BimalPaudel07

Bimal Paudel

2024-05-31

Question No. 6

```
set.seed(7)
n_samples <- 200
age <- c(10:99)
sex <- c('male', 'female')
education_level <- c('No education','Primary', 'Secondary', 'Beyond Secondary')
socio_economic_stataus <- c('Low', 'Middle', 'High')
body_mass_index <- c(14:38)
```

Question No. 7

```
data(airquality)

shapiro.test(airquality$Temp)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  airquality$Temp
## W = 0.97617, p-value = 0.009319
```

```
# p-value = 0.009319 < 0.05, hence it does not follow the normal distribution

data <- airquality[, c("Temp", "Month")]
row_names <- row.names(data)

# Step 2: Standardize the data
data_standardized <- scale(data)

# Step 3: Assign row names back to the standardized data
rownames(data_standardized) <- row_names

classical_state_disimilarity <- dist(data_standardized)
classical_mds <- cmdscale(classical_state_disimilarity)
summary(classical_mds)
```

```
##        V1                 V2
##  Min.   :-2.6298   Min.   :-1.6253
##  1st Qu.:-0.6365   1st Qu.:-0.5285
##  Median : 0.3109   Median :-0.1549
##  Mean   : 0.0000   Mean   : 0.0000
##  3rd Qu.: 0.7863   3rd Qu.: 0.4665
##  Max.   : 2.1310   Max.   : 2.1134
```

```r
# Perform Shapiro-Wilk test for each month separately
months <- unique(airquality$Month)

for (month in months) {
  temp_values <- airquality$Temp[airquality$Month == month]
  result <- shapiro.test(temp_values)
  cat("Shapiro-Wilk Test for Temp in Month", month, ":\n")
  print(result)
  cat("\n")
}
```

```
## Shapiro-Wilk Test for Temp in Month 5 :
##
##  Shapiro-Wilk normality test
##
## data:  temp_values
## W = 0.94771, p-value = 0.1349
##
##
## Shapiro-Wilk Test for Temp in Month 6 :
##
##  Shapiro-Wilk normality test
##
## data:  temp_values
## W = 0.97158, p-value = 0.5832
##
##
## Shapiro-Wilk Test for Temp in Month 7 :
##
##  Shapiro-Wilk normality test
##
## data:  temp_values
## W = 0.94579, p-value = 0.1194
##
##
## Shapiro-Wilk Test for Temp in Month 8 :
##
##  Shapiro-Wilk normality test
##
## data:  temp_values
## W = 0.96391, p-value = 0.3688
##
##
## Shapiro-Wilk Test for Temp in Month 9 :
##
##  Shapiro-Wilk normality test
##
## data:  temp_values
## W = 0.9513, p-value = 0.1831
```

```r
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(purrr)
```

```
##
## Attaching package: 'purrr'

## The following object is masked from 'package:base':
##
##     %||%
```

```r
# Group data by 'Month' and perform Shapiro-Wilk test on 'Temp'
results <- airquality %>%
  group_by(Month) %>%
  summarise(test_result = list(shapiro.test(Temp)))

# Print the results
results %>%
  mutate(p_value = map_dbl(test_result, "p.value")) %>%
  select(-test_result)
```

```
## # A tibble: 5 x 2
##   Month p_value
##   <int>   <dbl>
## 1     5   0.135
## 2     6   0.583
## 3     7   0.119
## 4     8   0.369
## 5     9   0.183
```

```r
# Interpretation: P-value for each month is above 0.005 hence we can conclude that the data are normall
```

Question No. 8

```r
library(car)
```

```
## Loading required package: carData
```

```
##
## Attaching package: 'car'

## The following object is masked from 'package:purrr':
##
##     some

## The following object is masked from 'package:dplyr':
##
##     recode
```

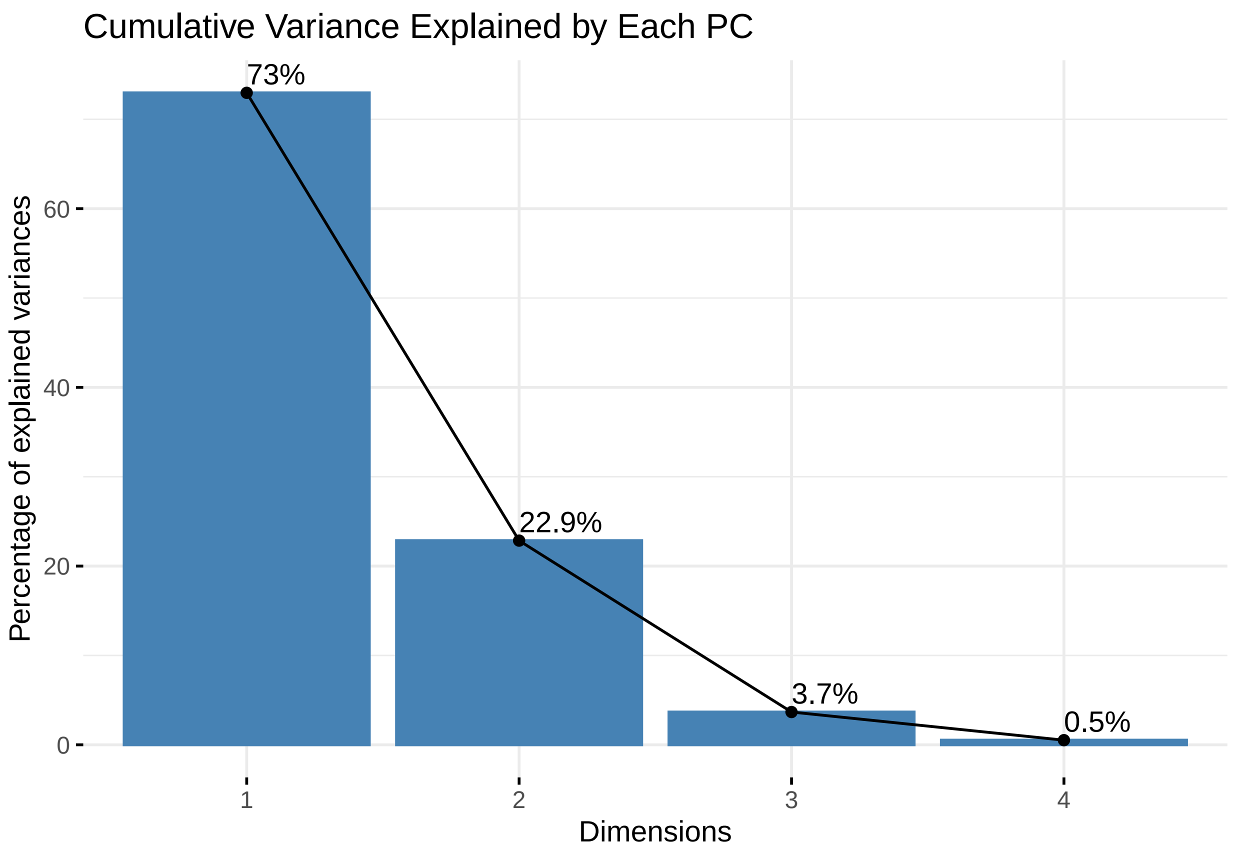
```r
data(Arrests)

a.sample <- sample(c(TRUE, FALSE), nrow(Arrests), replace=T, prob=c(0.8,0.2))
train <- Arrests[a.sample, ]
test <- Arrests[!a.sample, ]
```

Question No. 9

```r
library(ggplot2)
library(factoextra)
```

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

```r
data(iris)

data <- data.frame(Sepal.Length = iris$Sepal.Length, Sepal.Width = iris$Sepal.Width, Petal.Length = iris

row_names <- row.names(data)

# Step 2: Standardize the data
data_standardized <- scale(data)

# Step 3: Assign row names back to the standardized data
rownames(data_standardized) <- row_names

# Perform PCA: generating composite score
pca_model <- prcomp(data_standardized, center = TRUE, scale. = TRUE)

# Calculate and plot cumulative variance explained by each PC
fviz_eig(pca_model, addlabels = TRUE) +
  ggtitle("Cumulative Variance Explained by Each PC")
```



```r
# Check the summary to see how much variance each PC explains
summary(pca_model)
```

```
## Importance of components:
##                           PC1    PC2     PC3     PC4
## Standard deviation     1.7084 0.9560 0.38309 0.14393
## Proportion of Variance 0.7296 0.2285 0.03669 0.00518
## Cumulative Proportion  0.7296 0.9581 0.99482 1.00000
```

```r
# Calculate total variance explained by each principal component
var_explained = pca_model$sdev^2 / sum(pca_model$sdev^2)
```