Praktikum W2 Komstat

Harvest Walukow

2025-02-21

Nomor 1

```
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
```

```
df <- read.csv("D:/UNAIR/SEMESTER 4/Komstat/Life Expectancy Data.csv")
```

a. Jumlah negara berdasarkan status pada tahun 2015

```
jumlah_negara_status <- df %>%
  filter(Year == 2015) %>%
  group_by(Status) %>%
  summarize(Jumlah_Negara = n())

print(jumlah_negara_status)
```

```
## # A tibble: 2 x 2
##   Status     Jumlah_Negara
##   <chr>              <int>
## 1 Developed             32
## 2 Developing           151
```

b. Mean, Median, Min, Max Angka Harapan Hidup (Life Expectancy) dari negara Indonesia berdasarkan tahun

```r
statistik_indonesia <- df %>%
  filter(Country == "Indonesia") %>%
  group_by(Year) %>%
  summarize(
    Mean_Life_Expectancy = mean(Life.expectancy, na.rm = TRUE),
    Median_Life_Expectancy = median(Life.expectancy, na.rm = TRUE),
    Min_Life_Expectancy = min(Life.expectancy, na.rm = TRUE),
    Max_Life_Expectancy = max(Life.expectancy, na.rm = TRUE)
  )

print(statistik_indonesia)
```

```
## # A tibble: 16 x 5
##     Year Mean_Life_Expectancy Median_Life_Expectancy Min_Life_Expectancy
##    <int>                <dbl>                  <dbl>                <dbl>
##  1  2000                 66.3                   66.3                 66.3
##  2  2001                 66.5                   66.5                 66.5
##  3  2002                 66.7                   66.7                 66.7
##  4  2003                 66.9                   66.9                 66.9
##  5  2004                 65.3                   65.3                 65.3
##  6  2005                 67.2                   67.2                 67.2
##  7  2006                 67.3                   67.3                 67.3
##  8  2007                 67.5                   67.5                 67.5
##  9  2008                 67.7                   67.7                 67.7
## 10  2009                 67.9                   67.9                 67.9
## 11  2010                 68.1                   68.1                 68.1
## 12  2011                 68.3                   68.3                 68.3
## 13  2012                 68.5                   68.5                 68.5
## 14  2013                 68.7                   68.7                 68.7
## 15  2014                 68.9                   68.9                 68.9
## 16  2015                 69.1                   69.1                 69.1
## # i 1 more variable: Max_Life_Expectancy <dbl>
```

c. Mean, Median, Min, Max GDP berdasarkan status pada Tahun 2010

```r
df$GDP[is.na(df$GDP)] <- median(df$GDP, na.rm = TRUE)

statistik_gdp_status <- df %>%
  filter(Year == 2010) %>%
  group_by(Status) %>%
  summarize(
    Mean_GDP = mean(GDP, na.rm = TRUE),
    Median_GDP = median(GDP, na.rm = TRUE),
    Min_GDP = min(GDP, na.rm = TRUE),
    Max_GDP = max(GDP, na.rm = TRUE)
  )

print(statistik_gdp_status)
```

```
## # A tibble: 2 x 5
```

```
##   Status     Mean_GDP Median_GDP Min_GDP Max_GDP
##   <chr>         <dbl>      <dbl>   <dbl>   <dbl>
## 1 Developed    20794.      9812. 1326.    87647.
## 2 Developing    3621.      1767.    8.38  47447.
```

Nomor 2

```
library(readxl)
df2 <- read_excel("D:/UNAIR/SEMESTER 4/Komstat/Dataset Komputasi Statistika M2.xlsx")
df2[, 2:10] <- sapply(df2[, 2:10], as.numeric)
```

1. Buat vektor Y

```
Y <- df2$Y
print(Y)
```

```
##  [1]  976 2545  413 2238 1314  947 2184 2773 2243 1710 1217 1047 1132 1603 1353
## [16]  893  886 1509 1378  976 2545  413 2238 1314  947 2184 2773 2243 1710 1217
## [31] 1047 1132 1603 1353  893  886 1509 1378
```

```
is.vector(Y)
```

```
## [1] TRUE
```

2. Buat matriks X

```
X <- as.matrix(cbind(1, df2[, 2:10]))
print(X)
```

```
##       1       X1     X2      X3       X4     X5       X6       X7       X8
##  [1,] 1     5.97 306.00 835.000     2.61 257038    19.44 45785.00    -1.12
##  [2,] 1     7.66 329.00 482.000     4.20 517819     7.51     5.26     4.43
##  [3,] 1     9.63 308.00   1.116     0.52  56746     3.79     8.43     6.18
##  [4,] 1     7.82 335.00 711.000 45660.00 355797     8.71     5.45 45693.00
##  [5,] 1     7.43   0.28 569.000     3.20 361702    12.21     4.69    -6.16
##  [6,] 1     6.22 365.00 503.000     1.91 256294    13.47     4.32     3.51
##  [7,] 1     9.75 363.00   1.061     3.22 338320 45819.00     7.84     7.38
##  [8,] 1     6.50 301.00 775.000     6.24 732924     9.39 45812.00     4.53
##  [9,] 1     8.76   0.31   1.204     3.24 354260 45756.00     5.47     5.37
## [10,] 1     8.23 329.00   1.087 45692.00 446132    10.65     6.83 45904.00
## [11,] 1    10.65 381.00   4.577     0.37  40594     7.37     5.39     5.22
## [12,] 1    10.45 371.00   4.305     0.71  75615     7.23     4.38     3.95
## [13,] 1    11.67 398.00   5.514     0.48  55966     4.76     6.39     5.52
## [14,] 1    10.69 421.00   7.617 45840.00 227385     4.37     7.66     6.32
## [15,] 1    10.80 381.00   6.645     0.33  35382     5.98 45782.00     5.56
## [16,] 1     9.67   0.36   5.424     0.51  53031     6.37     6.18     6.22
```

```
## [17,] 1     9.29 322.00   4.448     0.59  62187     6.65     4.57 45997.00
## [18,] 1     8.33 273.00 783.000     3.31 350770    12.53 45783.00     5.56
## [19,] 1     6.87 349.00 633.000     2.75 316354 45817.00     4.97     4.43
## [20,] 1     7.94 354.00   6.668     1.83 223767    10.79     5.84     4.32
## [21,] 1     8.66 352.00 960.000     1.65 189431     9.84     4.33     3.89
## [22,] 1     7.68 368.00 773.000     6.53 729749     9.55     6.57     5.13
## [23,] 1     8.97 327.00   1.151     2.75 302098     9.71     4.83     5.82
## [24,] 1 45999.00 301.00 867.000     2.71 309967 45848.00     4.74     4.84
## [25,] 1     7.59   0.30 629.000     2.14 264905    14.15     2.48     3.19
## [26,] 1     7.82 338.00 414.000     1.44 166356 45882.00     3.65     5.54
## [27,] 1     6.88 305.00   1.079 45902.00 221816    13.93 45748.00     4.66
## [28,] 1     7.42 325.00   1.084     3.95 440348     8.96     5.91     5.32
## [29,] 1     7.77 336.00 680.000     2.33 271776     9.32     5.51     3.24
## [30,] 1     6.13 275.00 673.000     2.83 327078 46008.00     3.25     4.52
## [31,] 1 45813.00   0.28 801.000     2.38 243087    21.61 45964.00     2.31
## [32,] 1    10.77 373.00   2.905 45996.00 560491     5.36 45877.00     7.53
## [33,] 1     6.63 298.00 418.000     1.69 220789    11.78     3.38     4.39
## [34,] 1     5.93 266.00 545.000     2.77 341226    18.76     1.36 45964.00
## [35,] 1    10.51 388.00   8.595 45845.00 790186     4.72     7.62     6.51
## [36,] 1     7.89 331.00 592.000     1.80 211852    10.96     5.37     4.52
## [37,] 1     7.37 345.00 613.000     2.95 327614 45703.00     4.54     8.88
## [38,] 1     8.65 342.00 966.000     2.68 310483     6.71     6.65     5.22
##           X9
##  [1,]  15608
##  [2,]  33454
##  [3,]  56202
##  [4,]  21799
##  [5,]  46979
##  [6,]  18441
##  [7,]  81639
##  [8,]  39099
##  [9,]  22520
## [10,]  18599
## [11,]  34099
## [12,] 310631
## [13,]  55823
## [14,]  66986
## [15,]  39099
## [16,]  29704
## [17,]  36484
## [18,]  21471
## [19,]  20776
## [20,]  18702
## [21,]  20549
## [22,]  26857
## [23,]  56193
## [24,]  17496
## [25,]  16257
## [26,]  19771
## [27,]  14026
## [28,]  70012
## [29,]  15653
## [30,]  21323
## [31,]  14539
```

```
## [32,]  72080
## [33,]  20713
## [34,]  21918
## [35,] 150410
## [36,]  18313
## [37,]  39594
## [38,]  26073
```

```r
class(X)
```

```
## [1] "matrix" "array"
```

3. Beta

```r
beta <- solve(t(X) %*% X) %*% t(X) %*% Y
print(beta)
```

```
##             [,1]
## 1   1.243320e+03
## X1 -1.160623e-02
## X2  1.638422e-01
## X3  2.167799e-04
## X4  4.734132e-03
## X5  4.257792e-04
## X6  9.271238e-03
## X7  6.089117e-03
## X8  1.168165e-03
## X9 -1.574161e-03
```