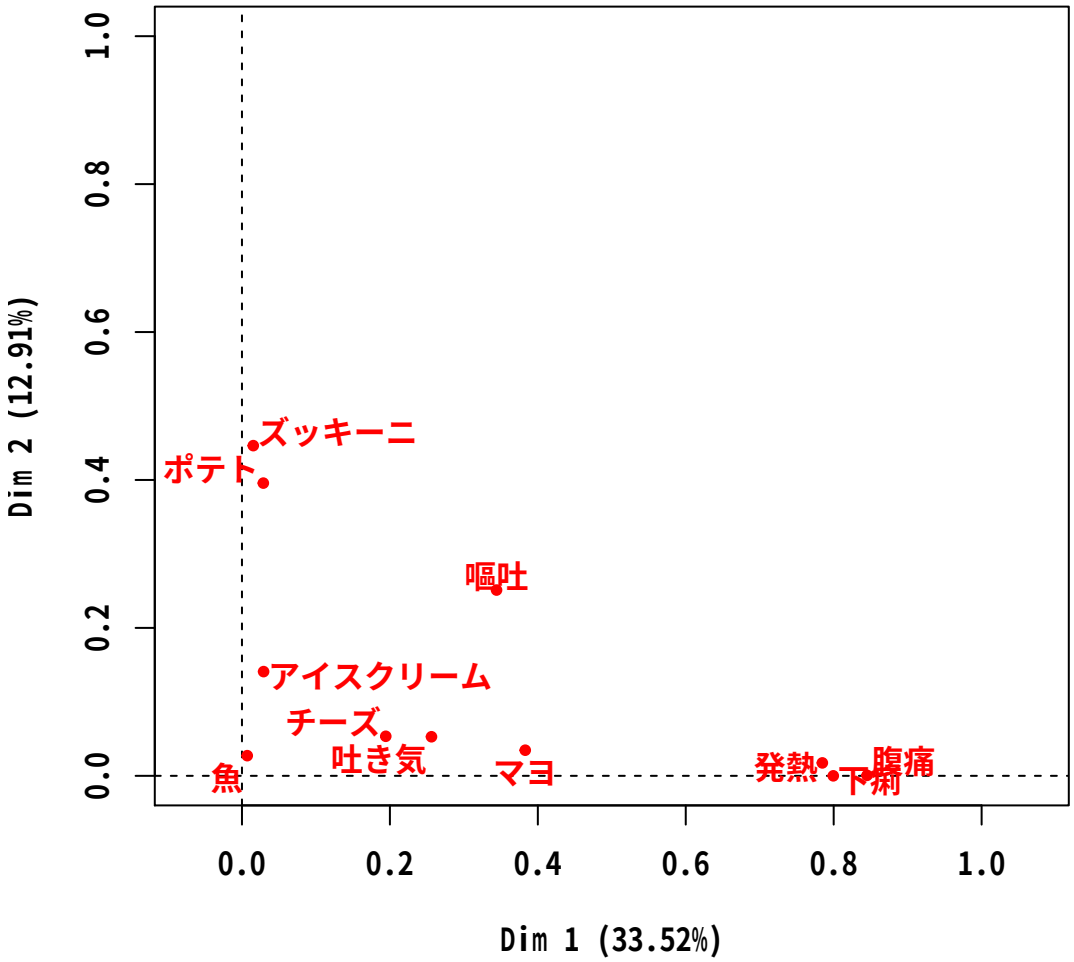
```
## Multiple Correspondence Analysis Results for variables
##  ===================================================
```

```
##   Name       Description
## 1 "$coord"   "Coordinates for categories"
## 2 "$cos2"    "Cos2 for categories"
## 3 "$contrib" "contributions of categories"
```

12 Correlation between variables and principal dimensions

Variables can be visualized as follow:

```
plot(res.mca, choix = "var")
```



- The plot above helps to identify variables that are the most correlated with each dimension. The squared correlations between variables and the dimensions are used as coordinates.
- It can be seen that, the variables Diarrhae, Abdominals and Fever are the most correlated with dimension 1. Similarly, the variables Courgette and Potato are the most correlated with dimension 2.

13 Coordinates of variable categories

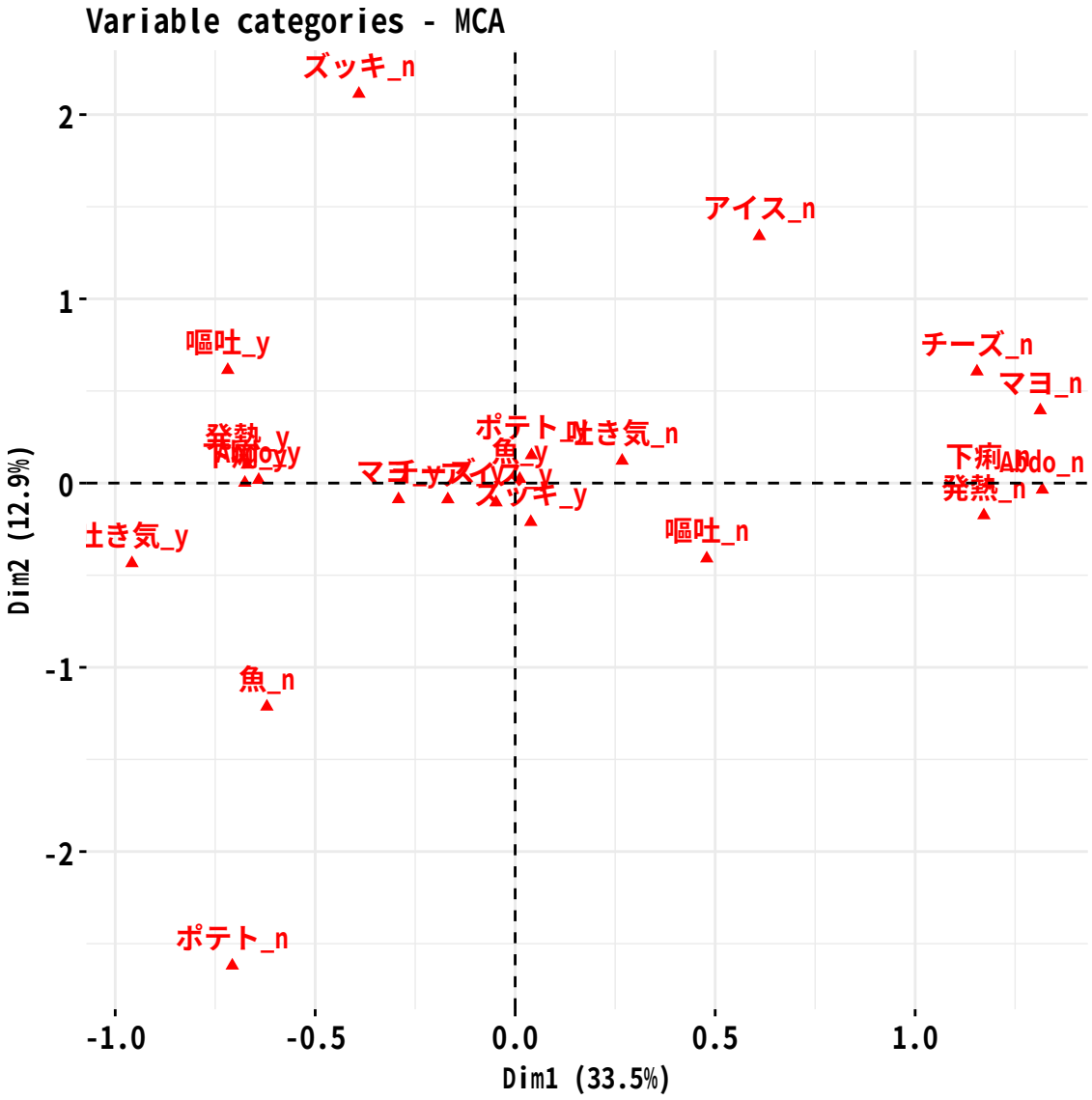
```
head(round(var$coord, 2))
```

```
##          Dim 1 Dim 2 Dim 3 Dim 4 Dim 5
## 吐き気_n  0.27  0.12 -0.27  0.03  0.07
```

```
## 吐き気_y -0.96 -0.43  0.95 -0.12 -0.26
## 嘔吐_n    0.48 -0.41  0.08  0.27  0.05
## 嘔吐_y   -0.72  0.61 -0.13 -0.41 -0.08
## Abdo_n    1.32 -0.04 -0.01 -0.15 -0.07
## Abdo_y   -0.64  0.02  0.00  0.07  0.03
```

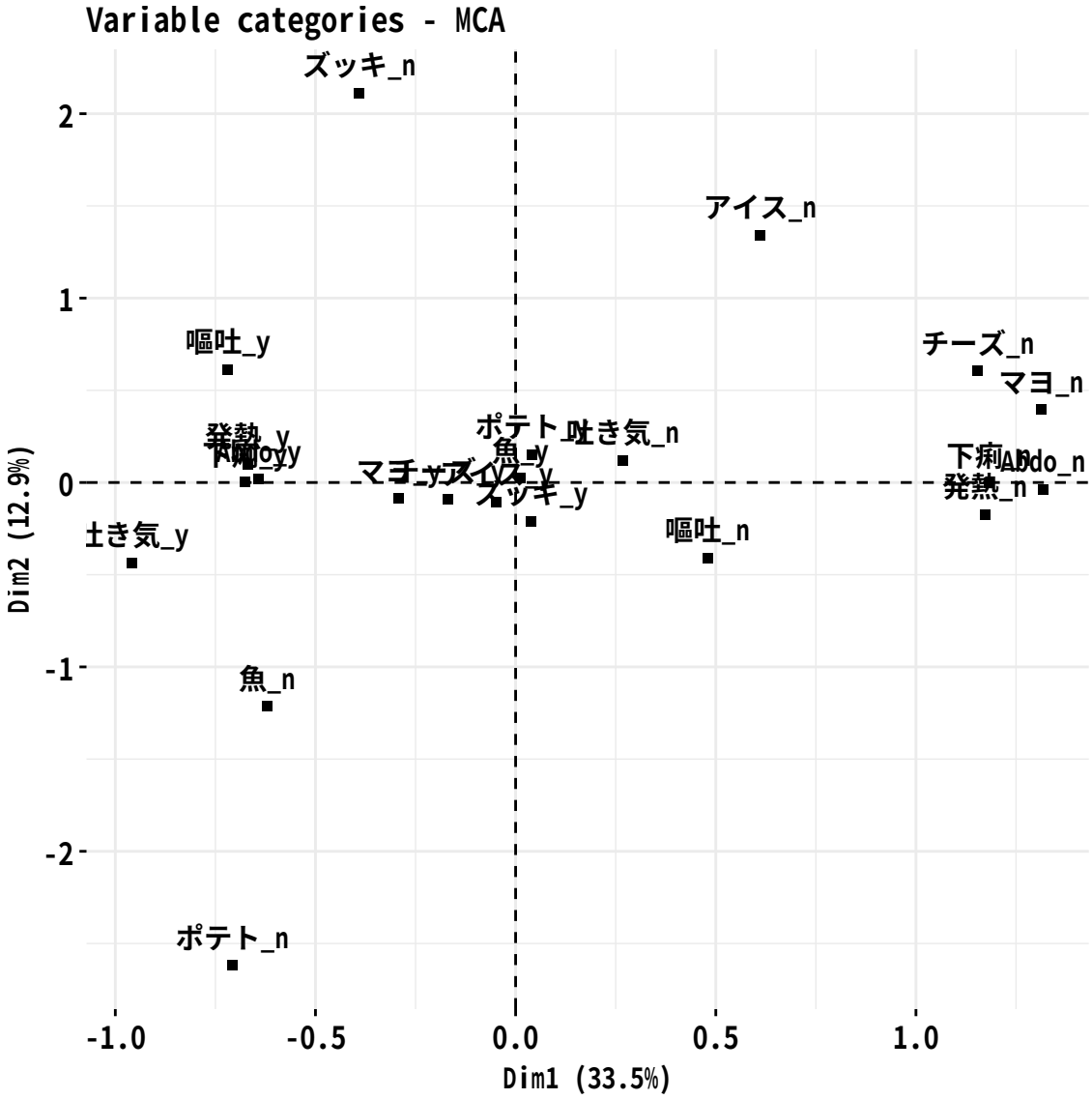
Use the function fviz_mca_var() [in factoextra] to visualize only variable categories:

```
# Default plot
fviz_mca_var(res.mca,font.family = "sans")
```



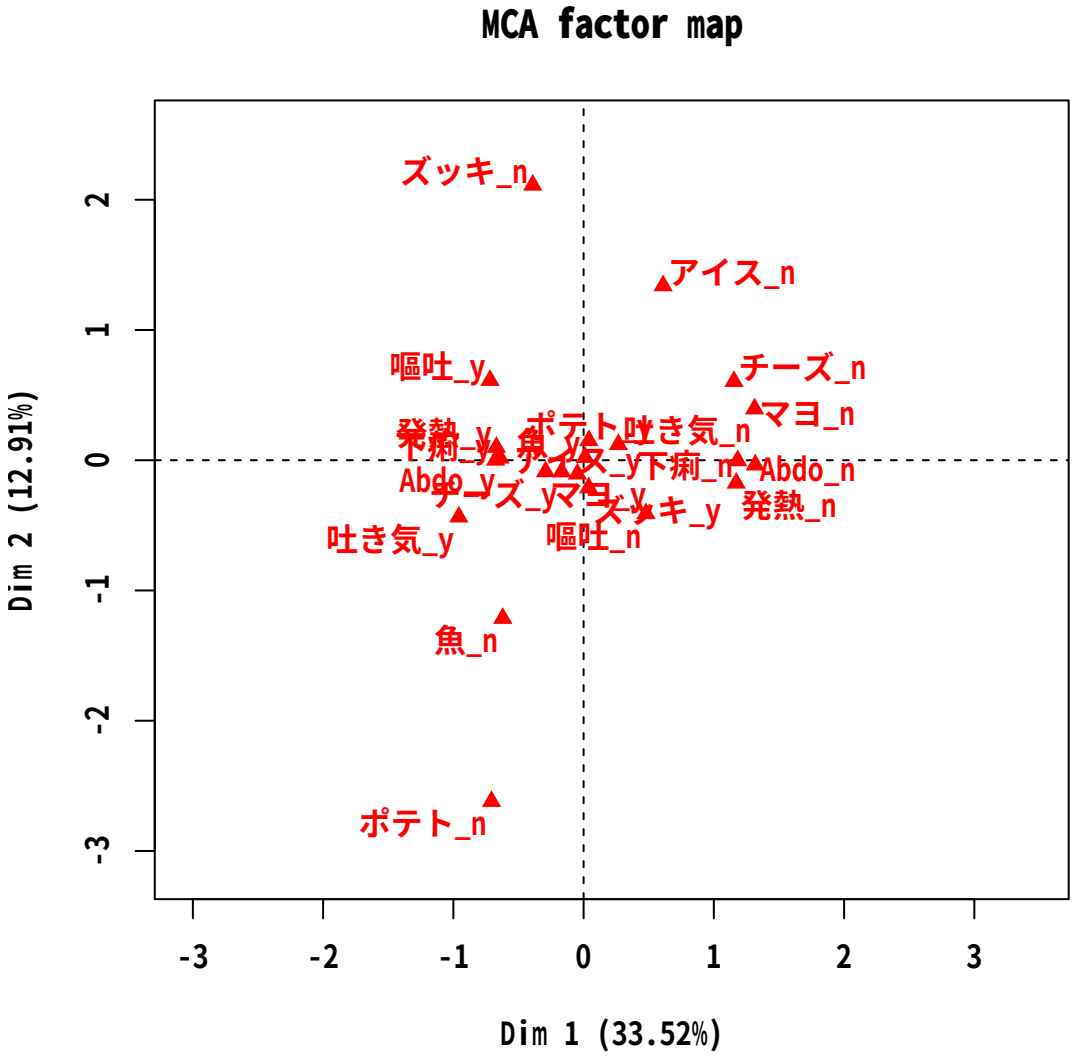
It's possible to change the color and the shape of the variable points using the arguments col.var and shape.var as follow:

```
fviz_mca_var(res.mca, col.var="black", shape.var = 15,font.family = "sans")
```



Note that, it's also possible to make the graph of variables only using FactoMineR base graph. The argument invisible is used to hide the individual points:

```
# Hide individuals
plot(res.mca, invisible="ind")
```



14　軸への変数カテゴリーの寄与

The contribution of the variable categories (in %) to the definition of the dimensions can be extracted as follow:

```
head(round(var$contrib,2))
```

```
##          Dim 1 Dim 2 Dim 3 Dim 4 Dim 5
## 吐き気_n  1.52  0.81  4.67  0.08  0.49
## 吐き気_y  5.43  2.91 16.73  0.30  1.76
## 嘔吐_n    3.73  7.07  0.36  4.26  0.19
## 嘔吐_y    5.60 10.61  0.54  6.39  0.29
## Abdo_n   15.42  0.03  0.00  0.73  0.18
## Abdo_y    7.50  0.01  0.00  0.36  0.09
```

The variable categories with the larger value, contribute the most to the definition of the dimensions.

The different categories in the table are:

```
categories <- rownames(var$coord)
length(categories)
```

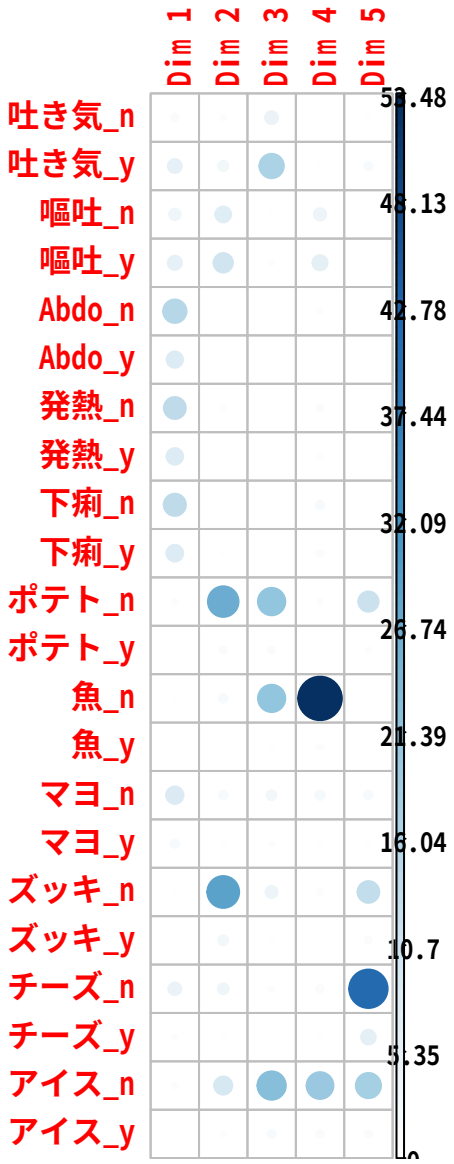
```
## [1] 22
```

```
print(categories)
```

```
##  [1] "吐き気_n" "吐き気_y" "嘔吐_n"   "嘔吐_y"   "Abdo_n"   "Abdo_y"
##  [7] "発熱_n"   "発熱_y"   "下痢_n"   "下痢_y"   "ポテト_n" "ポテト_y"
## [13] "魚_n"     "魚_y"     "マヨ_n"   "マヨ_y"   "ズッキ_n" "ズッキ_y"
## [19] "チーズ_n" "チーズ_y" "アイス_n" "アイス_y"
```

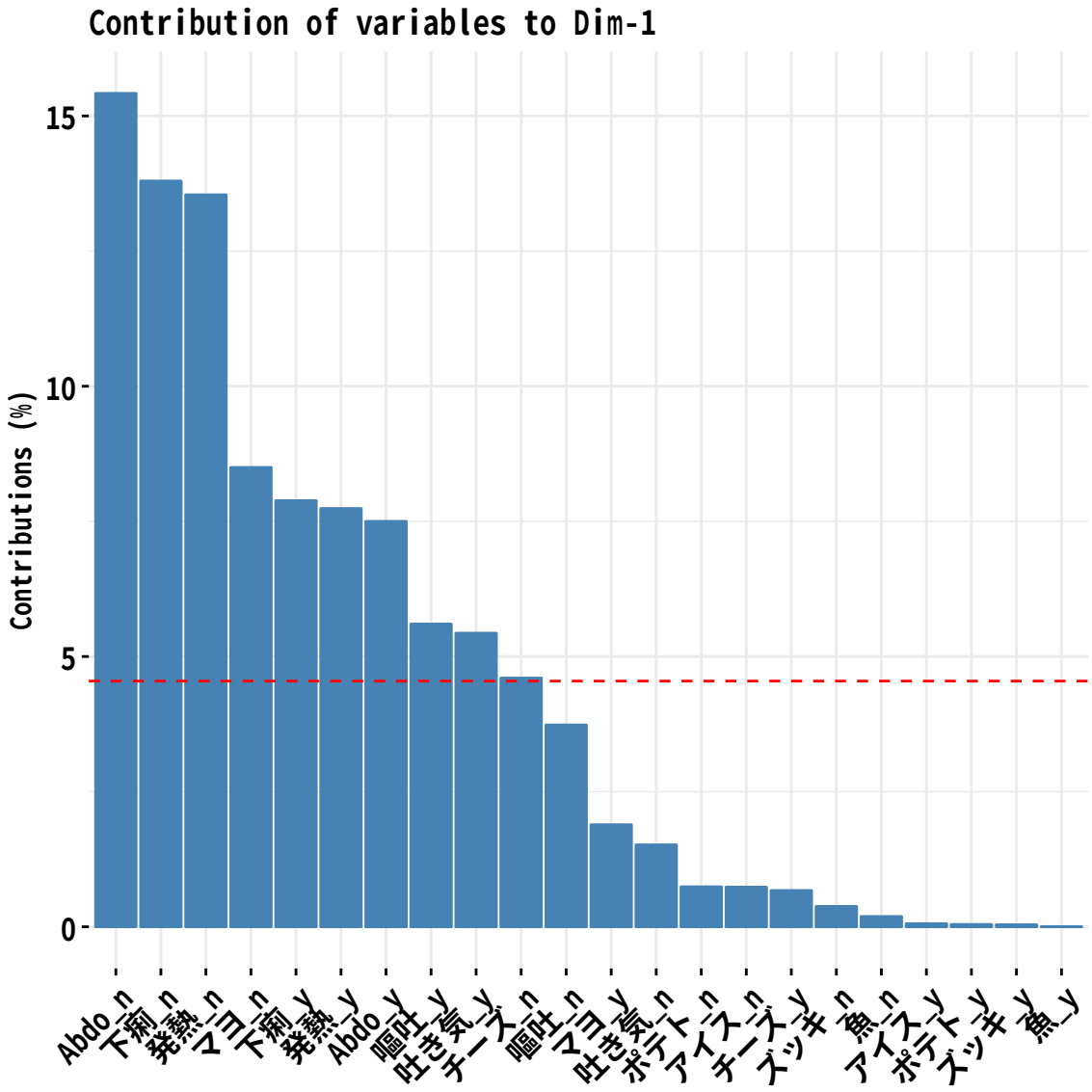
It's possible to use the function corrplot to highlight the most contributing variables for each dimension:

```
library("corrplot")
corrplot(var$contrib, is.corr = FALSE)
```



The function fviz_contrib()[in factoextra] can be used to draw a bar plot of variable contributions:

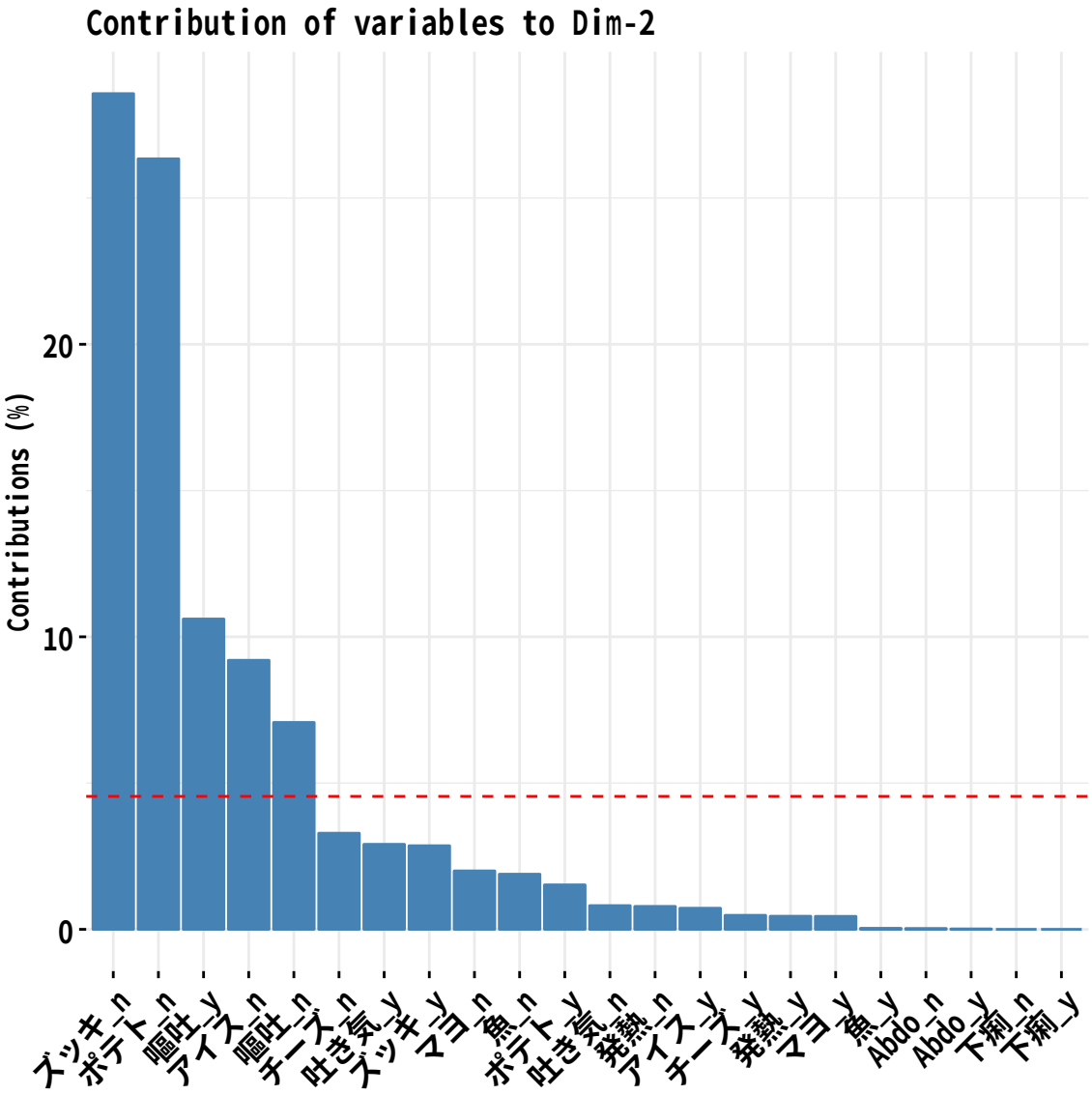
```
# Contributions of variables on Dim.1
fviz_contrib(res.mca, choice = "var", axes = 1, font.family = "sans") + theme(text = element_text(family = "sans"))
```



- If the contribution of variable categories were uniform, the expected value would be 1/number_of_categories = 1/22 = 4.5%.
- The red dashed line on the graph above indicates the expected average contribution. For a given dimension, any category with a contribution larger than this threshold could be considered as important in contributing to that dimension.

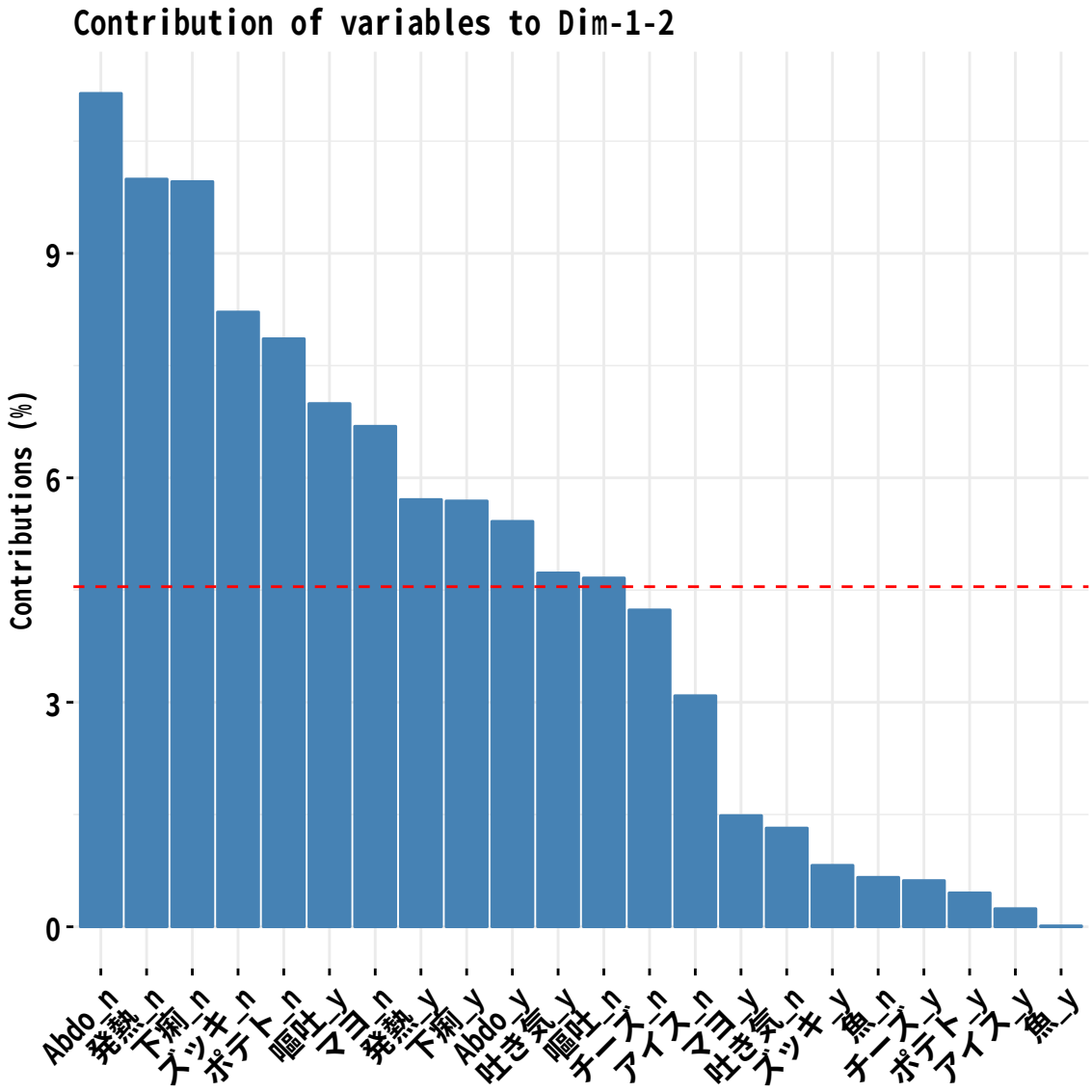
  It can be seen that the categories Abdo_n, Diarrhea_n, Fever_n and Mayo_n are the most important in the definition of the first dimension.

```
# Contributions of rows on Dim.2
fviz_contrib(res.mca, choice = "var", axes = 2, font.family = "sans") + theme(text = element_text(family = "sans"))
```



The row items Courg_n, Potato_n, Vomit_y and Icecream_n contribute the most to the dimension 2.

```
# Total contribution on Dim.1 and Dim.2
fviz_contrib(res.mca, choice = "var", axes = 1:2)  + theme(text = element_text(family = "sans"))
```

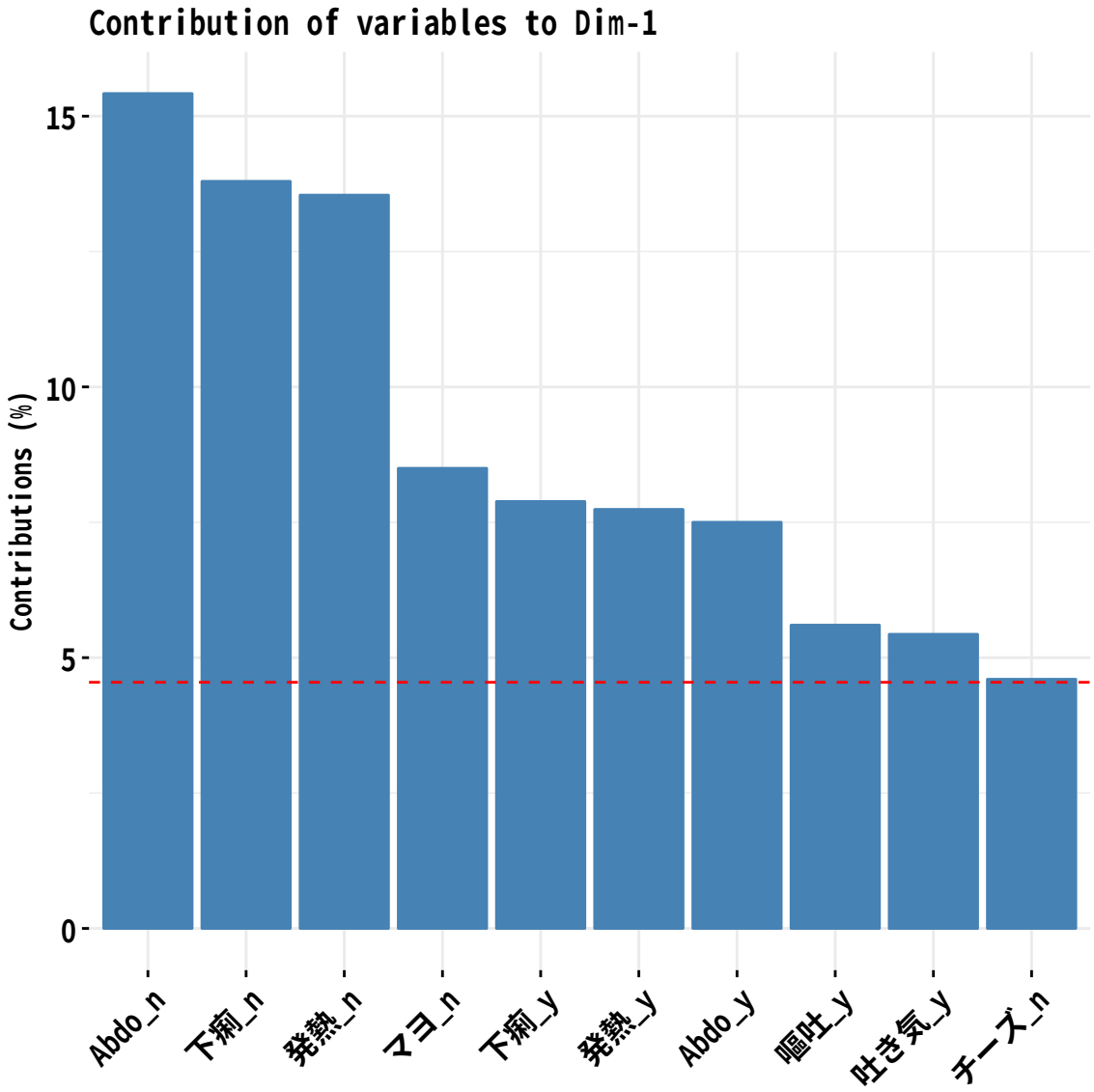


The total contribution of a category, on explaining the variations retained by Dim.1 and Dim.2, is calculated as follow : (C1 * Eig1) + (C2 * Eig2).

C1 and C2 are the contributions of the category to dimensions 1 and 2, respectively. Eig1 and Eig2 are the eigenvalues of dimensions 1 and 2, respectively. The expected average contribution of a category for Dim.1 and Dim.2 is : (4.5 * Eig1) + (4.5 * Eig2) = (4.50.34) + (4.50.13) = 2.12%

If your data contains many categories, the top contributing categories can be displayed as follow:

```
fviz_contrib(res.mca, choice = "var", axes = 1, top = 10) + theme(text = element_text(family = "sans"))
```

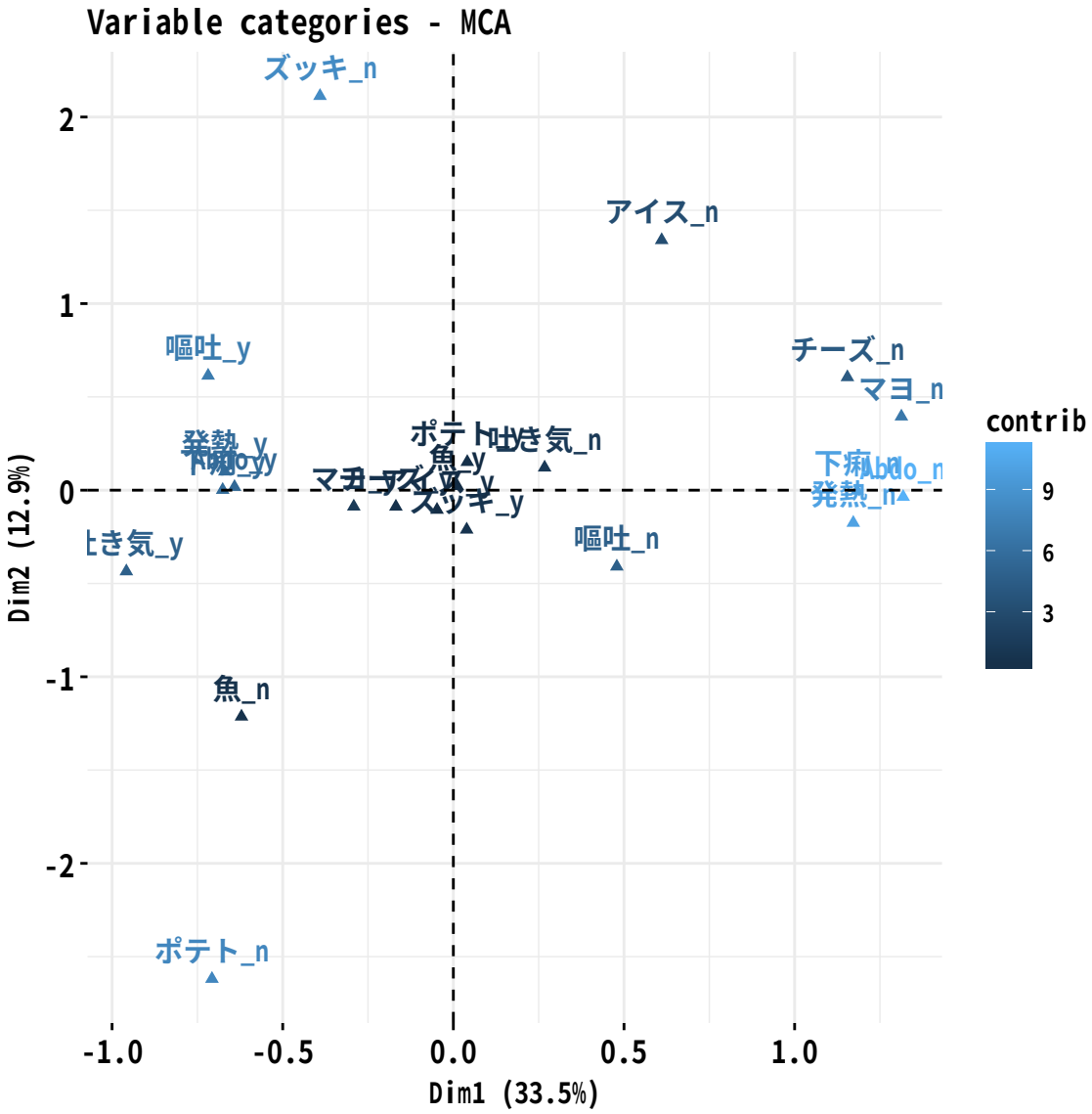


Read more about fviz_contrib(): fviz_contrib

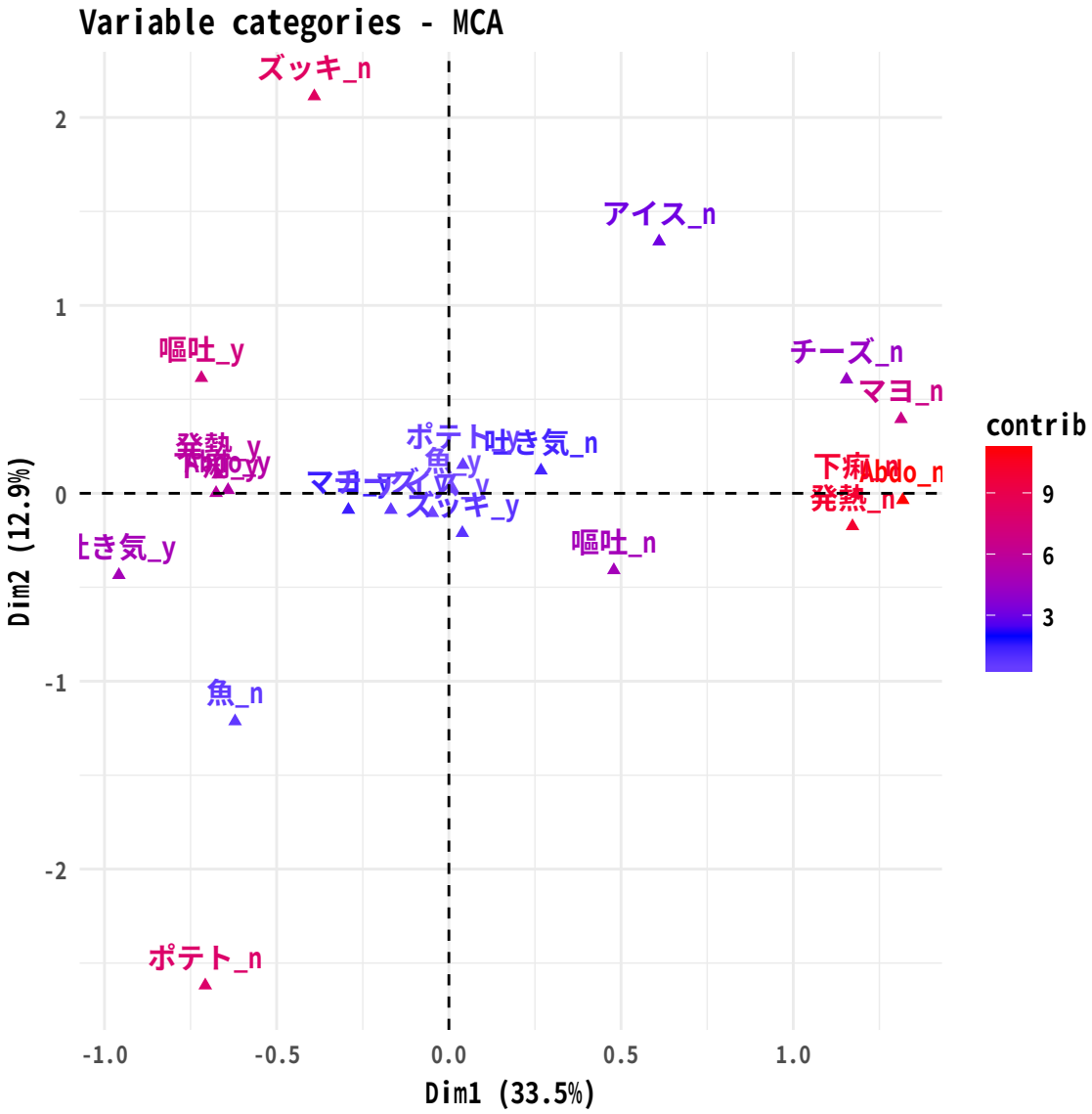
A second option is to draw a scatter plot of categories and to highlight categories according to the amount of their contributions. The function fviz_mca_var() is used.

Note that, using factoextra package, the color or the transparency of the variable categories can be automatically controlled by the value of their contributions, their cos2, their coordinates on x or y axis.

```
# Control category point colors using their contribution
# Possible values for the argument col.row are :
  # "cos2", "contrib", "coord", "x", "y"
fviz_mca_var(res.mca, col.var = "contrib",font.family = "sans")
```




```
# Change the gradient color
fviz_mca_var(res.mca, col.var="contrib", font.family = "sans")+
scale_color_gradient2(low="white", mid="blue",
                      high="red", midpoint=2)+theme_minimal()
```

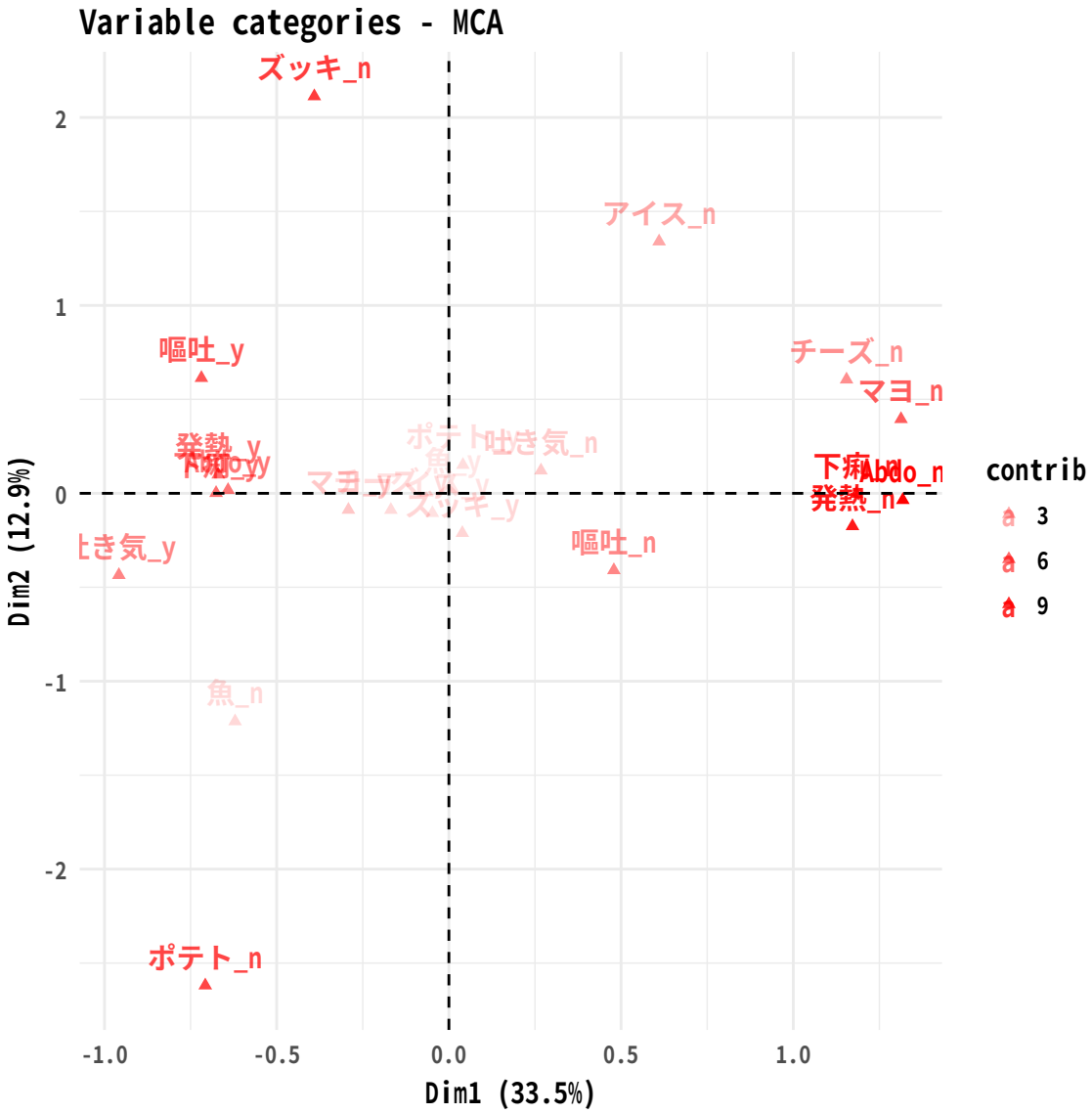


The scatter plot is also helpful to highlight the most important categories in the determination of the dimensions.

In addition we can have an idea of what pole of the dimensions the categories are actually contributing to. It is evident that the categories Abdo_n, Diarrhea_n, Fever_n and Mayo_n have an important contribution to the positive pole of the first dimension, while the categories Fever_y and Diarrhea_y have a major contribution to the negative pole of the first dimension; etc, ….

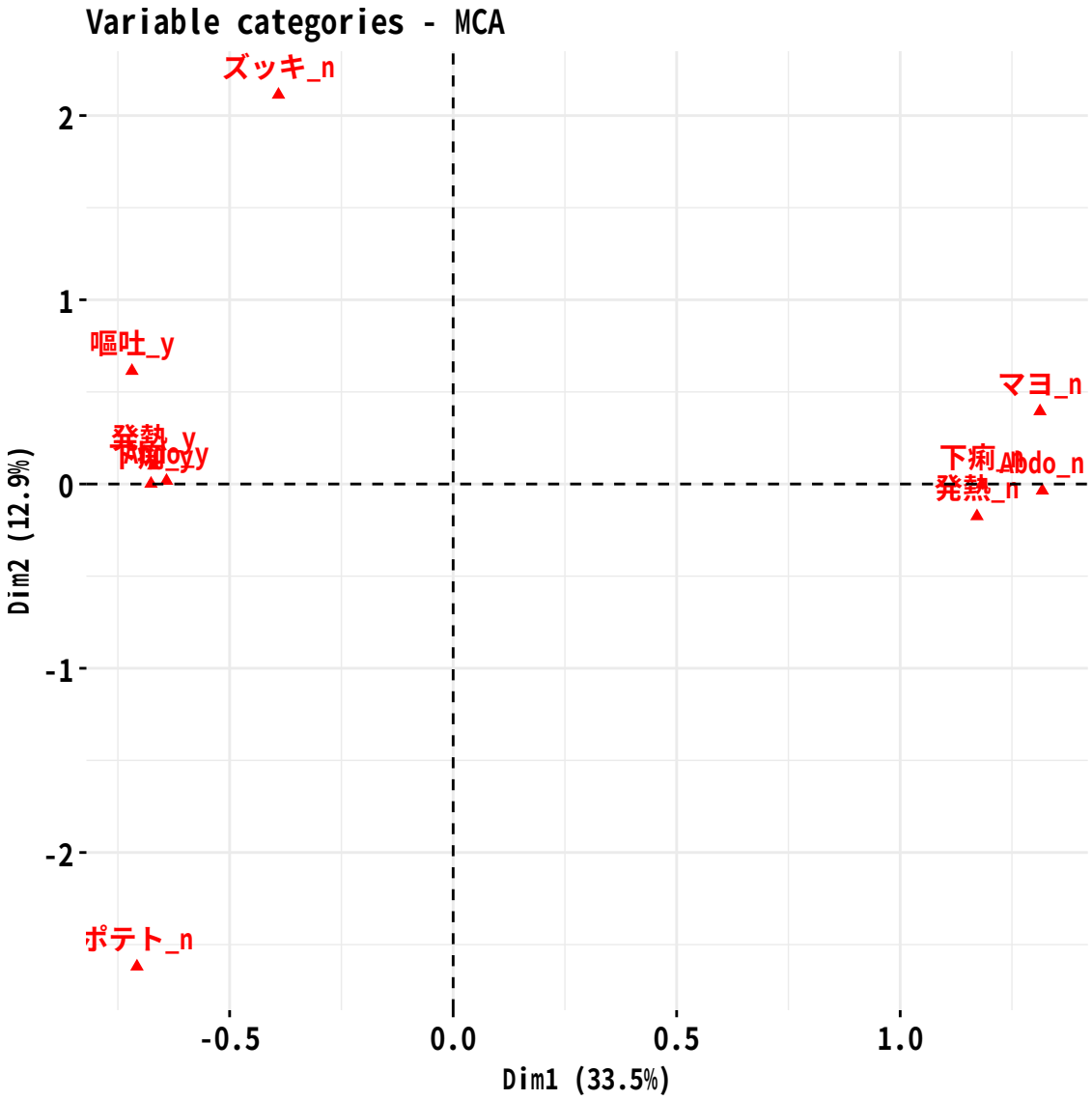
It's also possible to control automatically the transparency of variable categories by their contributions. The argument alpha.var is used:

```
# Control the transparency of categories using their contribution
# Possible values for the argument alpha.var are :
  # "cos2", "contrib", "coord", "x", "y"
fviz_mca_var(res.mca, alpha.var="contrib", font.family = "sans")+
  theme_minimal()
```



It's possible to select and display only the top contributing categories as illustrated in the R code below.

```
# Select the top 10 contributing categories
fviz_mca_var(res.mca, select.var=list(contrib=10), font.family = "sans")
```



Variable category/individual selections are discussed in details in the next sections

Read more about fviz_mca_var(): fviz_mca_var

15 Cos2 : The quality of representation of variable categories

The two dimensions 1 and 2 are sufficient to retain 46% of the total inertia contained in the data.

However, not all the points are equally well displayed in the two dimensions.

The quality of representation of the categories on the factor map is called the squared cosine (cos2) or the squared correlations.

The cos2 measures the degree of association between variable categories and a particular axis.

The cos2 of variable categories can be extracted as follow:

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

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

```
## 吐き気_n 0.2562007 0.0528025759 2.527485e-01 0.004084375 0.019466197
## 吐き気_y 0.2562007 0.0528025759 2.527485e-01 0.004084375 0.019466197
## 嘔吐_n   0.3442016 0.2511603912 1.070855e-02 0.112294813 0.004126898
## 嘔吐_y   0.3442016 0.2511603912 1.070855e-02 0.112294813 0.004126898
## Abdo_n   0.8451157 0.0006215864 1.262496e-05 0.011479077 0.002374929
## Abdo_y   0.8451157 0.0006215864 1.262496e-05 0.011479077 0.002374929
```

The values of the cos2 are comprised between 0 and 1.

The sum of the cos2 for rows on all the MCA dimensions is equal to one.

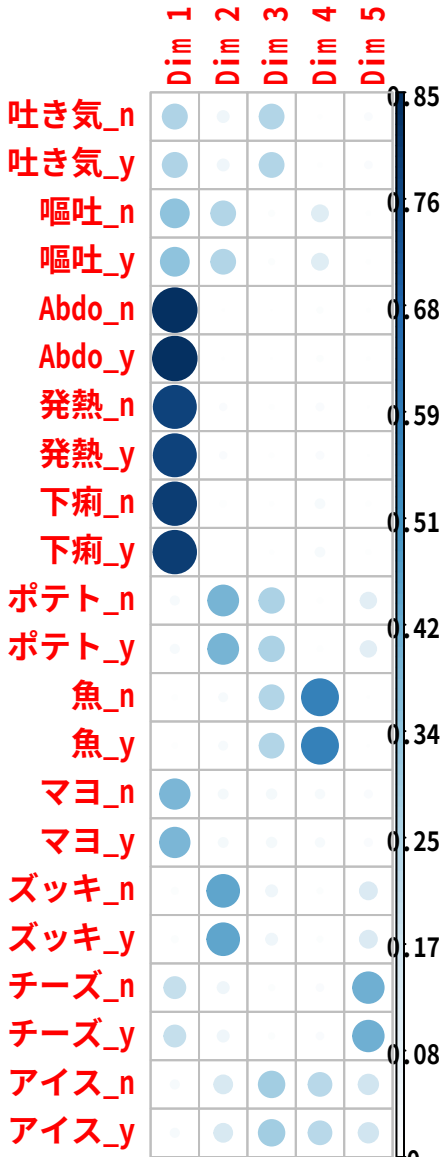
The quality of representation of a variable category or an individual in n dimensions is simply the sum of the squared cosine of that variable category or individual over the n dimensions.

If a variable category is well represented by two dimensions, the sum of the cos2 is closed to one.

For some of the categories, more than 2 dimensions are required to perfectly represent the data.

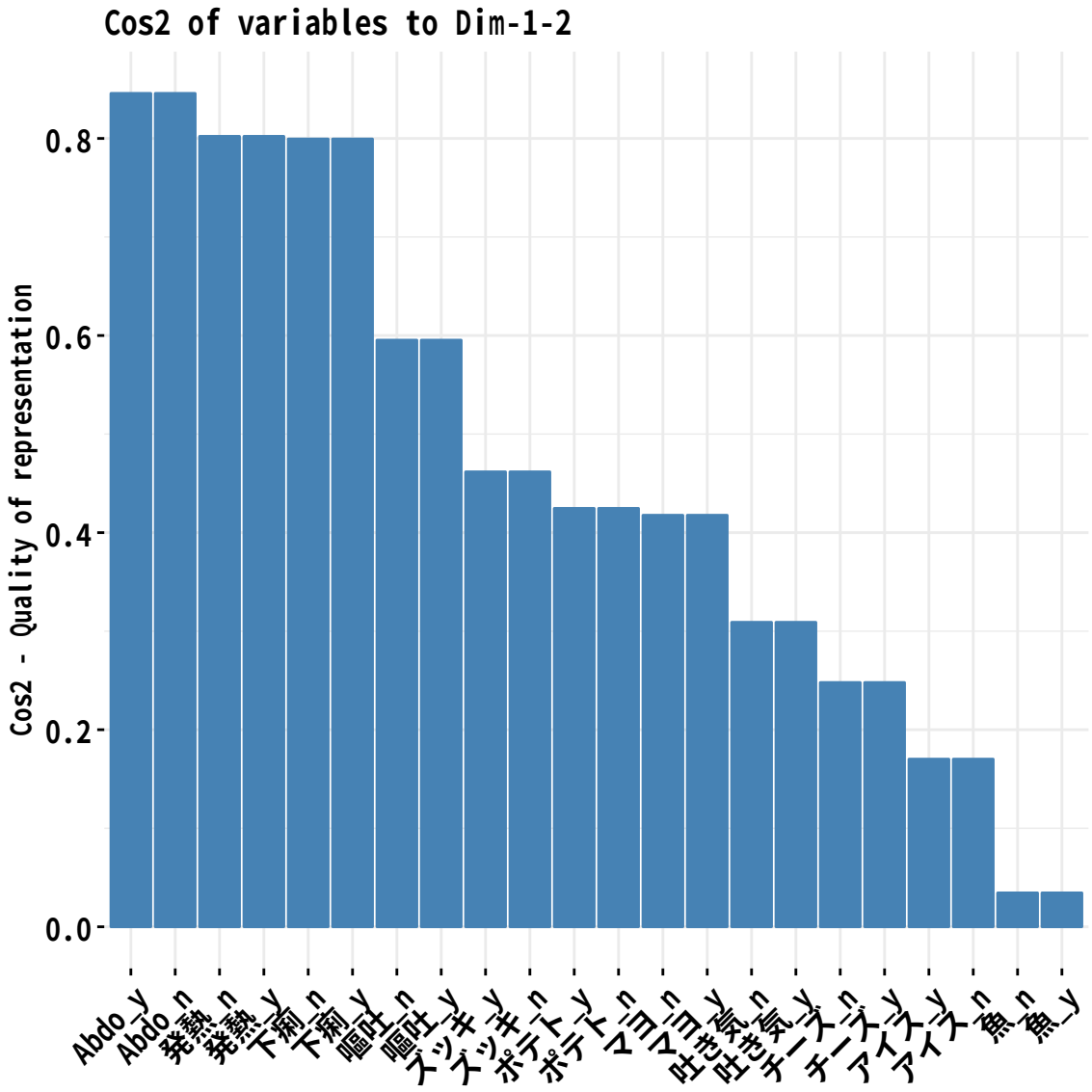
Visualize the cos2 of variable categories using corrplot:

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



The function fviz_cos2()[in factoextra] can be used to draw a bar plot of rows cos2:

```
# Cos2 of variable categories on Dim.1 and Dim.2
fviz_cos2(res.mca, choice = "var", axes = 1:2, font.family = "sans") + theme(text = element_text(family = "sans"))
```



Multiple Correspondence Analysis - R software and data mining

Note that, variable categories Fish_n, Fish_y, Icecream_n and Icecream_y are not very well represented by the first two dimensions. This implies that the position of the corresponding points on the scatter plot should be interpreted with some caution. A higher dimensional solution is probably necessary.

Read more about fviz_cos2(): fviz_cos2 Individuals

The function get_mca_ind()[in factoextra] is used to extract the results for individuals. This function returns a list containing the coordinates, the cos2 and the contributions of individuals:

```
ind <- get_mca_ind(res.mca)
ind
```

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

The result for individuals gives the same information as described for variable categories. For this reason, I'll just

displayed the result for individuals in this section without commenting.

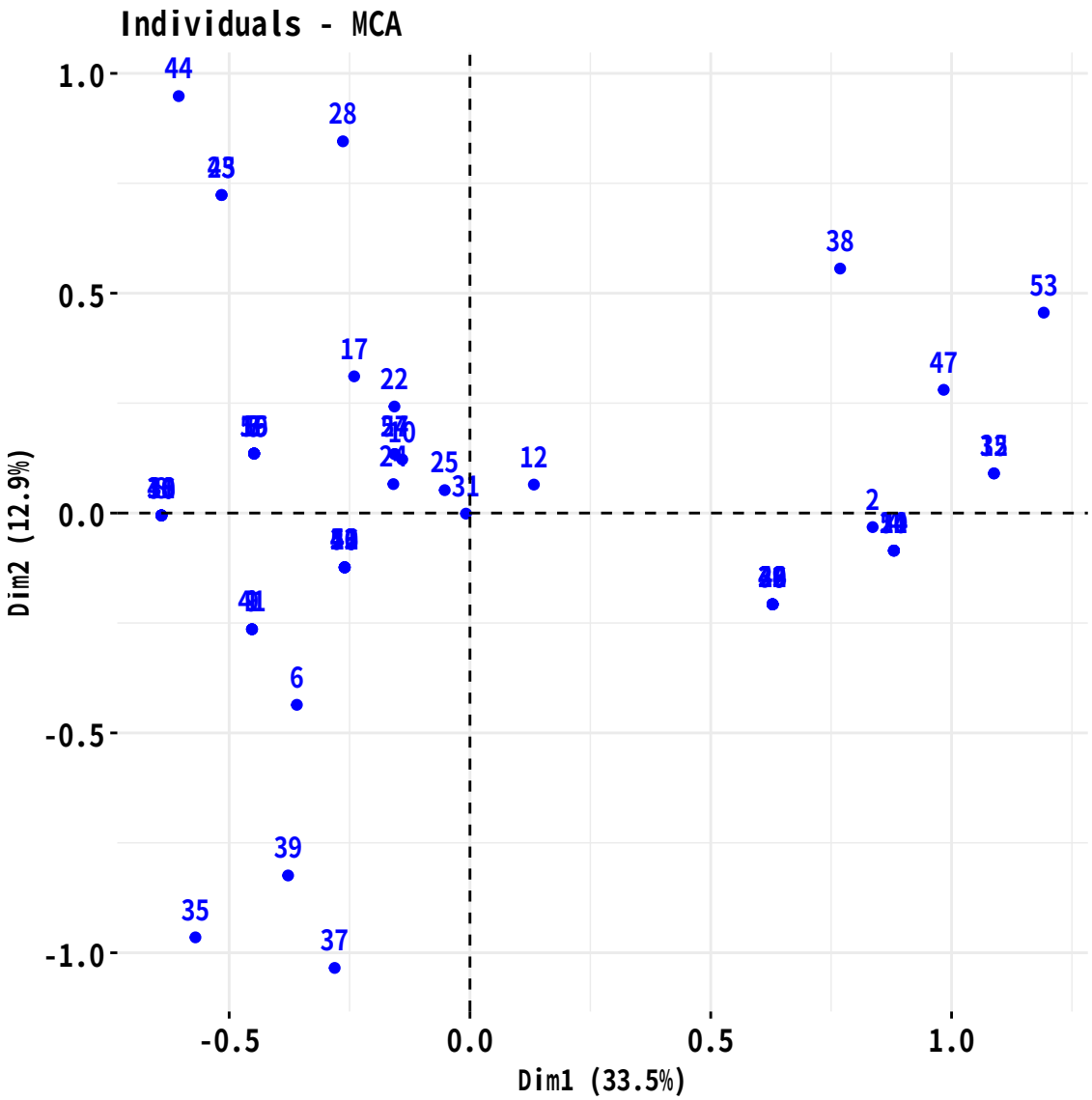
15.1 Coordinates of individuals

head(ind$coord)

	Dim 1	Dim 2	Dim 3	Dim 4	Dim 5
1	-0.4525811	-0.26415072	0.17151614	0.01369348	-0.11696806
2	0.8361700	-0.03193457	-0.07208249	-0.08550351	0.51978710
3	-0.4481892	0.13538726	-0.22484048	-0.14170168	-0.05004753
4	0.8803694	-0.08536230	-0.02052044	-0.07275873	-0.22935022
5	-0.4481892	0.13538726	-0.22484048	-0.14170168	-0.05004753
6	-0.3594324	-0.43604390	-1.20932223	1.72464616	0.04348157

Use the function fviz_mca_ind() [in factoextra] to visualize only column points:

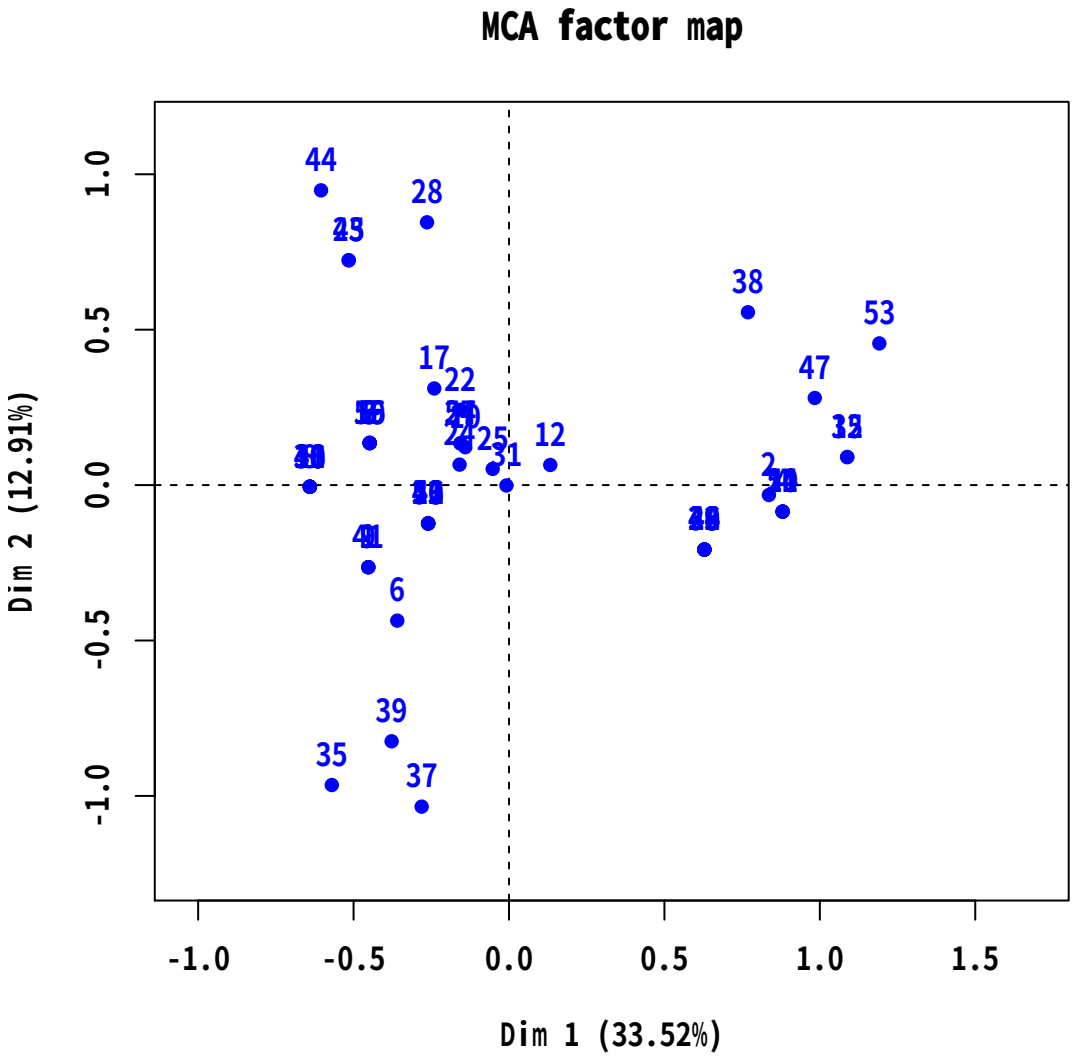
```
fviz_mca_ind(res.mca, font.family = "sans")
```



Read more about fviz_mca_ind(): fviz_mca_ind

Note that, it's also possible to make the graph of individuals only using FactoMineR base graph.The argument invisible is used to hide the variable categories on the factor map:

```
# Hide variable categories
plot(res.mca, invisible="var")
```



15.2 Contribution of individuals to the dimensions

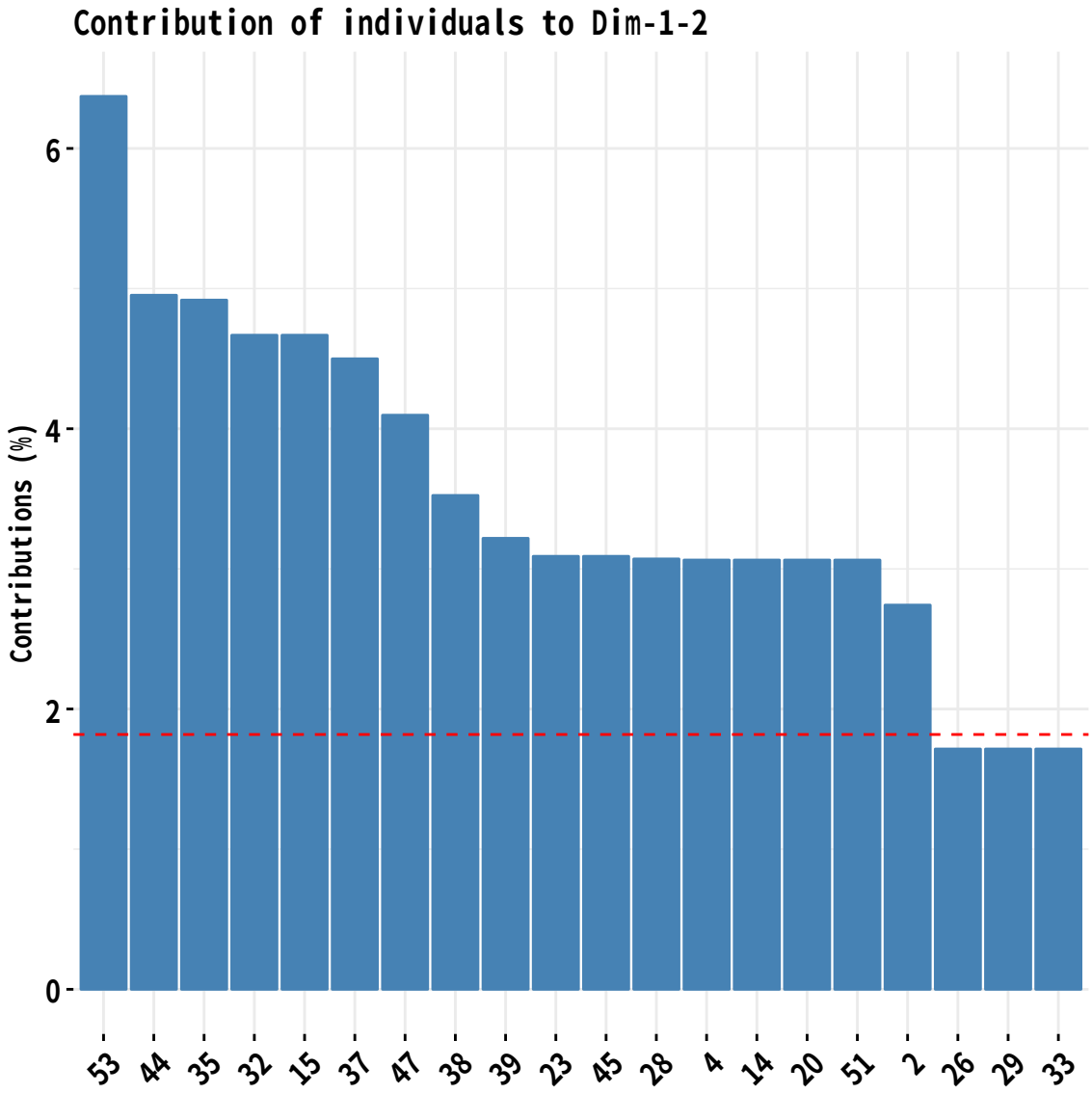
```
head(ind$contrib)
```

```
##      Dim 1      Dim 2       Dim 3       Dim 4      Dim 5
## 1 1.110927 0.98238297  0.498254685  0.003555817 0.31554778
## 2 3.792117 0.01435818  0.088003703  0.138637089 6.23134138
## 3 1.089470 0.25806722  0.856229950  0.380768961 0.05776914
## 4 4.203611 0.10259105  0.007132055  0.100387990 1.21319013
## 5 1.089470 0.25806722  0.856229950  0.380768961 0.05776914
## 6 0.700692 2.67693398 24.769968729 56.404214518 0.04360547
```

Note that, you can use the previously mentioned corrplot() function to visualize the contribution of individuals.

Use the function fviz_contrib()[in factoextra] to visualize column contributions on dimensions 1+2:

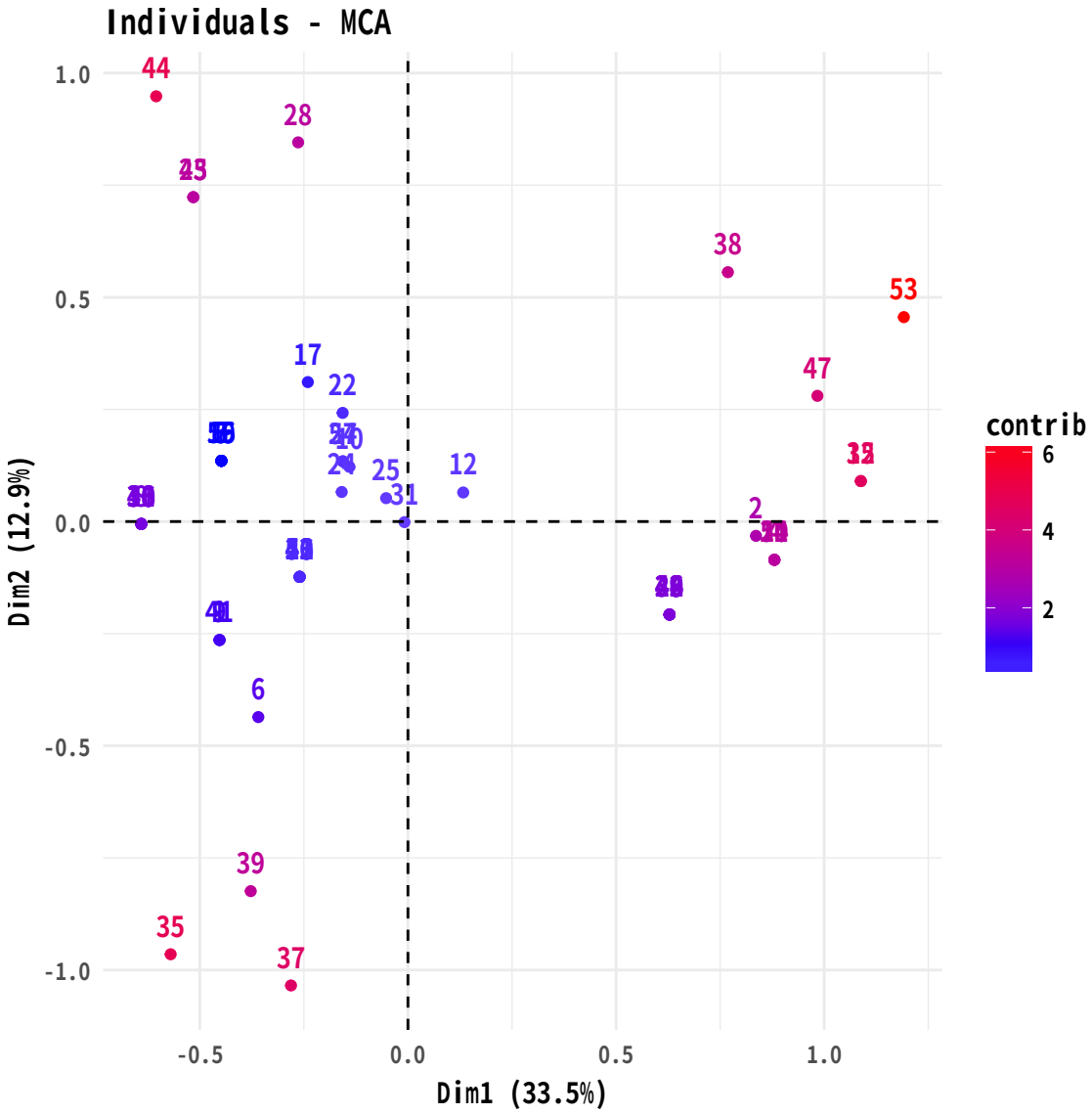
```
fviz_contrib(res.mca, choice = "ind", axes = 1:2, top = 20, font.family = "sans")
```



- If the individual contributions were uniform, the expected value would be 1/nrow(poison) = 1/55 = 1.8%.
- The expected average contribution (reference line) of a column for Dim.1 and Dim.2 is : (1.8 * Eig1) + (1.8 * Eig2) = (1.8 * 0.34) + (1.8 * 0.13) = 0.85%.

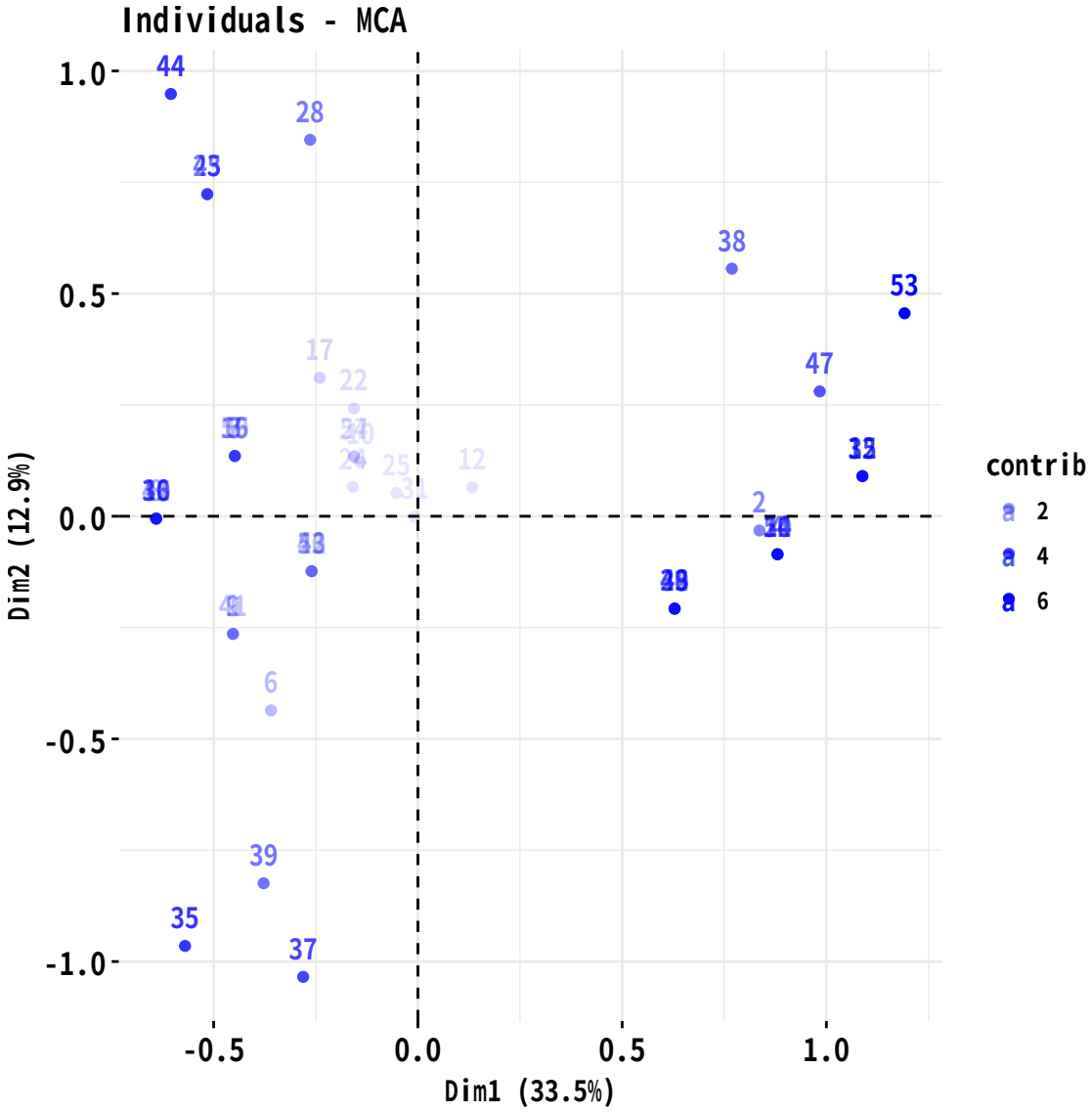
Draw a scatter plot of individuals points and highlight individuals according to the amount of their contributions. The function fviz_mca_ind() [in factoextra] is used:

```
# Control individual colors using their contribution
# Possible values for the argument col.ind are :
  # "cos2", "contrib", "coord", "x", "y"
fviz_mca_ind(res.mca, col.ind="contrib", font.family = "sans")+
scale_color_gradient2(low="white", mid="blue",
                      high="red", midpoint=0.85)+theme_minimal()
```



Note that, it's also possible to control automatically the transparency of individuals by their contributions using the argument alpha.ind:

```
# Control the transparency of individuals using their contribution
# Possible values for the argument alpha.ind are :
  # "cos2", "contrib", "coord", "x", "y"
fviz_mca_ind(res.mca, alpha.ind="contrib", font.family = "sans")
```



15.3 Cos2 : The quality of representation of individuals

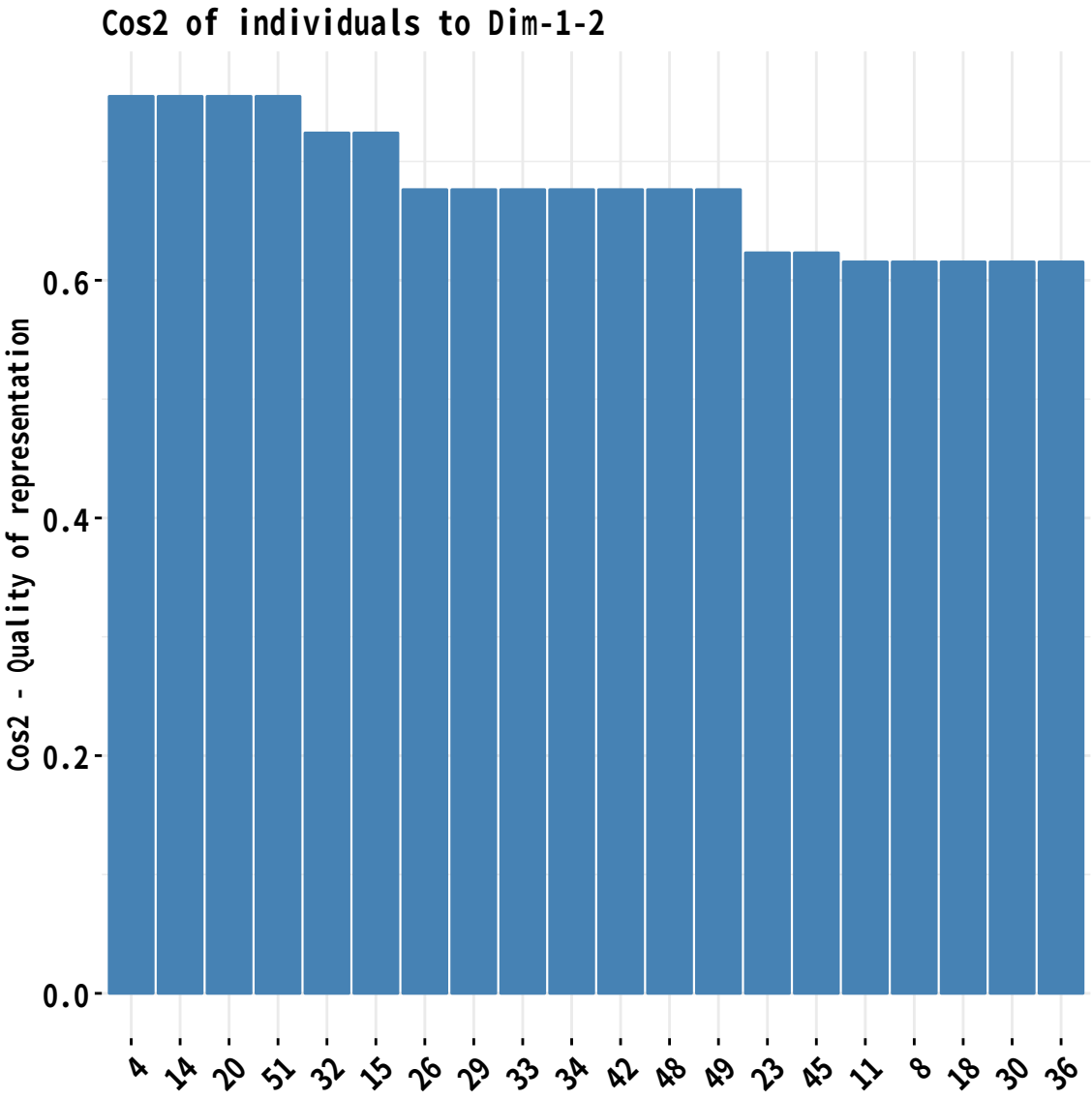
```
head(ind$cos2)
```

```
##        Dim 1        Dim 2        Dim 3        Dim 4        Dim 5
## 1 0.34652591 0.1180447167 0.0497683175 0.0003172275 0.0231460846
## 2 0.55589562 0.0008108236 0.0041310808 0.0058126211 0.2148103098
## 3 0.54813888 0.0500176790 0.1379484860 0.0547920948 0.0068349171
## 4 0.74773962 0.0070299584 0.0004062504 0.0051072923 0.0507479873
## 5 0.54813888 0.0500176790 0.1379484860 0.0547920948 0.0068349171
## 6 0.02485357 0.0365775483 0.2813443706 0.5722083217 0.0003637178
```

Note that, the value of the cos2 is between 0 and 1. A cos2 closed to 1 corresponds to a variable categories/ individuals that are well represented on the factor map.

The function fviz_cos2()[in factoextra] can be used to draw a bar plot of individuals cos2:

```
# Cos2 of individuals on Dim.1 and Dim.2
fviz_cos2(res.mca, choice = "ind", axes = 1:2, top = 20, font.family = "sans")
```



15.4 Change the color of individuals by groups

As mentioned above, our data contains supplementary qualitative variables: Columns 3 and 4 corresponding to the columns Sick and Sex, respectively. These factor variables will be used to color individuals by groups.

```
sick <- as.factor(poison$発症)
head(sick)
```

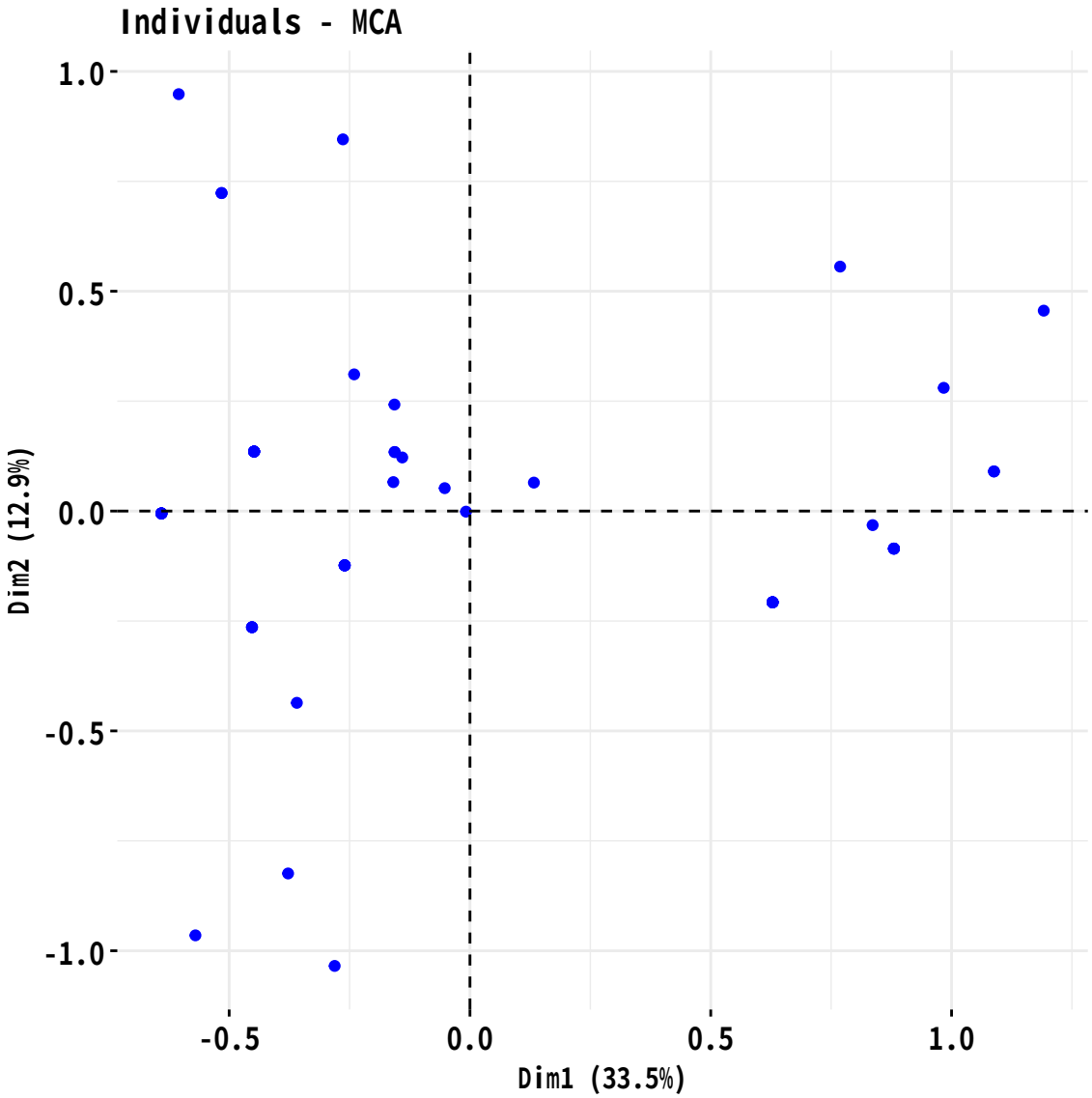
```
## [1] 発症_y 発症_n 発症_y 発症_n 発症_y 発症_y
## Levels: 発症_n 発症_y
```

```
sex <- as.factor(poison$性別)
head(sex)
```

```
## [1] 女性 女性 女性 女性 男性 男性
## Levels: 女性 男性
```

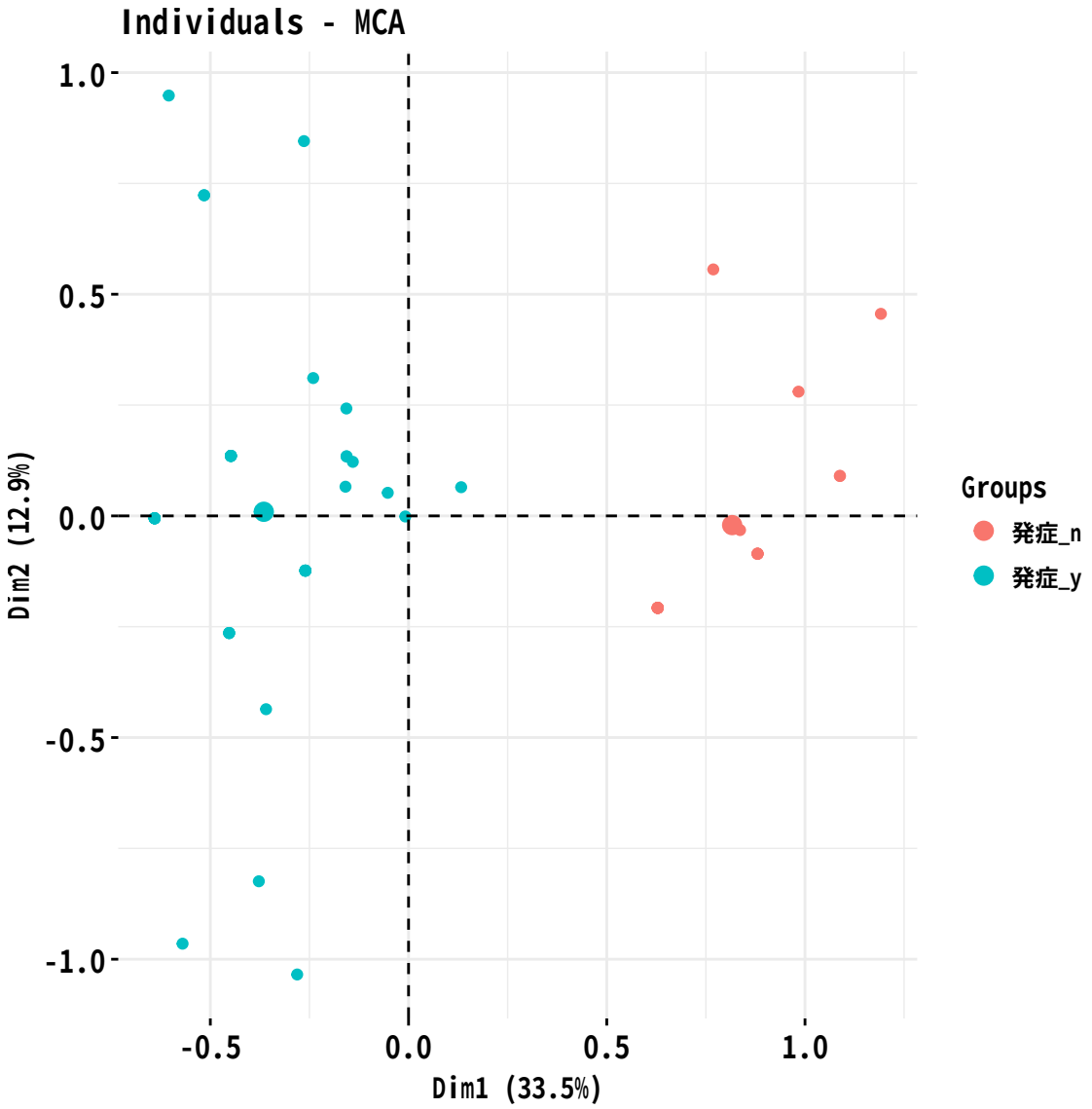
Individuals factor map :

```
# Default plot
fviz_mca_ind(res.mca, label ="none", font.family = "sans")
```



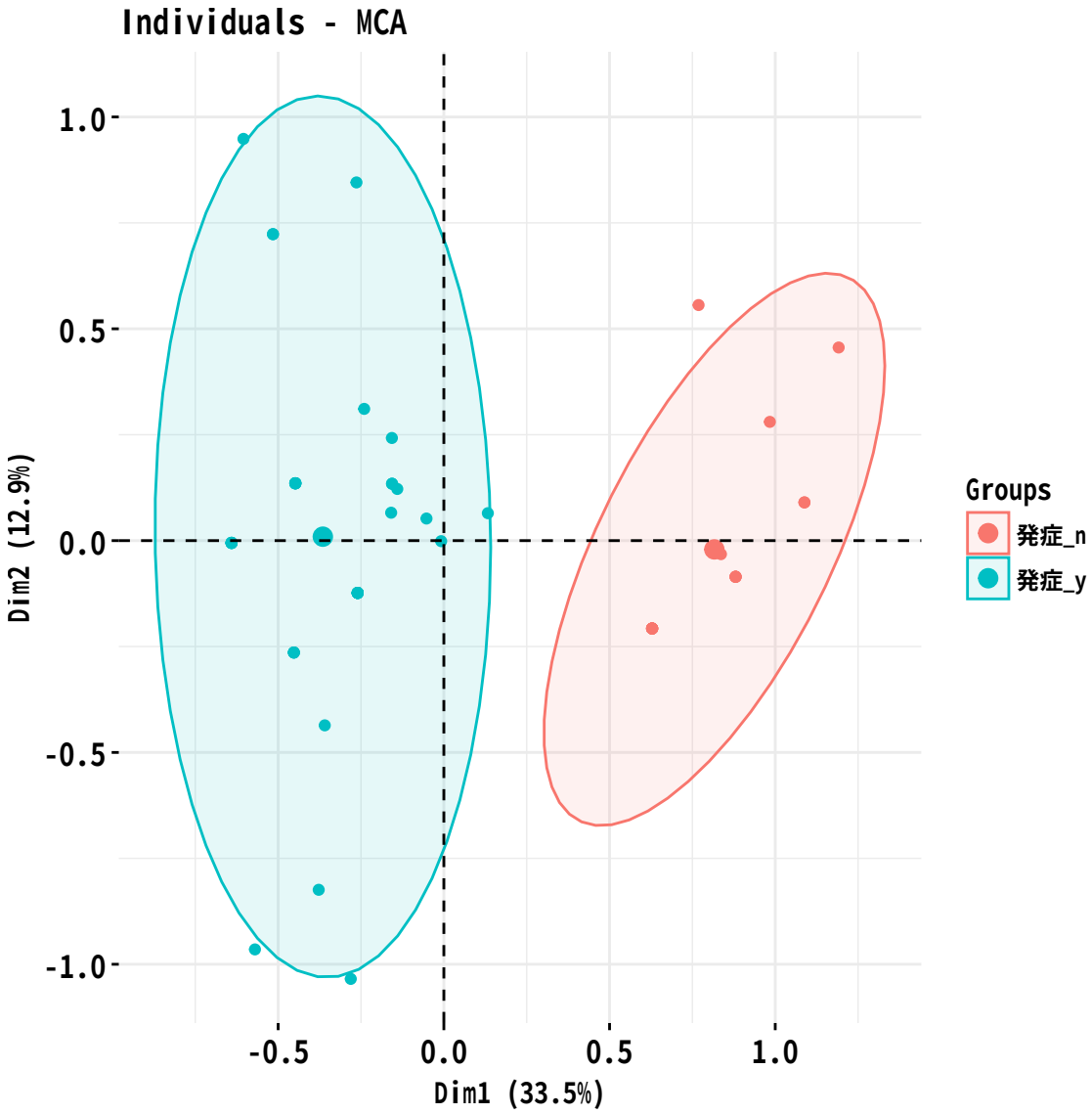
Change individual colors by groups using the levels of the variable sick. The argument habillage is used:

```
fviz_mca_ind(res.mca, label = "none", habillage=sick, font.family = "sans")
```



Add ellipses of point concentrations : the argument habillage is used to specify the factor variable for coloring the observations by groups.

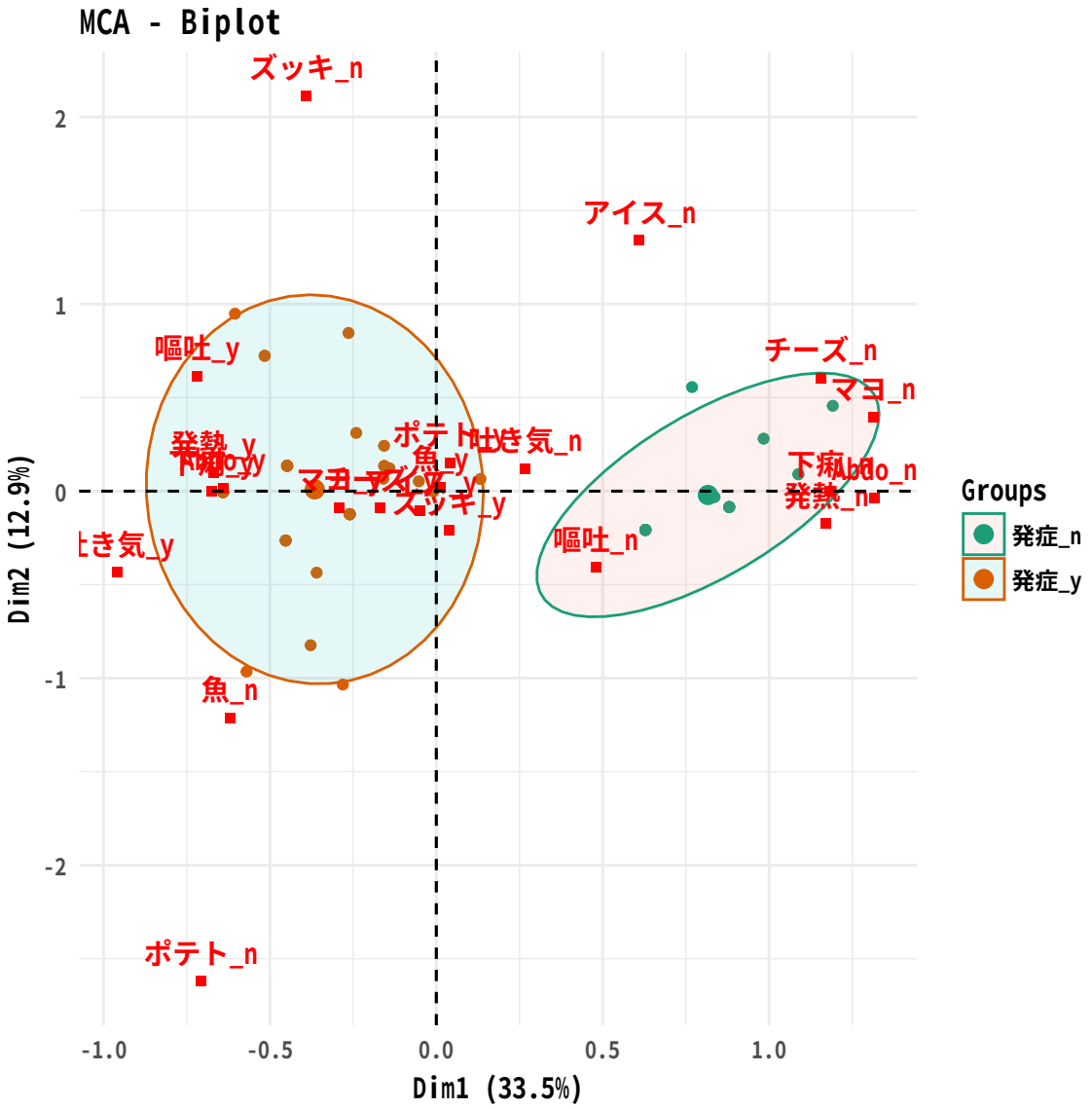
```
fviz_mca_ind(res.mca, label="none", habillage = sick,
             addEllipses = TRUE, ellipse.level = 0.95, font.family = "sans")
```



Now, let's :

- make a biplot of individuals and variable categories
- change the color of individuals by groups (sick levels)
- show only the labels for variables

```
fviz_mca_biplot(res.mca,
  habillage = sick, addEllipses = TRUE,
  label = "var", shape.var = 15, font.family = "sans") +
  scale_color_brewer(palette="Dark2")+
  theme_minimal() + theme(text = element_text(family = "sans"))
```



Error::active個体の数(52)とhabillage のfactor の長さ(55)が異なっている。legend で日本語が□になるのはフォントの問題ではなく、このエラーでひっかかってる可能性大。.add_ind_groups(X, df, habillage) でエラー: The number of active individuals is different from the length of the factor habillage. Please, remove the supplementary individuals in the variable habillage. sick[1:52] にして　+ theme(text = element_text(family = “sans”))　を指定したらOKとなった。

```
length(sick)
```

```
## [1] 55
```

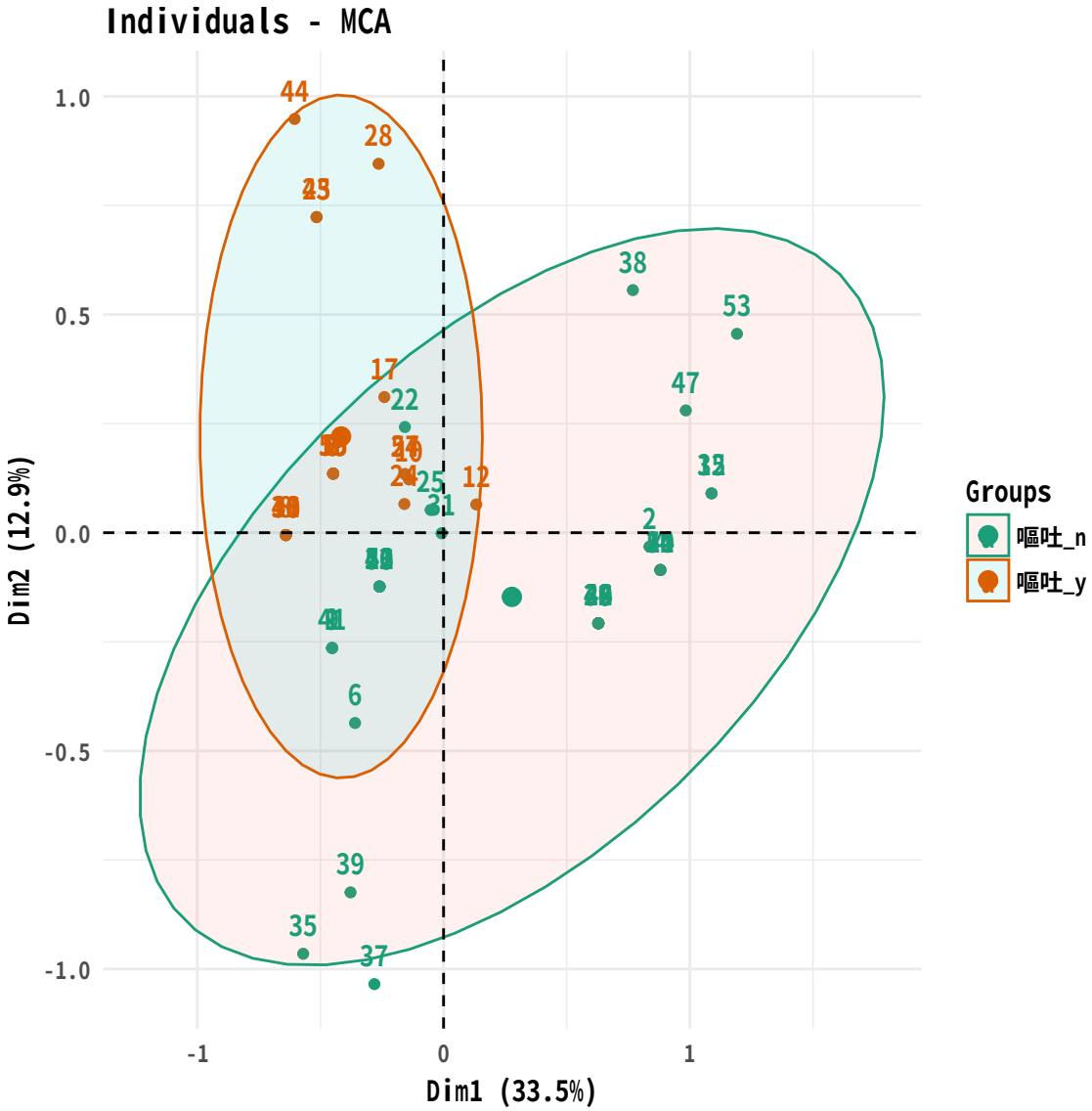
```
tail(res.mca$ind$coord)
```

```
##         Dim 1      Dim 2       Dim 3       Dim 4       Dim 5
## 50 -0.2601554 -0.1234026 -0.16623084  0.05912373 -0.00758947
## 51  0.8803694 -0.0853623 -0.02052044 -0.07275873 -0.22935022
## 52 -0.2601554 -0.1234026 -0.16623084  0.05912373 -0.00758947
## 53  1.1914531  0.4558336  0.64328804  0.59070046 -0.11487151
## 54 -0.1563462  0.1342913 -0.26102115 -0.25299536 -0.10342634
## 55 -0.4481892  0.1353873 -0.22484048 -0.14170168 -0.05004753
```

Note that, it’s possible to color the individuals using any of the qualitative variable in the initial data table (poison)

Let's color the individuals by groups using the levels of the variable Vomiting:

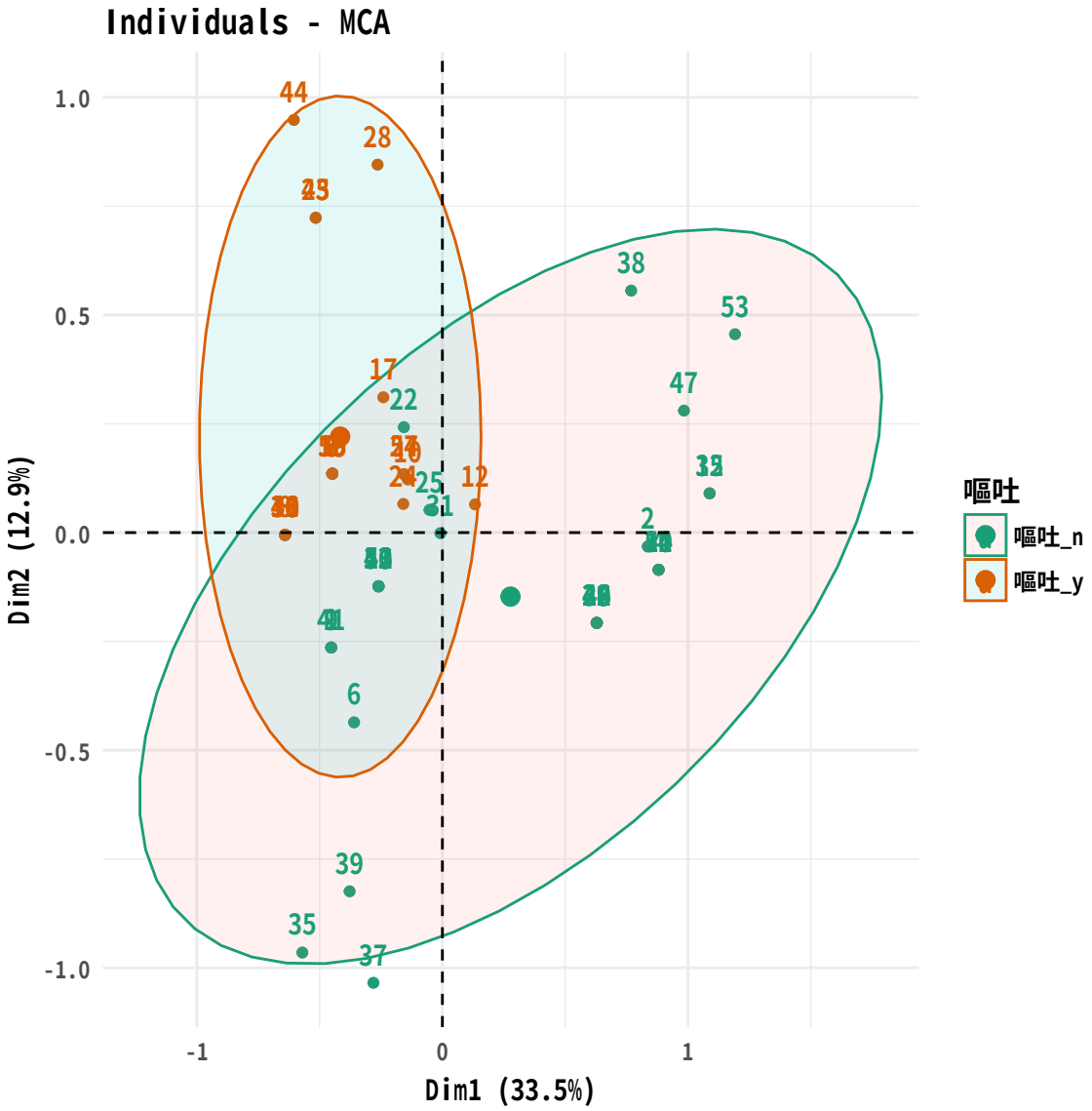
```
fviz_mca_ind(res.mca,
  habillage = poison$嘔吐, addEllipses = TRUE, font.family = "sans") +
  scale_color_brewer(palette="Dark2")+
  theme_minimal() + theme(text = element_text(family = "sans"))
```



It's also possible to use the index of the column as follow (habillage = 2):

列番号で指定することもできる。

```
fviz_mca_ind(res.mca,  # habillage = 2 (嘔吐)
  habillage = 2, addEllipses = TRUE,font.family = "sans") +
  scale_color_brewer(palette="Dark2")+
  theme_minimal() + theme(text = element_text(family = "sans"))
```




```
colnames(poison)
```

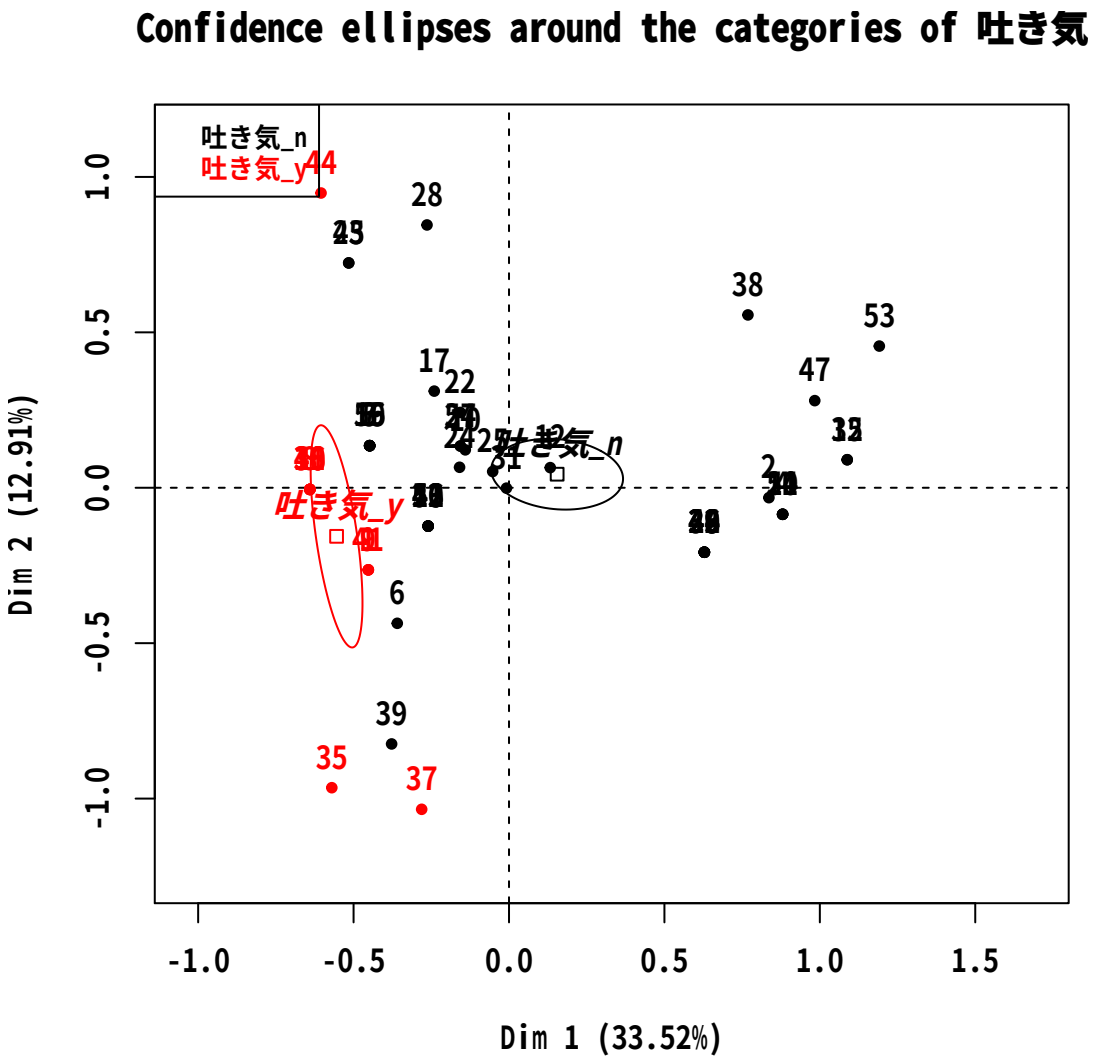
```
##  [1] "年齢"          "時刻"          "発症"          "性別"
##  [5] "吐き気"        "嘔吐"          "腹痛"          "発熱"
##  [9] "下痢"          "ポテト"        "魚"            "マヨ"
## [13] "ズッキーニ"    "チーズ"        "アイスクリーム"
```

You can also use the function plotellipses() [in FactoMineR] to draw confidence ellipses around the categories. The simplified format is:

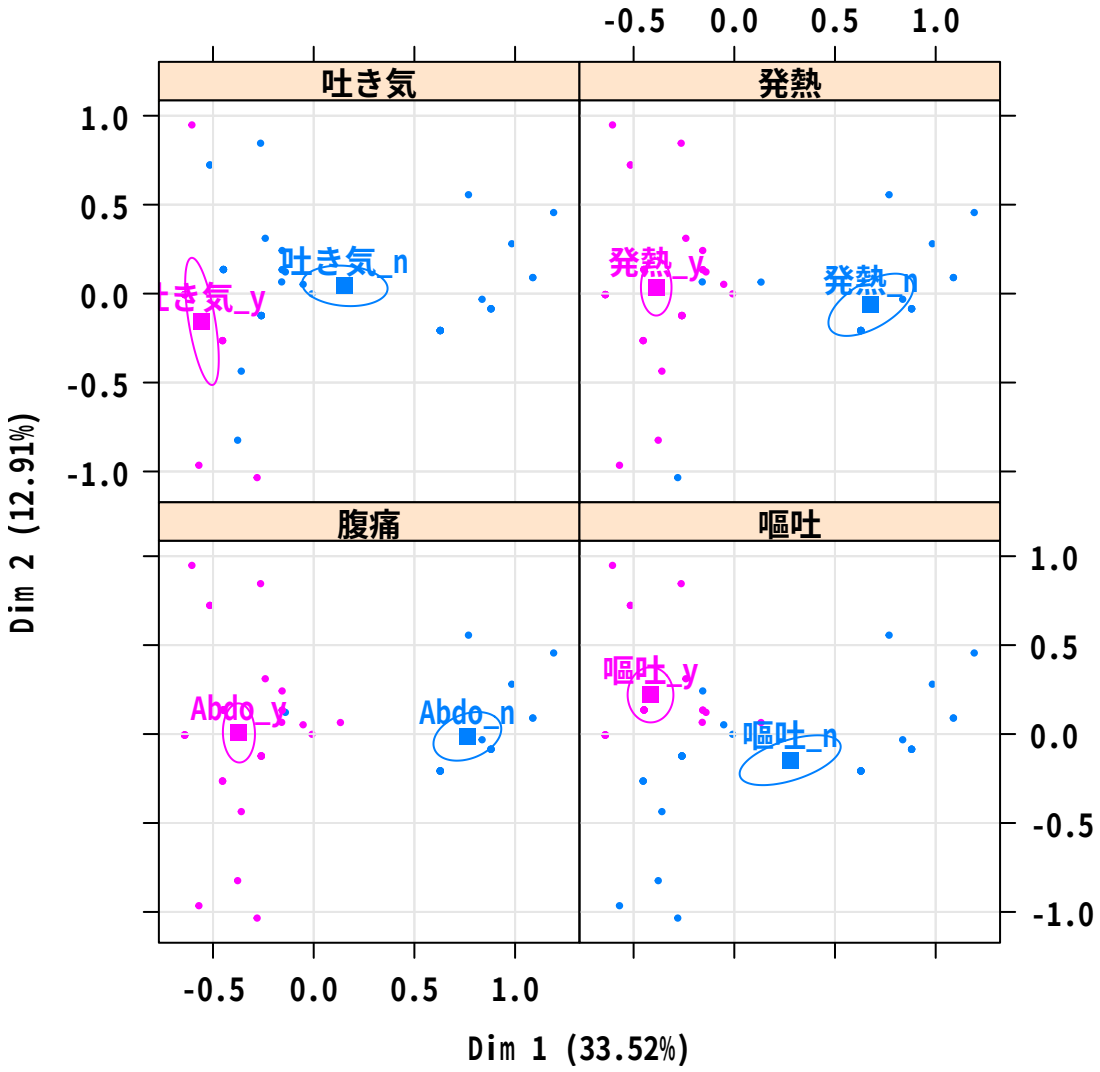
```
plotellipses(model, keepvar="all", axis =c(1,2))
```

- model: object of class MCA or PCA
- keppvar: a boolean or numeric vector of indexes of variables or a character vector of names of variables. If keepvar is "all", "quali" or "quali.sup", variables which are plotted are all the categorical variables, only those which are used to compute the dimensions (active variables) or only the supplementary categorical variables. If keepvar is a numeric vector of indexes or a character vector of names of variables, only relevant variables are plotted.

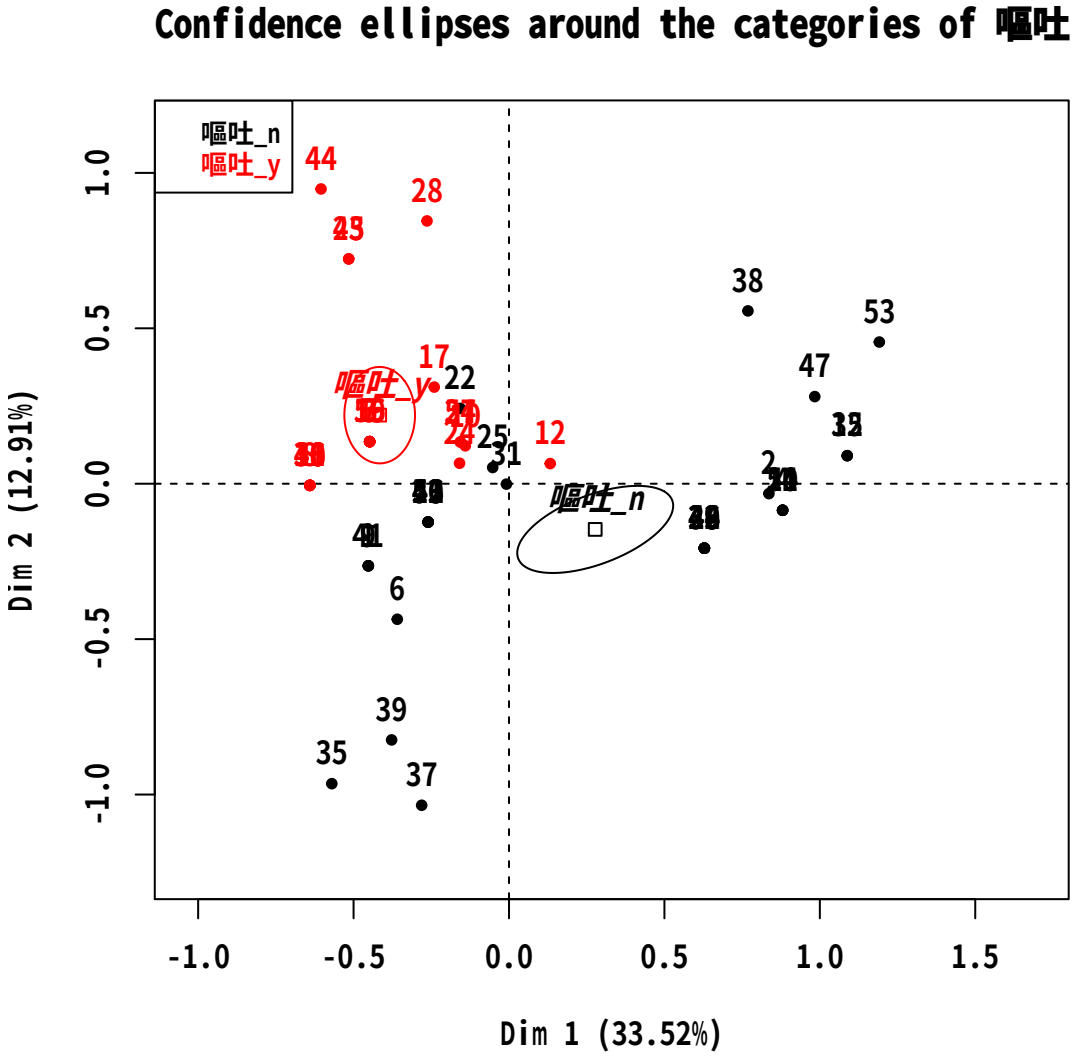
```
plotellipses(res.mca, keepvar=1)
```



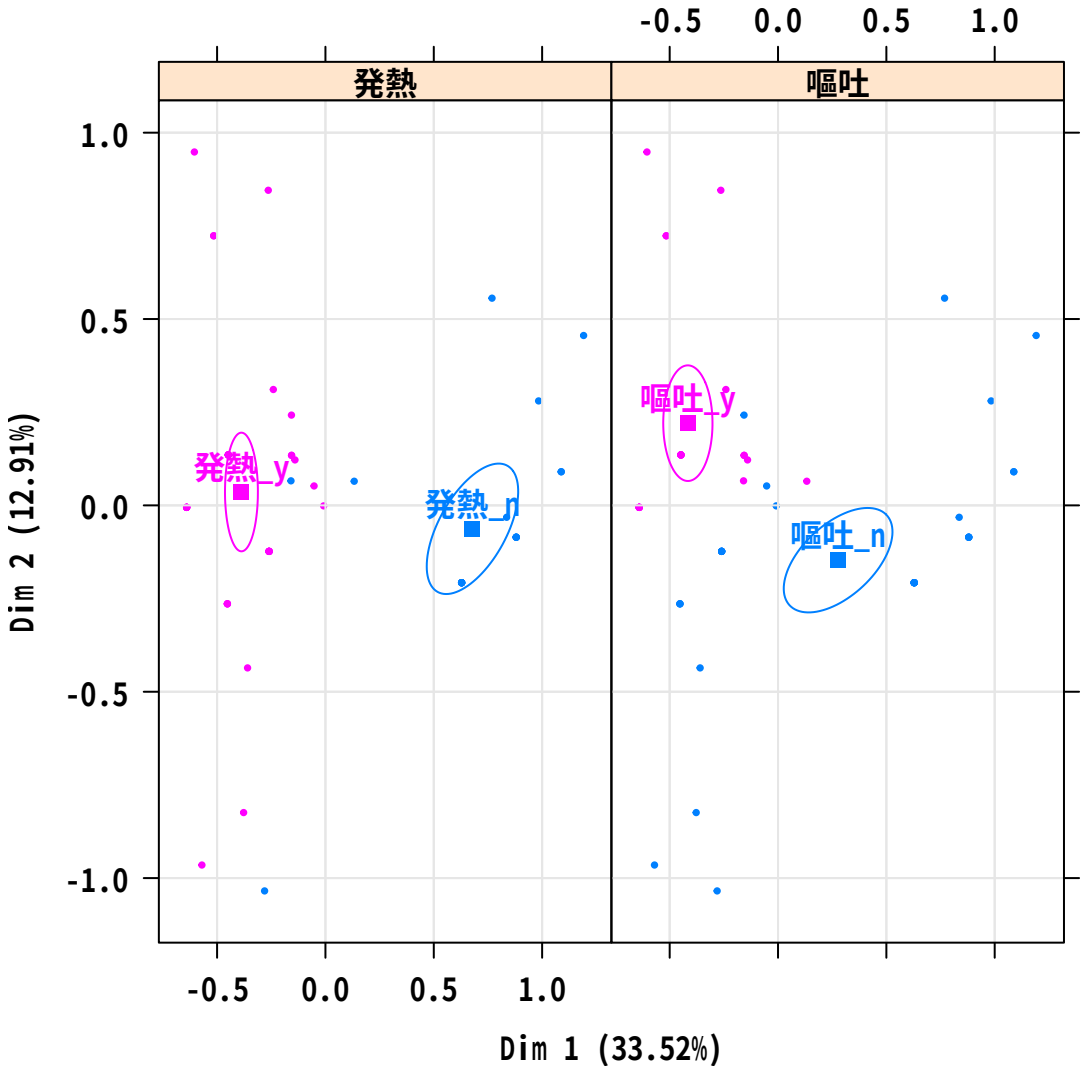

```
plotellipses(res.mca, keepvar=1:4)
```



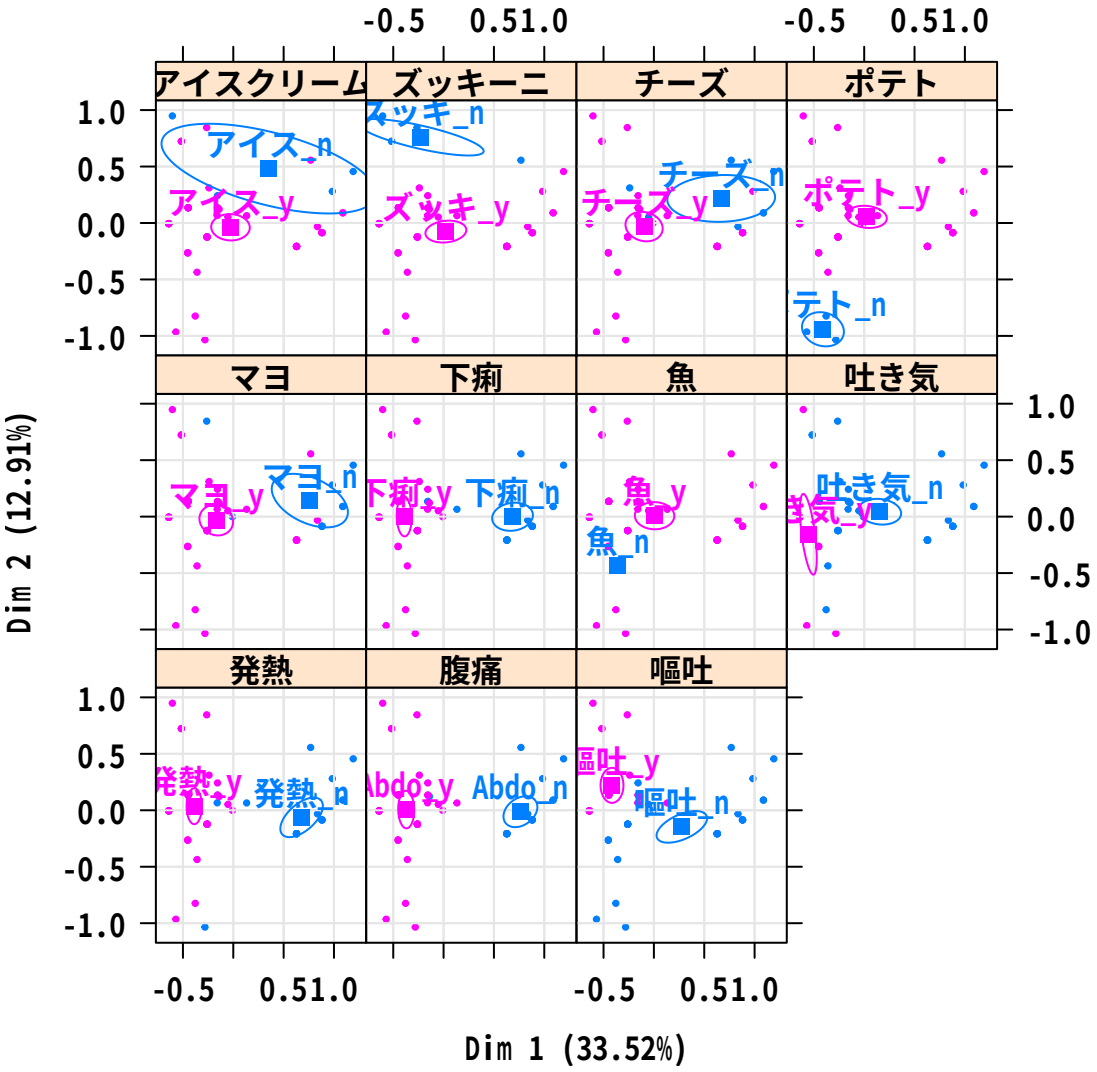

```
plotellipses(res.mca, keepvar="嘔吐")
```




```
plotellipses(res.mca, keepvar=c("嘔吐", "発熱"))
```




```
plotellipses(res.mca, keepvar="all")
```



16 MCA using supplementary individuals and variables

As described above, the data set poison contains:

- supplementary continuous variables (quanti.sup = 1:2, columns 1 and 2 corresponding to the columns Sick and Sex, respectively)
- supplementary qualitative variables (quali.sup = 3:4, corresponding to the columns Sick and Sex, respectively). This factor variables are used to color individuals by groups

The data doesn't contain supplementary individuals. However for demonstration, we'll use the individuals 53:55 as supplementary individuals. The coordinates of these individuals will be predicted from the parameters of the MCA on the active individuals (1:52)

Supplementary variables and individuals are not used for the determination of the principal dimensions. Their coordinates are predicted using only the information provided by the performed multiple correspondence analysis on active variables/individuals.

To specify supplementary individuals and variables, the function MCA() can be used as follow :

```
MCA(X,  ncp = 5, ind.sup = NULL,
    quanti.sup=NULL, quali.sup=NULL, graph=TRUE, axes = c(1,2))
```

- X : a data frame. Rows are individuals and columns are variables.
- ncp : number of dimensions kept in the final results.
- ind.sup : a numeric vector specifying the indexes of the supplementary individuals
- quanti.sup, quali.sup : a numeric vector specifying, respectively, the indexes of the quantitative and qualitative variables
- graph : a logical value. If TRUE a graph is displayed.
- axes : a vector of length 2 specifying the components to be plotted

Example of usage :

```
res.mca <- MCA(poison, ind.sup=53:55,
               quanti.sup = 1:2, quali.sup = 3:4,  graph=FALSE)
```

The summary of the MCA is :

```
summary(res.mca, nb.dec = 2, ncp = 2)
```

```
##
## Call:
## MCA(X = poison, ind.sup = 53:55, quanti.sup = 1:2, quali.sup = 3:4,
##      graph = FALSE)
##
##
## Eigenvalues
##                        Dim.1  Dim.2  Dim.3  Dim.4  Dim.5  Dim.6  Dim.7
## Variance                0.33   0.13   0.11   0.10   0.09   0.07   0.06
## % of var.              32.88  13.04  10.63   9.67   8.60   6.66   6.40
## Cumulative % of var.   32.88  45.92  56.56  66.23  74.83  81.49  87.89
##                        Dim.8  Dim.9 Dim.10 Dim.11
## Variance                0.06   0.04   0.01   0.01
## % of var.               5.94   3.89   1.33   0.95
## Cumulative % of var.   93.83  97.72  99.05 100.00
##
## Individuals (the 10 first)
##          Dim.1   ctr  cos2   Dim.2   ctr  cos2
## 1      | -0.44  1.14  0.35 | -0.27  1.10  0.13 |
## 2      |  0.85  4.23  0.54 | -0.01  0.00  0.00 |
## 3      | -0.43  1.09  0.50 |  0.13  0.24  0.04 |
## 4      |  0.91  4.81  0.77 | -0.03  0.01  0.00 |
## 5      | -0.43  1.09  0.50 |  0.13  0.24  0.04 |
## 6      | -0.34  0.67  0.02 | -0.45  2.93  0.04 |
## 7      | -0.43  1.09  0.50 |  0.13  0.24  0.04 |
## 8      | -0.63  2.32  0.61 | -0.02  0.00  0.00 |
## 9      | -0.44  1.14  0.35 | -0.27  1.10  0.13 |
## 10     | -0.12  0.08  0.03 |  0.14  0.27  0.04 |
##
## Supplementary individuals
##          Dim.1  cos2   Dim.2  cos2
## 53     |  1.08  0.36 |  0.52  0.08 |
## 54     | -0.12  0.03 |  0.14  0.04 |
## 55     | -0.43  0.50 |  0.13  0.04 |
##
## Categories (the 10 first)
##            Dim.1   ctr  cos2 v.test   Dim.2   ctr  cos2 v.test
## 吐き気_n |  0.29  1.78  0.28   3.77 |  0.13  0.94  0.06   1.72 |
## 吐き気_y | -0.97  5.94  0.28  -3.77 | -0.44  3.12  0.06  -1.72 |
```

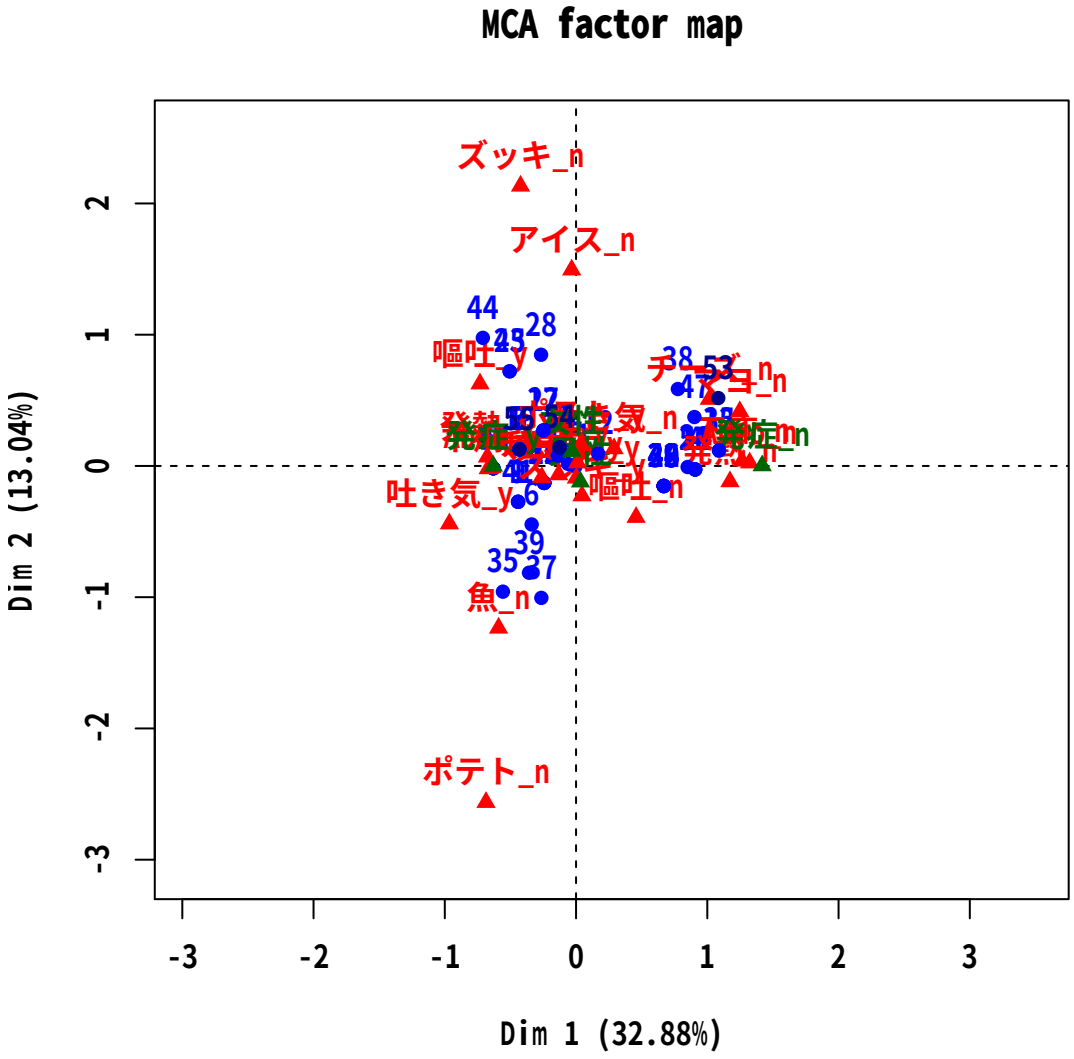
```
## 嘔吐_n   |  0.46  3.56  0.33   4.13 | -0.39  6.57  0.24  -3.53 |
## 嘔吐_y   | -0.73  5.70  0.33  -4.13 |  0.63 10.51  0.24   3.53 |
## Abdo_n   |  1.32 15.80  0.85   6.58 |  0.02  0.01  0.00   0.12 |
## Abdo_y   | -0.64  7.68  0.85  -6.58 | -0.01  0.01  0.00  -0.12 |
## 発熱_n   |  1.17 13.89  0.79   6.35 | -0.12  0.36  0.01  -0.65 |
## 発熱_y   | -0.68  8.00  0.79  -6.35 |  0.07  0.21  0.01   0.65 |
## 下痢_n   |  1.26 15.31  0.85   6.57 |  0.04  0.04  0.00   0.20 |
## 下痢_y   | -0.67  8.10  0.85  -6.57 | -0.02  0.02  0.00  -0.20 |
##
## Categorical variables (eta2)
##              Dim.1 Dim.2
## 吐き気     |  0.28  0.06 |
## 嘔吐       |  0.33  0.24 |
## 腹痛       |  0.85  0.00 |
## 発熱       |  0.79  0.01 |
## 下痢       |  0.85  0.00 |
## ポテト     |  0.03  0.40 |
## 魚         |  0.01  0.03 |
## マヨ       |  0.33  0.04 |
## ズッキーニ |  0.02  0.48 |
## チーズ     |  0.13  0.03 |
##
## Supplementary categories
##          Dim.1  cos2 v.test   Dim.2  cos2 v.test
## 発症_n |  1.42  0.89   6.75 |  0.00  0.00   0.01 |
## 発症_y | -0.63  0.89  -6.75 |  0.00  0.00  -0.01 |
## 女性   | -0.03  0.00  -0.23 |  0.11  0.01   0.83 |
## 男性   |  0.03  0.00   0.23 | -0.12  0.01  -0.83 |
##
## Supplementary categorical variables (eta2)
##          Dim.1 Dim.2
## 発症   |  0.89  0.00 |
## 性別   |  0.00  0.01 |
##
## Supplementary continuous variables
##          Dim.1   Dim.2
## 年齢   |  0.00 | -0.01 |
## 時刻   | -0.84 | -0.08 |
```

For the supplementary individuals/variable categories, the coordinates and the quality of representation (cos2) on the factor maps are shown. They don't contribute to the dimensions.

16.1 Make a biplot of individuals and variable categories

FactomineR base graph:

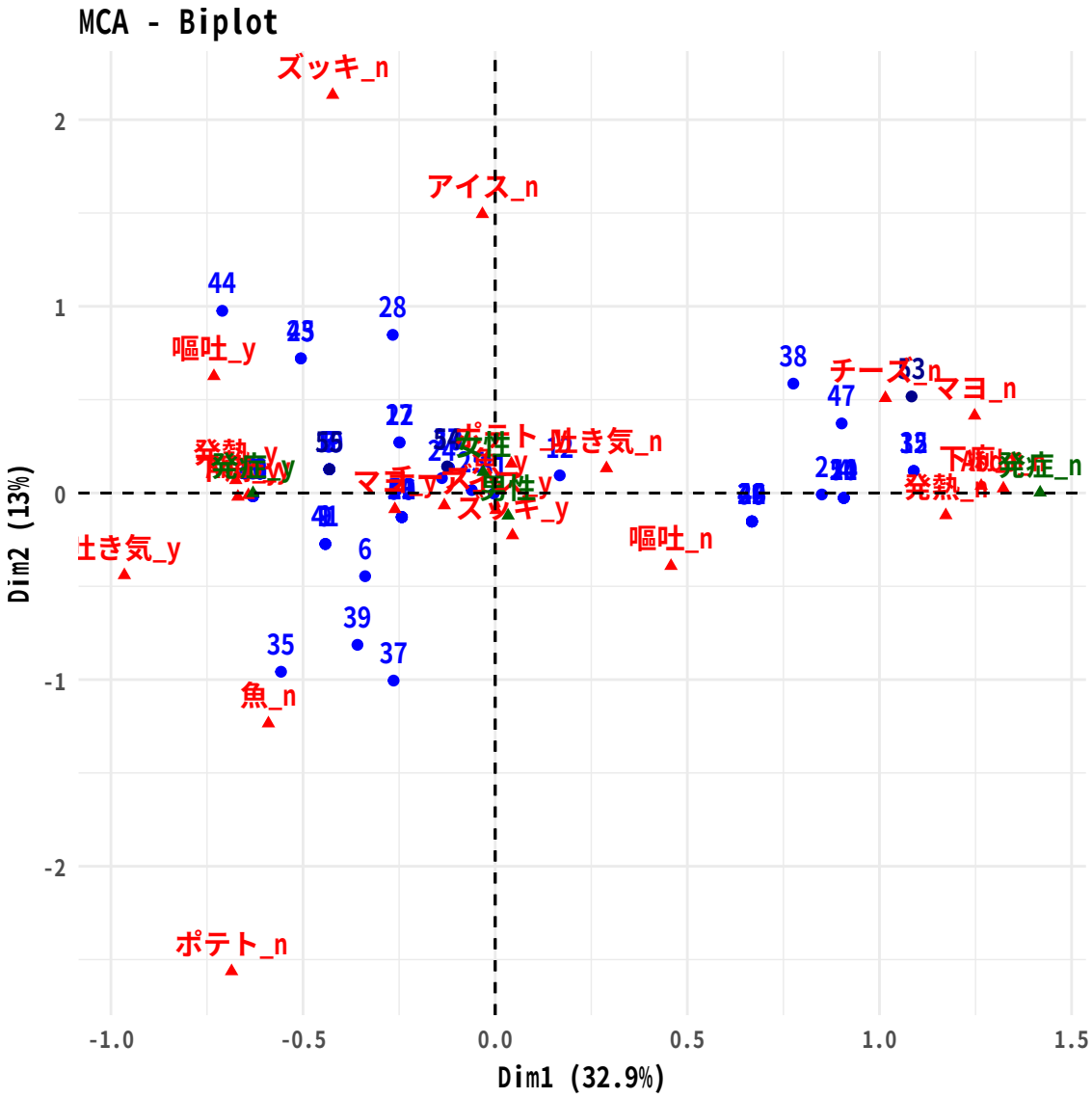
```
plot(res.mca)
```



- Active individuals are in blue - Supplementary individuals are in darkblue - Active variable categories are in red - Supplementary variable categories are in darkgreen

Use factoextra:

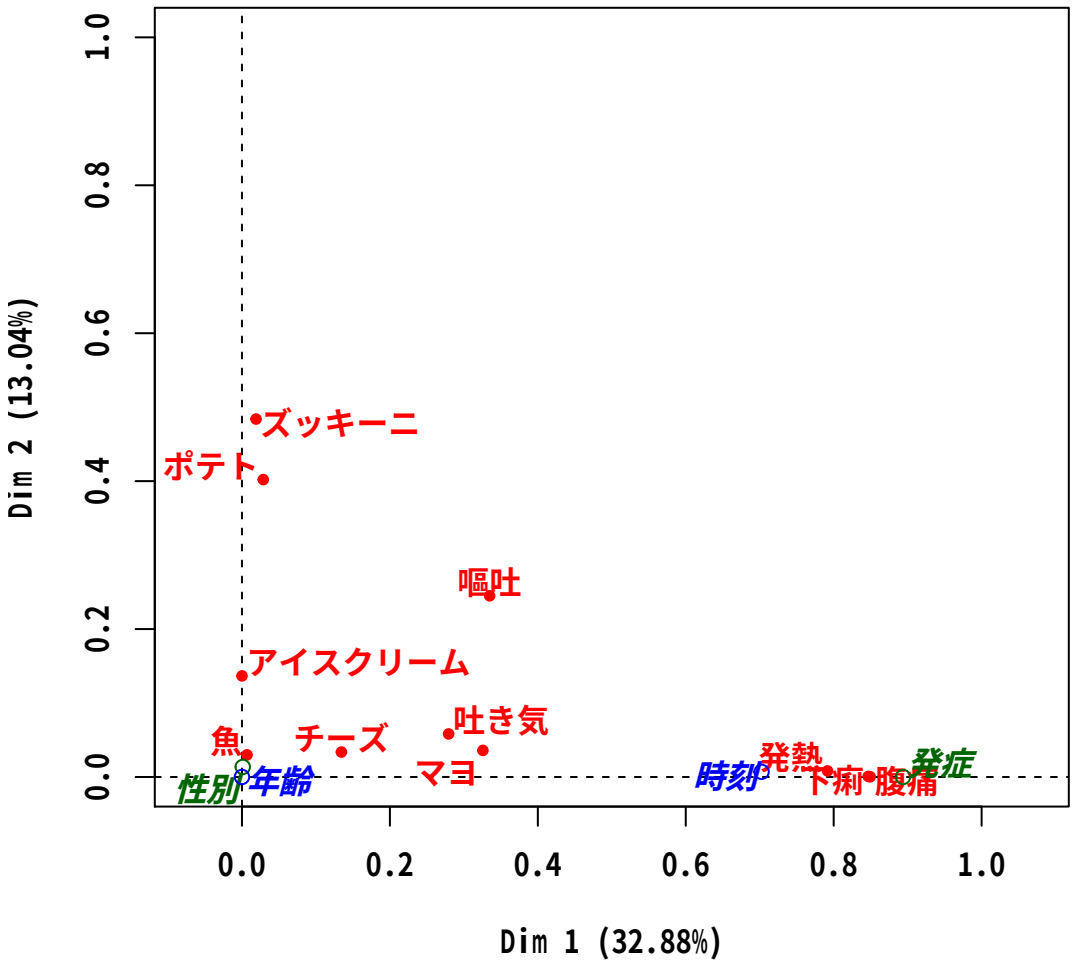
```
fviz_mca_biplot(res.mca, font.family = "sans") +
  theme_minimal()
```



16.2 Visualize supplementary variables

The graph below highlight the correlation between variables (active & supplementary) and dimensions:

```
plot(res.mca, choix ="var")
```



16.3 Supplementary qualitative variable categories

All the results (coordinates, cos2, v.test and eta2) for the supplementary qualitative variable categories can be extracted as follow :

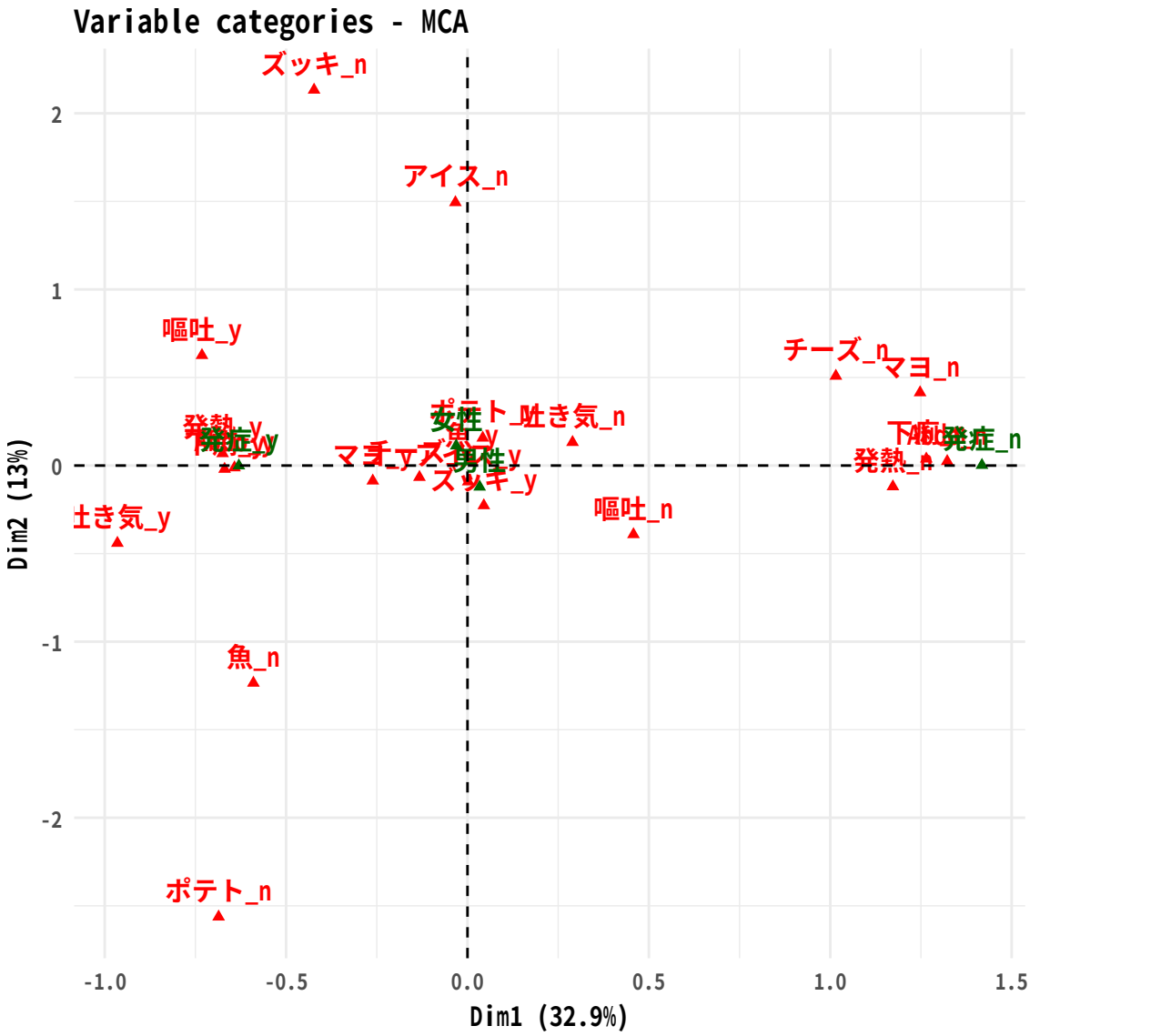
```
res.mca$quali.sup
```

```
## $coord
##              Dim 1         Dim 2       Dim 3        Dim 4       Dim 5
## 発症_n  1.41809140  0.0020394048  0.13199139 -0.016036841 -0.08354663
## 発症_y -0.63026284 -0.0009064021 -0.05866284  0.007127485  0.03713184
## 女性   -0.03108147  0.1123143957  0.05033124 -0.055927173 -0.06832928
## 男性    0.03356798 -0.1212995474 -0.05435774  0.060401347  0.07379562
##
## $cos2
##              Dim 1        Dim 2       Dim 3        Dim 4       Dim 5
## 発症_n 0.893770319 1.848521e-06 0.007742990 0.0001143023 0.003102240
## 発症_y 0.893770319 1.848521e-06 0.007742990 0.0001143023 0.003102240
## 女性   0.001043342 1.362369e-02 0.002735892 0.0033780765 0.005042401
## 男性   0.001043342 1.362369e-02 0.002735892 0.0033780765 0.005042401
##
## $v.test
##             Dim 1        Dim 2      Dim 3       Dim 4      Dim 5
```

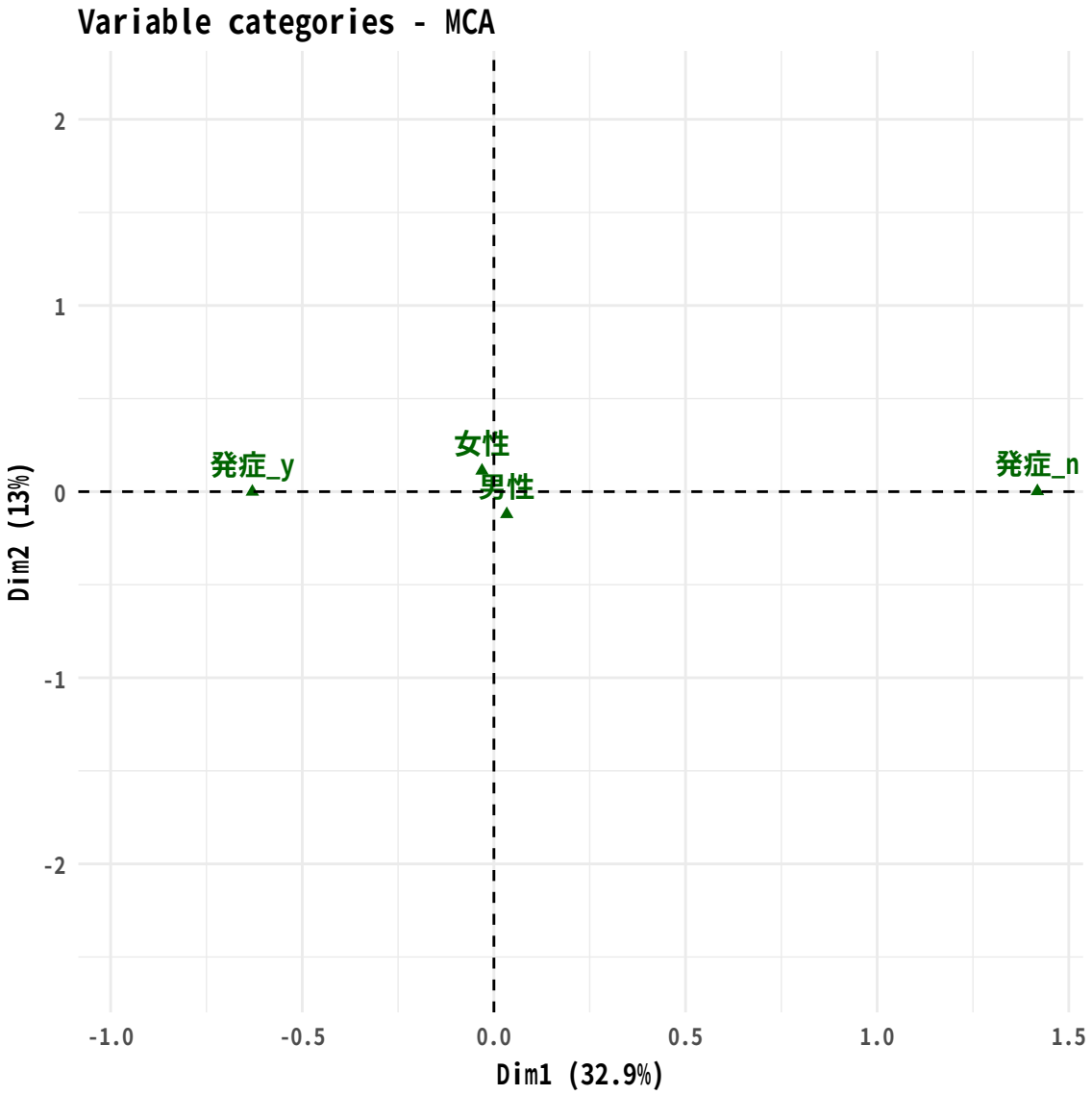
```
## 発症_n  6.7514655  0.009709509  0.6284047 -0.07635063 -0.3977615
## 発症_y -6.7514655 -0.009709509 -0.6284047  0.07635063  0.3977615
## 女性   -0.2306739  0.833551410  0.3735378 -0.41506855 -0.5071119
## 男性    0.2306739 -0.833551410 -0.3735378  0.41506855  0.5071119
##
## $eta2
##            Dim 1        Dim 2       Dim 3        Dim 4       Dim 5
## 発症 0.893770319 1.848521e-06 0.007742990 0.0001143023 0.003102240
## 性別 0.001043342 1.362369e-02 0.002735892 0.0033780765 0.005042401
```

Factor map :

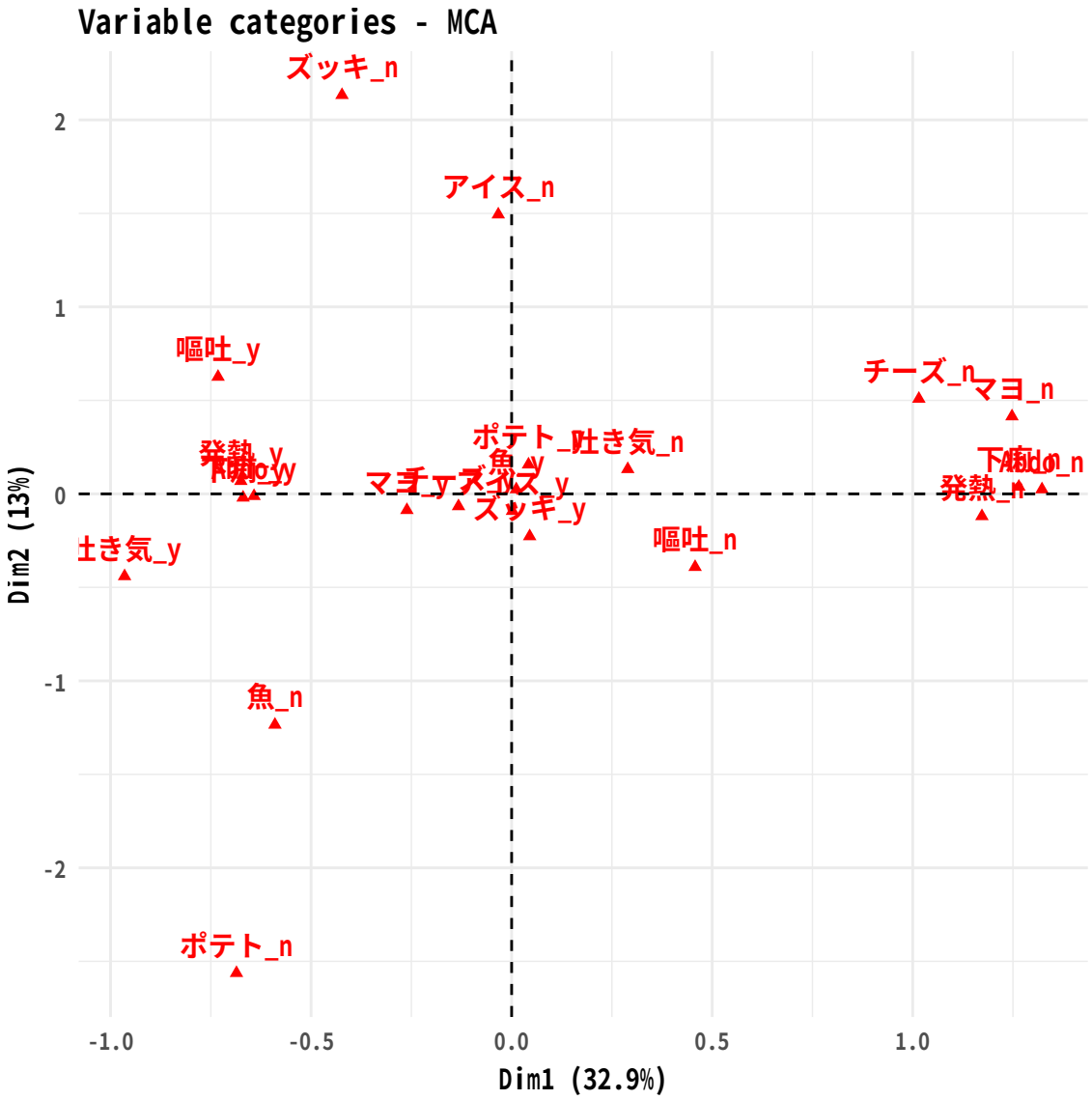
```
fviz_mca_var(res.mca, font.family = "sans") + theme_minimal()
```




```
# Hide active variables
fviz_mca_var(res.mca, invisible ="var", font.family = "sans") +
  theme_minimal()+ theme(text = element_text(family = "sans"))
```




```
# Hide supplementary qualitative variables
fviz_mca_var(res.mca, invisible ="quali.sup", font.family = "sans") +
  theme_minimal()
```



Supplementary variable categories are shown in darkgreen color. ## Supplementary quantitative variables

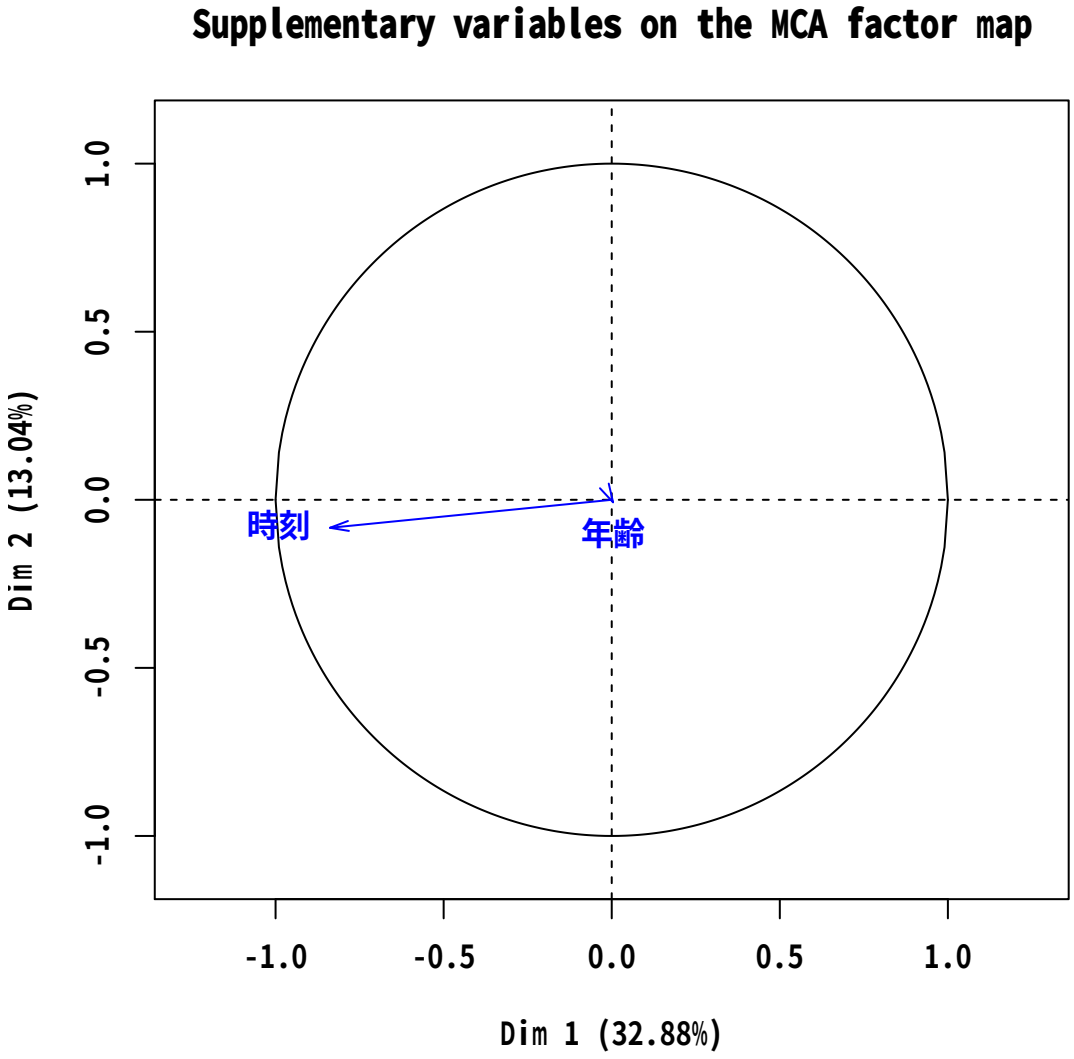
The coordinates of supplementary quantitative variables are:

```
res.mca$quanti
```

```
## $coord
##             Dim 1       Dim 2       Dim 3       Dim 4       Dim 5
## 年齢  0.003934896 -0.00741340 -0.26494536  0.20015501  0.02928483
## 時刻 -0.838158507 -0.08330586 -0.08718851 -0.08421599 -0.02316931
```

Graph using FactoMineR base graph:

```
plot(res.mca, choix="quanti.sup")
```



16.4 Visualize supplementary individuals

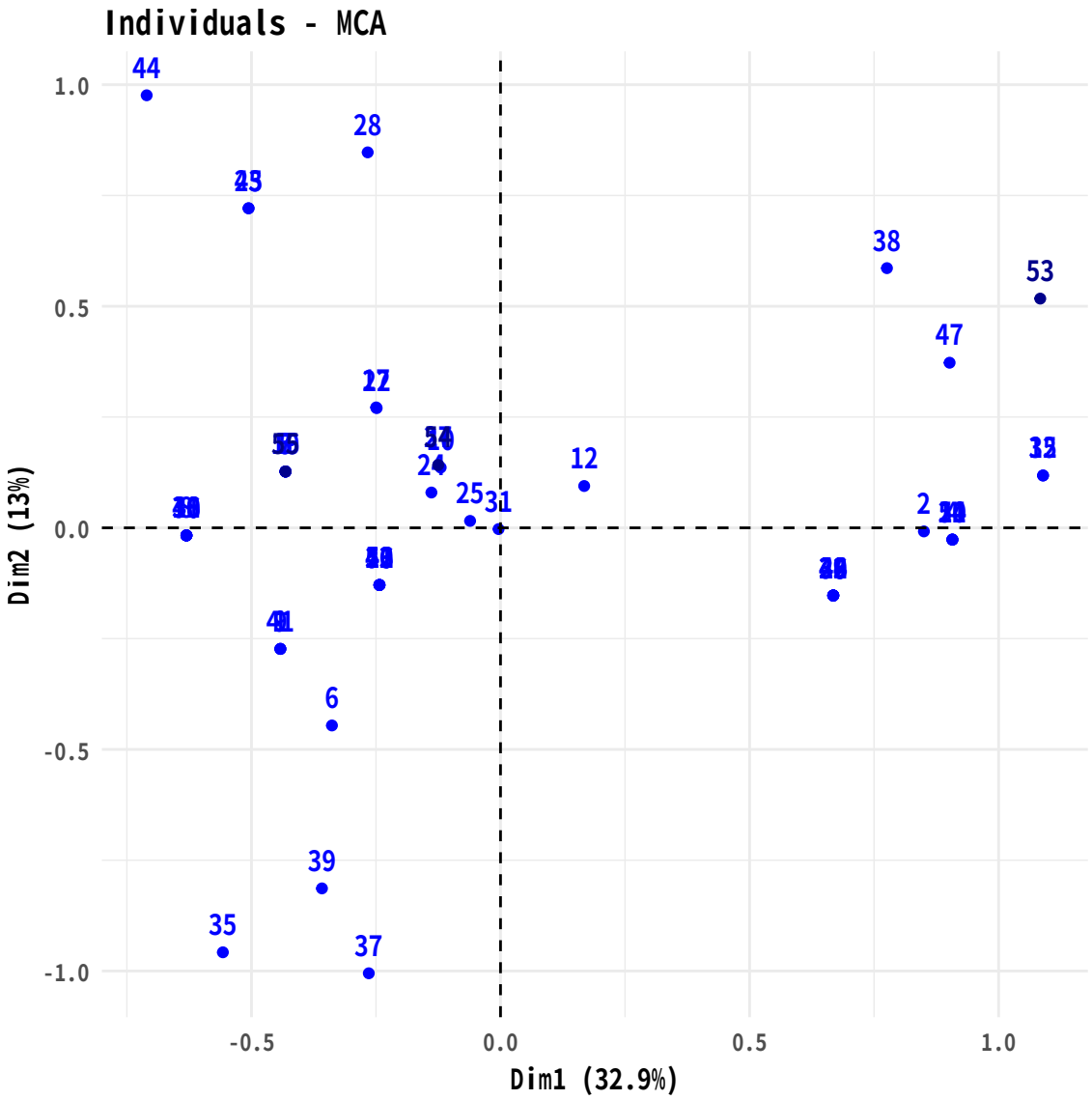
The results for supplementary individuals can be extracted as follow :

```
res.mca$ind.sup
```

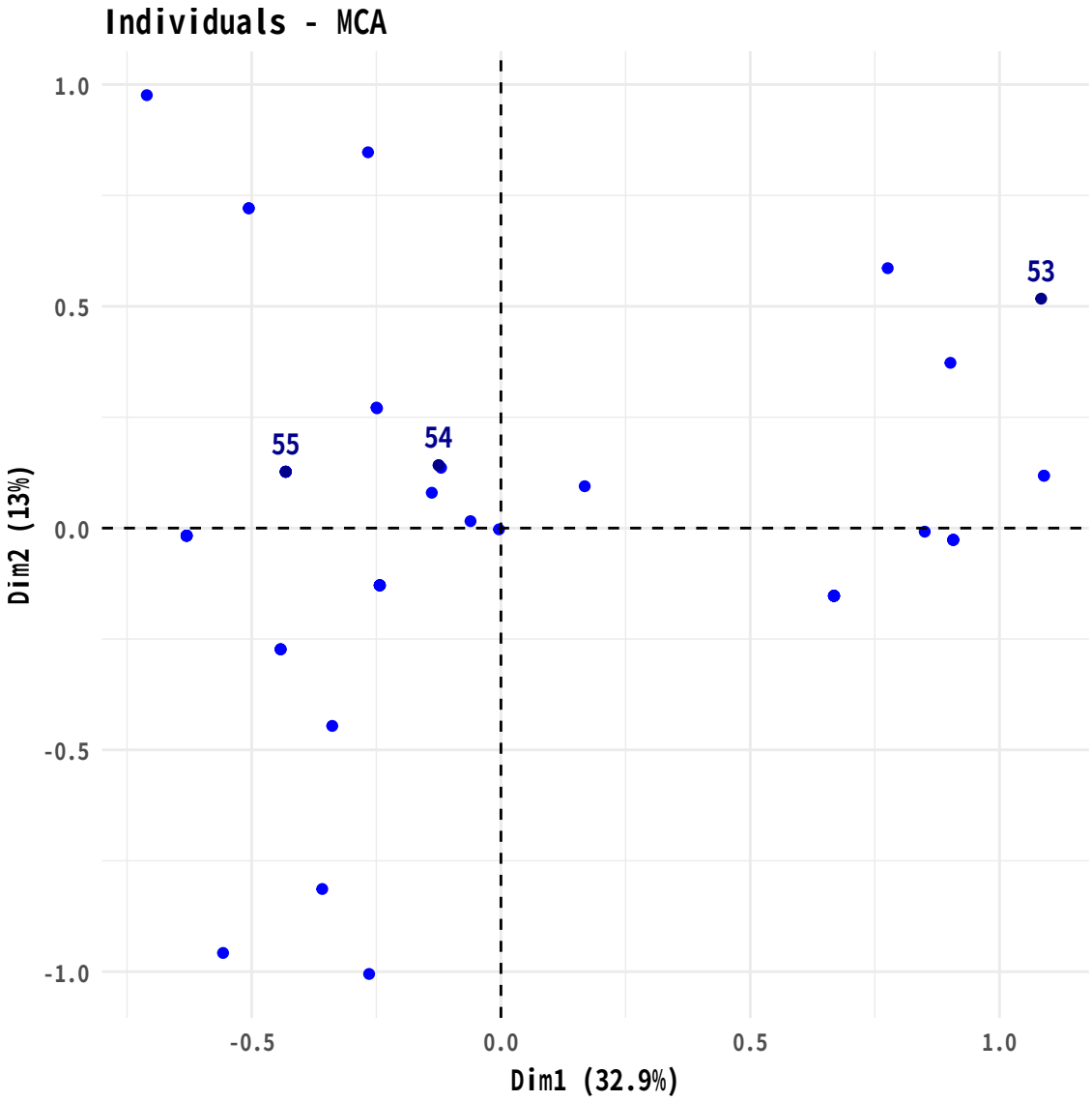
```
## $coord
##         Dim 1     Dim 2      Dim 3      Dim 4      Dim 5
## 53  1.0835684 0.5172478  0.5794063  0.5390903  0.4553650
## 54 -0.1249473 0.1417271 -0.1765234 -0.1526587 -0.2779565
## 55 -0.4315948 0.1270468 -0.2071580 -0.1186804 -0.1891760
##
## $cos2
##         Dim 1      Dim 2      Dim 3      Dim 4      Dim 5
## 53 0.36304957 0.08272764 0.10380536 0.08986204 0.06411692
## 54 0.03157652 0.04062716 0.06302535 0.04713607 0.15626590
## 55 0.50232519 0.04352713 0.11572730 0.03798314 0.09650827
```

Factor map for individuals:

```
fviz_mca_ind(res.mca, font.family = "sans") +
  theme_minimal()
```




```
# Show the label of ind.sup only
fviz_mca_ind(res.mca, label="ind.sup",font.family = "sans") +
  theme_minimal()
```



Supplementary individuals are shown in darkblue.

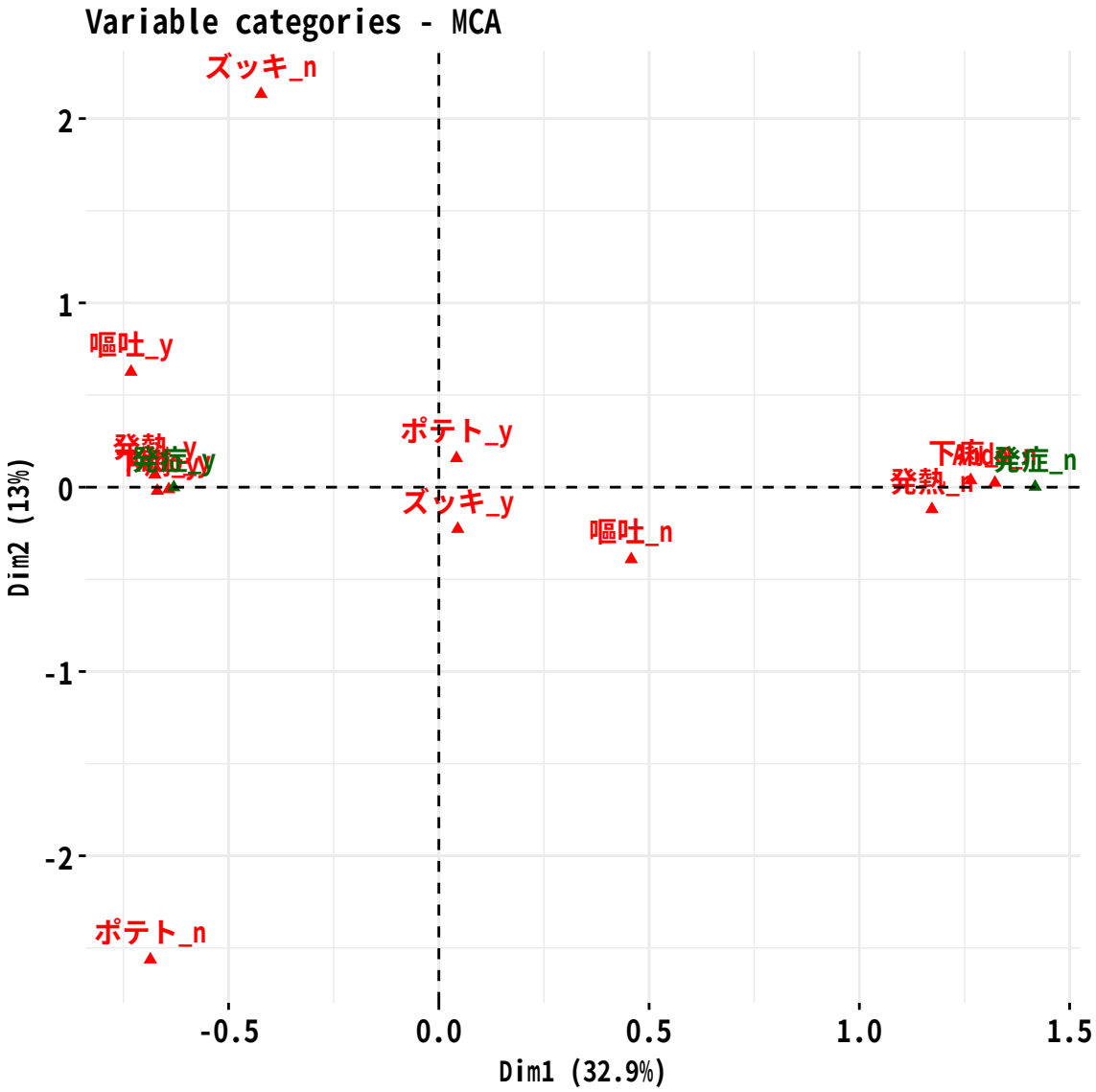
17 Filter the MCA result

If you have many individuals/variable categories, it's possible to visualize only some of them using the arguments select.ind and select.var.

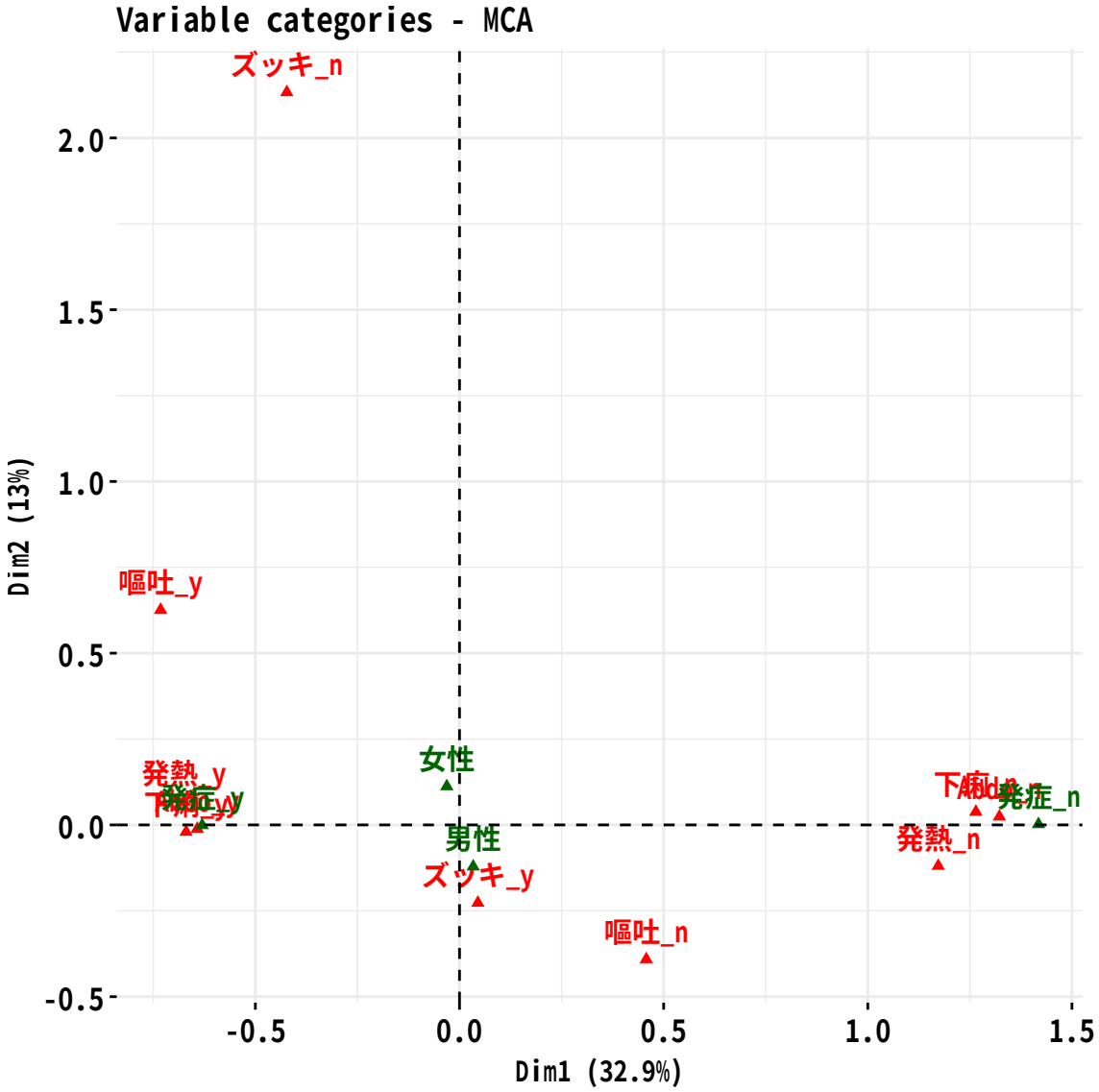
select.ind, select.var: a selection of individuals/variable categories to be drawn. Allowed values are NULL or a list containing the arguments name, cos2 or contrib:

- name: is a character vector containing individuals/variable category names to be drawn
- cos2: if cos2 is in [0, 1], ex: 0.6, then individuals/variable categories with a cos2 > 0.6 are drawn
- if cos2 > 1, ex: 5, then the top 5 active individuals/variable categories and top 5 supplementary columns/rows with the highest cos2 are drawn
- contrib: if contrib > 1, ex: 5, then the top 5 individuals/variable categories with the highest cos2 are drawn

```
# Visualize variable categories with cos2 >= 0.4
fviz_mca_var(res.mca, select.var = list(cos2 = 0.4), font.family = "sans")
```

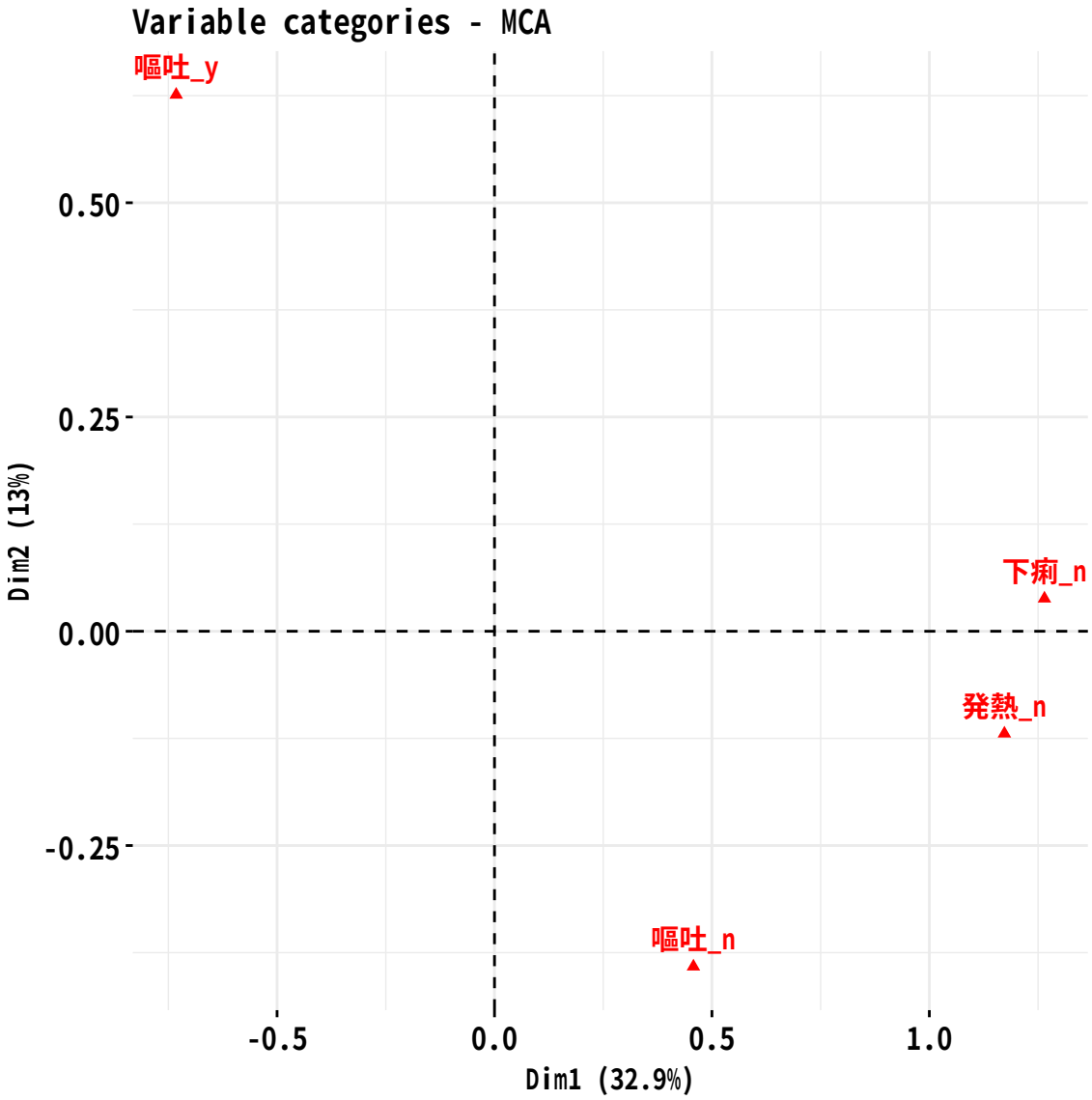



```
# Top 10 active variables with the highest cos2
fviz_mca_var(res.mca, select.var= list(cos2 = 10), font.family = "sans")
```

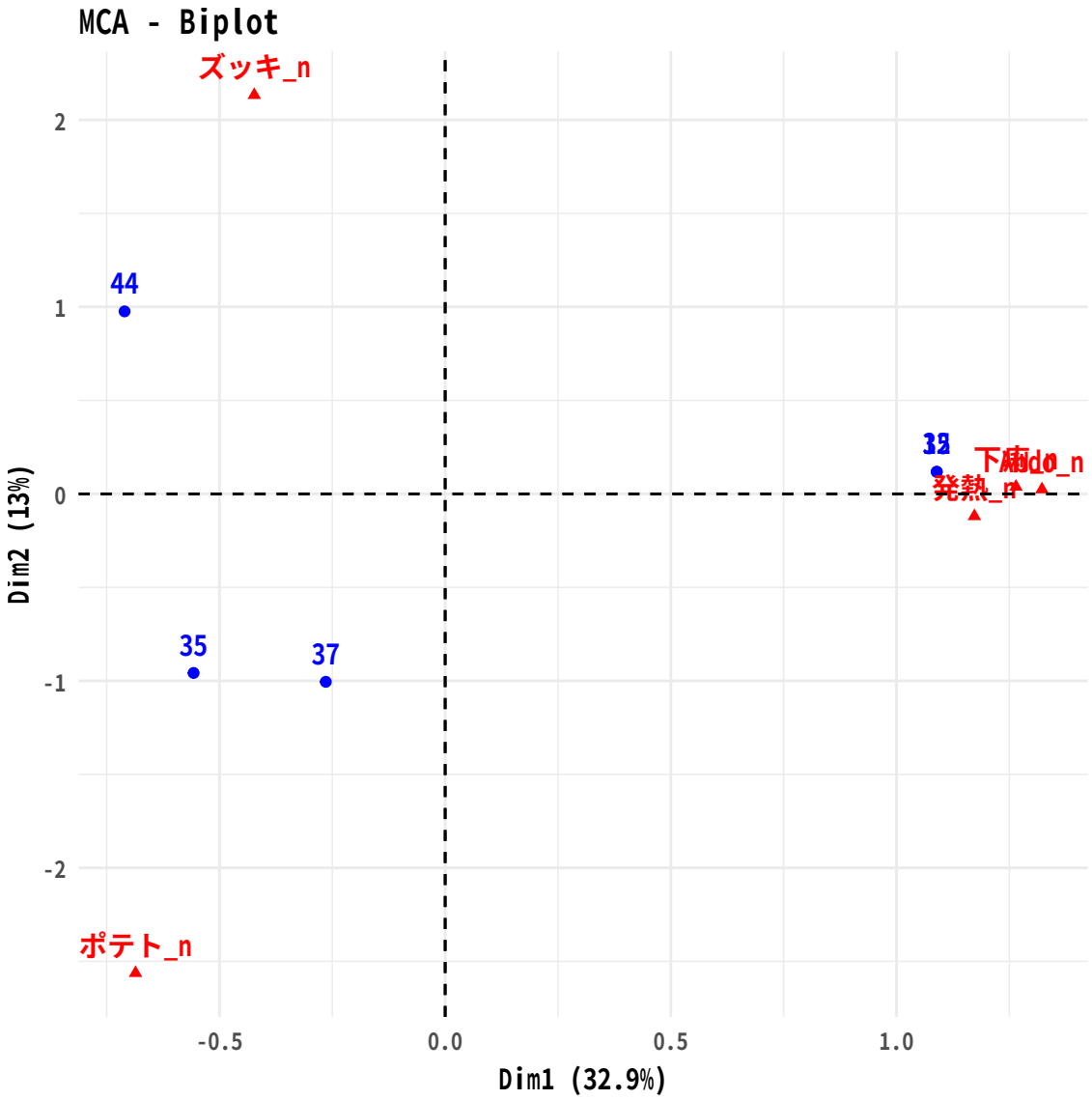


The top 10 active individuals and the top 10 supplementary individuals are shown.

```
# Select by names
name <- list(name = c("発熱_n", "腹痛_y", "下痢_n", "発熱_Y", "嘔吐_y", "嘔吐_n"))
fviz_mca_var(res.mca, select.var = name,font.family = "sans")
```




```
#top 5 contributing individuals and variable categories
fviz_mca_biplot(res.mca, select.ind = list(contrib = 5),
               select.var = list(contrib = 5),font.family = "sans") +
  theme_minimal()
```



Supplementary individuals/variable categories are not shown because they don't contribute to the construction of the axes.

18　次元記述 Dimension description

The function dimdesc() can be used to identify the most correlated variables with a given dimension.

A simplified format is :

```
dimdesc(res, axes = 1:2, proba = 0.05)
```

- res : an object of class MCA
- axes : a numeric vector specifying the dimensions to be described
- prob : the significance level

Example of usage :

```
res.desc <- dimdesc(res.mca, axes = c(1,2))
# Description of dimension 1
```

```
res.desc$`Dim 1`
```

```
## $quanti
##      correlation     p.value
## 時刻  -0.8381585 9.12658e-15
##
## $quali
##                R2      p.value
## 発症   0.8937703 5.368221e-26
## 腹痛   0.8493262 3.429439e-22
## 下痢   0.8467702 5.229788e-22
## 発熱   0.7916690 1.168654e-18
## 嘔吐   0.3348718 7.001487e-06
## マヨ   0.3257425 9.967995e-06
## 吐き気 0.2794053 5.623583e-05
## チーズ 0.1344785 7.495656e-03
##
## $category
##            Estimate      p.value
## 発症_n    0.5872910 5.368221e-26
## Abdo_n    0.5632879 3.429439e-22
## 下痢_n    0.5545730 5.229788e-22
## 発熱_n    0.5297728 1.168654e-18
## 嘔吐_n    0.3410366 7.001487e-06
## マヨ_n    0.4325471 9.967995e-06
## 吐き気_n  0.3597065 5.623583e-05
## チーズ_n  0.3290968 7.495656e-03
## チーズ_y -0.3290968 7.495656e-03
## 吐き気_y -0.3597065 5.623583e-05
## マヨ_y   -0.4325471 9.967995e-06
## 嘔吐_y   -0.3410366 7.001487e-06
## 発熱_y   -0.5297728 1.168654e-18
## 下痢_y   -0.5545730 5.229788e-22
## Abdo_y   -0.5632879 3.429439e-22
## 発症_y   -0.5872910 5.368221e-26
```

```
# Description of dimension 2
res.desc$`Dim 2`
```

```
## $quali
##                       R2      p.value
## ズッキーニ     0.4839477 1.039252e-08
## ポテト         0.4020987 4.489421e-07
## 嘔吐           0.2449186 1.917736e-04
## アイスクリーム 0.1366683 6.989716e-03
##
## $category
##            Estimate      p.value
## ズッキ_n  0.4261065 1.039252e-08
## ポテト_y  0.4910893 4.489421e-07
## 嘔吐_y    0.1836850 1.917736e-04
## アイス_n  0.2863045 6.989716e-03
## アイス_y -0.2863045 6.989716e-03
## 嘔吐_n   -0.1836850 1.917736e-04
## ポテト_n -0.4910893 4.489421e-07
```

```
## ズッキ_y -0.4261065 1.039252e-08
```

19　付録 カイ2乗検定

これでみると、帰無仮説が棄却されるのは、マヨとチーズ。

```
chisq.test(with(poison, table(発症,ポテト)))
```

```
## Warning in chisq.test(with(poison, table(発症, ポテト))): Chi-squared
## approximation may be incorrect
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  with(poison, table(発症, ポテト))
## X-squared = 0.3014, df = 1, p-value = 0.583
```

```
chisq.test(with(poison, table(発症,魚)))
```

```
## Warning in chisq.test(with(poison, table(発症, 魚))): Chi-squared
## approximation may be incorrect
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  with(poison, table(発症, 魚))
## X-squared = 1.2618e-30, df = 1, p-value = 1
```

```
chisq.test(with(poison, table(発症,マヨ)))
```

```
## Warning in chisq.test(with(poison, table(発症, マヨ))): Chi-squared
## approximation may be incorrect
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  with(poison, table(発症, マヨ))
## X-squared = 11.126, df = 1, p-value = 0.0008512
```

```
chisq.test(with(poison, table(発症,ズッキーニ)))
```

```
## Warning in chisq.test(with(poison, table(発症, ズッキーニ))): Chi-squared
## approximation may be incorrect
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  with(poison, table(発症, ズッキーニ))
## X-squared = 0.0021285, df = 1, p-value = 0.9632
```

```
chisq.test(with(poison, table(発症,チーズ)))
```

```
## Warning in chisq.test(with(poison, table(発症, チーズ))): Chi-squared
## approximation may be incorrect
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  with(poison, table(発症, チーズ))
## X-squared = 4.1841, df = 1, p-value = 0.04081
```

```
chisq.test(with(poison, table(発症,アイスクリーム)))
```

```
## Warning in chisq.test(with(poison, table(発症, アイスクリーム))): Chi-
## squared approximation may be incorrect
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  with(poison, table(発症, アイスクリーム))
## X-squared = 0.087748, df = 1, p-value = 0.7671
```

20　実行環境

This analysis has been performed using R software (ver. 3.2.1), FactoMineR (ver. 1.30) and factoextra (ver. 1.0.2)

この分析は、以下の環境で実行されました。- R software ver 3.4.1 Mac - FactoMineR 1.35 - factextra 1.0.4.999 - ggplubr 0.1.2.999

```
sessionInfo()
```

```
## R version 3.4.0 (2017-04-21)
## Platform: x86_64-apple-darwin15.6.0 (64-bit)
## Running under: OS X El Capitan 10.11.6
##
## Matrix products: default
## BLAS: /Library/Frameworks/R.framework/Versions/3.4/Resources/lib/libRblas.0.dylib
## LAPACK: /Library/Frameworks/R.framework/Versions/3.4/Resources/lib/libRlapack.dylib
##
## locale:
## [1] ja_JP.UTF-8/ja_JP.UTF-8/ja_JP.UTF-8/C/ja_JP.UTF-8/ja_JP.UTF-8
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base
##
## other attached packages:
##  [1] corrplot_0.77        dplyr_0.5.0          purrr_0.2.2.2
##  [4] readr_1.1.0          tidyr_0.6.3          tibble_1.3.1
##  [7] tidyverse_1.1.1      car_2.1-4            lattice_0.20-35
## [10] ggpubr_0.1.2.999     magrittr_1.5         factoextra_1.0.4.999
## [13] ggplot2_2.2.1        FactoMineR_1.35
##
## loaded via a namespace (and not attached):
##  [1] reshape2_1.4.2       splines_3.4.0        haven_1.0.0
##  [4] colorspace_1.3-2     htmltools_0.3.5      yaml_2.1.14
##  [7] mgcv_1.8-17          rlang_0.1.1          nloptr_1.0.4
## [10] foreign_0.8-67       DBI_0.6-1            RColorBrewer_1.1-2
## [13] readxl_1.0.0         modelr_0.1.0         plyr_1.8.4
## [16] stringr_1.2.0        MatrixModels_0.4-1   cellranger_1.1.0
## [19] munsell_0.4.3        gtable_0.2.0         rvest_0.3.2
## [22] leaps_3.0            psych_1.7.3.21       evaluate_0.10
## [25] labeling_0.3         knitr_1.15.1         forcats_0.2.0
## [28] SparseM_1.76         quantreg_5.33        pbkrtest_0.4-7
## [31] parallel_3.4.0       broom_0.4.2          Rcpp_0.12.11
## [34] scales_0.4.1         backports_1.0.5      flashClust_1.01-2
```

```
## [37] scatterplot3d_0.3-39 jsonlite_1.4         lme4_1.1-13
## [40] ellipse_0.3-8        mnormt_1.5-5         hms_0.3
## [43] digest_0.6.12        stringi_1.1.5        ggrepel_0.6.5
## [46] grid_3.4.0           rprojroot_1.2        tools_3.4.0
## [49] lazyeval_0.2.0       cluster_2.0.6        MASS_7.3-47
## [52] Matrix_1.2-9         xml2_1.1.1           lubridate_1.6.0
## [55] assertthat_0.2.0     minqa_1.2.4          rmarkdown_1.4
## [58] httr_1.2.1           R6_2.2.0             nnet_7.3-12
## [61] nlme_3.1-131         compiler_3.4.0
```

21　参考文献およびより理解するために

- Bendixen M.1995, Compositional perceptual mapping using chi-squared tree analysis and Correspondence Analysis, «Journal of Marketing Management», 11, 571-581.
- Bendixen M. 2003, A Practical Guide to the Use of Correspondence Analysis in Marketing Research, Marketing Bulletin, 2003, 14, Technical Note 2. http://marketing-bulletin.massey.ac.nz/V14/MB_V14_T2_Bendixen.pdf
- Greenacre M.. Contribution biplots. http://www.econ.upf.edu/docs/papers/downloads/1162.pdf François Husson, http://factominer.free.fr/contact/index.html

22　提案

- Principal component analysis in R : prcomp() vs. princomp() - R software and data mining
- Correspondence Analysis in R: The Ultimate Guide for the Analysis, the Visualization and the Interpretation - R software and data mining
- Principal Component Analysis: How to reveal the most important variables in your data? - R software and data mining
- FactoMineR and factoextra : Principal Component Analysis Visualization - R software and data mining
- Principal component analysis : the basics you should read - R software and data mining ade4 and factoextra : Principal Component Analysis - R software and data mining
- Correspondence analysis basics - R software and data mining
- Factor analysis
- ca package and factoextra : Correspondence Analysis - R software and data mining
- ade4 and factoextra : Correspondence Analysis - R software and data mining
- MASS package and factoextra : Correspondence Analysis - R software and data mining