Simple linear Regression in R

1- Create a data frame

↳ Write data manually

```
ourdata <- data.frame( x = c (x1, x2, x3,...), # I.V.
                       y = c (y1 , y2 , y3,...) # D.V.
)
```

↳ Load data from a file

```
library (redxl)
our data <- read_excel ('ourdata.xlsx')
```

2- Display the data

```
ourdata
```

3- Plot the data

```
plot (ourdata$x ,ourdata$y , xlab='...',ylab='...')
```

4- Fit a linear model

```
ourmodel <- lm (y ~ x , data = ourdata )
```

5- Display the model summary

```
Summary (ourmodel)
```

6- Get the coefficients ($\hat{\beta}_0$ and $\hat{\beta}_1$)

```
slope <- ourmodel$coefficients[2]
intercept <- ourmodel$coefficients[1]
```

7- Plot the regression line

```
plot(ourdata$x , ourdata$y , main= 'Linear Regressi
     col= "blue" , xlab= "..." , ylab= "...")
abline(a = intercept , b= slope, col= "red")
```

8- Calculate SST , SSR , SSE

```
y <- ourdata$y
y_bar <- mean(y)
y_hat <- ourmodel$fitted.values

SST <- sum((y - y_bar)^2)

SSE <- sum((y - y_hat)^2)

SSR <- sum((y_hat - y_bar)^2)
```

9- Calculate R^2

$\left(R^2 = \frac{SSR}{SST}\right)$

```
r_squared <- SSR / SST
```

10- Calculate standard error S

$\left(S = \sqrt{\frac{SSE}{n-2}}\right)$

```
S <- sqrt(SSE/(length(y)-2))
```

!! mmkn yes2alo 3an l Error variance estimator (S^2)

```
s_squared <- SSE/(length(y)-2)
```

11. Make Predictions

```
new_ x     <- c (newx1 , newx2 , ...)
predicted_y <- intercept + slope * new_x
```

or

```
predict (model , newdata = data.frame (x = c (x1, x2, ...)))
```

12. Calculate the correlation coefficient (r)

(to justify a linear rlt)

```
correlation_coefficient <- cor (ourdata$x , ourdata$y)
```

!! $r = \dfrac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum (x_i - \bar{x})^2 \sum (y_i - \bar{y})^2}}$

13. Calculate the variance Estimators of $\hat{\beta}_0$ and $\hat{\beta}_1$ ($S\hat{\beta}_0$) and ($S\hat{\beta}_1$)

!! $S\hat{\beta}_0 = S \times \sqrt{\dfrac{\sum x_i^2}{n \sum (x_i - \bar{x})^2}}$ - $S\hat{\beta}_1 = \dfrac{S}{\sqrt{\sum (x_i - \bar{x})^2}}$

```
#find the mean of x  (x̄)
x_bar <- mean (x)
# find the standar error of estimate (s) (part 10)
# Calculate the variance Estimator of β̂0 (Sβ̂0)
SB0<- s * sqrt(sum(x^2)/ (length (y) * sum((x - x_bar)^2)))

# Calculate the variance estimator of β̂1 (Sβ̂1)
SB1<- s /sqrt (sum((x - x_bar)^2))
```

14. Confidence Interval for $\hat{\beta}_0$ and $\hat{\beta}_1$

```
# for β̂0 :
confidence_interval <- confint(object=model,
                               parm="Intercept", level=0.95)
# for β̂1
confidence_interval <- confint(object=model, parm="model$x",
                               level=0.95)
```

15. Hypothesis testing for $\hat{\beta}_0$ / $\hat{\beta}_1$

hon hadofna njib l t-value wl t-critical w n2arendun

t-value :

for $\hat{\beta}_0$: $t = \frac{\hat{\beta}_0}{S\hat{\beta}_0}$ | for $\hat{\beta}_1$: $t = \frac{\hat{\beta}_1}{S\hat{\beta}_1}$

A. For $\hat{\beta}_0$

```
# Find t_value
t <- model$coefficient[1] / SB0
```

t_critical bl 3ode mnjiba mn table bl R ela function

```
t_critical <- qt(df= length(y) - 2 , p = 0.25)
```

mn2arendun : ez t > t-critical => reject Ho

B. for $\hat{\beta}_1$

```
t <- model$coefficient[2] / SB1
t_critical <- qt(df= length(y) - 2 , p=0.25)
```

16. ANOVA

```
anova(our model)
```