R Notebook

Hide

```
library(readxl)


file_path <- "C:/Users/KHOOBTEK/Desktop/Final_Data_Translated.xlsx"


data <- read_excel(file_path)


head(data)
```

Year	Overall Index	Interest Rate	Inflation Rate	Exchan
<dbl>	<dbl>	<dbl>	<dbl>	
1379	2978.30	14.79	0.140	
1380	3758.80	13.80	0.130	
1381	5062.80	13.00	0.158	
1382	11379.37	13.00	0.156	
1383	12113.01	13.00	0.152	
1384	9459.36	13.00	0.104	

6 rows

Hide

```
NA
```

Hide

```


summary(data)
```

```
      Year       Overall Index      Interest Rate    Inflation Rate    Exchange R
ate         Oil Price
 Min.   :1379   Min.   :   2978   Min.   :13.00   Min.   :0.0900   Min.   :
800.0   Min.   : 18.00
```

```
 1st Qu.:1385   1st Qu.:  10017   1st Qu.:14.95   1st Qu.:0.1227   1st Qu.: 
917.5   1st Qu.: 34.75  
 Median :1390   Median :  31973   Median :16.00   Median :0.1570   Median : 1
905.5   Median : 54.50  
 Mean   :1390   Mean   : 337073   Mean   :17.36   Mean   :0.2217   Mean   : 7
001.6   Mean   : 60.25  
 3rd Qu.:1396   3rd Qu.: 116882   3rd Qu.:20.00   3rd Qu.:0.3155   3rd Qu.: 4
950.0   3rd Qu.: 83.75  
 Max.   :1402   Max.   :2195092   Max.   :25.00   Max.   :0.5230   Max.   :50
900.0   Max.   :108.00  
```

Hide

```r
if (!requireNamespace("nortest", quietly = TRUE)) {
  install.packages("nortest")
}
library(nortest)

for (var in c("Overall Index", "Interest Rate", "Inflation Rate", "Exchange R
ate", "Oil Price")) {
  cat("\n==============================\n")
  cat("Variable:", var, "\n")
  cat("==============================\n\n")

  ad_result <- ad.test(data[[var]])
  print(ad_result)
}
```

```
==============================
Variable: Overall Index
==============================



    Anderson-Darling normality test


data:  data[[var]]
A = 4.8328, p-value = 2.293e-12
```

```
==============================
Variable: Interest Rate
==============================


    Anderson-Darling normality test

data:  data[[var]]
A = 0.89836, p-value = 0.01833


==============================
Variable: Inflation Rate
==============================


    Anderson-Darling normality test

data:  data[[var]]
A = 1.2371, p-value = 0.002512


==============================
Variable: Exchange Rate
==============================


    Anderson-Darling normality test

data:  data[[var]]
A = 3.8774, p-value = 5.5e-10
```

```
==============================
Variable: Oil Price
==============================


    Anderson-Darling normality test

data:  data[[var]]
A = 0.50074, p-value = 0.188
```

Hide

```r
for (var in c("Overall Index", "Interest Rate", "Inflation Rate", "Exchange Rate", "Oil Price")) {
  cat("\n==============================\n")
  cat("Variable:", var, "\n")
  cat("==============================\n\n")

  ks_result <- ks.test(data[[var]], "pnorm", mean(data[[var]], na.rm = TRUE), sd(data[[var]], na.rm = TRUE))
  print(ks_result)
}
```

```
==============================
Variable: Overall Index
==============================


    Exact one-sample Kolmogorov-Smirnov test

data:  data[[var]]
D = 0.39358, p-value = 0.0007315
alternative hypothesis: two-sided
```

```
==============================
Variable: Interest Rate
==============================
Warning: ties should not be present for the one-sample Kolmogorov-Smirnov tes
t


	Asymptotic one-sample Kolmogorov-Smirnov test


data:  data[[var]]
D = 0.18137, p-value = 0.4088
alternative hypothesis: two-sided



==============================
Variable: Inflation Rate
==============================
Warning: ties should not be present for the one-sample Kolmogorov-Smirnov tes
t


	Asymptotic one-sample Kolmogorov-Smirnov test


data:  data[[var]]
D = 0.23483, p-value = 0.1417
alternative hypothesis: two-sided



==============================
Variable: Exchange Rate
==============================



	Exact one-sample Kolmogorov-Smirnov test
```

```
data:  data[[var]]
D = 0.3353, p-value = 0.006578
alternative hypothesis: two-sided



==============================
Variable: Oil Price
==============================
Warning: ties should not be present for the one-sample Kolmogorov-Smirnov tes
t


    Asymptotic one-sample Kolmogorov-Smirnov test


data:  data[[var]]
D = 0.12972, p-value = 0.8141
alternative hypothesis: two-sided
```

Hide

```
for (var in c("Overall Index", "Interest Rate", "Inflation Rate", "Exchange R
ate", "Oil Price")) {
  cat("\n==============================\n")
  cat("Variable:", var, "\n")
  cat("==============================\n\n")

  shapiro_result <- shapiro.test(data[[var]])
  print(shapiro_result)
}
```

Hide

```
data$Year_Group <- cut(
  data$Year,
  breaks = c(1379, 1383, 1387, 1391, 1395, 1399, 1403),
  labels = c("1379-1382", "1383-1386", "1387-1390", "1391-1394", "1395-1398",
"1399-1402"),
  include.lowest = TRUE,
  right = FALSE
)
```

Hide

```
data$Year_Group <- cut(
  data$Year,
  breaks = c(1379, 1385, 1391, 1397, 1403),  # مرزهای بازه‌ها برای 4 گروه 6 تا
یی
  labels = c("1379-1384", "1385-1390", "1391-1396", "1397-1402"),  # برچسب‌ها
  include.lowest = TRUE,  # شامل اولین مقدار
  right = FALSE           # کران راست باز نیست (چپ بسته است)
)
```

Hide

```
data$Year_Group <- cut(
  data$Year,
  breaks = seq(1379, 1403, by = 3),  # مرزهای بازه‌ها برای 8 گروه 3 تایی
  labels = c("1379-1381", "1382-1384", "1385-1387", "1388-1390",
             "1391-1393", "1394-1396", "1397-1399", "1400-1402"),  # برچسب‌ها
  include.lowest = TRUE,  # شامل اولین مقدار
  right = FALSE           # کران راست باز نیست (چپ بسته است)
)
```

Hide

```
table(data$Year_Group)
```

```
1379-1381 1382-1384 1385-1387 1388-1390 1391-1393 1394-1396 1397-1399 1400-14
02
        3         3         3         3         3         3         3
3
```

Hide

```
numeric_vars <- c("Overall Index", "Interest Rate", "Inflation Rate", "Exchan
ge Rate", "Oil Price")


for (var in numeric_vars) {
  cat("\n==============================\n")
  cat("Variable:", var, "\n")
  cat("==============================\n\n")

  cat(">>> Bartlett Test <<<\n")
  print(
    bartlett.test(
      as.formula(paste0("`", var, "` ~ Year_Group")),
      data = data
    )
  )
  cat("\n\n")
}
```

```

==============================
Variable: Overall Index
==============================

>>> Bartlett Test <<<

    Bartlett test of homogeneity of variances

data:  Overall Index by Year_Group
```

```
Bartlett's K-squared = 81.114, df = 7, p-value = 8.164e-15




==============================
Variable: Interest Rate
==============================

>>> Bartlett Test <<<

    Bartlett test of homogeneity of variances

data:  Interest Rate by Year_Group
Bartlett's K-squared = Inf, df = 7, p-value < 2.2e-16




==============================
Variable: Inflation Rate
==============================

>>> Bartlett Test <<<

    Bartlett test of homogeneity of variances

data:  Inflation Rate by Year_Group
Bartlett's K-squared = 8.7885, df = 7, p-value = 0.2682




==============================
Variable: Exchange Rate
==============================

>>> Bartlett Test <<<

    Bartlett test of homogeneity of variances

data:  Exchange Rate by Year_Group
Bartlett's K-squared = 82.511, df = 7, p-value = 4.233e-15




==============================
Variable: Oil Price
==============================

>>> Bartlett Test <<<

    Bartlett test of homogeneity of variances

data:  Oil Price by Year_Group
Bartlett's K-squared = 5.4547, df = 7, p-value = 0.6047
```

Hide

```
# نصب و بارگذاری بسته car (در صورت نیاز)
if (!requireNamespace("car", quietly = TRUE)) {
  install.packages("car")
}
library(car)
```

```
Loading required package: carData
```

Hide

```r
# آزمون Levene برای هر متغیر
numeric_vars <- c("Overall Index", "Interest Rate", "Inflation Rate", "Exchange Rate", "Oil Price")


for (var in numeric_vars) {
  cat("\n==============================\n")
  cat("Variable:", var, "\n")
  cat("==============================\n\n")

  cat(">>> Levene Test <<<\n")
  print(
    leveneTest(
      as.formula(paste0("`", var, "` ~ Year_Group")),
      data = data
    )
  )
  cat("\n\n")
}
```

```

==============================
Variable: Overall Index
==============================


>>> Levene Test <<<
Levene's Test for Homogeneity of Variance (center = median)
      Df F value  Pr(>F)
group  7  2.2105 0.08954 .
      16
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1



==============================
Variable: Interest Rate
==============================

>>> Levene Test <<<
Levene's Test for Homogeneity of Variance (center = median)
      Df F value Pr(>F)
group  7  1.5764 0.2128
      16



==============================
Variable: Inflation Rate
==============================

>>> Levene Test <<<
Levene's Test for Homogeneity of Variance (center = median)
      Df F value Pr(>F)
group  7  0.6186 0.7334
      16



==============================
Variable: Exchange Rate
==============================

>>> Levene Test <<<
Levene's Test for Homogeneity of Variance (center = median)
      Df F value Pr(>F)
group  7  1.1343 0.3904
      16
```

```
==============================
Variable: Oil Price
==============================

>>> Levene Test <<<
Levene's Test for Homogeneity of Variance (center = median)
      Df F value Pr(>F)
group  7  0.3057  0.941
      16
```

Hide

```
# آزمون Fligner-Killeen برای هر متغیر
for (var in numeric_vars) {
  cat("\n==============================\n")
  cat("Variable:", var, "\n")
  cat("==============================\n\n")

  cat(">>> Fligner-Killeen Test <<<\n")
  print(
    fligner.test(
      as.formula(paste0("`", var, "` ~ Year_Group")),
      data = data
    )
  )
  cat("\n\n")
}
```

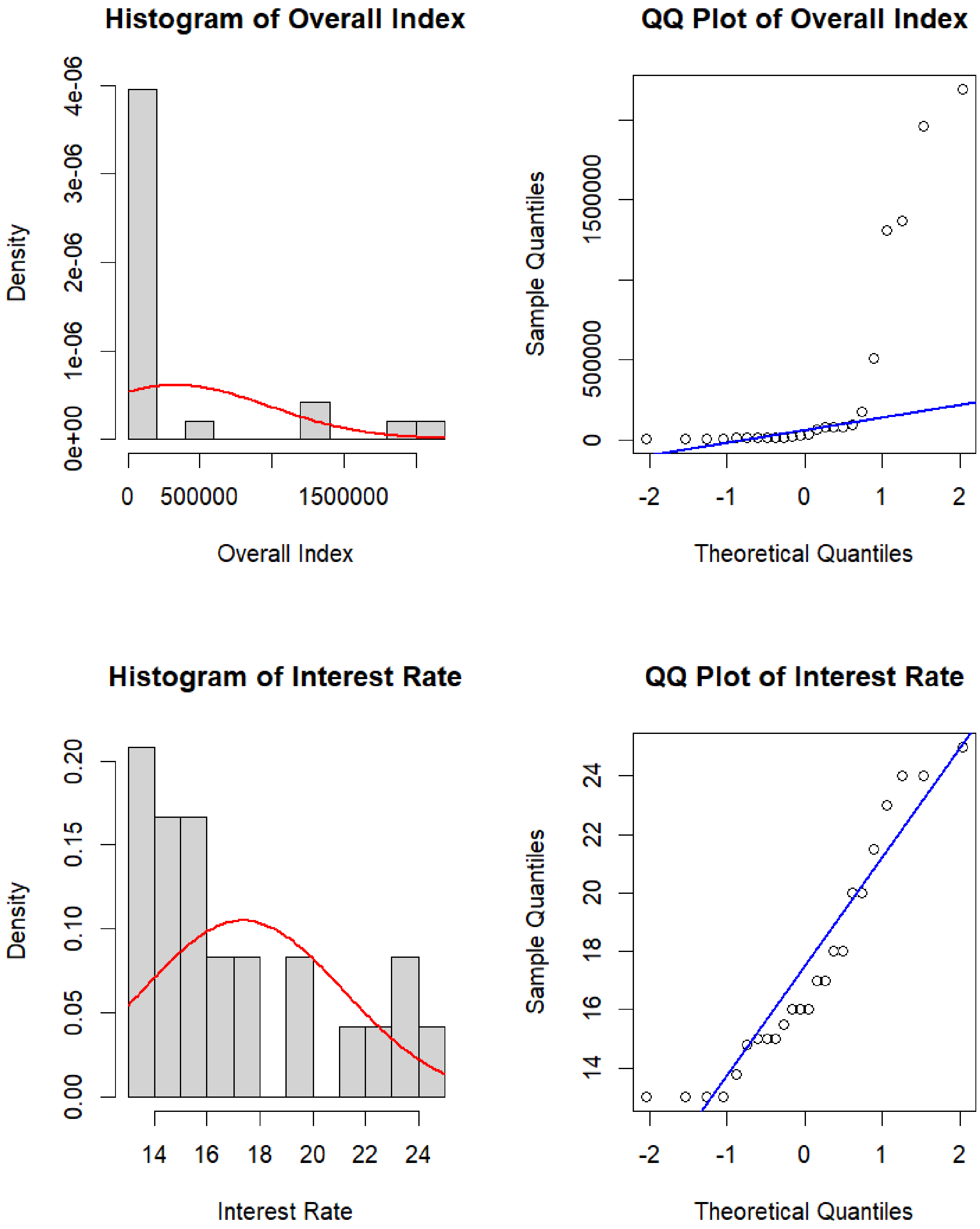
```
==============================
Variable: Overall Index
==============================
>>> Fligner-Killeen Test <<<

    Fligner-Killeen test of homogeneity of variances

data:  Overall Index by Year_Group
Fligner-Killeen:med chi-squared = 6.623, df = 7, p-value = 0.4692




==============================
Variable: Interest Rate
==============================

>>> Fligner-Killeen Test <<<

    Fligner-Killeen test of homogeneity of variances

data:  Interest Rate by Year_Group
Fligner-Killeen:med chi-squared = 8.7819, df = 7, p-value = 0.2687




==============================
Variable: Inflation Rate
==============================

>>> Fligner-Killeen Test <<<

    Fligner-Killeen test of homogeneity of variances

data:  Inflation Rate by Year_Group
```

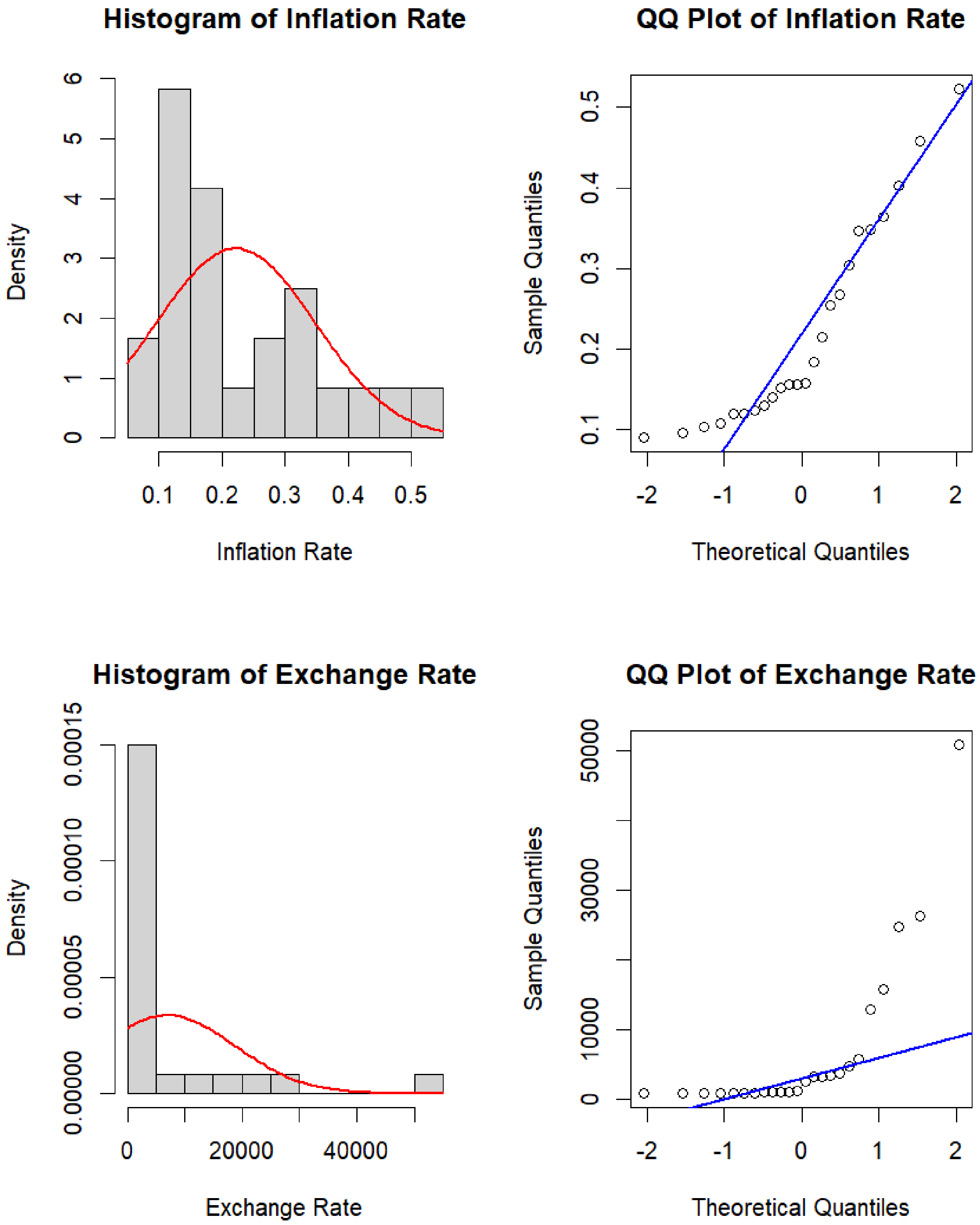
```
Fligner-Killeen:med chi-squared = 3.0459, df = 7, p-value = 0.8807




==============================
Variable: Exchange Rate
==============================

>>> Fligner-Killeen Test <<<

    Fligner-Killeen test of homogeneity of variances

data:  Exchange Rate by Year_Group
Fligner-Killeen:med chi-squared = 6.5089, df = 7, p-value = 0.4817




==============================
Variable: Oil Price
==============================

>>> Fligner-Killeen Test <<<

    Fligner-Killeen test of homogeneity of variances

data:  Oil Price by Year_Group
Fligner-Killeen:med chi-squared = 2.0776, df = 7, p-value = 0.9554
```

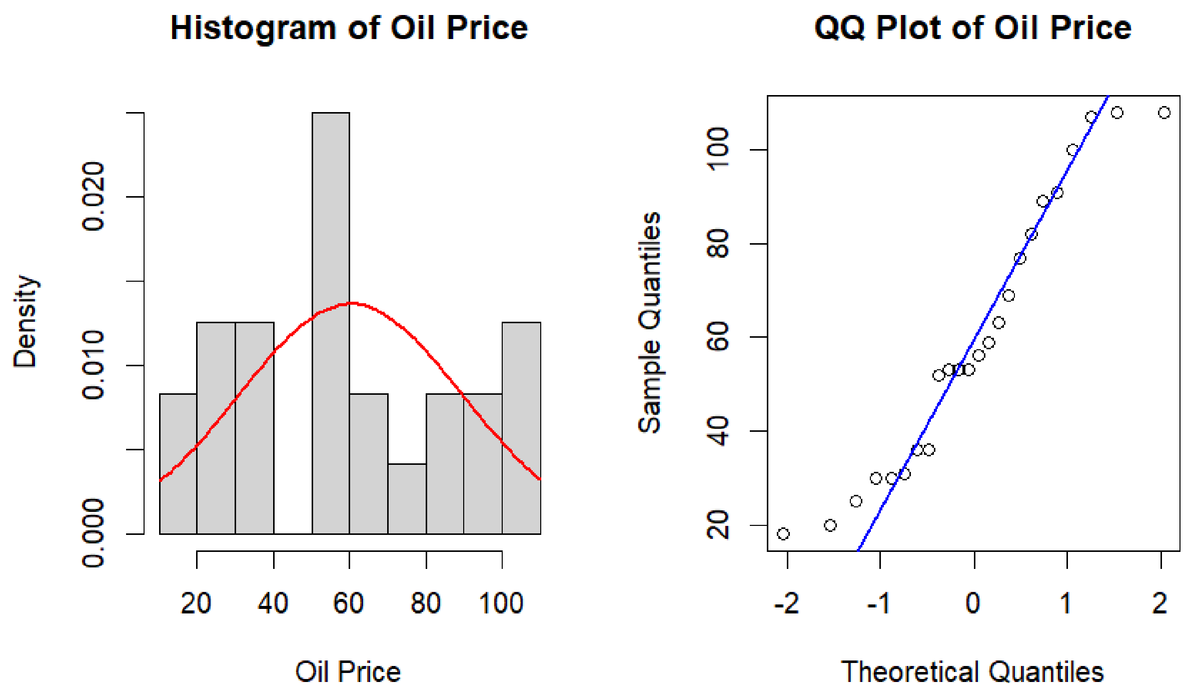
Hide

```
numeric_vars <- c("Overall Index", "Interest Rate",
                  "Inflation Rate", "Exchange Rate", "Oil Price")
```

```r
for (var in numeric_vars) {


  # برای مشاهده توزیع QQ ترسیم هیستوگرام و نمودار
  par(mfrow = c(1, 2))
  hist(data[[var]],
       main = paste("Histogram of", var),
       xlab = var,
       col = "lightgray",
       breaks = 10,
       prob = TRUE)
  curve(dnorm(x, mean(data[[var]]), sd(data[[var]])),
        col = "red", lwd = 2, add = TRUE)

  qqnorm(data[[var]], main = paste("QQ Plot of", var))
  qqline(data[[var]], col = "blue", lwd = 2)

  # بازنشانی پارامترهای گرافیکی
  par(mfrow = c(1, 1))


}
```


Histogram of Overall Index
Density
0e+00
1e-06
2e-06
3e-06
4e-06
0
500000
1500000
Overall Index
QQ Plot of Overall Index
Sample Quantiles
0
500000
1500000
-2
-1
0
1
2
Theoretical Quantiles
Histogram of Interest Rate
Density
0.00
0.05
0.10
0.15
0.20
14
16
18
20
22
24
Interest Rate
QQ Plot of Interest Rate
Sample Quantiles
14
16
18
20
22
24
-2
-1
0
1
2
Theoretical Quantiles


Histogram of Inflation Rate
Density
0
1
2
3
4
5
6
0.1
0.2
0.3
0.4
0.5
Inflation Rate
QQ Plot of Inflation Rate
Sample Quantiles
0.1
0.2
0.3
0.4
0.5
-2
-1
0
1
2
Theoretical Quantiles
Histogram of Exchange Rate
Density
0.00000
0.00005
0.00010
0.00015
0
20000
40000
Exchange Rate
QQ Plot of Exchange Rate
Sample Quantiles
0
10000
30000
50000
-2
-1
0
1
2
Theoretical Quantiles



Hide

```
NA

NA
```

Hide

```
# کتابخانه های لازم

library(car)


# Overall Index

data$Overall_Index_Transformed <- bcPower(data$`Overall Index`, lambda = -0.1
8)


# Exchange Rate (Log Transform)

data$Exchange_Rate_Transformed <- log(data$`Exchange Rate` + 1)  # جلوگیری از
log(0)
```

```
# Inflation Rate
data$Inflation_Rate_Transformed <- bcPower(data$`Inflation Rate`, lambda = -0
.476)

# Interest Rate (Inverse Transform)
data$Interest_Rate_Transformed <- 1 / data$`Interest Rate`

# Oil Price
data$Oil_Price_Transformed <- bcPower(data$`Oil Price`, lambda = 0.426)

# بررسی نرمالیت نهایی
transformed_vars <- c(
  "Overall_Index_Transformed",
  "Exchange_Rate_Transformed",
  "Inflation_Rate_Transformed",
  "Interest_Rate_Transformed",
  "Oil_Price_Transformed"
)

for (var in transformed_vars) {
  cat("\n=== Normality Check for", var, "===\n")
  print(shapiro.test(data[[var]]))
}
```

```
=== Normality Check for Overall_Index_Transformed ===

    Shapiro-Wilk normality test

data:  data[[var]]
W = 0.94795, p-value = 0.2445


=== Normality Check for Exchange_Rate_Transformed ===

    Shapiro-Wilk normality test
```

```
data:  data[[var]]
W = 0.8397, p-value = 0.001416


=== Normality Check for Inflation_Rate_Transformed ===

    Shapiro-Wilk normality test

data:  data[[var]]
W = 0.94206, p-value = 0.1813


=== Normality Check for Interest_Rate_Transformed ===

    Shapiro-Wilk normality test

data:  data[[var]]
W = 0.93403, p-value = 0.12


=== Normality Check for Oil_Price_Transformed ===

    Shapiro-Wilk normality test

data:  data[[var]]
W = 0.94605, p-value = 0.2221
```

Hide

```
# آزمون‌های بارتلت، لوین و فلیگنر-کیلین برای متغیرهای تبدیل‌شده# آزمون‌های بارتل
ت، لوین و فلیگنر-کیلین برای متغیرهای تبدیل‌شده
numeric_vars <- transformed_vars

for (var in numeric_vars) {
  cat("\n==============================\n")
  cat("Variable:", var, "\n")
  cat("==============================\n\n")

  cat(">>> Bartlett Test <<<\n")
  print(
    bartlett.test(
      as.formula(paste0(var, " ~ Year_Group")),
      data = data
    )
  )
  cat("\n\n")
}
```

```
==============================
Variable: Overall_Index_Transformed
==============================

>>> Bartlett Test <<<

    Bartlett test of homogeneity of variances

data:  Overall_Index_Transformed by Year_Group
Bartlett's K-squared = 9.2669, df = 7, p-value = 0.2341




==============================
Variable: Exchange_Rate_Transformed
==============================

>>> Bartlett Test <<<

    Bartlett test of homogeneity of variances
```

```
data:  Exchange_Rate_Transformed by Year_Group
Bartlett's K-squared = 26.309, df = 7, p-value = 0.0004435




==============================
Variable: Inflation_Rate_Transformed
==============================

>>> Bartlett Test <<<

    Bartlett test of homogeneity of variances

data:  Inflation_Rate_Transformed by Year_Group
Bartlett's K-squared = 7.3421, df = 7, p-value = 0.3942




==============================
Variable: Interest_Rate_Transformed
==============================

>>> Bartlett Test <<<

    Bartlett test of homogeneity of variances

data:  Interest_Rate_Transformed by Year_Group
Bartlett's K-squared = Inf, df = 7, p-value < 2.2e-16
```

```

==============================
Variable: Oil_Price_Transformed
==============================

>>> Bartlett Test <<<

    Bartlett test of homogeneity of variances

data:  Oil_Price_Transformed by Year_Group
Bartlett's K-squared = 3.6369, df = 7, p-value = 0.8205
```

Hide

```
for (var in numeric_vars) {
  cat("\n==============================\n")
  cat("Variable:", var, "\n")
  cat("==============================\n\n")

  cat(">>> Levene Test <<<\n")
  print(
    leveneTest(
      as.formula(paste0(var, " ~ Year_Group")),
      data = data
    )
  )
  cat("\n\n")
}
```

```

==============================
Variable: Overall_Index_Transformed
==============================

>>> Levene Test <<<
Levene's Test for Homogeneity of Variance (center = median)
      Df F value Pr(>F)
```

```
group  7  0.8478 0.5652
      16



==============================
Variable: Exchange_Rate_Transformed
==============================

>>> Levene Test <<<
Levene's Test for Homogeneity of Variance (center = median)
      Df F value Pr(>F)
group  7  0.9707 0.4843
      16



==============================
Variable: Inflation_Rate_Transformed
==============================

>>> Levene Test <<<
Levene's Test for Homogeneity of Variance (center = median)
      Df F value Pr(>F)
group  7  0.4781 0.8366
      16



==============================
Variable: Interest_Rate_Transformed
==============================

>>> Levene Test <<<
```

```
Levene's Test for Homogeneity of Variance (center = median)
      Df F value Pr(>F)
group  7  1.2333 0.3415
      16



==============================
Variable: Oil_Price_Transformed
==============================

>>> Levene Test <<<
Levene's Test for Homogeneity of Variance (center = median)
      Df F value Pr(>F)
group  7  0.2205 0.9749
      16
```

Hide

```
for (var in numeric_vars) {
  cat("\n==============================\n")
  cat("Variable:", var, "\n")
  cat("==============================\n\n")

  cat(">>> Fligner-Killeen Test <<<\n")
  print(
    fligner.test(
      as.formula(paste0(var, " ~ Year_Group")),
      data = data
    )
  )
  cat("\n\n")
}
```

```

==============================
Variable: Overall_Index_Transformed
```

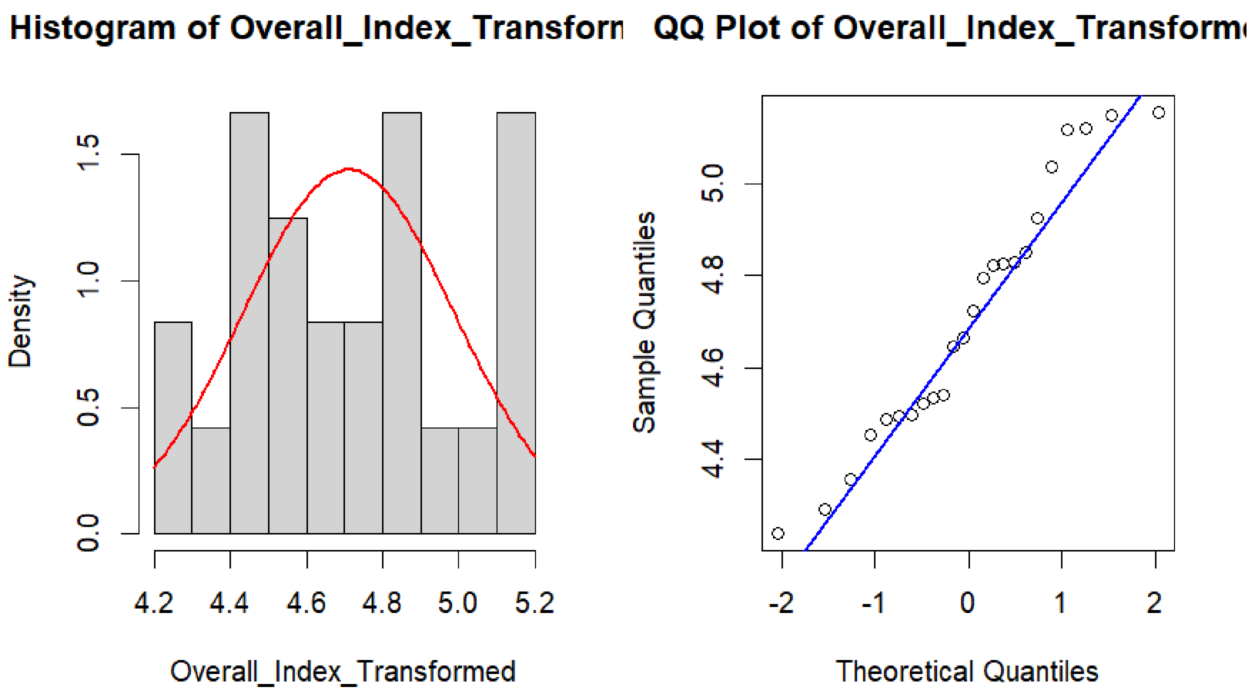
```
==============================

>>> Fligner-Killeen Test <<<

    Fligner-Killeen test of homogeneity of variances

data:  Overall_Index_Transformed by Year_Group
Fligner-Killeen:med chi-squared = 4.7814, df = 7, p-value = 0.6866



==============================
Variable: Exchange_Rate_Transformed
==============================

>>> Fligner-Killeen Test <<<

    Fligner-Killeen test of homogeneity of variances

data:  Exchange_Rate_Transformed by Year_Group
Fligner-Killeen:med chi-squared = 5.136, df = 7, p-value = 0.6434



==============================
Variable: Inflation_Rate_Transformed
==============================

>>> Fligner-Killeen Test <<<

    Fligner-Killeen test of homogeneity of variances
```

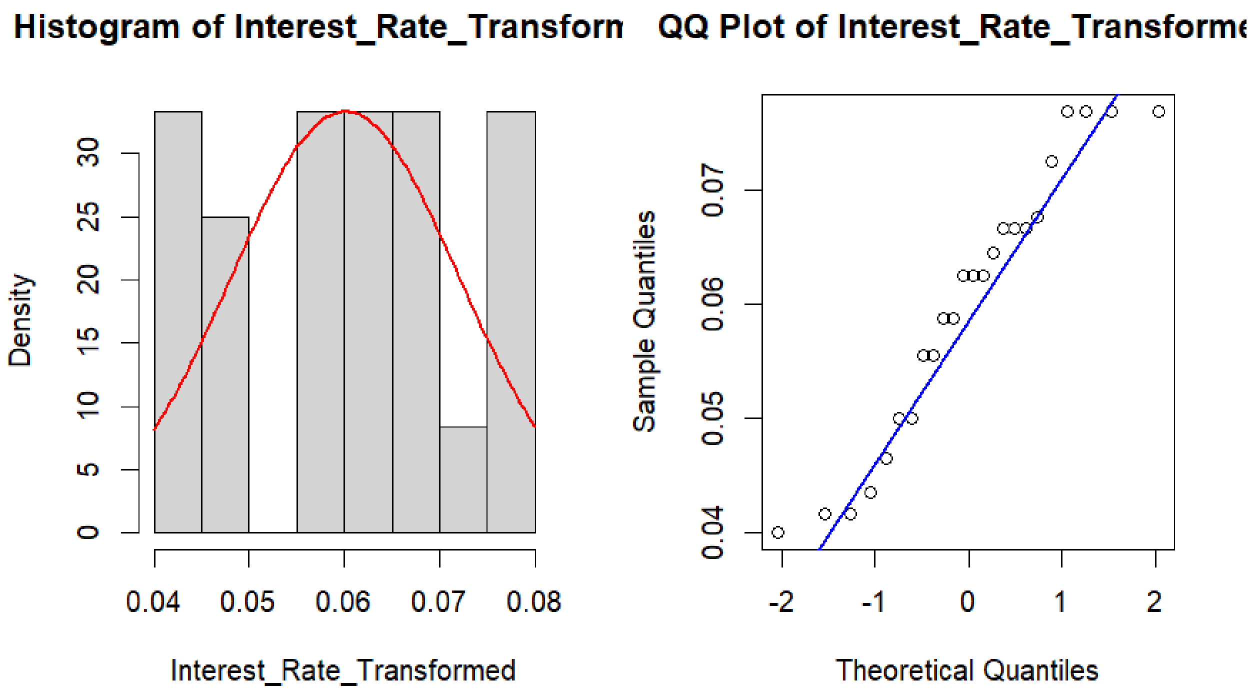
```
data:  Inflation_Rate_Transformed by Year_Group
Fligner-Killeen:med chi-squared = 3.2318, df = 7, p-value = 0.8628




==============================
Variable: Interest_Rate_Transformed
==============================

>>> Fligner-Killeen Test <<<

    Fligner-Killeen test of homogeneity of variances

data:  Interest_Rate_Transformed by Year_Group
Fligner-Killeen:med chi-squared = 7.7125, df = 7, p-value = 0.3586




==============================
Variable: Oil_Price_Transformed
==============================

>>> Fligner-Killeen Test <<<

    Fligner-Killeen test of homogeneity of variances

data:  Oil_Price_Transformed by Year_Group
Fligner-Killeen:med chi-squared = 1.4953, df = 7, p-value = 0.9825
```

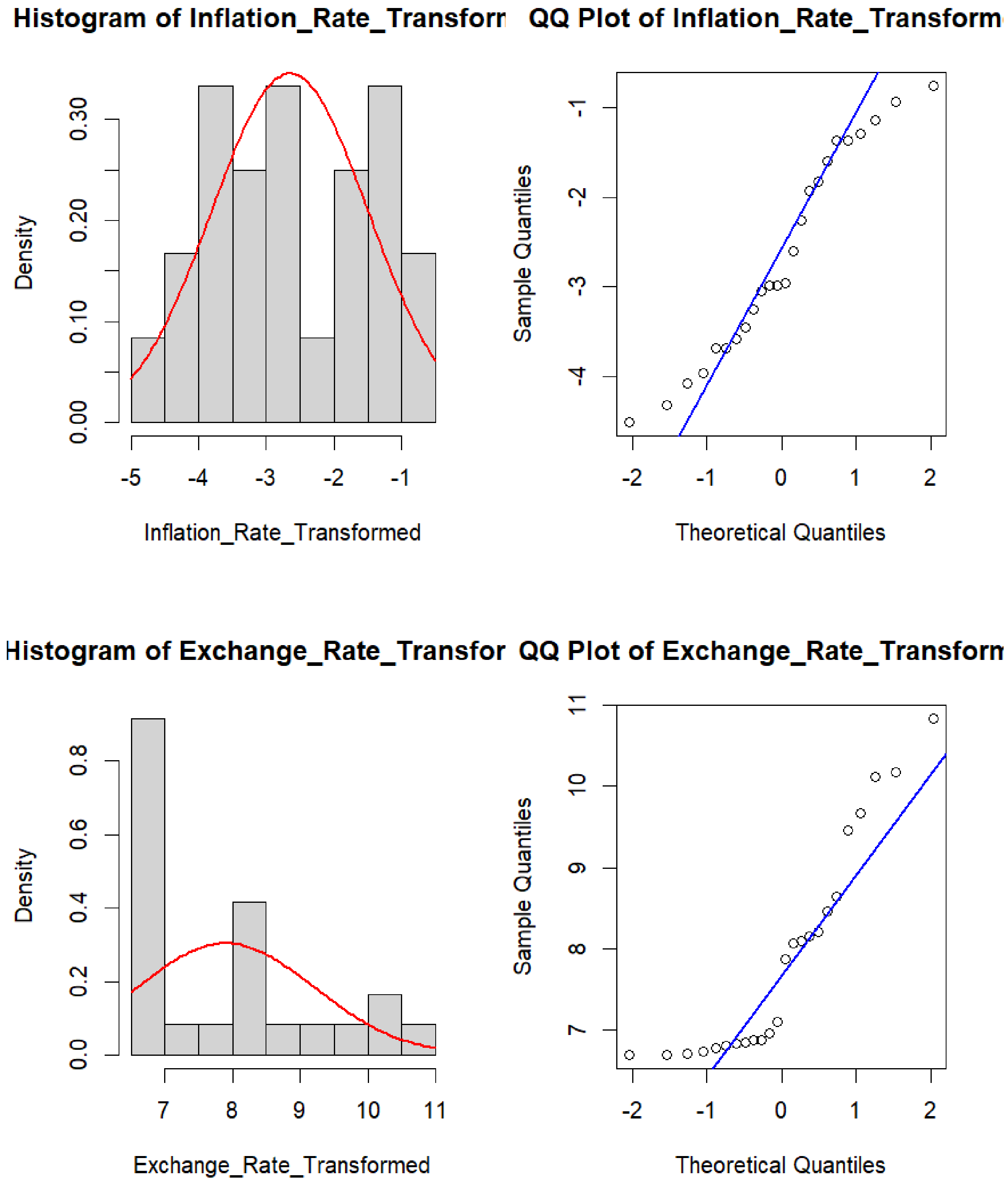
Hide

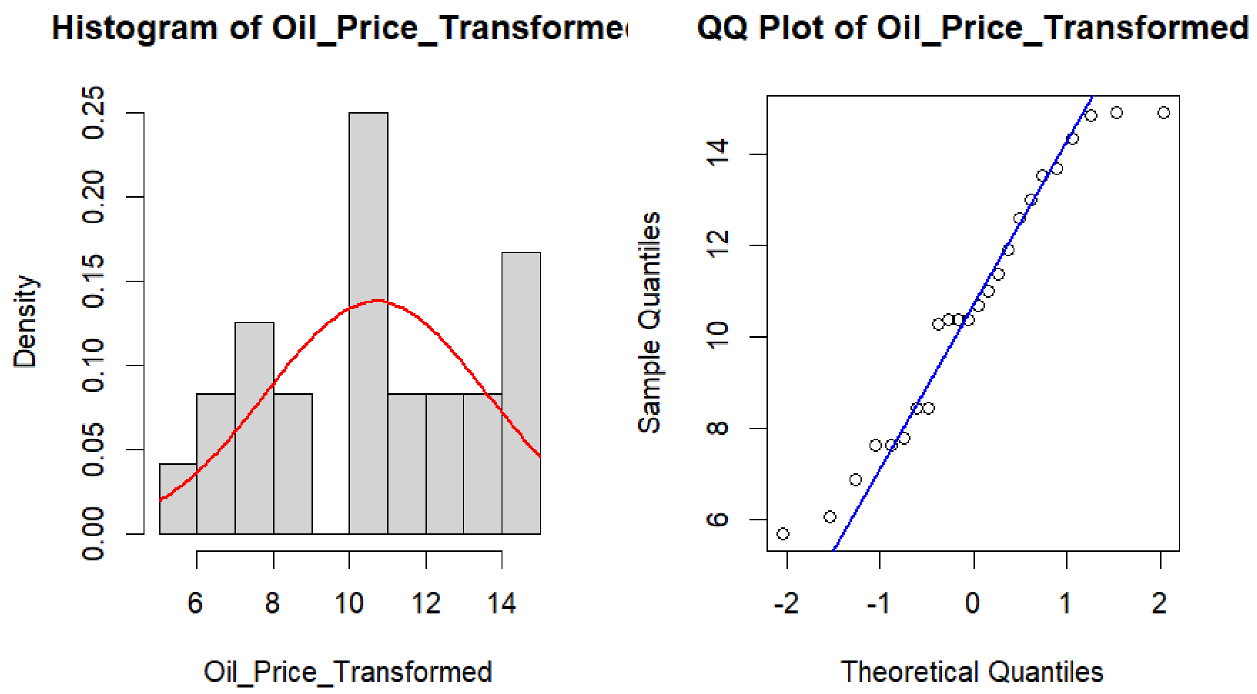
```
numeric_vars <- c("Overall_Index_Transformed", "Interest_Rate_Transformed",
                  "Inflation_Rate_Transformed", "Exchange_Rate_Transformed",
"Oil_Price_Transformed")
```

```r
for (var in numeric_vars) {


  # برای مشاهده توزیع QQ ترسیم هیستوگرام و نمودار
  par(mfrow = c(1, 2))
  hist(data[[var]],
       main = paste("Histogram of", var),
       xlab = var,
       col = "lightgray",
       breaks = 10,
       prob = TRUE)
  curve(dnorm(x, mean(data[[var]]), sd(data[[var]])),
        col = "red", lwd = 2, add = TRUE)

  qqnorm(data[[var]], main = paste("QQ Plot of", var))
  qqline(data[[var]], col = "blue", lwd = 2)

  # بازنشانی پارامترهای گرافیکی
  par(mfrow = c(1, 1))


}
```


Histogram of Overall_Index_Transforn
QQ Plot of Overall_Index_Transform
Density
0.0
0.5
1.0
1.5
4.2
4.4
4.6
4.8
5.0
5.2
Overall_Index_Transformed
Sample Quantiles
4.4
4.6
4.8
5.0
-2
-1
0
1
2
Theoretical Quantiles


Histogram of Interest_Rate_Transforn
QQ Plot of Interest_Rate_Transforme
Density
0
5
10
15
20
25
30
0.04
0.05
0.06
0.07
0.08
Interest_Rate_Transformed
Sample Quantiles
0.04
0.05
0.06
0.07
-2
-1
0
1
2
Theoretical Quantiles


Histogram of Inflation_Rate_Transforn
Density
0.00
0.10
0.20
0.30
-5
-4
-3
-2
-1
Inflation_Rate_Transformed
QQ Plot of Inflation_Rate_Transform
Sample Quantiles
-4
-3
-2
-1
-2
-1
0
1
2
Theoretical Quantiles
Histogram of Exchange_Rate_Transfor
Density
0.0
0.2
0.4
0.6
0.8
7
8
9
10
11
Exchange_Rate_Transformed
QQ Plot of Exchange_Rate_Transforn
Sample Quantiles
7
8
9
10
11
-2
-1
0
1
2
Theoretical Quantiles



Hide

```
# بررسی VIF (Variance Inflation Factor) برای مدل رگرسیون
library(car)

# مطمئن شوید که ستون‌های داده شما به درستی نام‌گذاری شده‌اند و وجود دارند
# جایگزین کردن نام ستون‌ها با نام‌های صحیح موجود در داده شما
model_temp <- lm(`Overall_Index_Transformed` ~ `Interest_Rate_Transformed` +
                   `Inflation_Rate_Transformed` + `Exchange_Rate_Transformed`
+
                   `Oil_Price_Transformed`, data = data)

# محاسبه VIF
vif_values <- vif(model_temp)
```

```
# چاپ VIF
cat("\n==============================\n")
```

```

==============================
```

Hide

```
cat("Variance Inflation Factors (VIF):\n")
```

```
Variance Inflation Factors (VIF):
```

Hide

```
cat("==============================\n\n")
```

```
==============================
```

Hide

```
print(vif_values)
```

```
 Interest_Rate_Transformed Inflation_Rate_Transformed  Exchange_Rate_Transfor
med      Oil_Price_Transformed 

                  3.001090                   1.850678                   3.185
271                   1.338095 
```

Hide

```
# گام 6: ساخت مدل رگرسیون خطی اصلی
model <- lm(`Overall_Index_Transformed` ~ `Interest_Rate_Transformed` +
              `Inflation_Rate_Transformed` + `Exchange_Rate_Transformed` +
              `Oil_Price_Transformed`, data = data)


# نمایش خلاصه ای از مدل
cat("\n==============================\n")
```

```

==============================
```

Hide

```
cat("نتایج مدل رگرسیون خطی اولیه (با متغیرهای تبدیل‌شده)\n")
```

```
نتایج مدل رگرسیون خطی اولیه (با متغیرهای تبدیل‌شده)
```

Hide

```
cat("==============================\n\n")
```

```
==============================
```

Hide

```
summary(model)
```

```

Call:
lm(formula = Overall_Index_Transformed ~ Interest_Rate_Transformed + 
    Inflation_Rate_Transformed + Exchange_Rate_Transformed + 
    Oil_Price_Transformed, data = data)

Residuals:
      Min        1Q    Median        3Q       Max 
-0.156347 -0.037062 -0.000229  0.058333  0.103735 

Coefficients:
                            Estimate Std. Error t value Pr(>|t|)    
(Intercept)                 2.836386   0.351022   8.080 1.44e-07 ***
Interest_Rate_Transformed   0.171163   2.468426   0.069   0.9454    
Inflation_Rate_Transformed -0.023808   0.020098  -1.185   0.2508    
Exchange_Rate_Transformed   0.204104   0.023258   8.776 4.13e-08 ***
Oil_Price_Transformed       0.017663   0.006831   2.586   0.0181 *  
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.08186 on 19 degrees of freedom
Multiple R-squared:  0.9279,	Adjusted R-squared:  0.9128 
F-statistic: 61.17 on 4 and 19 DF,  p-value: 1.379e-10
```

Hide

```
residuals_model <- residuals(model)
```

```
# آزمون شاپیرو-ویلک بر روی پسماندها
shapiro_res <- shapiro.test(residuals_model)
cat("\n==============================\n")
```

```

==============================
```

Hide

```
cat("Shapiro-Wilk Test on Residuals\n")
```

```
Shapiro-Wilk Test on Residuals
```

Hide

```
cat("==============================\n\n")
```

```
==============================
```

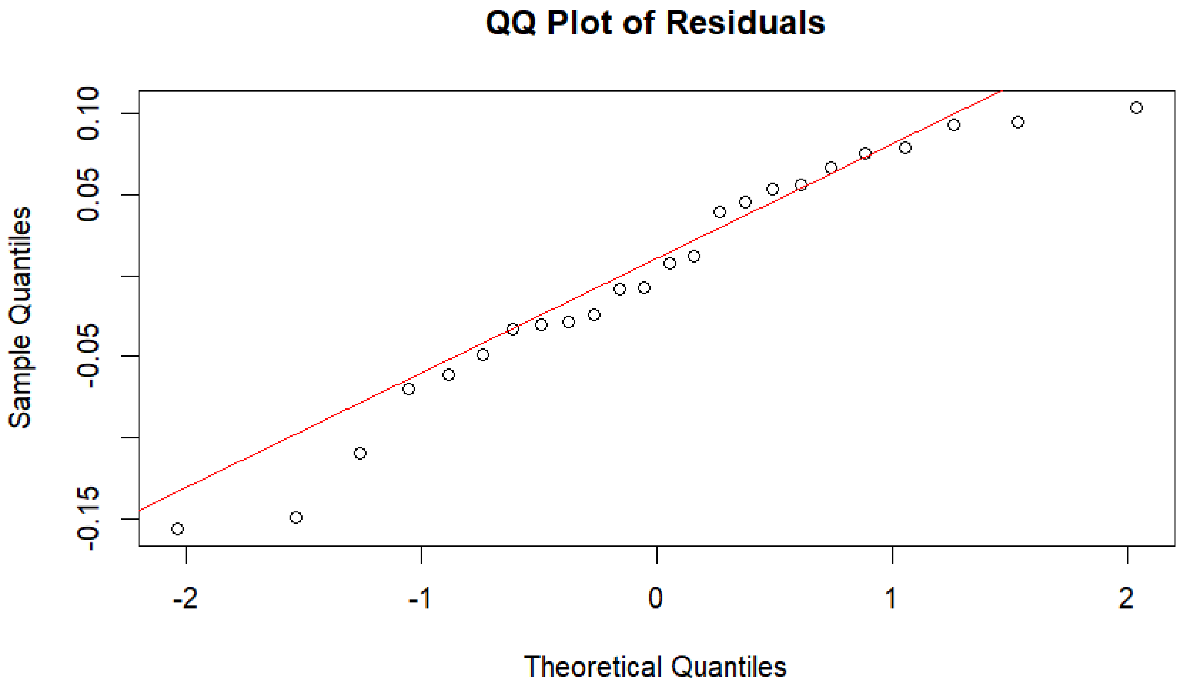
Hide

```
print(shapiro_res)
```

```

    Shapiro-Wilk normality test

data:  residuals_model
W = 0.94657, p-value = 0.228
```

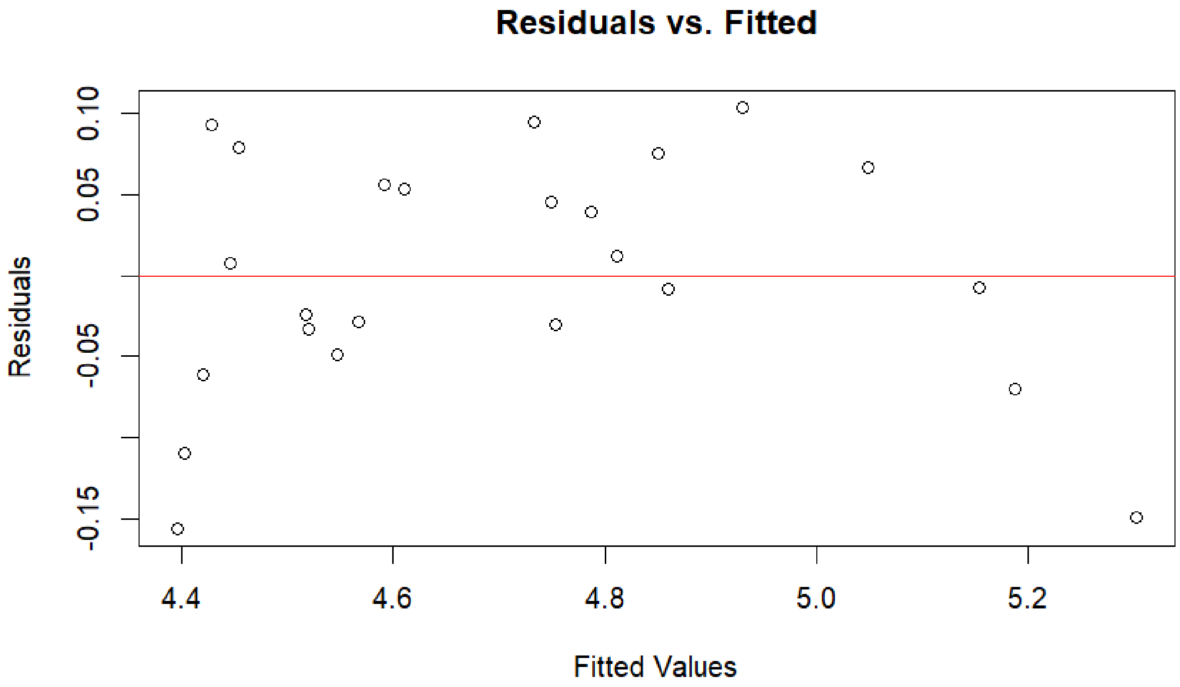
Hide

```
# QQ Plot پسماندها
qqnorm(residuals_model, main = "QQ Plot of Residuals")
qqline(residuals_model, col = "red")
```



Hide

```
fitted_values <- fitted(model)


plot(fitted_values, residuals_model,
     xlab = "Fitted Values",
     ylab = "Residuals",
     main = "Residuals vs. Fitted")
abline(h = 0, col = "red")
```



Hide

```
# اگر الگوی قیفی شکل یا روند خاصی در نمودار دیده شد، احتمال واریانس ناهمسان و
جود دارد.
```

Hide

```
if (!requireNamespace("lmtest", quietly = TRUE)) {
  install.packages("lmtest")
}
library(lmtest)
```

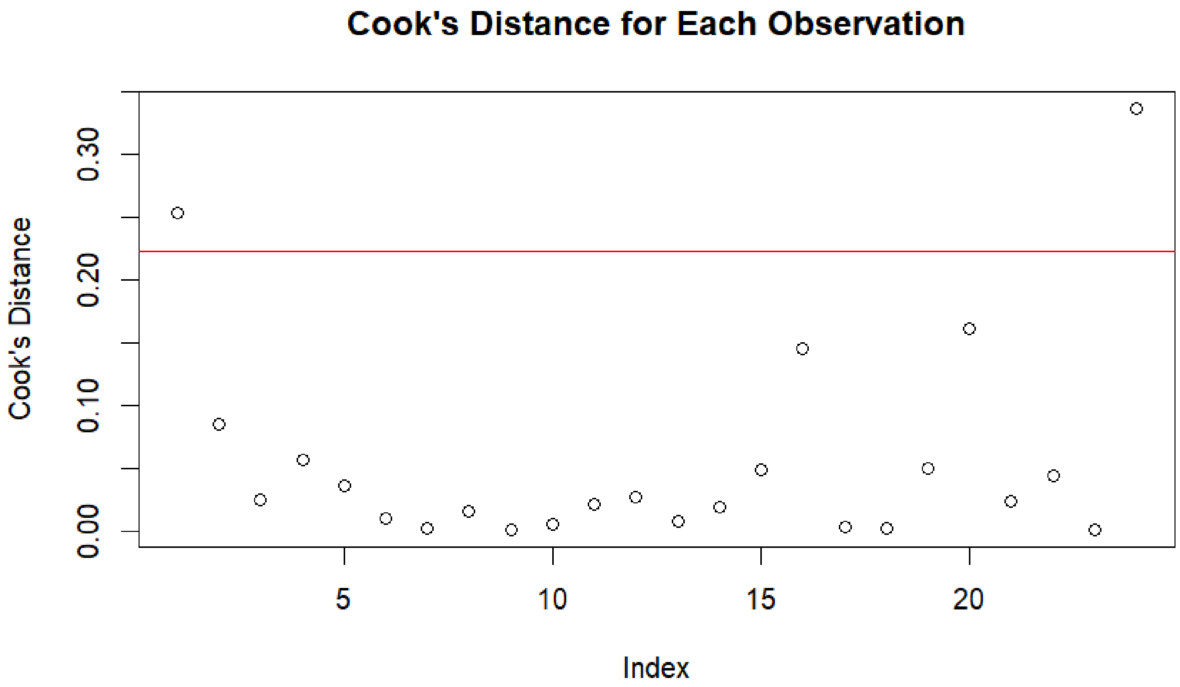
```
Loading required package: zoo


Attaching package: ‘zoo’


The following objects are masked from ‘package:base’:
    as.Date, as.Date.numeric
```

Hide

```
bptest(model)  # آزمون بروس-پگان


    studentized Breusch-Pagan test


data:  model
BP = 7.5887, df = 4, p-value = 0.1079
```

Hide

```
cook_values <- cooks.distance(model)
plot(cook_values,
     ylab = "Cook's Distance",
     main = "Cook's Distance for Each Observation")
abline(h = 4/(nrow(data) - length(coef(model)) - 1), col = "red")
```



Hide

```
influential_points <- which(cooks.distance(model_temp) > 0.2)
print(influential_points)
```

```
 1 24
 1 24
```

Hide

```
data_cleaned <- data[-influential_points, ]
model_temp_cleaned <- lm(`Overall_Index_Transformed` ~ `Interest_Rate_Transformed` +
                         `Inflation_Rate_Transformed` + `Exchange_Rate_Transformed` +
                         `Oil_Price_Transformed`, data = data_cleaned)
summary(model_temp_cleaned)
```

```

Call:
lm(formula = Overall_Index_Transformed ~ Interest_Rate_Transformed +
    Inflation_Rate_Transformed + Exchange_Rate_Transformed +
    Oil_Price_Transformed, data = data_cleaned)

Residuals:
      Min        1Q    Median        3Q       Max
-0.131290 -0.031205 -0.006619  0.053827  0.077457

Coefficients:
                           Estimate Std. Error t value Pr(>|t|)
(Intercept)                 2.89862    0.29787   9.731 2.31e-08 ***
Interest_Rate_Transformed  -0.20025    1.99285  -0.100   0.9211
Inflation_Rate_Transformed -0.01784    0.01595  -1.119   0.2788
Exchange_Rate_Transformed   0.20952    0.02001  10.470 7.87e-09 ***
Oil_Price_Transformed       0.01285    0.00586   2.193   0.0425 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

```
Residual standard error: 0.06458 on 17 degrees of freedom
Multiple R-squared:  0.9474,    Adjusted R-squared:  0.935 
F-statistic: 76.55 on 4 and 17 DF,  p-value: 1.217e-10
```

Hide

```
# انجام تحلیل واریانس برای مدل رگرسیون
anova_results <- anova(model_temp_cleaned)

# نمایش جدول ANOVA
cat("\n==============================\n")
```

```

==============================
```

Hide

```
cat("جدول تحلیل واریانس (ANOVA Table)\n")
```

```
جدول تحلیل واریانس (ANOVA Table)
```

Hide

```
cat("==============================\n\n")
```

```
==============================
```

Hide

```
print(anova_results)
```

```
Analysis of Variance Table

Response: Overall_Index_Transformed
                           Df  Sum Sq Mean Sq  F value    Pr(>F)    
Interest_Rate_Transformed   1 0.77165 0.77165 185.0428 1.445e-10 ***
Inflation_Rate_Transformed  1 0.04619 0.04619  11.0761  0.003981 ** 
Exchange_Rate_Transformed   1 0.43891 0.43891 105.2531 1.062e-08 ***
Oil_Price_Transformed       1 0.02005 0.02005   4.8092  0.042507 *  
Residuals                  17 0.07089 0.00417                       
```

```
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Hide

```
# توضیحات تکمیلی:
cat("\n--------------------------------------------------------\n")
```

```

--------------------------------------------------------
```

Hide

```
cat("تفسیر جدول ANOVA:\n")
```

```
تفسیر جدول ANOVA:
```

Hide

```
cat("- ستون 'Sum Sq': مجموع مربعات برای هر متغیر.\n")
```

```
- ستون 'Sum Sq': مجموع مربعات برای هر متغیر.
```

Hide

```
cat("- ستون 'Mean Sq': میانگین مربعات برای هر متغیر.\n")
```

```
- ستون 'Mean Sq': میانگین مربعات برای هر متغیر.
```

Hide

```
cat("- ستون 'F value': مقدار آماره F برای هر متغیر.\n")
```

```
- ستون 'F value': مقدار آماره F برای هر متغیر.
```

Hide

```
cat("- ستون 'Pr(>F)': مقدار p برای آزمون فرضیه اثرگذاری.\n")
```

```
- ستون 'Pr(>F)': مقدار p برای آزمون فرضیه اثرگذاری.
```

Hide

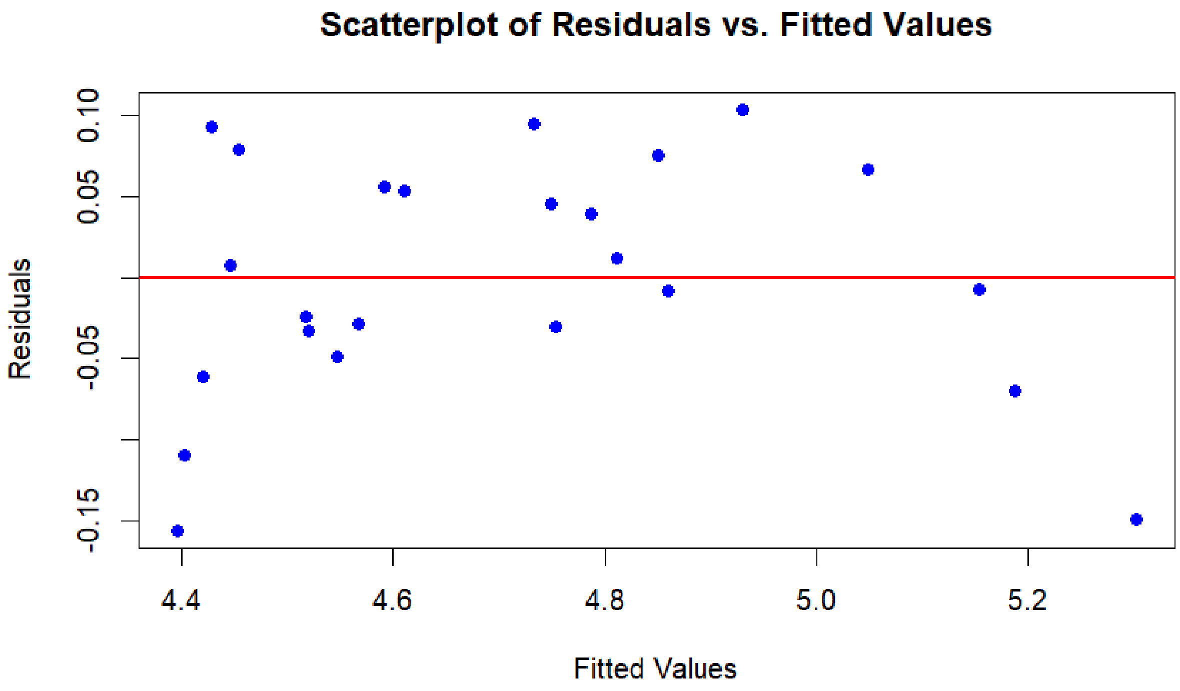
```
cat("--------------------------------------------------------\n\n")
```

```
--------------------------------------------------------
```

Hide

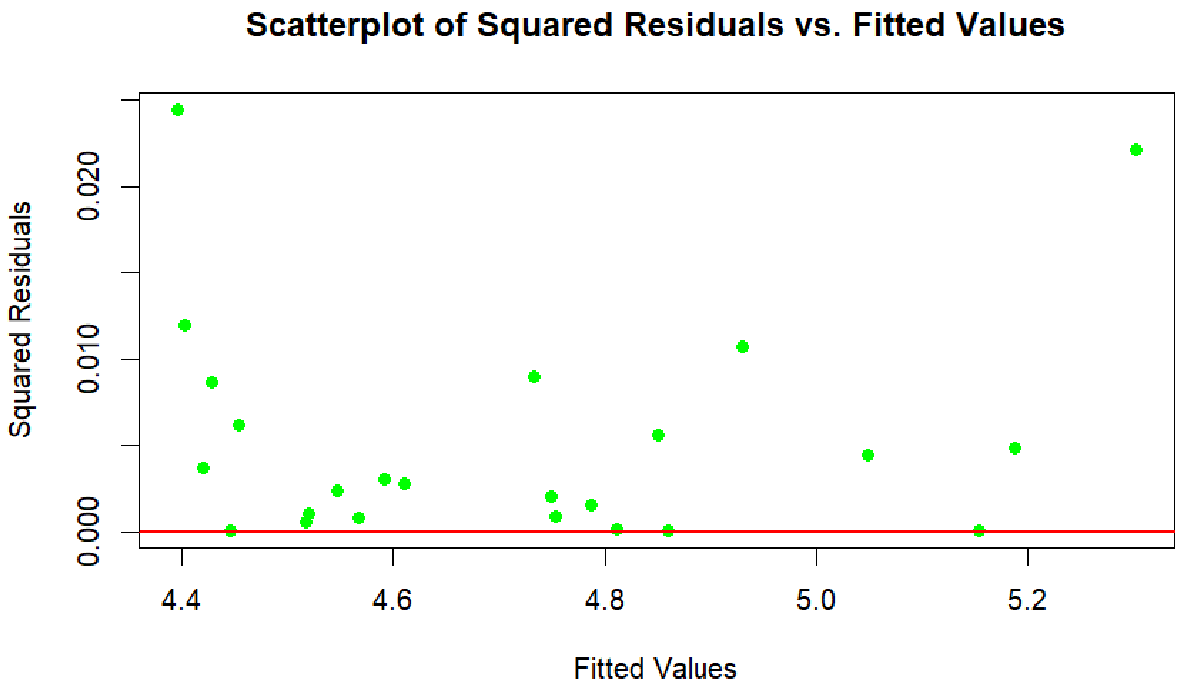
```
# دریافت مقادیر باقی‌مانده‌ها و نقاط برازش شده
```

```
residuals_model <- residuals(model)
fitted_values <- fitted(model)

# 1. نمودار پراکنش بین باقی‌مانده‌ها و نقاط برازش شده پاسخ
plot(fitted_values, residuals_model,
     xlab = "Fitted Values",
     ylab = "Residuals",
     main = "Scatterplot of Residuals vs. Fitted Values",
     pch = 19,
     col = "blue")
abline(h = 0, col = "red", lwd = 2)
```



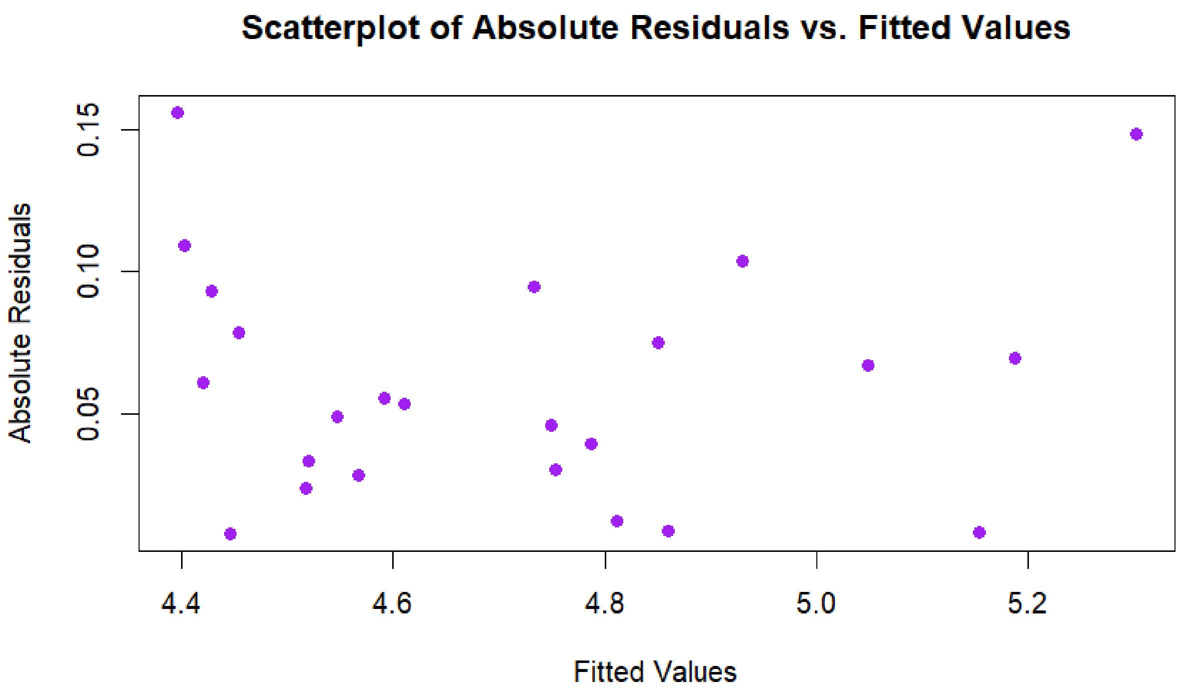
Hide

```

# 2. نمودار پراکنش بین توان دوم باقی‌مانده‌ها و نقاط برازش شده پاسخ
plot(fitted_values, residuals_model^2,
     xlab = "Fitted Values",
     ylab = "Squared Residuals",
     main = "Scatterplot of Squared Residuals vs. Fitted Values",
     pch = 19,
     col = "green")
abline(h = 0, col = "red", lwd = 2)
```



Hide

```
# 3. نمودار پراکنش بین قدر مطلق باقی‌مانده‌ها و نقاط برازش شده پاسخ
plot(fitted_values, abs(residuals_model),
      xlab = "Fitted Values",
      ylab = "Absolute Residuals",
      main = "Scatterplot of Absolute Residuals vs. Fitted Values",
      pch = 19,
      col = "purple")
abline(h = 0, col = "red", lwd = 2)
```



Hide

```
# اجرای آزمون گلدفیلد-کوانت

goldfeld_quandt_test <- gqtest(model_temp_cleaned, order.by = fitted(model_te
mp_cleaned))


# نمایش نتایج

cat("\n==============================\n")
```

```

==============================
```

Hide

```
cat("نتایج آزمون گلدفیلد-کوانت (Goldfeld-Quandt)\n")
```

```
نتایج آزمون گلدفیلد-کوانت (Goldfeld-Quandt)
```

Hide

```
cat("==============================\n\n")
```

```
==============================
```

Hide

```
print(goldfeld_quandt_test)
```

```

	Goldfeld-Quandt test

data:  model_temp_cleaned
GQ = 0.27958, df1 = 6, df2 = 6, p-value = 0.9269
alternative hypothesis: variance increases from segment 1 to 2
```

Hide

```
# بارگذاری کتابخانه مورد نیاز
if (!requireNamespace("lmtest", quietly = TRUE)) {
  install.packages("lmtest")
}
library(lmtest)

# اجرای آزمون Breusch-Pagan برای مدل تمیز شده
bp_test <- bptest(model_temp_cleaned)

# نمایش نتایج آزمون
cat("\n==============================\n")
```

```

==============================
```

Hide

```
cat("نتایج آزمون Breusch-Pagan (BP Test)\n")
```

```
نتایج آزمون Breusch-Pagan (BP Test)
```

Hide

```
cat("==============================\n\n")
```

```
==============================
```

Hide

```
print(bp_test)
```

```

	studentized Breusch-Pagan test

data:  model_temp_cleaned
BP = 3.677, df = 4, p-value = 0.4515
```

Hide

```
# راهنمای تفسیر
cat("\n--------------------------------------------------------\n")
```

```

--------------------------------------------------------
```

Hide

```
cat("راهنمای تفسیر:\n")
```

```
راهنمای تفسیر:
```

Hide

```
cat("- اگر p-value < 0.05 باشد، فرضیه صفر (همگنی واریانس) رد می‌شود.\n")
```

```
- اگر p-value < 0.05 باشد، فرضیه صفر (همگنی واریانس) رد می‌شود.
```

Hide

```
cat("- اگر p-value > 0.05 باشد، فرضیه صفر پذیرفته می‌شود (واریانس همگن است).\n
")
```

```
- اگر p-value > 0.05 باشد، فرضیه صفر پذیرفته می‌شود (واریانس همگن است).
```

Hide

```
cat("--------------------------------------------------------\n\n")
```

```
--------------------------------------------------------
```

Hide

```
# نصب کتابخانه مورد نیاز
if (!requireNamespace("sandwich", quietly = TRUE)) {
  install.packages("sandwich")
}
if (!requireNamespace("lmtest", quietly = TRUE)) {
  install.packages("lmtest")
}
library(sandwich)
library(lmtest)

# محاسبه ضرایب استوار
robust_se <- coeftest(model_temp_cleaned, vcov = vcovHC(model_temp_cleaned, type = "HC3"))

# نمایش ضرایب استوار
cat("\n==============================\n")
```

```

==============================
```

Hide

```
cat("ضرایب رگرسیونی با خطای استاندارد استوار\n")
```

```
ضرایب رگرسیونی با خطای استاندارد استوار
```

Hide

```
cat("==============================\n\n")
```

```
==============================
```

Hide

```
print(robust_se)
```

```

t test of coefficients:

                            Estimate Std. Error t value  Pr(>|t|)    
(Intercept)                2.8986228  0.2622252 11.0539 3.497e-09 ***
Interest_Rate_Transformed -0.2002462  1.8217187 -0.1099    0.9138    
Inflation_Rate_Transformed -0.0178413  0.0112854 -1.5809    0.1323    
Exchange_Rate_Transformed  0.2095238  0.0176502 11.8709 1.187e-09 ***
Oil_Price_Transformed      0.0128507  0.0077685  1.6542    0.1164    
```

```
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Hide

```
# راهنمای تفسیر:
cat("\n--------------------------------------------------------\n")
```

```

--------------------------------------------------------
```

Hide

```
cat("راهنمای تفسیر ضرایب:\n")
```

```
راهنمای تفسیر ضرایب:
```

Hide

```
cat("- ضرایب استوار در صورت وجود ناهمگنی واریانس قابل اعتمادتر هستند.\n")
```

```
- ضرایب استوار در صورت وجود ناهمگنی واریانس قابل اعتمادتر هستند.
```

Hide

```
cat("- در صورتی که ضرایب معنادار باقی بمانند، مدل قابل قبول است.\n")
```

```
- در صورتی که ضرایب معنادار باقی بمانند، مدل قابل قبول است.
```

Hide

```
cat("--------------------------------------------------------\n")
```

```
--------------------------------------------------------
```

Hide

```
# تعیین وزن‌ها (معکوس باقی‌مانده‌ها به عنوان تقریبی از واریانس)
weights <- 1 / (abs(residuals(model_temp_cleaned)) + 1e-6)


# مدل با استفاده از رگرسیون وزنی
wls_model <- lm(`Overall_Index_Transformed` ~ `Interest_Rate_Transformed` +
                  `Inflation_Rate_Transformed` + `Exchange_Rate_Transformed`
+
                  `Oil_Price_Transformed`, data = data_cleaned, weights = wei
ghts)
```

```
# نمایش خلاصه مدل وزنی
cat("\n==============================\n")
```

```
==============================
```

Hide

```
cat("خلاصه مدل رگرسیون وزنی\n")
```

```
خلاصه مدل رگرسیون وزنی
```

Hide

```
cat("==============================\n\n")
```

```
==============================
```

Hide

```
summary(wls_model)
```

```

Call:
lm(formula = Overall_Index_Transformed ~ Interest_Rate_Transformed + 
    Inflation_Rate_Transformed + Exchange_Rate_Transformed + 
    Oil_Price_Transformed, data = data_cleaned, weights = weights)

Weighted Residuals:
     Min       1Q   Median       3Q      Max 
-0.36170 -0.14777 -0.02628  0.20256  0.30573 

Coefficients:
                            Estimate Std. Error t value Pr(>|t|)    
(Intercept)                 2.950378   0.180463  16.349 7.84e-12 ***
Interest_Rate_Transformed  -0.806682   1.284992  -0.628   0.5385    
Inflation_Rate_Transformed -0.016733   0.006888  -2.429   0.0265 *  
Exchange_Rate_Transformed   0.210794   0.012538  16.812 5.00e-12 ***
Oil_Price_Transformed       0.010754   0.004558   2.359   0.0305 *  
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

```
Residual standard error: 0.2456 on 17 degrees of freedom
Multiple R-squared:  0.969, Adjusted R-squared:  0.9617
F-statistic: 132.9 on 4 and 17 DF,  p-value: 1.384e-12
```

Hide

```
# آزمون واریانس باقی‌مانده‌ها در مدل وزنی
cat("\n==============================\n")
```

```

==============================
```

Hide

```
cat("آزمون همگنی واریانس در مدل وزنی\n")
```

```
آزمون همگنی واریانس در مدل وزنی
```

Hide

```
cat("==============================\n\n")
```

```
==============================
```

Hide

```
bptest(wls_model)
```

```

	studentized Breusch-Pagan test

data:  wls_model
BP = 1031.6, df = 4, p-value < 2.2e-16
```

Hide

```
# نصب کتابخانه مورد نیاز
if (!requireNamespace("MASS", quietly = TRUE)) {
  install.packages("MASS")
}
library(MASS)

# مدل مقاوم با استفاده از الگوریتم Huber
```

```
robust_model <- rlm(`Overall_Index_Transformed` ~ `Interest_Rate_Transformed`
+
                      `Inflation_Rate_Transformed` + `Exchange_Rate_Transform
ed` +
                      `Oil_Price_Transformed`, data = data_cleaned)


# خلاصه مدل مقاوم
cat("\n==============================\n")
```

```

==============================
```

Hide

```
cat("خلاصه مدل مقاوم\n")
```

```
خلاصه مدل مقاوم
```

Hide

```
cat("==============================\n\n")
```

```
==============================
```

Hide

```
summary(robust_model)
```

```

Call: rlm(formula = Overall_Index_Transformed ~ Interest_Rate_Transformed +
    Inflation_Rate_Transformed + Exchange_Rate_Transformed +
    Oil_Price_Transformed, data = data_cleaned)
Residuals:
      Min        1Q    Median        3Q       Max
-0.142652 -0.032023 -0.007332  0.051666  0.068809


Coefficients:
                           Value   Std. Error t value
(Intercept)                 2.9449  0.3043     9.6776
Interest_Rate_Transformed  -0.2915  2.0358    -0.1432
Inflation_Rate_Transformed -0.0170  0.0163    -1.0447
Exchange_Rate_Transformed   0.2072  0.0204    10.1353
Oil_Price_Transformed       0.0112  0.0060     1.8732
```

```
Residual standard error: 0.0652 on 17 degrees of freedom
```