Test 82: F-test for Testing Linearity of Regression

# Load Required Functions:  
source(file = "muFunc.R")  
  
# Load Required Packages:  
wants <- c("dplyr", "psych", "pacman", "ggpubr")  
has <- wants %in% rownames(x = installed.packages())  
if(any(!has)) install.packages(wants[!has])  
pacman::p\_load(char = wants)  
  
# Load Required Data:  
data <- read.csv(file = "data/Data\_Test\_82.csv", header = TRUE)  
  
set.seed(1)  
nSample <- data %>% sample\_n(size = 35)  
  
# Show Data:  
headTail(x = nSample, top = 2, bottom = 2)

## Year prec\_mashhad prec\_sabzevar  
## 1 2015 207.6 120.4  
## 2 1962 196.8 123.6  
## ... ... ... ...  
## 34 1988 306.7 285.9  
## 35 1994 175.1 140.4

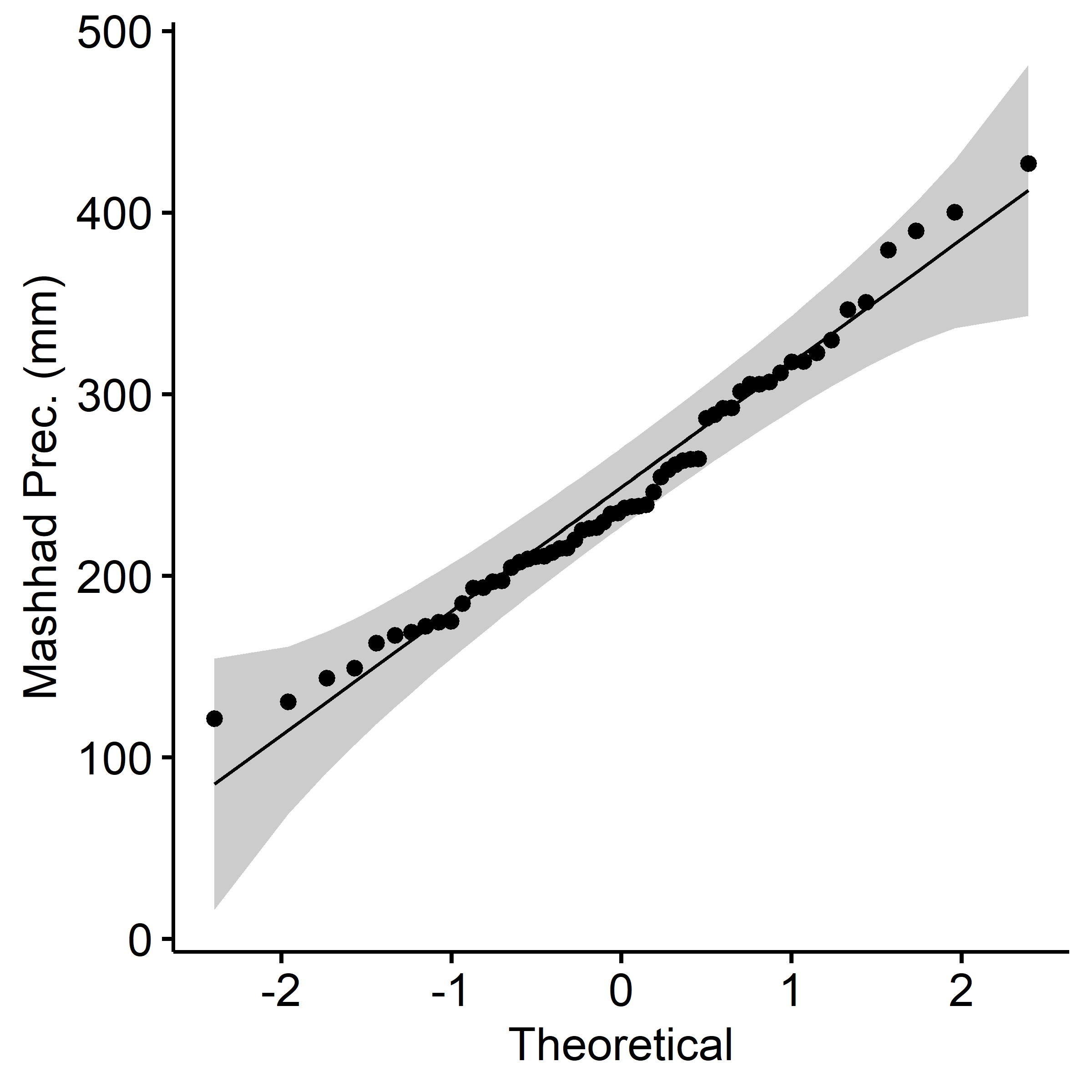
# Assumption Checking:  
# 1. It is assumed that the x and y values originate from a bivariate normal distribution,  
# and that the relationship is linear.  
  
# Use Shapiro-Wilk Normality Test:  
shapiro.test(x = data$prec\_mashhad)

##   
## Shapiro-Wilk normality test  
##   
## data: data$prec\_mashhad  
## W = 0.97234, p-value = 0.1897

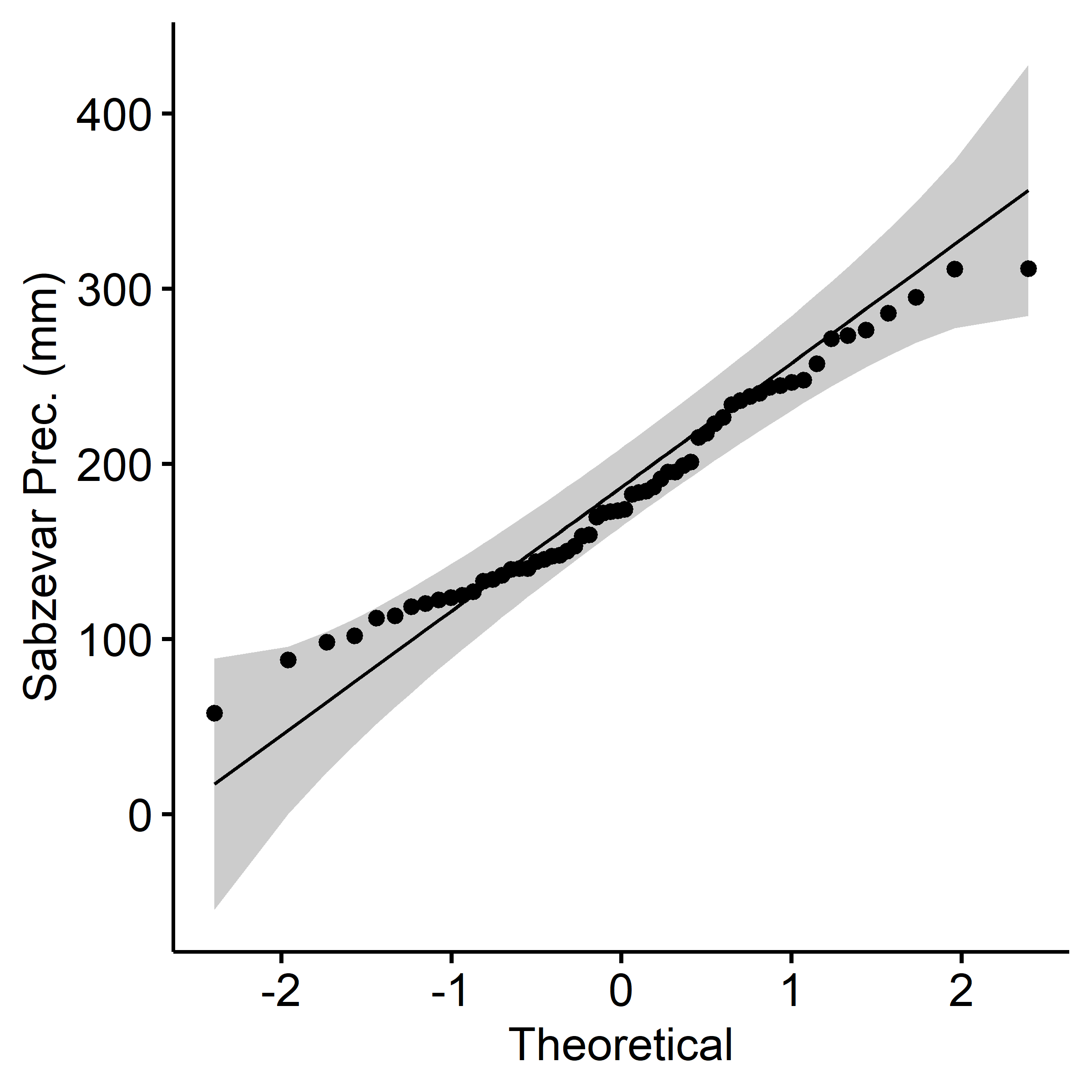
shapiro.test(x = data$prec\_sabzevar)

##   
## Shapiro-Wilk normality test  
##   
## data: data$prec\_sabzevar  
## W = 0.97163, p-value = 0.1753

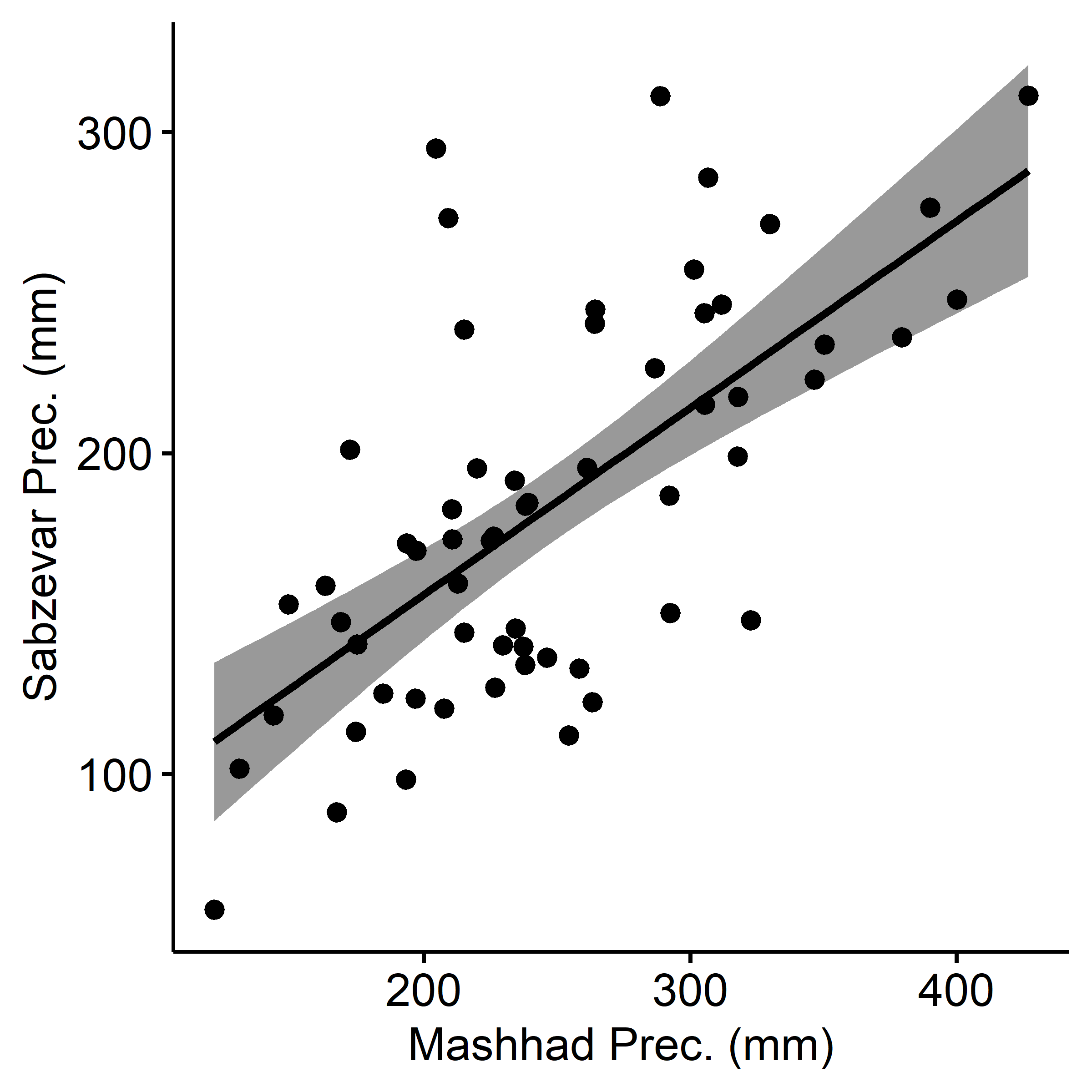
# Normality Plot:  
ggqqplot(data = data$prec\_mashhad, ylab = "Mashhad Prec. (mm)")



ggqqplot(data = data$prec\_sabzevar, ylab = "Sabzevar Prec. (mm)")



# Visualize Data Using Scatter Plots:  
data %>% ggscatter(x = "prec\_mashhad", y = "prec\_sabzevar",   
 add = "reg.line", conf.int = TRUE,  
 xlab = "Mashhad Prec. (mm)", ylab = "Sabzevar Prec. (mm)")



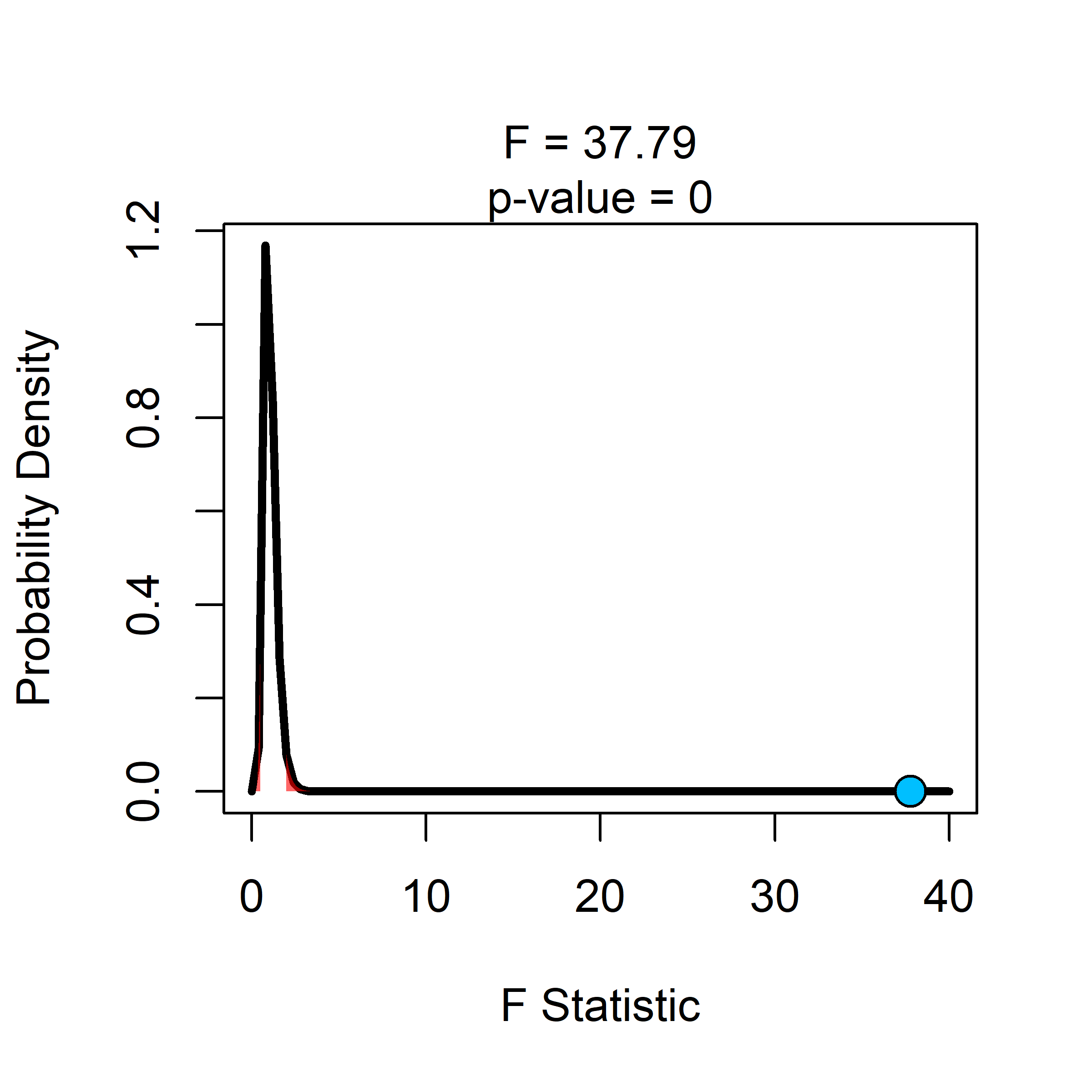
# Method 1:---------------------------------------------------------------------  
# X Variable  
x <- nSample$prec\_mashhad  
xbar <- mean(x)  
  
# Y Variable  
y <- nSample$prec\_sabzevar  
ybar <- mean(y)  
  
# Length Data  
n <- length(y)  
  
# Regression Coefficient  
b <- (sum(x \* y) - ((sum(x) \* sum(y)) / n)) / (sum(x ^ 2) - (sum(x) ^ 2 / n))  
  
# Total Sum of Squares  
SST <- sum((y - ybar) ^ 2)  
  
# Regression Sum of Squares  
SSR <- sum((b \* (x - xbar)) ^ 2)  
  
# Residual Sum of Squares  
SSE <- sum((y - ybar - (b \* (x - xbar))) ^ 2)  
  
# Mean Squares  
SSEbar <- SSE / (n - 2)  
SSRbar <- SSR / 1  
  
# Ratio of Mean Squares  
F\_statistic = SSRbar / SSEbar  
  
# Critical Value of F for (1, n − 2) Degrees of Freedom  
alpha = 0.05  
qf(p = (1 - alpha), df1 = 1, df2 = n - 2)

## [1] 4.139252

# Method 2:---------------------------------------------------------------------  
mod <- lm(formula = prec\_sabzevar ~ prec\_mashhad, data = nSample)  
  
summary(mod)

##   
## Call:  
## lm(formula = prec\_sabzevar ~ prec\_mashhad, data = nSample)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -87.260 -30.830 1.491 25.157 119.382   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 3.8909 30.9626 0.126 0.901   
## prec\_mashhad 0.7167 0.1166 6.147 6.26e-07 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 46.32 on 33 degrees of freedom  
## Multiple R-squared: 0.5338, Adjusted R-squared: 0.5197   
## F-statistic: 37.79 on 1 and 33 DF, p-value: 6.259e-07

plotDistStat(dist = "F",  
 df1 = length(x = nSample$prec\_mashhad) - 1,  
 df2 = length(x = nSample$prec\_sabzevar) - 1,  
 to = 40,  
 alpha\_level = 0.05,  
 statistic\_point = extract\_lm\_F\_p(modelobject = mod)[["F\_statistics"]],  
 p\_value = extract\_lm\_F\_p(modelobject = mod)[["p\_value"]])



# Assumption Checking:  
# 1. The error terms are normally and independently distributed with mean zero.  
shapiro.test(x = mod$residuals)

##   
## Shapiro-Wilk normality test  
##   
## data: mod$residuals  
## W = 0.97721, p-value = 0.6667

mean(mod$residuals)

## [1] 1.421135e-15