Principal Component Analysis (PCA) with Protein data

Humaun Farid Sohag

2025-08-13

#Step1. Load Required Libraries

```
# Install packages if not already installed
required_packages <- c("corrr", "ggcorrplot", "FactoMineR", "factoextra", "ibawds")
lapply(required_packages, function(pkg) {
  if (!require(pkg, character.only = TRUE)) install.packages(pkg)
  library(pkg, character.only = TRUE)
})
```

```
## [[1]]
## [1] "corrr"     "stats"     "graphics"  "grDevices" "utils"     "datasets"
## [7] "methods"   "base"
##
## [[2]]
##  [1] "ggcorrplot" "ggplot2"    "corrr"      "stats"      "graphics"
##  [6] "grDevices"  "utils"      "datasets"   "methods"    "base"
##
## [[3]]
##  [1] "FactoMineR" "ggcorrplot" "ggplot2"    "corrr"      "stats"
##  [6] "graphics"   "grDevices"  "utils"      "datasets"   "methods"
## [11] "base"
##
## [[4]]
##  [1] "factoextra" "FactoMineR" "ggcorrplot" "ggplot2"    "corrr"
##  [6] "stats"      "graphics"   "grDevices"  "utils"      "datasets"
## [11] "methods"    "base"
##
## [[5]]
##  [1] "ibawds"     "dslabs"     "factoextra" "FactoMineR" "ggcorrplot"
##  [6] "ggplot2"    "corrr"      "stats"      "graphics"   "grDevices"
## [11] "utils"      "datasets"   "methods"    "base"
```

#Step2. Load and Inspect Data

```
data(protein)  # From 'ibawds' package
protein_data <- protein
cat("Dataset Dimensions:", dim(protein_data), "\n")
```

```
## Dataset Dimensions: 25 10
```

```r
str(protein_data)
```

```
## tibble [25 x 10] (S3: tbl_df/tbl/data.frame)
##  $ country   : chr [1:25] "Albania" "Austria" "Belgium" "Bulgaria" ...
##  $ red_meat  : num [1:25] 10.1 8.9 13.5 7.8 9.7 10.6 8.4 9.5 18 10.2 ...
##  $ white_meat: num [1:25] 1.4 14 9.3 6 11.4 10.8 11.6 4.9 9.9 3 ...
##  $ eggs      : num [1:25] 0.5 4.3 4.1 1.6 2.8 3.7 3.7 2.7 3.3 2.8 ...
##  $ milk      : num [1:25] 8.9 19.9 17.5 8.3 12.5 25 11.1 33.7 19.5 17.6 ...
##  $ fish      : num [1:25] 0.2 2.1 4.5 1.2 2 9.9 5.4 5.8 5.7 5.9 ...
##  $ cereals   : num [1:25] 42.3 28 26.6 56.7 34.3 21.9 24.6 26.3 28.1 41.7 ...
##  $ starch    : num [1:25] 0.6 3.6 5.7 1.1 5 4.8 6.5 5.1 4.8 2.2 ...
##  $ nuts      : num [1:25] 5.5 1.3 2.1 3.7 1.1 0.7 0.8 1 2.4 7.8 ...
##  $ fruit_veg : num [1:25] 1.7 4.3 4 4.2 4 2.4 3.6 1.4 6.5 6.5 ...
```

```r
head(protein_data)
```

```
## # A tibble: 6 x 10
##   country   red_meat white_meat  eggs  milk  fish cereals starch  nuts fruit_veg
##   <chr>        <dbl>      <dbl> <dbl> <dbl> <dbl>   <dbl>  <dbl> <dbl>     <dbl>
## 1 Albania       10.1        1.4   0.5   8.9   0.2    42.3    0.6   5.5       1.7
## 2 Austria        8.9       14     4.3  19.9   2.1    28      3.6   1.3       4.3
## 3 Belgium       13.5        9.3   4.1  17.5   4.5    26.6    5.7   2.1       4
## 4 Bulgaria       7.8        6     1.6   8.3   1.2    56.7    1.1   3.7       4.2
## 5 Czechosl~      9.7       11.4   2.8  12.5   2      34.3    5     1.1       4
## 6 Denmark       10.6       10.8   3.7  25     9.9    21.9    4.8   0.7       2.4
```

#Step3. Check for Missing Values

```r
#3. Check for Missing Values
missing_counts <- colSums(is.na(protein_data))
print(missing_counts)
```

```
##    country   red_meat white_meat       eggs       milk       fish    cereals
##          0          0          0          0          0          0          0
##     starch       nuts  fruit_veg
##          0          0          0
```

#Step4. Select Numerical Variables

```r
numerical_data <- protein_data[, 2:10]
head(numerical_data)
```

```
## # A tibble: 6 x 9
##   red_meat white_meat  eggs  milk  fish cereals starch  nuts fruit_veg
##      <dbl>      <dbl> <dbl> <dbl> <dbl>   <dbl>  <dbl> <dbl>     <dbl>
## 1     10.1        1.4   0.5   8.9   0.2    42.3    0.6   5.5       1.7
## 2      8.9       14     4.3  19.9   2.1    28      3.6   1.3       4.3
## 3     13.5        9.3   4.1  17.5   4.5    26.6    5.7   2.1       4
## 4      7.8        6     1.6   8.3   1.2    56.7    1.1   3.7       4.2
## 5      9.7       11.4   2.8  12.5   2      34.3    5     1.1       4
## 6     10.6       10.8   3.7  25     9.9    21.9    4.8   0.7       2.4
```

#Step5. PCA Computation

```r
pca_model <- princomp(numerical_data, cor = TRUE) # cor=TRUE standardizes the data
summary(pca_model)  # Proportion of variance explained
```

```
## Importance of components:
##                           Comp.1    Comp.2    Comp.3    Comp.4    Comp.5
## Standard deviation     2.0016087 1.2786710 1.0620355 0.9770691 0.6810568
## Proportion of Variance 0.4451597 0.1816666 0.1253244 0.1060738 0.0515376
## Cumulative Proportion  0.4451597 0.6268263 0.7521507 0.8582245 0.9097621
##                            Comp.6     Comp.7     Comp.8     Comp.9
## Standard deviation     0.57020257 0.52115865 0.34101599 0.31482043
## Proportion of Variance 0.03612566 0.03017848 0.01292132 0.01101243
## Cumulative Proportion  0.94588776 0.97606624 0.98898757 1.00000000
```

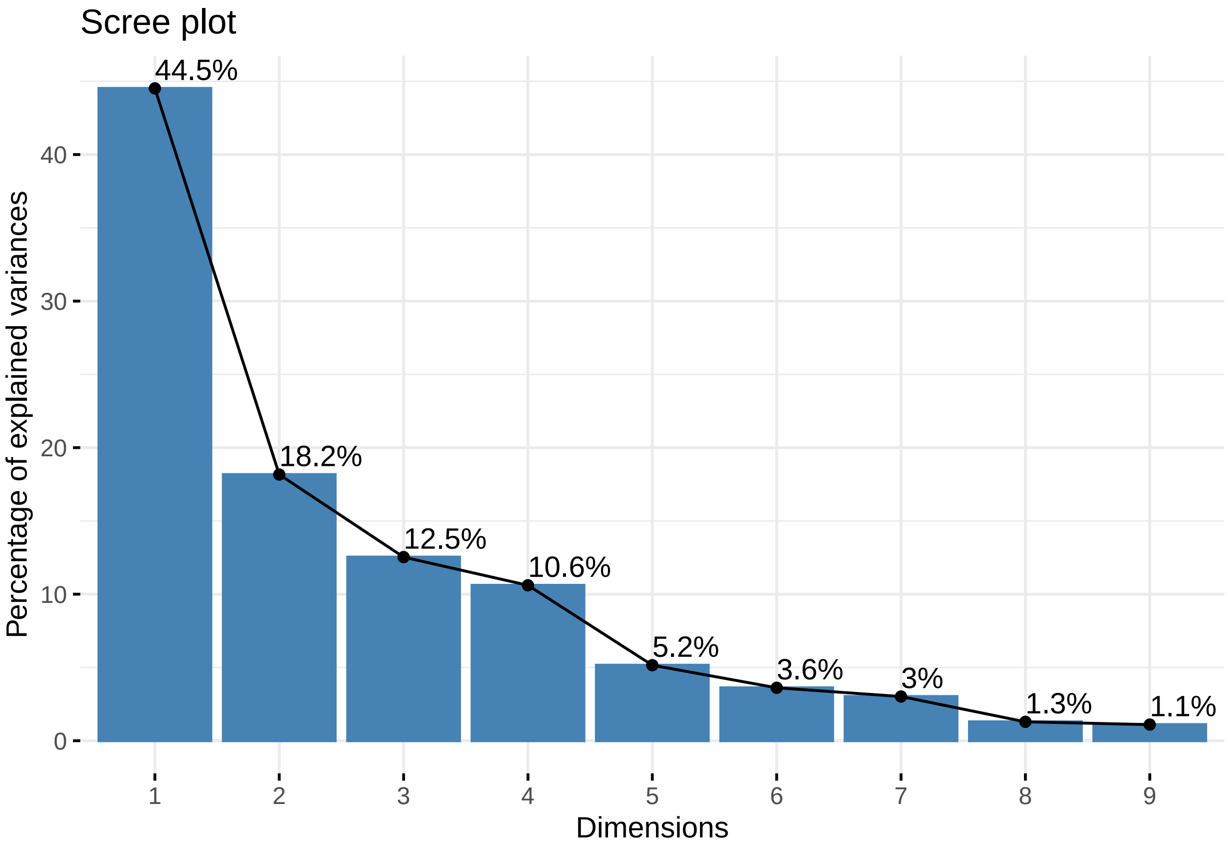
```r
pca_model$loadings[, 1:2]  # Loadings for first two PCs
```

```
##                Comp.1      Comp.2
## red_meat    0.3026094  0.05625165
## white_meat  0.3105562  0.23685334
## eggs        0.4266785  0.03533576
## milk        0.3777273  0.18458877
## fish        0.1356499 -0.64681970
## cereals    -0.4377434  0.23348508
## starch      0.2972477 -0.35282564
## nuts       -0.4203344 -0.14331056
## fruit_veg  -0.1104199 -0.53619004
```

#6. PCA Visualization

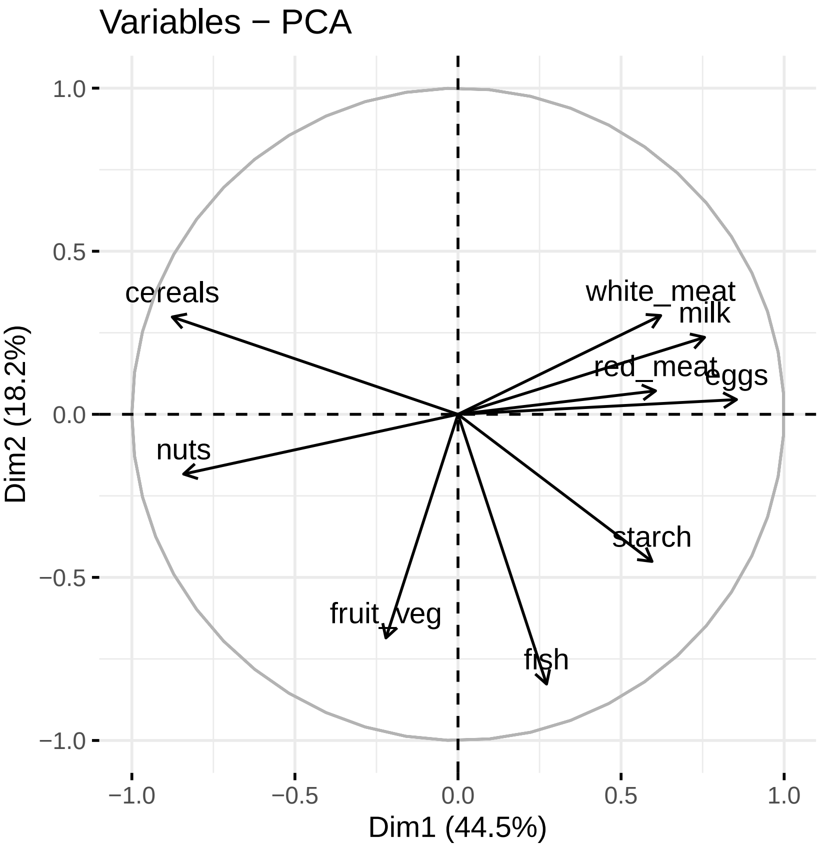
Scree plot (Eigenvalues)

```r
fviz_eig(pca_model, addlabels = TRUE)
```



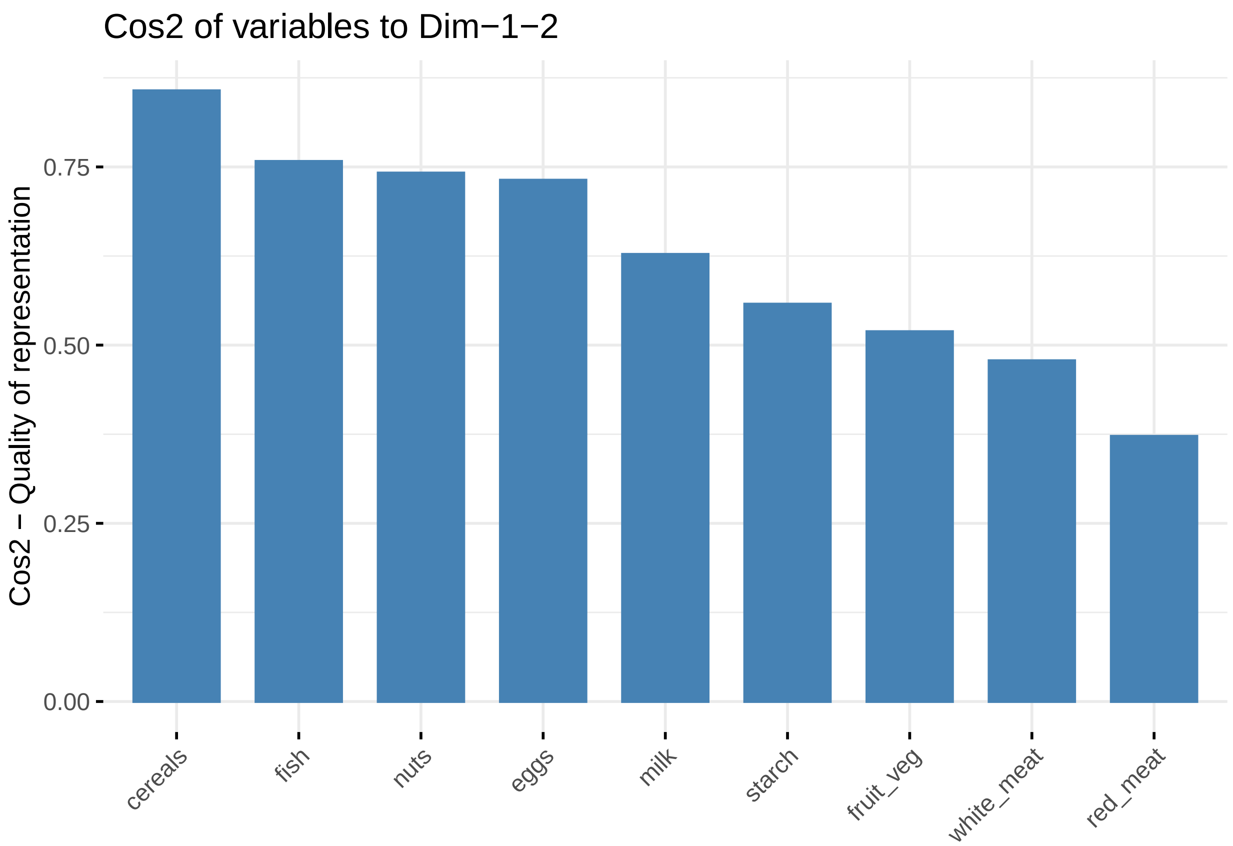
Variable correlation plot (PCA Biplot)

```
fviz_pca_var(pca_model, col.var = "black")
```



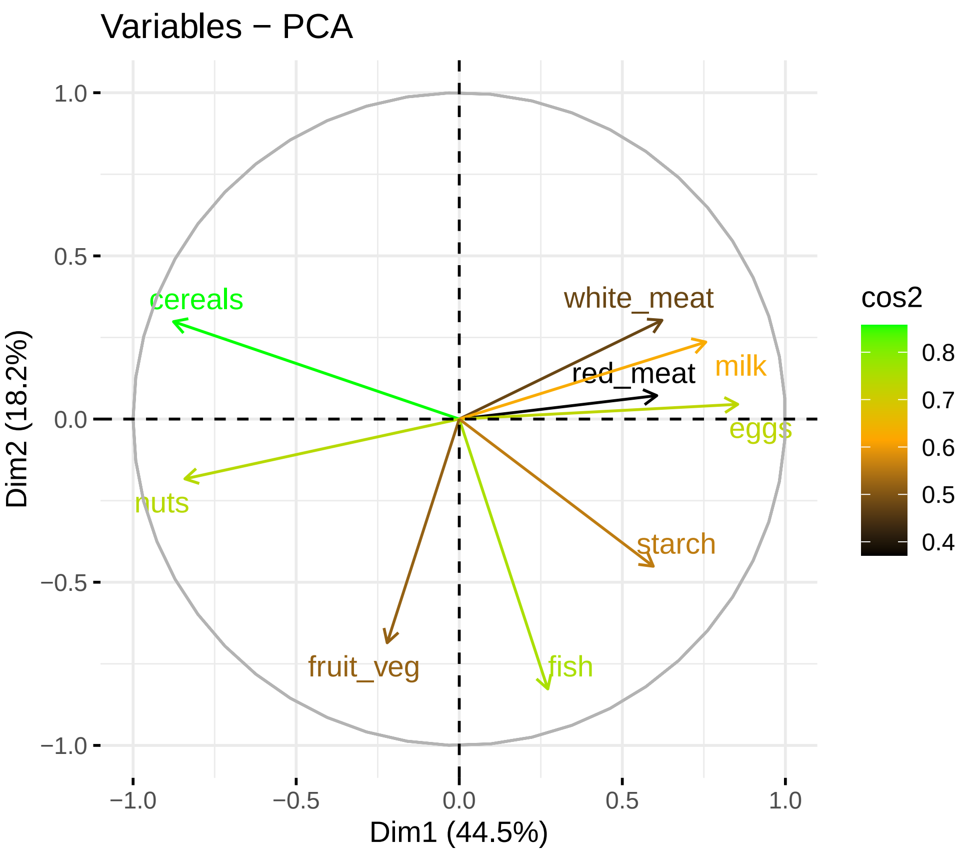
Cos²: Quality of representation on PC1 & PC2

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



PCA variable plot with Cos² coloring

```
fviz_pca_var(
  pca_model,
  col.var = "cos2",
  gradient.cols = c("black", "orange", "green"),
  repel = TRUE
)
```


Variables − PCA
Dim2 (18.2%)
Dim1 (44.5%)
1.0
0.5
0.0
−0.5
−1.0
−1.0
−0.5
0.0
0.5
1.0
cereals
white_meat
red_meat
milk
eggs
nuts
starch
fruit_veg
fish
cos2
0.8
0.7
0.6
0.5
0.4