Виконали: Кузьменко Юрій, Болотов Єгор

Лабораторна робота №8

Модельна діагностика.

Опис dataset

Назва dataset:

Spotify Top 10000 Streamed Songs

Link на dataset:

https://www.kaggle.com/datasets/rakkesharv/spotify-top-10000-streamed-songs

Опис dataset та постановку задачі:

Це набір даних, зібраний з веб-сайту Spotify, котрий містить потоки виконавця та кількість прослiховувань (було взято саме топ-10000) Основна мета: вплив факторів на популярність пісні й дізнатись найпопулярніших виконавців та треки.

Змінні та їх опис:

Position - Spotify Ranking

Artist Name - Artist Name

Song Name - Song Name

Days - No of days since the release of the song

Top 10 (xTimes) - No of times inside top 10

Peak Position - Peak position attained

Peak Position (xTimes) - No of times Peak position attained

Peak Streams - Total no of streams during Peak position

Total Streams - Total song streams

```
df <- read_csv("../Spotify_final_dataset.csv")

y <- df$Top_ten_times
x1 <- df$Peak_position_times
x2 <- df$Peak_streams
x3 <- df$Days
x4 <- df$Total_streams
mod <- lm(y ~ x1 + x2 + x3 + x4);
```

Завдання 1: Метод головних компонент (Principal Component Analysis – PCA).

(A) Підготовка до методу PCA (всі змінні мають тип num);

```
df_num <- df[, c(1, 4:9)]
```

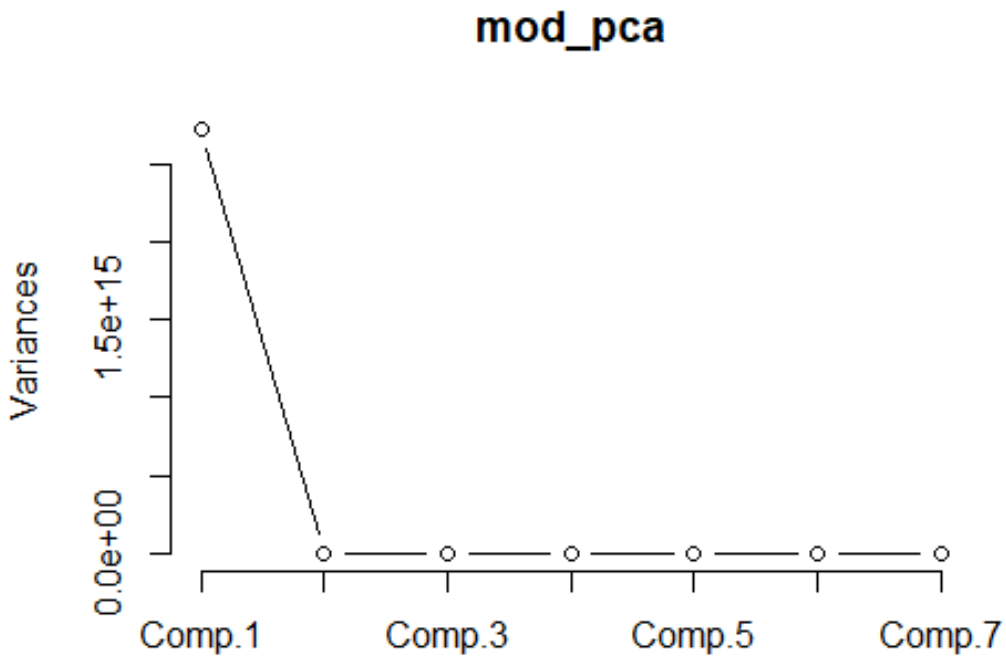
(B) Застосувати PCA mod_pca <- princomp(data, fix_sign = TRUE);

```
mod_pca <- princomp(df_num, fix_sign = TRUE)
```

(C) Розподіл дисперсій кожної компоненти

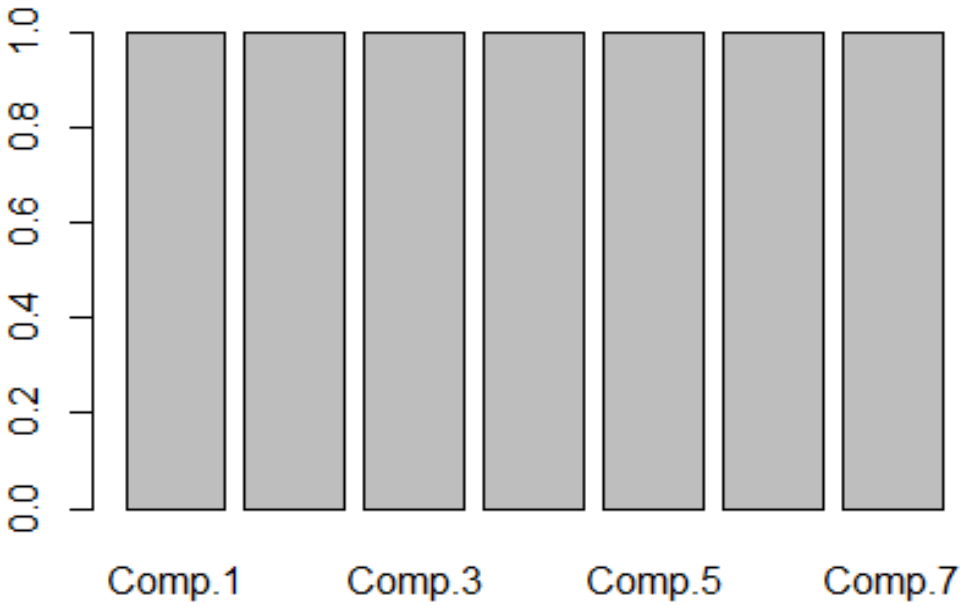
o plot(mod_pca, type = “l”).

```
plot(mod_pca, type = "l")
```



о Альтернативна діаграма сукупної відсоткової дисперсії
barplot(cumsum(mod_pca$sdev^2)/sum(mod_pca$sdev^2));

```
barplot(cumsum(mod_pca$sdev^2) / sum(mod_pca$sdev^2));
```



о висновок про кількість основних компонент

```
summary(mod_pca)

## Importance of components:
##                             Comp.1       Comp.2       Comp.3       Comp.4
## Standard deviation     5.219223e+07 5.622330e+05 2.542397e+03 4.107639e+01
## Proportion of Variance 9.998840e-01 1.160301e-04 2.372601e-09 6.193304e-13
## Cumulative Proportion  9.998840e-01 1.000000e+00 1.000000e+00 1.000000e+00
##                             Comp.5       Comp.6       Comp.7
## Standard deviation     3.260434e+01 9.378343e+00 2.483549e+00
## Proportion of Variance 3.902013e-13 3.228420e-14 2.264036e-15
## Cumulative Proportion  1.000000e+00 1.000000e+00 1.000000e+00
```

У summary PCA вказано стандартне відхмлення, частка дисперсії й кумулятивна частинка для наших компонентів (усього їх 7).

Завдання 2: Відновлення даних.

(A) Відновлення даних з усіх основних компонент;

```
mod_pcaStd <- princomp(x = df_num, cor = TRUE, fix_sign = TRUE)
```

(B) Центрування та стандартизованих змінних

o scale(laliga, center = TRUE, scale = FALSE) %*% A

```
n <- nrow(df_num)
eig <- eigen(cov(df_num) * (n - 1) / n)
A <- eig$vectors

df_num_scaled <- scale(df_num, center = TRUE, scale = FALSE) %*% A
```

(C) Метод PCA для стандартизованих змінних

o princomp(x = laliga, cor = TRUE, fix_sign = TRUE)

```
princomp(x = df_num, cor = TRUE, fix_sign = TRUE)
```

```
## Call:
## princomp(x = df_num, cor = TRUE, fix_sign = TRUE)
##
## Standard deviations:
##    Comp.1    Comp.2    Comp.3    Comp.4    Comp.5    Comp.6    Comp.7
## 1.9637688 1.1983825 0.9122442 0.6602553 0.4951206 0.4007021 0.1834620
##
##  7  variables and  11084 observations.
```

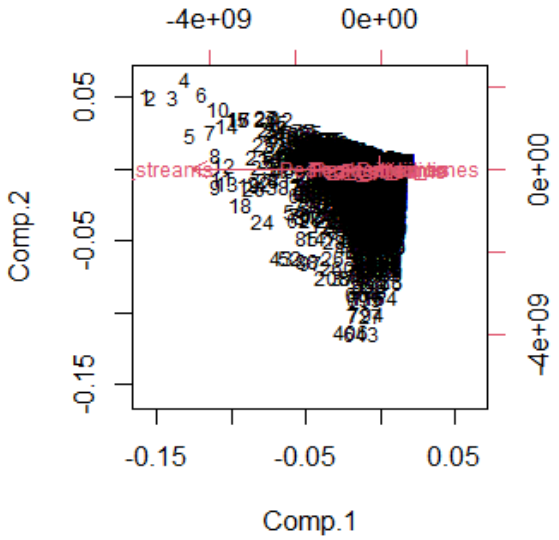
7 змінних й 11084 спостережень (значень для кожної змінної).

Завдання 3: Графічне подання змінних.

(A) Графічне подання змінних через 2-ві перші основні компоненти biplot(*, cex = 0.75)

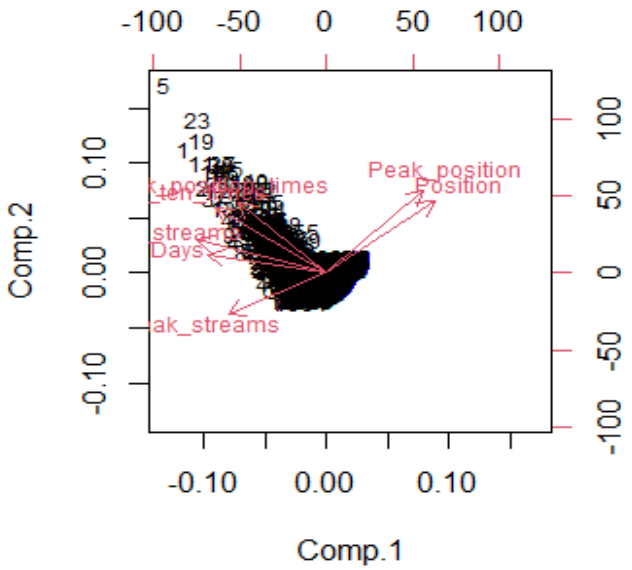
о для звичайних даних

```
biplot(mod_pca, cex = 0.75)
```



о стандартизованих;

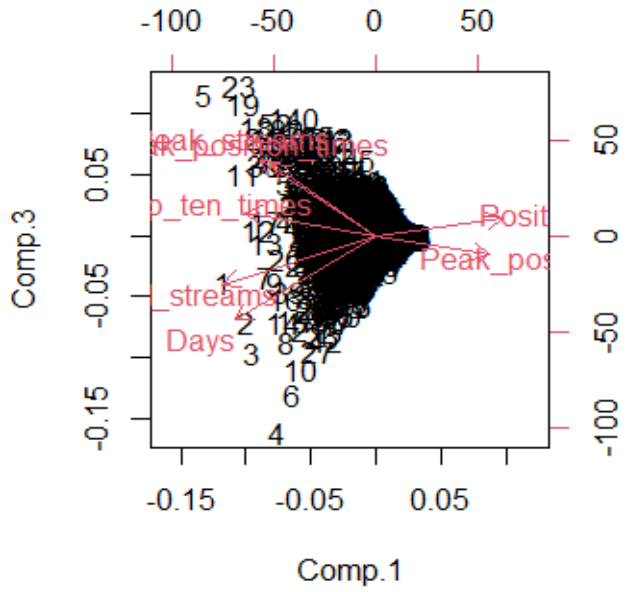
```
biplot(mod_pcaStd, cex = 0.75)
```



(B) Представлення за вказаними компонентами

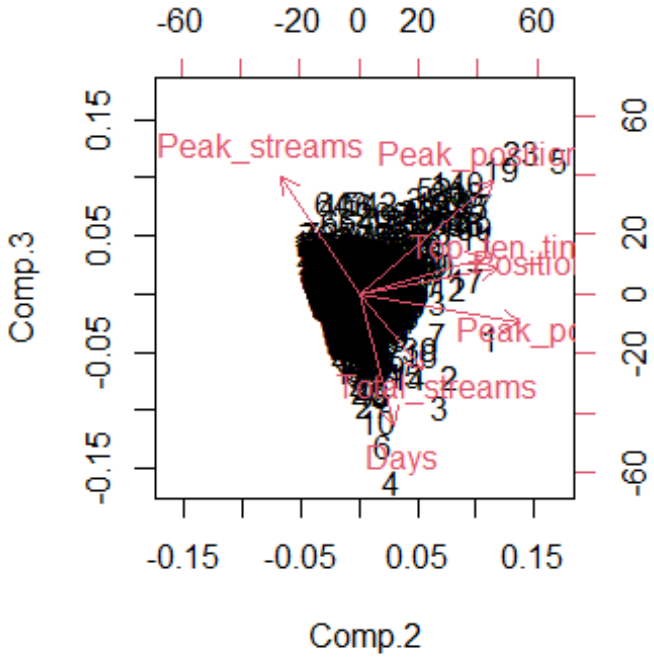
o biplot(pcaLaligaStd, choices = c(1, 3))

```
biplot(mod_pcaStd, choices = c(1, 3))
```



o biplot(pcaLaligaStd, choices = c(2, 3))

```
biplot(mod_pcaStd, choices = c(2, 3))
```



Завдання 4: Моделювання.

(A) Побудувати модель на базі основних компонент modPCA <- lm(Y ~ Comp.1 + Comp.2, Comp.3, dataPCA)

```
dfPCA <- data.frame("Y" = y, cbind(mod_pca$scores))

modPCA <- lm(Y ~ Comp.1 + Comp.2 + Comp.3 , data=dfPCA)
```

(B) Порівняйте результат PCA з mod <- lm(Y ~ x1 +x2 +x3+x4, data)

```
summary(modPCA)

##
## Call:
## lm(formula = Y ~ Comp.1 + Comp.2 + Comp.3, data = dfPCA)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -156.413   -1.246    0.257    1.476  181.070
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.714e+00  9.740e-02   27.86   <2e-16 ***
## Comp.1      -2.205e-07  1.866e-09 -118.16   <2e-16 ***
## Comp.2      -1.755e-06  1.732e-07  -10.13   <2e-16 ***
## Comp.3       8.744e-04  3.831e-05   22.82   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 10.25 on 11080 degrees of freedom
## Multiple R-squared:  0.5683, Adjusted R-squared:  0.5682
## F-statistic:  4862 on 3 and 11080 DF,  p-value: < 2.2e-16

summary(mod)

##
## Call:
## lm(formula = y ~ x1 + x2 + x3 + x4)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -155.584   -0.171   -0.039    0.073  140.212
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  3.315e-01  1.115e-01   2.972  0.00297 **
## x1           1.548e+00  2.733e-02  56.648  < 2e-16 ***
## x2          -1.175e-06  1.465e-07  -8.021 1.16e-15 ***
## x3          -6.552e-02  1.837e-03 -35.672  < 2e-16 ***
## x4           3.238e-07  5.166e-09  62.682  < 2e-16 ***
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.219 on 11079 degrees of freedom
## Multiple R-squared:  0.7227, Adjusted R-squared:  0.7226
## F-statistic:  7218 on 4 and 11079 DF,  p-value: < 2.2e-16
```

Порівнюючи моделі можна вкахати, що mod описує дані краще, бо скоріш за все mod краще результує дані бо має +1 фічу яка є важливою.