Contoh 1.

Data tentang Sel darah merah (Y) dan Hemoglobin (X)

Y(juta/ml)	7,3	6,9	8,3	5,4	6,7	5,1	7,0	8,5	7,8
X(g/dl)	93	96	108	86	92	80	96	117	95

Y(juta/ml)	7,4	7,6	7,8	6,8
X (g/dl)	92	96	108	92

a. Tentukan model prediksi nya!

b. Lakukan uji asumsi model regresi linear

c. Lakukan deteksi pencilannya

d. Lakukan uji signifikansi model regresi linear baik simultan maupun parsial

e. Tentukan koefisien determinasi

Sofware R untuk contoh Regresi sederhana

Perintah-perintah yang digunakan sebagai berikut.

1. **Cara memasukkan data dalam R, Output datanya tunjukkan!**
2. Cara menentuksn model prediksinya
3. Cara menentukan output uji asumsinya, Normalitas, Homogenitas, non autokorelasi
4. Cara menentukan chek outliernya
5. Cara memunculkan tabel anava nya baik yg simultasn maupun parsial dab koefisien determinasinya

Petunjuk :

1. Cara Memasukkan Data dalam R
   - Masukkan data pada microsoft excel (variabel dependen dan independen)
   - Save file dengan tipe CSV (Comma delimited)
   - Buka software R
   - Masukkan syntax berikut untuk memunculkan data

     ```
     > data=read.csv(file.choose())
     > data
     ```

2. **Metode Kuadrat Terkecil (MKT)**

```
> model.mkt=lm(Y~.,data=data)
> summary(model.mkt)
Call:
lm(formula = Y ~ ., data = data)
Residuals:
    Min       1Q   Median       3Q      Max
-0.81464 -0.36812 -0.04741  0.46380  0.78621
Coefficients:
            Estimate   Std. Error   t value   Pr(>|t|)
(Intercept) -1.42170   1.47808      -0.962    0.356789
X            0.08879   0.01529       5.808    0.000118 ***
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
Residual standard error: 0.5167 on 11 degrees of freedom
Multiple R-squared:  0.7541,   Adjusted R-squared:  0.7318
F-statistic: 33.74 on 1 and 11 DF,  p-value: 0.0001178
```

3. Uji asumsi klasik

```
> library(car)
> library(MASS)
```

Uji Normalitas (Kolmogorov Smirnov)

```
> residu=resid(model.mkt)
> ks.test(residu,"pnorm")

        One-sample Kolmogorov-Smirnov test

data:  residu
D = 0.21587, p-value = 0.5115
alternative hypothesis: two-sided
```

Uji Homo Kedastisitas(Breusch Pagan)

```
> library(lmtest)
> bptest(model.mkt)

  studentized Breusch-Pagan test

data:  model.mkt
BP = 1.5598, df = 1, p-value = 0.2117
```

Uji Non autokorelasi (Durbin Watson)

```
> dwtest(model.mkt)

  Durbin-Watson test

data:  model.mkt
DW = 1.7804, p-value = 0.3515
alternative hypothesis: true autocorrelation is greater than 0
```

3. <u>Deteksi Pencilan (DFFITS)</u>

```
> dffits(model.mkt)
    1         2         3         4         5          6
 0.286440 -0.113225  0.135619 -0.869868 -0.029349  -0.941920
    7         8         9        10        11        12
-0.057010 -1.146838  0.501229  0.440577  0.289418 -0.388506
   13
 0.0325623
> nilai.pembanding.dffits=2*(sqrt(2/13))
> nilai.pembanding.dffits
```

[1] 0.7844645

$|(DFFITS)_i| > 2\sqrt{p/n}$ dengan $p = k + 1$ dan k adalah banyaknya variabel independen

4. Uji Simultan, Parsial dan Koefisien Determinasi

Petunjuk : Uji simultan, parsial, dan koefisien determinasi dapat dilihat pada output mkt.

```
> model.mkt=lm(Y~.,data=data)
> summary(model.mkt)
Call:
lm(formula = Y ~ ., data = data)
Residuals:
     Min       1Q   Median       3Q      Max
-0.81464 -0.36812 -0.04741  0.46380  0.78621
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -1.42170    1.47808  -0.962 0.356789
X            0.08879    0.01529   5.808 0.000118 ***
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
Residual standard error: 0.5167 on 11 degrees of freedom
Multiple R-squared:  0.7541,	Adjusted R-squared:  0.7318
F-statistic: 33.74 on 1 and 11 DF,  p-value: 0.0001178
```

Contoh 2

Data Produksi Jagung Kabupaten Karanganyar tahun 2016

Data berikut adalah dataProduksi Jagung Karanganyar (Y) dan yang dipengaruhi oleh Luas Lahan (X1) dan Curah Hujan (X2)

Kecamatan	Produksi	Luas Lahan	Curah Hujan
Jatipuro	4624	663	13,8
Jatiyoso	7533	1084	0
Jumapolo	7167	1030	11,58
Jumantono	6932	1000	15
Matesih	273	38	11,42
Tawangmangu	604	86	14,25
Ngargoyoso	836	120	9,5
KarangPandan	283	40	16,7
Karanganyar	485	69	18,5
Colomadu	777	110	14,42
Gondangrejo	481	69	0
KebakKramat	180	26	17,83
Mojogedang	3836	537	13,8
Kerjo	2443	350	16,3
Jenawi	2957	423	21,8

a.Tentukan model prediksinya

b. Lakukan uji asumsi klasik dan lakukan cek diasnotik untuk Data pada contoh Tabel 2.2.

c. Lakukan Uji Signifikansi Simultan dan Parsial Data pada Tabel 2.2

d. Tentukan Koefisien determinasi untuk data Tabel 2.2

Dik tolong yg ini juga, sesuai no satu hanya ujiasumsinya ditambah uji non multikolinearitas ya

Sofware R: Uji Asumsi, Deteksi Autlier , MKT, Uji Sifnifikansi untuk data Tabel 2.2

Tabel 2.2. DATA JAGUNG KARANGANYAR.

1. Cara Memasukkan Data dalam R

- Masukkan data pada microsoft excel (variabel dependen dan independen)
- Save file dengan tipe CSV (Comma delimited)
- Buka software R

- Masukkan syntax berikut untuk memunculkan data

  > data=read.csv(file.choose())

  > data

2. Metode Kuadrat Terkecil

```
> library(car)
> library(MASS)
> model.mkt=lm(Y~.,data=data)
> summary(model.mkt)

Call:
lm(formula = Y ~ ., data = data)

Residuals:
    Min      1Q  Median      3Q     Max
-33.394  -7.737  -4.989  -0.447  90.891

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   2.8147    21.3890   0.132    0.897
X1            6.9537     0.0200 347.692   <2e-16 ***
X2            0.5899     1.2983   0.454    0.658
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' '

Residual standard error: 28.71 on 12 degrees of freedom
Multiple R-squared:  0.9999,      Adjusted R-squared:  0.9999
F-statistic: 6.415e+04 on 2 and 12 DF,  p-value: < 2.2e-16
```

3. Uji Asumsi Klasik

> residumkt=resid(model.mkt)
> library(dwtest)
> library(car)
> library(lmtest)

Uji Normalitas

- **Shapiro-Wilk Test**

> shapiro.test(residumkt)

Shapiro-Wilk normality test

data: residumkt
W = 0.5661, p-value = 1.264e-05.

- **Kolmogorov-Smirnov Test**

```
> ks.test(residumkt,"pnorm")

One-sample Kolmogorov-Smirnov test

data:  residu
D = 0.61428, p-value = 5.66e-06
alternative hypothesis: two-sided
```

Uji Homoskesdastistias (Breusch-Pagan Test)

```
> bptest(model.mkt)

        studentized Breusch-Pagan test

data:  mkt
BP = 0.55062, df = 2, p-value = 0.7593
```

Uji Non - Autokorelasi

```
> dwtest(model.mkt)

        Durbin-Watson test

  data:  mkt
  DW = 2.2403, p-value = 0.6026
  alternative hypothesis: true autocorrelation is greater than 0
```

Uji Non-Multikolinearitas

```
> vif(model.mkt)
      X1       X2
1.061932 1.061932
```

4. **Deteksi Outlier**

```
> dffits(model.mkt)
      1         2         3         4          5         6         7         8
 0.0341 -0.380423 -0.116377 -0.899296   -0.01125  -0.06347  -0.09855 -0.10877
      9        10        11        12         13        14        15
-0.129536  0.00898 -0.0809 -0.215621  3.170750 -0.03512 -0.002142
> nilai.pembanding.dffits= 2*(sqrt(3/15))
> nilai.pembanding.dffits
[1] 0.8944272
```

5. **Uji Simultan dan Parsial**

Petunjuk : Uji simultan dan parsial dapat dilihat pada output mkt.

```
> model.mkt=lm(Y~.,data=data)
> summary(model.mkt)

Call:
lm(formula = Y ~ ., data = data)

Residuals:
    Min      1Q  Median      3Q     Max 
-33.394  -7.737  -4.989  -0.447  90.891 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)   2.8147    21.3890   0.132    0.897    
X1            6.9537     0.0200 347.692   <2e-16 ***
```

```
X2             0.5899     1.2983   0.454    0.658
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’

Residual standard error: 28.71 on 12 degrees of freedom
Multiple R-squared:  0.9999,	Adjusted R-squared:  0.9999
F-statistic: 6.415e+04 on 2 and 12 DF,  p-value: < 2.2e-16
```

6. **Koefisien Determinasi**

Petunjuk : Koefisien determinasi dapat dilihat juga pada output mkt.

```
> model.mkt=lm(Y~.,data=data)
> summary(model.mkt)

Call:
lm(formula = Y ~ ., data = data)

Residuals:
    Min      1Q  Median      3Q     Max
-33.394  -7.737  -4.989  -0.447  90.891

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   2.8147    21.3890   0.132    0.897
X1            6.9537     0.0200 347.692   <2e-16 ***
X2            0.5899     1.2983   0.454    0.658
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’

Residual standard error: 28.71 on 12 degrees of freedom
Multiple R-squared:  0.9999,	Adjusted R-squared:  0.9999
F-statistic: 6.415e+04 on 2 and 12 DF,  p-value: < 2.2e-16
```

Praktikum 3

Dari contoh 2, tolong cara estimasi M, S, LTS, LMS , MM dan hasilnya dik

1. **Estimasi LMS**

Iterasi 1

```
> library(dplyr)
> mkt=lm(Y~.,data=data)
> summary(mkt)

Call:
lm(formula = Y ~ ., data = data)

Residuals:
    Min      1Q  Median      3Q     Max
-33.394  -7.737  -4.989  -0.447  90.891

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   2.8147    21.3890   0.132    0.897
X1            6.9537     0.0200 347.692   <2e-16 ***
X2            0.5899     1.2983   0.454    0.658
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 28.71 on 12 degrees of freedom
Multiple R-squared:  0.9999,Adjusted R-squared:  0.9999
```

```
F-statistic: 6.415e+04 on 2 and 12 DF,  p-value: < 2.2e-16
```

Iterasi 2

```
> residumkt=resid(mkt)
> residukuadrat=(residumkt)^2
> med_1=median(residukuadrat)
> urutkan_1=arrange(data,residukuadrat)
> h1=(15/2)+((3+1)/2)
> h1
[1] 9.5
> data1=slice(urutkan_1,1:10)
> mkt1=lm(Y~.,data=data1)
> summary(mkt1)

Call:
lm(formula = Y ~ ., data = data1)

Residuals:
    Min      1Q  Median      3Q     Max 
-4.2803 -1.5889 -0.3601  2.0736  5.3802 

Coefficients:
             Estimate Std. Error  t value Pr(>|t|)    
(Intercept) -1.482118   2.655915   -0.558  0.59420    
X1           6.951997   0.002909 2389.603  < 2e-16 ***
X2           0.791867   0.170540    4.643  0.00236 ** 
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 3.429 on 7 degrees of freedom
Multiple R-squared:      1,	Adjusted R-squared:      1 
F-statistic: 2.954e+06 on 2 and 7 DF,  p-value: < 2.2e-16
```

Iterasi 3

```
> residumkt1=resid(mkt1)
> reskuadrat_1=residumkt1^2
> sigmareskuadrat_1=sum(reskuadrat_1)
> med_2=median(reskuadrat_1)
> med_2
[1] 4.065701
> urutkan_2=arrange(data1,reskuadrat_1)
> h2=(10/2)+((3+1)/2)
> h2
[1] 7
> data2=slice(urutkan_2,1:7)
> mkt2=lm(Y~.,data=data2)
> summary(mkt2)

Call:
lm(formula = Y ~ ., data = data2)

Residuals:
      1       2       3       4       5       6       7 
 0.7842  0.5129 -0.4987 -0.2698 -1.9724  1.0272  0.4167 

Coefficients:
            Estimate Std. Error  t value Pr(>|t|)    
(Intercept) 1.139614   1.067714    1.067 0.345954    
X1          6.948460   0.001184 5866.211 5.07e-15 ***
X2          0.728341   0.064425   11.305 0.000349 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

```
Residual standard error: 1.257 on 4 degrees of freedom
Multiple R-squared:      1, Adjusted R-squared:      1 
F-statistic: 1.801e+07 on 2 and 4 DF,  p-value: 1.234e-14
```

Iterasi 4

```
> residumkt2=resid(mkt2)
> reskuadrat_2=residumkt2^2
> med_3=median(reskuadrat_2)
> h3=(7/2)+((3+1)/2)
> h3
[1] 5.5
> med_3
[1] 0.2630418
> urutkan_3=arrange(data2,reskuadrat_2)
> data3=slice(urutkan_3,1:6)
> mkt3=lm(Y~.,data=data3)
> summary(mkt3)

Call:
lm(formula = Y ~ ., data = data3)

Residuals:
       1        2        3        4        5        6 
-0.21949  0.39939 -0.90847  0.16369  0.04522  0.51966 

Coefficients:
             Estimate Std. Error   t value Pr(>|t|)    
(Intercept) 1.1614924  0.5660810     2.052 0.132533    
X1          6.9483930  0.0006283 11059.656 1.63e-12 ***
X2          0.7625256  0.0356454    21.392 0.000224 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.6663 on 3 degrees of freedom
Multiple R-squared:      1, Adjusted R-squared:      1 
F-statistic: 6.352e+07 on 2 and 3 DF,  p-value: 3.629e-12
```

Iterasi 5

```
> residumkt_3=resid(mkt3)
> reskuadrat_3=residumkt_3^2
> med_4=median(reskuadrat_3)
> med_4
[1] 0.1038441
> h4=(6/2)+((3+1)/2)
> h4
[1] 5
> urutkan_4=arrange(data3,reskuadrat_3)
> data4=slice(urutkan_4,1:5)
> mkt4=lm(Y~.,data=data4)
> summary(mkt4)

Call:
lm(formula = Y ~ ., data = data4)

Residuals:
        1         2         3         4         5 
-0.198892  0.214131 -0.138571 -0.005393  0.128726 

Coefficients:
             Estimate Std. Error   t value Pr(>|t|)
```

```
(Intercept) 1.5992952  0.2309534     6.925 0.020224 *  
X1          6.9479145  0.0002556 27179.753 1.35e-09 ***
X2          0.7629257  0.0131703    57.928 0.000298 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.2462 on 2 degrees of freedom
Multiple R-squared:      1,	Adjusted R-squared:      1 
F-statistic: 3.806e+08 on 2 and 2 DF,  p-value: 2.627e-09
```

Iterasi 6

```
> residumkt4=resid(mkt4)
> reskuadrat_4=residumkt4^2
> med_5=median(reskuadrat_4)
> med_5
[1] 0.01920185
> h5=(5/2)+((3+1)/2)
> h5
[1] 4.5
> urutkan_5=arrange(data4,reskuadrat_4)
> data5=slice(urutkan_5,1:5)
> mkt5=lm(Y~.,data=data5)
> summary(mkt5)

Call:
lm(formula = Y ~ ., data = data5)

Residuals:
        1         2         3         4         5 
-0.005393  0.128726 -0.138571 -0.198892  0.214131 

Coefficients:
             Estimate Std. Error   t value Pr(>|t|)    
(Intercept) 1.5992952  0.2309534     6.925 0.020224 *  
X1          6.9479145  0.0002556 27179.753 1.35e-09 ***
X2          0.7629257  0.0131703    57.928 0.000298 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.2462 on 2 degrees of freedom
Multiple R-squared:      1,	Adjusted R-squared:      1 
F-statistic: 3.806e+08 on 2 and 2 DF,  p-value: 2.627e-09
```

Mencari Pembobot LMS

```
> sigma=1.4826*(1+(5/(15-3)))*sqrt(med_5)
> sigma
[1] 0.291047
> psi_e=weights(mkt)
> e.star=residumkt/sigma
> for(i in 1:15){
+   if(abs(e.star[i])<=2.5){psi_e[i]=e.star[i]}
+   else if (e.star[i]>2.5){psi_e[i]=2.5}
+   else psi_e[i]=(-2.5)
+ }
> psi_e
 [1]  2.5000000 -2.5000000 -2.5000000 -2.5000000 -2.5000000 -2.5000000 -2.5000000
 [8] -2.5000000 -2.5000000  2.5000000 -2.5000000 -2.5000000  2.5000000 -2.5000000
[15] -0.3512486
> w=psi_e/e.star
> w
          1           2           3           4           5           6 
0.267397760 0.094992198 0.145847243 0.021788680 0.917850738 0.138827056 
          7           8           9          10          11          12 
```

```
0.105971053 0.093108873 0.085254304 0.946207070 0.448551012 0.051498158
         13          14          15
0.008005361 0.224881692 1.000000000
```

Model Estimasi LMS

```
> model.lms=lm(Y~.,data=data,weight=w)
> summary(model.lms)

Call:
lm(formula = Y ~ ., data = data, weights = w)

Weighted Residuals:
    Min      1Q  Median      3Q     Max 
-4.5710 -1.9697 -0.9556  0.5793  8.2902 

Coefficients:
            Estimate Std. Error  t value Pr(>|t|)    
(Intercept)  0.66857    3.75280    0.178   0.8616    
X1           6.95206    0.00574 1211.086   <2e-16 ***
X2           0.68254    0.24711    2.762   0.0172 *  
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3.26 on 12 degrees of freedom
Multiple R-squared:      1,	Adjusted R-squared:      1 
F-statistic: 7.62e+05 on 2 and 12 DF,  p-value: < 2.2e-16
```

2. <u>**Estimasi LTS**</u>

```
> mkt=lm(Y~.,data=data)
> residumkt=resid(mkt)
> reskuadrat=residumkt^2
> sigma.reskuadrat=sum(reskuadrat)
> h1=(15/2+((2+2)/2))
> h1
[1] 9.5
> urutan_1=arrange(data,reskuadrat)
> data1=slice(urutan_1,1:10)
> mkt_1=lm(Y~.,data=data1)
> resmkt1=resid(mkt_1)
> reskuarat1=resmkt1^2
> sigma.reskuadrat1=sum(reskuarat1)
> sigma.reskuadrat1
[1] 84.82376
>  urutan_2=arrange(data1,reskuarat1)
>  h2=(10/2)+((2+2)/2)
>  h2
```

```
[1] 7
> data2=slice(urutan_2,1:7)
> mkt_2=lm(Y~.,data=data2)
> resmkt2=resid(mkt_2)
> reskuadrat2=resmkt2^2
> sigma.reskuadrat2=sum(reskuadrat2)
> sigma.reskuadrat2
[1] 28.76901
> urutan_3=arrange(data2,reskuadrat2)
> h3=(7/2)+((2+2)/2)
> h3
[1] 5.5
> data3=slice(urutan_3,1:6)
> mkt_3=lm(Y~.,data=data3)
> resmkt3=resid(mkt_3)
> reskuadrat3=resmkt3^2
> sigma.reskuadrat3=sum(reskuadrat3)
> sigma.reskuadrat3
[1] 19.35617
> urutan_4=arrange(data3,reskuadrat3)
> urutan_4=arrange(data3,reskuadrat3)
> h4=(6/2)+((2+2)/2)
> h4
[1] 5
> data4=slice(urutan_4,1:5)
> mkt_4=lm(Y~.,data=data4)
> resmkt4=resid(mkt_4)
> reskuadrat4=resmkt4^2
> sigma.reskuadrat4=sum(reskuadrat4)
> sigma.reskuadrat4
[1] 4.872157
> urutan_5=arrange(data4,reskuadrat4)
> h5=(5/2)+((2+2)/2)
> h5
```

```
[1] 4.5
>  data5=slice(urutan_5,1:5)
>  mkt_5=lm(Y~.,data=data5)
>  resmkt5=resid(mkt_5)
>  reskuadrat5=resmkt5^2
>  sigma.reskuadrat5=sum(reskuadrat5)
>  sigma.reskuadrat5
[1] 4.872157
> r=2.5
> chn<- 1/(qnorm(20/30))
> dhn=1/sqrt(1-(30/(5*chn))*dnorm(1/(chn)))
> slts<-dhn*sqrt((1/5)*sigma.reskuadrat5)
> e=resmk
> w=weights(mkt_5)
> for (i in 1:5){
+ if((e[i]/slts)>2.5){w[i]=0}
+ else w[i]=1
+ }
> w
[1] 1 1 1 1 1
> model.lts2=lm(Y~.,data=data5,weights=w)
> summary(model.lts2)
Call:
lm(formula = Y ~ ., data = data5, weights = w)
Residuals:
       1        2        3        4        5 
 0.03365  0.36924 -0.48455 -1.45860  1.54025 
Coefficients:
            Estimate Std. Error  t value Pr(>|t|)    
(Intercept) 1.210898   1.594199    0.760    0.527    
X1          6.948114   0.002141 3245.743 9.49e-08 ***
X2          0.260425   0.123231    2.113    0.169    
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

```
Residual standard error: 1.561 on 2 degrees of freedom
Multiple R-squared:      1,      Adjusted R-squared:      1
F-statistic: 8.037e+06 on 2 and 2 DF,  p-value: 1.244e-07
```