UAS EVD

Faiz Iqbal I'tishom

2024-12-11

Import Library

```
library(readxl)
library(readr)
library(skedastic)
library(lmtest)
```

```
## Warning: package 'lmtest' was built under R version 4.4.2
```

```
## Loading required package: zoo
```

```
## Warning: package 'zoo' was built under R version 4.4.2
```

```
##
## Attaching package: 'zoo'
```

```
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```
library(regclass)
```

```
## Loading required package: bestglm
```

```
## Warning: package 'bestglm' was built under R version 4.4.2
```

```
## Loading required package: leaps
```

```
## Warning: package 'leaps' was built under R version 4.4.2
```

```
## Loading required package: VGAM
```

```
## Warning: package 'VGAM' was built under R version 4.4.2
```

```
## Loading required package: stats4
```

```
## Loading required package: splines
```

```
##
## Attaching package: 'VGAM'

## The following object is masked from 'package:lmtest':
##
##     lrtest

## Loading required package: rpart

## Warning: package 'rpart' was built under R version 4.4.2

## Loading required package: randomForest

## Warning: package 'randomForest' was built under R version 4.4.2

## randomForest 4.7-1.2

## Type rfNews() to see new features/changes/bug fixes.

## Important regclass change from 1.3:
## All functions that had a . in the name now have an _
## all.correlations -> all_correlations, cor.demo -> cor_demo, etc.
```

```
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 4.4.2

##
## Attaching package: 'ggplot2'

## The following object is masked from 'package:randomForest':
##
##     margin
```

```
library(reshape2)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following object is masked from 'package:randomForest':
##
##     combine

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```r
library(nlme)
```

```
##
## Attaching package: 'nlme'

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

```r
library(ggpubr)
```

```
## Warning: package 'ggpubr' was built under R version 4.4.2
```

```r
library(boot)
```

```
## Warning: package 'boot' was built under R version 4.4.2

##
## Attaching package: 'boot'

## The following objects are masked from 'package:VGAM':
##
##     logit, simplex
```

```r
library(MASS)
```

```
##
## Attaching package: 'MASS'

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

```r
library(robustbase)
```

```
## Warning: package 'robustbase' was built under R version 4.4.2

##
## Attaching package: 'robustbase'

## The following object is masked from 'package:boot':
##
##     salinity
```

```r
library(caret)
```

```
## Warning: package 'caret' was built under R version 4.4.2
```

```
## Loading required package: lattice

##
## Attaching package: 'lattice'

## The following object is masked from 'package:boot':
##
##     melanoma

## The following object is masked from 'package:regclass':
##
##     qq

##
## Attaching package: 'caret'

## The following object is masked from 'package:VGAM':
##
##     predictors
```

```
library(leaps)
library(nnet)
library(ROCR)
```

```
## Warning: package 'ROCR' was built under R version 4.4.2
```

```
library(maps)
```

```
## Warning: package 'maps' was built under R version 4.4.2
```

Data GDP

```
gdp <- read_csv("C:/Users/Lenovo-MPL018/Downloads/Data Fix/gdp.csv")
```

```
## Rows: 66 Columns: 4
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (2): Entity, Code
## dbl (2): Year, GDP per capita, PPP (constant 2017 international $)
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
#View(gdp)
```

Data FDI

```
fdi <- read_csv("C:/Users/Lenovo-MPL018/Downloads/Data Fix/foreign-direct-investment-net-inflows-as-sha
```

```
## Rows: 66 Columns: 4
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (2): Entity, Code
## dbl (2): Year, Foreign direct investment, net inflows (% of GDP)
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
#View(fdi)
```

Data Years of Schooling

```
yos <- read_csv("C:/Users/Lenovo-MPL018/Downloads/Data Fix/Years of Schooling.csv")
```

```
## Rows: 66 Columns: 4
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (2): Entity, Code
## dbl (2): Year, Expected years of schooling
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
#View(yos)
```

Data HDI

```
hdi <- read_csv("C:/Users/Lenovo-MPL018/Downloads/Data Fix/human-development-index.csv")
```

```
## Rows: 66 Columns: 4
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (2): Entity, Code
## dbl (2): Year, Human Development Index
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
#View(hdi)
```

Subset Data

```
negara = hdi$Entity
tahun = hdi$Year
y = hdi$`Human Development Index`
x1 = gdp$`GDP per capita, PPP (constant 2017 international $)`
x2 = fdi$`Foreign direct investment, net inflows (% of GDP)`
x3 = yos$`Expected years of schooling`
```

Subset data kembali untuk mempermudah pengujian

```
subset_data <- data.frame(y, x1, x2, x3)
```

Statistika Deskriptif

```
summary(subset_data)
```

```
##        y                x1               x2               x3
##  Min.   :0.5660   Min.   :  3076   Min.   :-1.753   Min.   :10.19
##  1st Qu.:0.6150   1st Qu.:  7842   1st Qu.: 2.244   1st Qu.:12.06
##  Median :0.7115   Median : 11687   Median : 3.235   Median :13.08
##  Mean   :0.7198   Mean   : 24173   Mean   : 6.303   Mean   :13.29
##  3rd Qu.:0.8017   3rd Qu.: 25914   3rd Qu.: 5.624   3rd Qu.:13.93
##  Max.   :0.9490   Max.   :108036   Max.   :32.691   Max.   :16.90
```

Check Missing Value

```
missing_summary <- sapply(subset_data, function(x) sum(is.na(x)))
missing_percentage <- sapply(subset_data, function(x) mean(is.na(x)) * 100)
missing_values <- data.frame(
  Variable = names(missing_summary),
  MissingCount = missing_summary,
  MissingPercentage = missing_percentage
)
print("Summary of Missing Values:")
```

```
## [1] "Summary of Missing Values:"
```

```
print(missing_values)
```

```
##    Variable MissingCount MissingPercentage
## y         y            0                 0
## x1       x1            0                 0
## x2       x2            0                 0
## x3       x3            0                 0
```

Outlier dengan IQR

```
detect_outliers <- function(column) {
  Q1 <- quantile(column, 0.25, na.rm = TRUE)
  Q3 <- quantile(column, 0.75, na.rm = TRUE)
  IQR <- Q3 - Q1
  lower_bound <- Q1 - 1.5 * IQR
  upper_bound <- Q3 + 1.5 * IQR
  sum(column < lower_bound | column > upper_bound, na.rm = TRUE)
}

outlier_summary <- sapply(subset_data, function(x) {
  if (is.numeric(x)) detect_outliers(x) else NA
})
```

```r
outliers <- data.frame(
  Variable = names(outlier_summary),
  OutlierCount = outlier_summary
)
print("Summary of Outliers:")
```

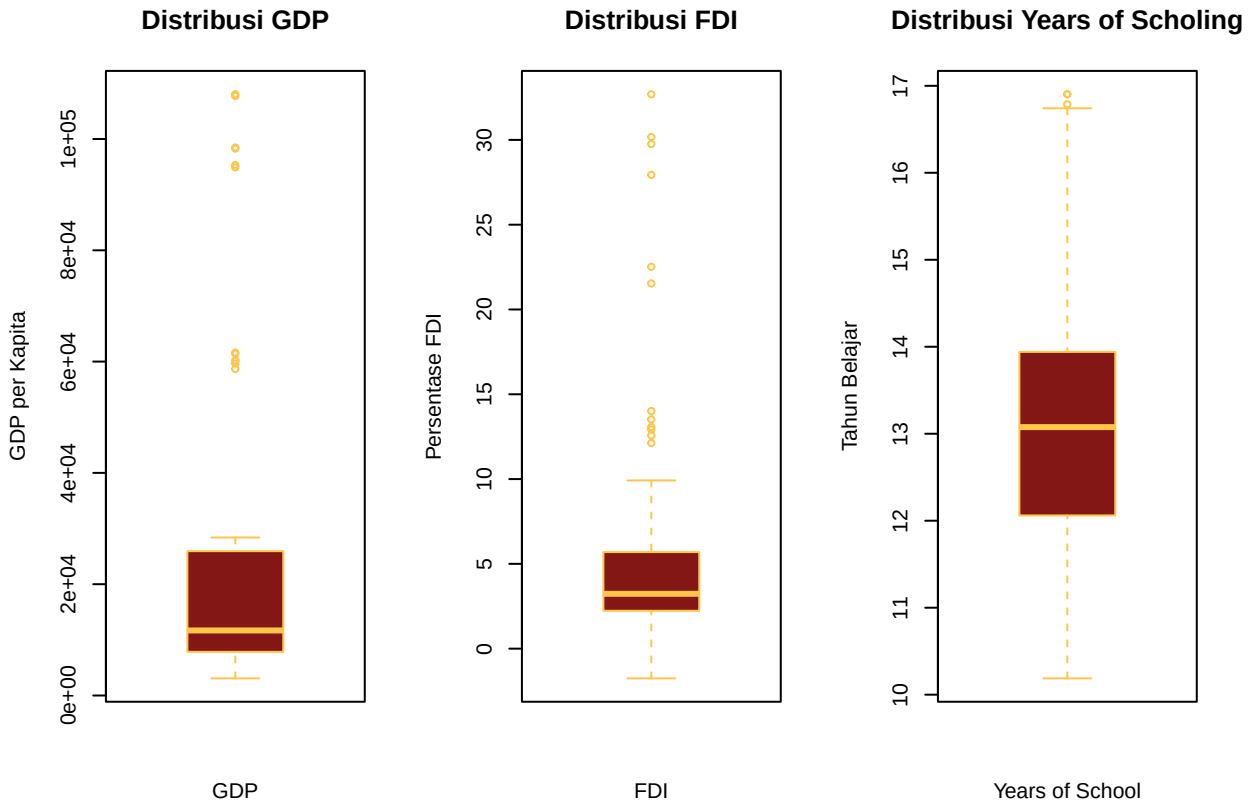
```
## [1] "Summary of Outliers:"
```

```r
print(outliers)
```

```
##    Variable OutlierCount
## y         y            0
## x1       x1           12
## x2       x2           12
## x3       x3            5
```

Outlier dengan Boxplot

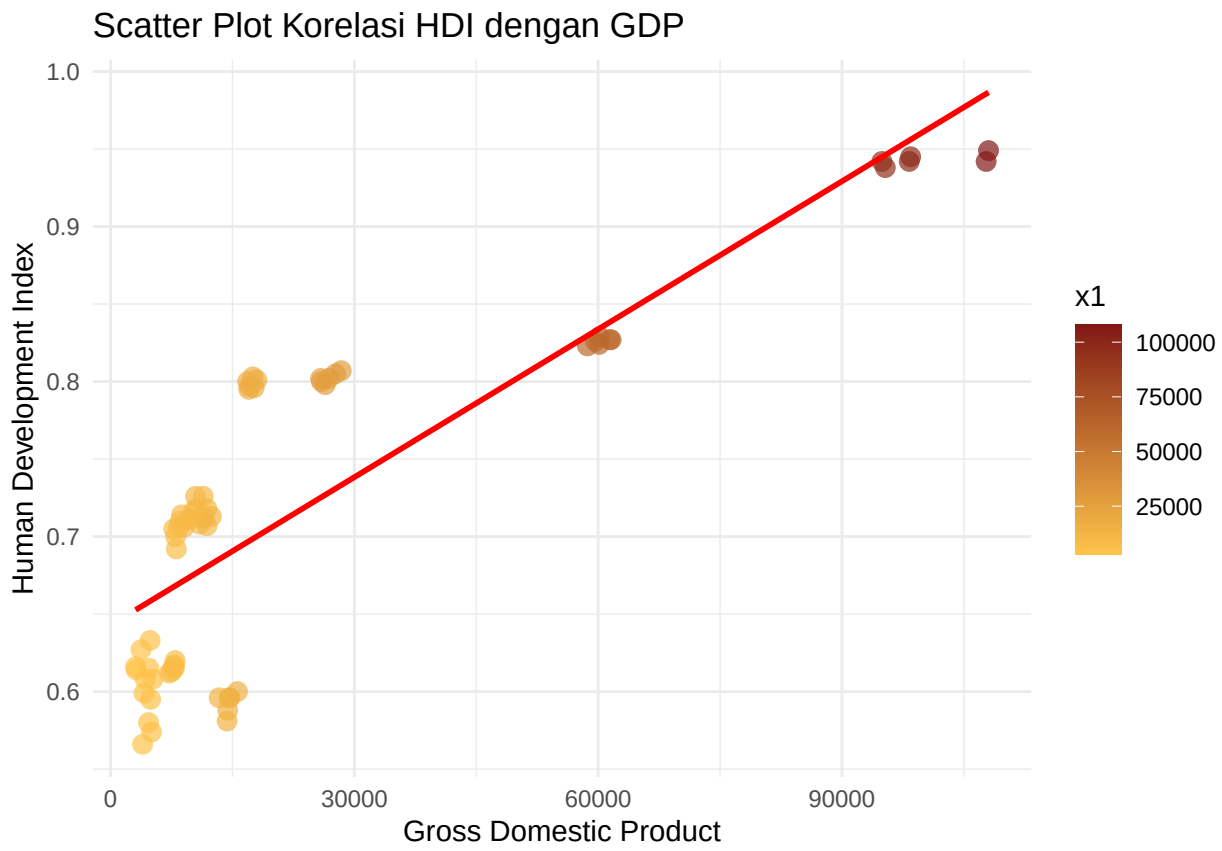
```r
par(mfrow=c(1,3))

 boxplot(subset_data$x1,
 main="Distribusi GDP",
 xlab="GDP",
 ylab="GDP per Kapita",
 col="#821716",
 border="#ffc54a",
 horizontal=FALSE)

 boxplot(subset_data$x2,
 main="Distribusi FDI",
 xlab="FDI",
 ylab="Persentase FDI",
 col="#821716",
 border="#ffc54a",
 horizontal=FALSE)

 boxplot(subset_data$x3,
 main="Distribusi Years of Scholing",
 xlab="Years of School",
 ylab="Tahun Belajar",
 col="#821716",
 border="#ffc54a",
 horizontal=FALSE)
```



Scatter Plot

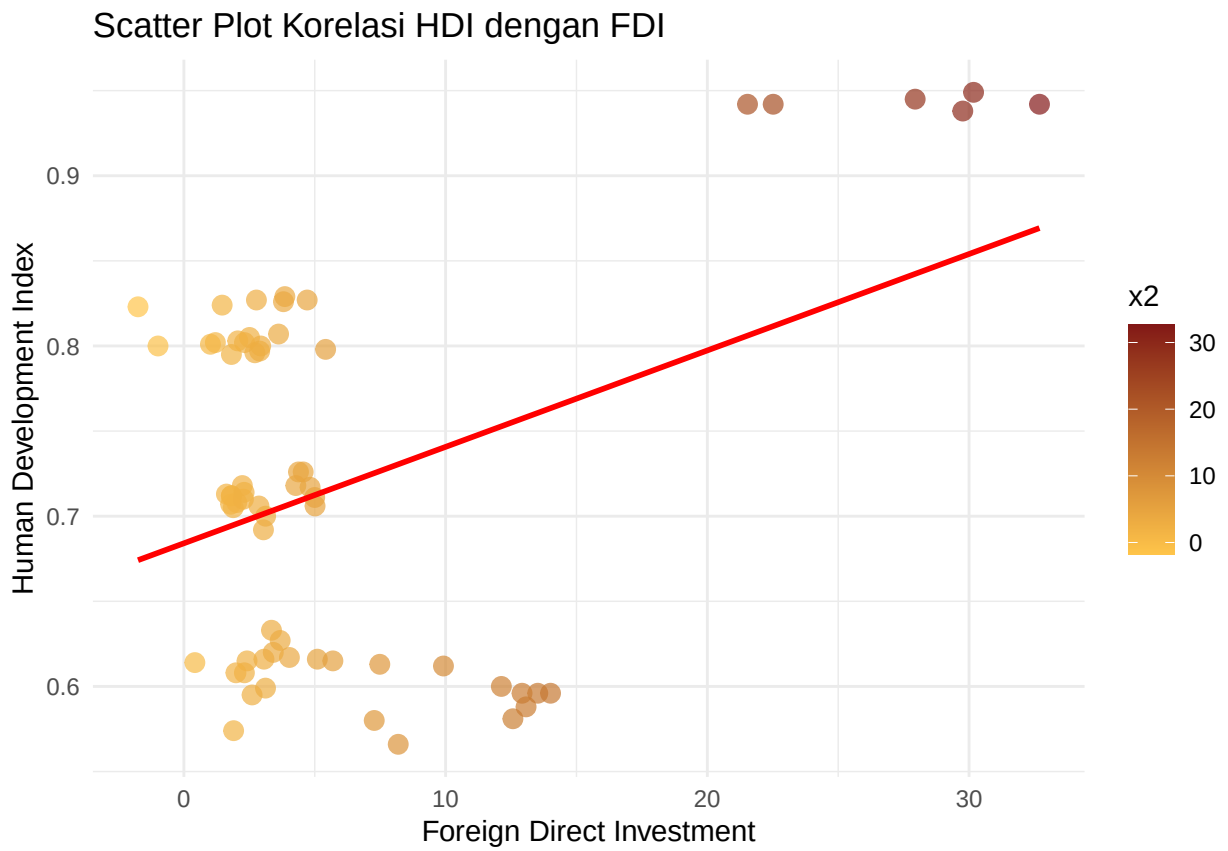
```
par(mfrow=c(1,3))

ggplot(subset_data, aes(x = x1, y = y, color = x1)) +
  geom_point(size = 3, alpha = 0.7) +
  geom_smooth(method = "lm", se = FALSE, color = "red") +
  scale_color_gradient(low = "#ffc54a", high = "#821716") +
  labs(title = "Scatter Plot Korelasi HDI dengan GDP", x = "Gross Domestic Product", y = "Human Develop
  theme_minimal()
```

```
## `geom_smooth()` using formula = 'y ~ x'
```



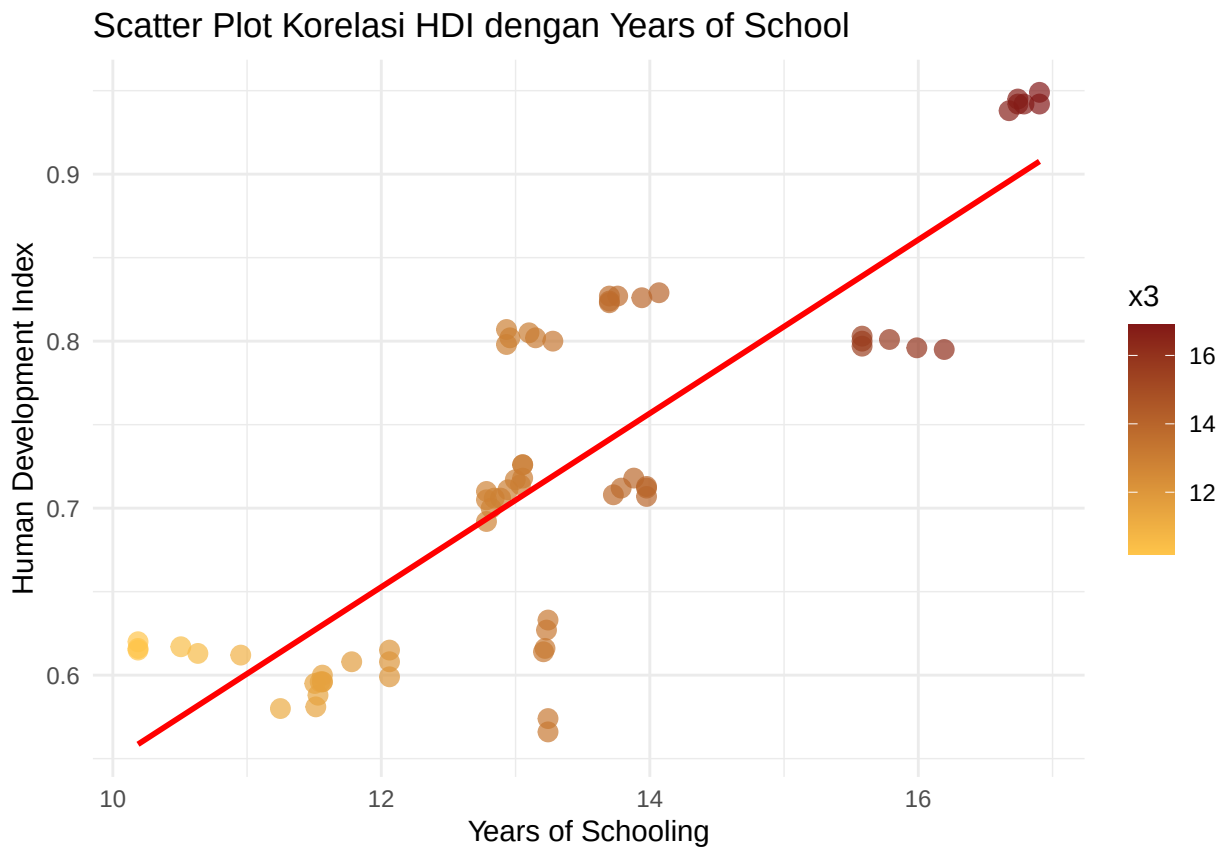

```
#scatter plot y vs x2 (FDI)
ggplot(subset_data, aes(x = x2, y = y, color = x2)) +
  geom_point(size = 3, alpha = 0.7) +
  geom_smooth(method = "lm", se = FALSE, color = "red") +
  scale_color_gradient(low = "#ffc54a", high = "#821716") +
  labs(title = "Scatter Plot Korelasi HDI dengan FDI", x = "Foreign Direct Investment", y = "Human Devel
  theme_minimal()
```

```
## `geom_smooth()` using formula = 'y ~ x'
```




```
#scatter plot y vs x3 (YOS)
ggplot(subset_data, aes(x = x3, y = y, color = x3)) +
  geom_point(size = 3, alpha = 0.7) +
  geom_smooth(method = "lm", se = FALSE, color = "red") +
  scale_color_gradient(low = "#ffc54a", high = "#821716") +
  labs(title = "Scatter Plot Korelasi HDI dengan Years of School", x = "Years of Schooling", y = "Human
  theme_minimal()
```

```
## `geom_smooth()` using formula = 'y ~ x'
```

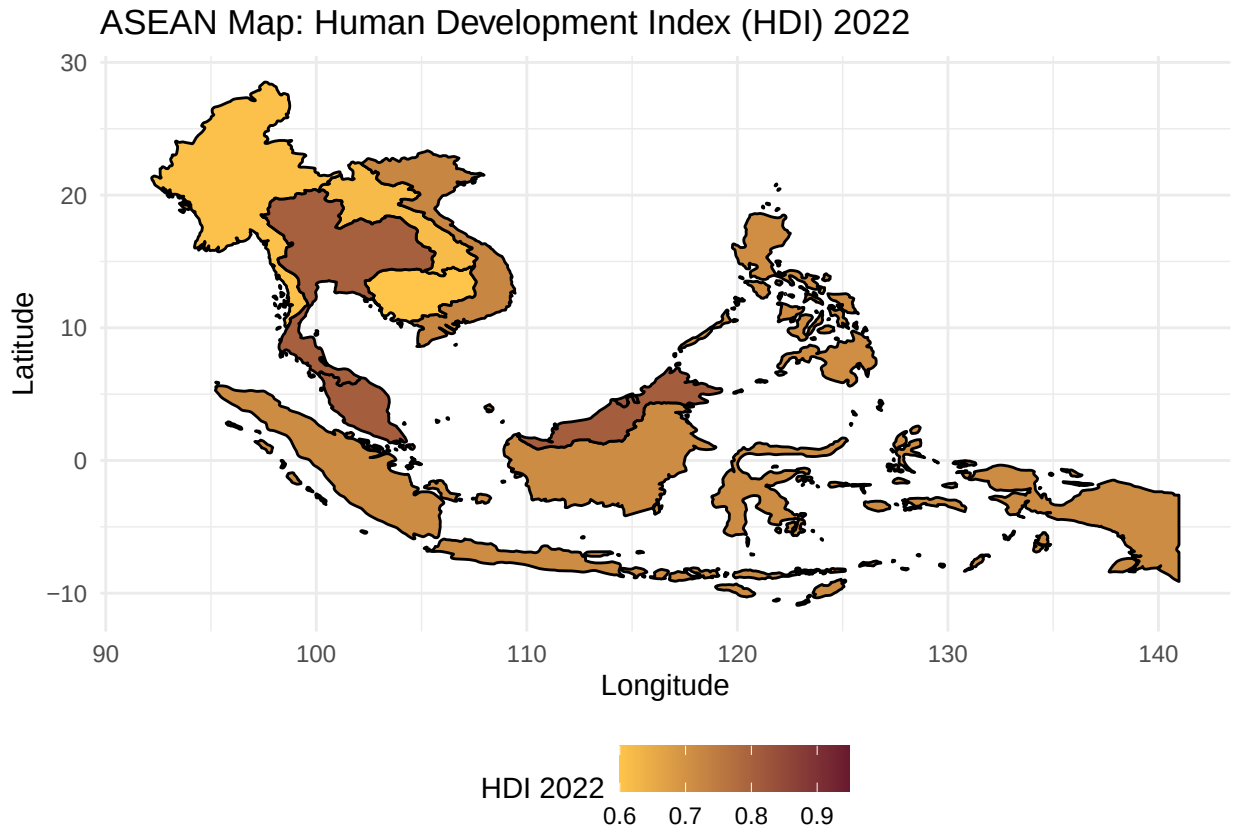



```
world_map <- map_data("world")

asean_map <- subset(world_map, region %in% c("Indonesia", "Malaysia", "Brunei", "Singapore", "Thailand"
negara <- hdi$Entity
tahun <- hdi$Year
y = hdi$`Human Development Index`
x1 = gdp$`GDP per capita, PPP (constant 2017 international $)`
x2 = fdi$`Foreign direct investment, net inflows (% of GDP)`
x3 = yos$`Expected years of schooling`

subset_data <- data.frame(negara, y, x1, x2, x3)

subset_2022 <- subset_data %>%
  filter(tahun == 2022)

subset_2022 <- subset_2022 %>%
  rename(region = negara)

asean_map <- asean_map %>%
  left_join(subset_2022, by = "region")

head(asean_map)
```

```
##       long      lat group order region subregion     y      x1        x2
## 1 115.0267 4.899707   240 14053 Brunei      East 0.823 58669.9 -1.752934
```

```
## 2 115.1400 4.899756   240 14054 Brunei      East 0.823 58669.9 -1.752934
## 3 115.1684 4.866699   240 14055 Brunei      East 0.823 58669.9 -1.752934
## 4 115.2279 4.750586   240 14056 Brunei      East 0.823 58669.9 -1.752934
## 5 115.2667 4.633984   240 14057 Brunei      East 0.823 58669.9 -1.752934
## 6 115.2793 4.456347   240 14058 Brunei      East 0.823 58669.9 -1.752934
##         x3
## 1 13.69862
## 2 13.69862
## 3 13.69862
## 4 13.69862
## 5 13.69862
## 6 13.69862
```

```
ggplot(data = asean_map, aes(x = long, y = lat, group = group, fill = y)) +
  geom_polygon(color = "black") +
  scale_fill_gradient(low = "#ffc54a", high = "#69192F", name = "HDI 2022") +
  labs(title = "ASEAN Map: Human Development Index (HDI) 2022",
       x = "Longitude", y = "Latitude") +
  theme_minimal() +
  theme(legend.position = "bottom")
```



Model OLS

```
ols <- lm(y ~ x1 + x2 + x3, data = subset_data)
ols
```

```
##
## Call:
## lm(formula = y ~ x1 + x2 + x3, data = subset_data)
##
## Coefficients:
## (Intercept)           x1           x2           x3
##   3.211e-01    2.983e-06   -4.716e-03    2.681e-02
```

```
summary(ols)
```

```
##
## Call:
## lm(formula = y ~ x1 + x2 + x3, data = subset_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.10822 -0.02754 -0.00129  0.02599  0.07692
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  3.211e-01  4.698e-02   6.834 4.16e-09 ***
## x1           2.983e-06  3.035e-07   9.826 2.93e-14 ***
## x2          -4.716e-03  9.143e-04  -5.158 2.78e-06 ***
## x3           2.681e-02  3.759e-03   7.133 1.27e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.03847 on 62 degrees of freedom
## Multiple R-squared:  0.8821, Adjusted R-squared:  0.8764
## F-statistic: 154.7 on 3 and 62 DF,  p-value: < 2.2e-16
```

Uji Homoskedastisitas

```
glejser(ols)
```

```
## # A tibble: 1 x 4
##   statistic p.value parameter alternative
##       <dbl>   <dbl>     <dbl> <chr>
## 1      7.11  0.0684         3 greater
```

Uji Autokorelasi

```
dwtest(y ~ x1 + x2 + x3, data = subset_data)
```

```
##
##  Durbin-Watson test
##
## data:  y ~ x1 + x2 + x3
## DW = 0.39888, p-value < 2.2e-16
## alternative hypothesis: true autocorrelation is greater than 0
```

Uji Multikolinearitas

```r
VIF(ols)
```

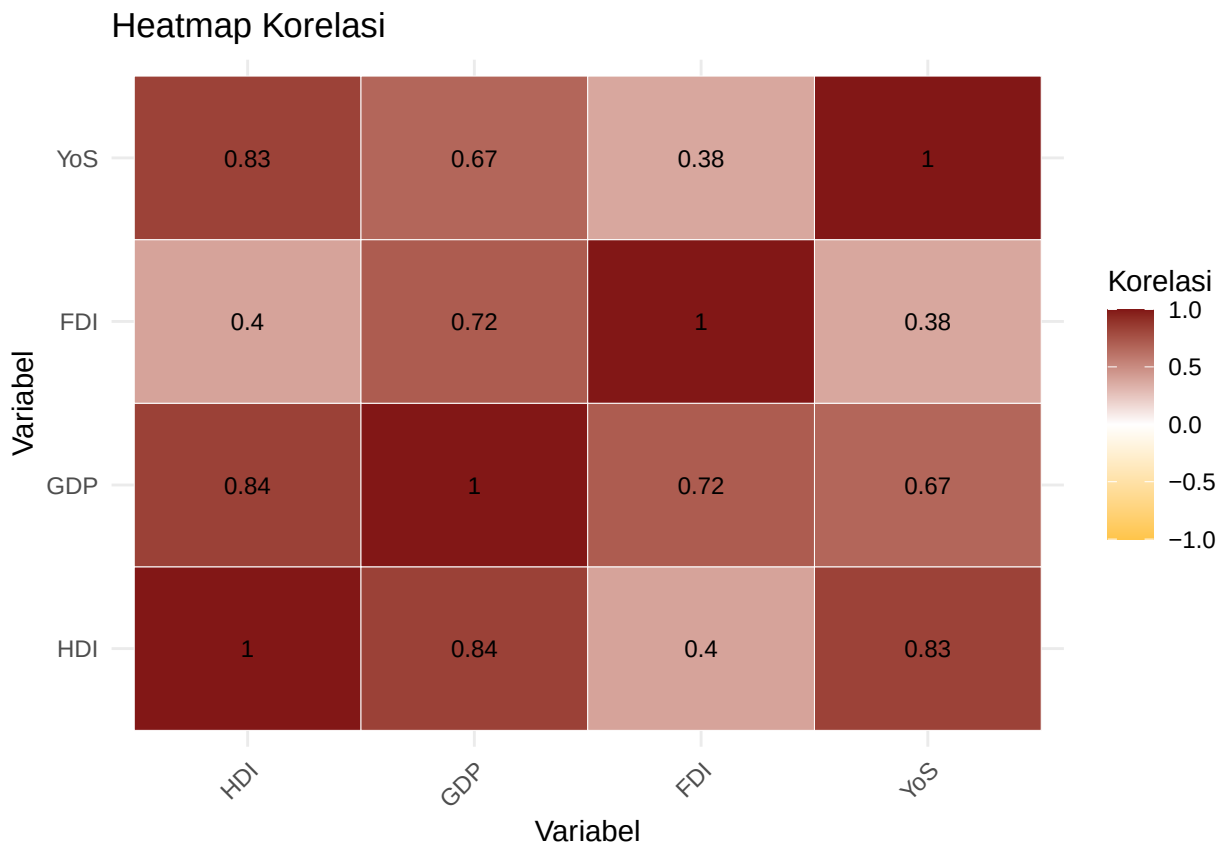
```
##       x1       x2       x3
## 3.350128 2.143404 1.901018
```

```r
numerik_data <- subset_data[sapply(subset_data, is.numeric)]
cor_matrix <- cor(numerik_data, use = "complete.obs")
```

```r
cor_data <- melt(cor_matrix)
colnames(cor_data) <- c("Variable1", "Variable2", "Correlation")
```

Heatmap

```r
heatmap_plot <- ggplot(cor_data, aes(x = Variable1, y = Variable2, fill = Correlation)) +
  geom_tile(color = "white") +
  scale_fill_gradient2(low = "#ffc54a", high = "#821716", mid = "white",
                       midpoint = 0, limit = c(-1, 1), space = "Lab",
                       name = "Korelasi") +
  geom_text(aes(label = round(Correlation, 2)), color = "black", size = 3) +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust = 1)) +
  labs(title = "Heatmap Korelasi", x = "Variabel", y = "Variabel") +
  scale_x_discrete(labels = c("y" = "HDI", "x1" = "GDP", "x2" = "FDI", "x3" = "YoS")) +
  scale_y_discrete(labels = c("y" = "HDI", "x1" = "GDP", "x2" = "FDI", "x3" = "YoS"))

print(heatmap_plot)
```

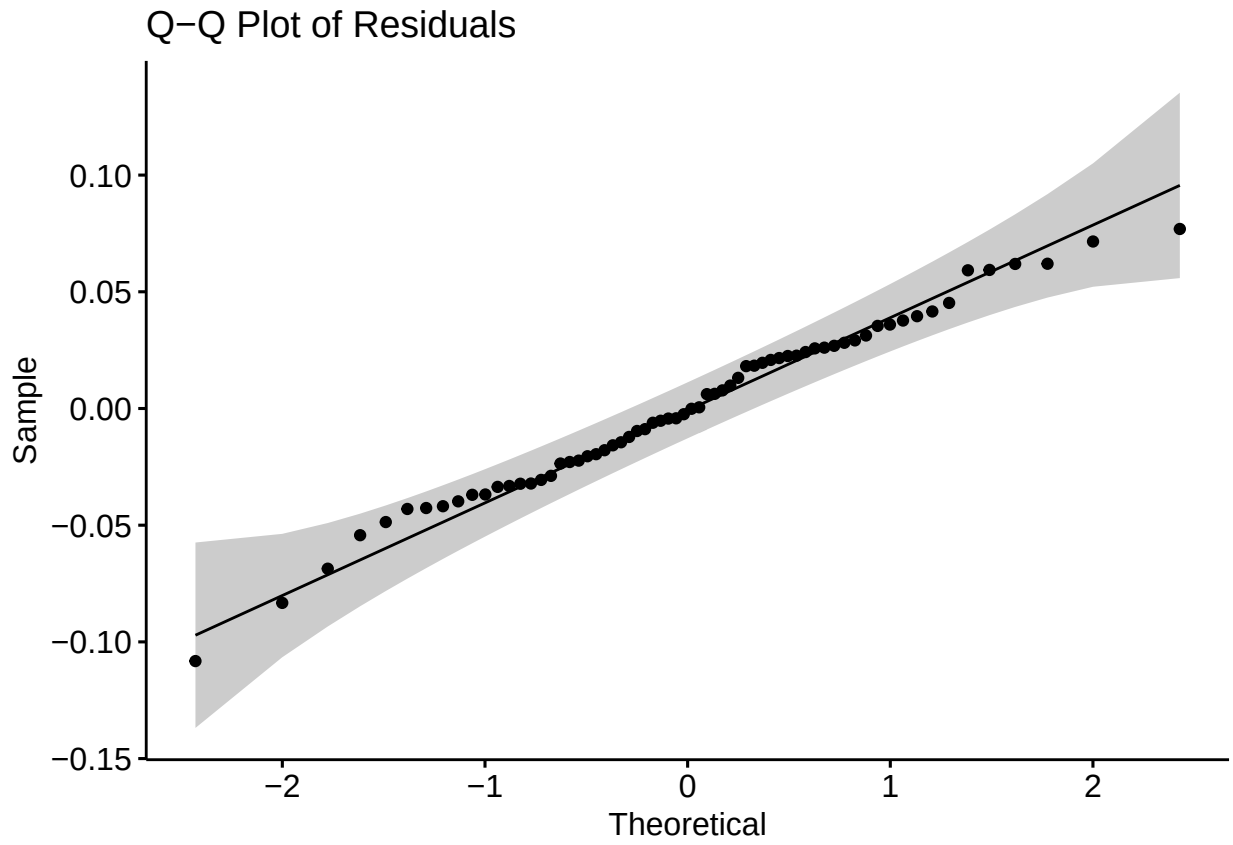


Uji Normalitas

```
residuals <- resid(ols)

shapiro_test <- shapiro.test(residuals)
print(shapiro_test)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  residuals
## W = 0.98621, p-value = 0.6766
```

```
ggqqplot(residuals, title = "Q-Q Plot of Residuals")
```



Model Logistik

```
data_logistik <- gdp %>%
  inner_join(fdi, by = c("Entity", "Year")) %>%
  inner_join(yos, by = c("Entity", "Year")) %>%
  inner_join(hdi, by = c("Entity", "Year"))
```

Subset Data Logistik

```
data_logistik <- data_logistik %>%
  rename(GDP = `GDP per capita, PPP (constant 2017 international $)`,
         FDI = `Foreign direct investment, net inflows (% of GDP)`,
         YOS = `Expected years of schooling`,
         HDI = `Human Development Index`)
```

```
View(data_logistik)
```

Pengkategorian berdasarkan skor HDI

```
data_logistik <- data_logistik %>%
  mutate(HDI_Category = case_when(
    HDI < 0.55 ~ 0,
    HDI >= 0.55 & HDI < 0.70 ~ 1,
    HDI >= 0.70 & HDI < 0.90 ~ 2,
    HDI >= 0.90 ~ 3
  ))
```

Penglabelan

```
data_logistik$HDI_Category <- factor(data_logistik$HDI_Category, levels = c(0, 1, 2, 3),
                                     labels = c("Rendah", "Menengah", "Tinggi", "Sangat Tinggi"))
```

Statistika logistik

```
summary(data_logistik)
```

```
##     Entity             Code.x               Year           GDP
##  Length:60          Length:60          Min.   :2017   Min.   :  3076
##  Class :character   Class :character   1st Qu.:2018   1st Qu.:  7771
##  Mode  :character   Mode  :character   Median :2020   Median : 11169
##                                        Mean   :2020   Mean   : 25140
##                                        3rd Qu.:2021   3rd Qu.: 26514
##                                        Max.   :2022   Max.   :108036
##     Code.y               FDI           Code.x.x              YOS
##  Length:60          Min.   :-1.753   Length:60          Min.   :10.19
##  Class :character   1st Qu.: 2.048   Class :character   1st Qu.:12.81
##  Mode  :character   Median : 3.050   Mode  :character   Median :13.21
##                     Mean   : 5.629                      Mean   :13.46
##                     3rd Qu.: 4.865                      3rd Qu.:13.98
##                     Max.   :32.691                      Max.   :16.90
##    Code.y.y             HDI               HDI_Category
##  Length:60          Min.   :0.5660   Rendah       : 0
##  Class :character   1st Qu.:0.6192   Menengah     :19
##  Mode  :character   Median :0.7135   Tinggi       :35
##                     Mean   :0.7325   Sangat Tinggi: 6
##                     3rd Qu.:0.8023
##                     Max.   :0.9490
```

Drop Kolom Tidak Berguna

```
drop_columns <- c("Code.x", "Code.y", "Code.x.x", "Code.y.y")
```

Membuat kolom kategori baru berdasarkan kriteria yang ditentukan

```
table(data_logistik$HDI_Category)
```

```
##
##        Rendah      Menengah        Tinggi Sangat Tinggi
##             0            19            35             6
```

Membuat data train

```
set.seed(123) # Untuk replikasi hasil
trainIndex <- createDataPartition(data_logistik$HDI_Category, p = 0.8, list = FALSE)
```

```
## Warning in createDataPartition(data_logistik$HDI_Category, p = 0.8, list =
## FALSE): Some classes have no records ( Rendah ) and these will be ignored
```

```r
trainData <- data_logistik[trainIndex, ]
testData <- data_logistik[-trainIndex, ]
```

membuat model multinomial

```r
multinom_model <- multinom(HDI_Category ~ GDP + FDI + YOS, data = trainData)
```

```
## Warning in multinom(HDI_Category ~ GDP + FDI + YOS, data = trainData): group
## 'Rendah' is empty
```

```
## # weights:  15 (8 variable)
## initial  value 53.832002
## iter  10 value 14.965779
## iter  20 value 5.024363
## iter  30 value 2.123252
## iter  40 value 2.121561
## iter  50 value 2.109523
## iter  60 value 2.090559
## iter  70 value 2.076995
## iter  80 value 2.075737
## iter  90 value 2.017651
## iter 100 value 2.013734
## final  value 2.013734
## stopped after 100 iterations
```

Menampilkan hasilnya

```r
summary(multinom_model)
```

```
## Call:
## multinom(formula = HDI_Category ~ GDP + FDI + YOS, data = trainData)
##
## Coefficients:
##               (Intercept)         GDP       FDI        YOS
## Tinggi        -129.487185 0.004630179 -1.935285   7.721643
## Sangat Tinggi   -5.311504 0.005903294  8.015832 -16.031849
##
## Std. Errors:
##                (Intercept)          GDP          FDI          YOS
## Tinggi        1.752973e-08 1.430782e-04 5.453577e-08 2.246260e-07
## Sangat Tinggi 8.057560e-38 7.640962e-33 1.810861e-36 1.352051e-36
##
## Residual Deviance: 4.027468
## AIC: 20.02747
```

Transformasi koefisien

```r
exp(coef(multinom_model))
```

```
##                (Intercept)      GDP          FDI          YOS
## Tinggi        5.813397e-57 1.004641    0.1443832 2.256665e+03
## Sangat Tinggi 4.934500e-03 1.005921 3028.5281178 1.090075e-07
```

membuat data predict

```
testData$Predicted <- predict(multinom_model, newdata = testData, type = "class")
```

Menampilkan hasil matrix regresi logistik

```
conf_matrix <- confusionMatrix(testData$Predicted, testData$HDI_Category)
```

```
## Warning in levels(reference) != levels(data): longer object length is not a
## multiple of shorter object length
```

```
## Warning in confusionMatrix.default(testData$Predicted, testData$HDI_Category):
## Levels are not in the same order for reference and data. Refactoring data to
## match.
```

```
## Registered S3 methods overwritten by 'proxy':
##   method               from
##   print.registry_field registry
##   print.registry_entry registry
```

```
print(conf_matrix)
```

```
## Confusion Matrix and Statistics
##
##                Reference
## Prediction      Rendah Menengah Tinggi Sangat Tinggi
##   Rendah             0        0      0             0
##   Menengah           0        3      0             0
##   Tinggi             0        0      7             0
##   Sangat Tinggi      0        0      0             1
##
## Overall Statistics
##
##                Accuracy : 1
##                  95% CI : (0.7151, 1)
##     No Information Rate : 0.6364
##     P-Value [Acc > NIR] : 0.00693
##
##                   Kappa : 1
##
##  Mcnemar's Test P-Value : NA
##
## Statistics by Class:
##
##                      Class: Rendah Class: Menengah Class: Tinggi
## Sensitivity                     NA          1.0000        1.0000
## Specificity                      1          1.0000        1.0000
## Pos Pred Value                  NA          1.0000        1.0000
## Neg Pred Value                  NA          1.0000        1.0000
## Prevalence                       0          0.2727        0.6364
## Detection Rate                   0          0.2727        0.6364
## Detection Prevalence             0          0.2727        0.6364
```

```
## Balanced Accuracy               NA          1.0000        1.0000
##                       Class: Sangat Tinggi
## Sensitivity                        1.00000
## Specificity                        1.00000
## Pos Pred Value                     1.00000
## Neg Pred Value                     1.00000
## Prevalence                         0.09091
## Detection Rate                     0.09091
## Detection Prevalence               0.09091
## Balanced Accuracy                  1.00000
```