Resistant Line

This Markdown is licensed under the Assistant Lecturer AED-A

M. Bondan Vitto R. and ***Kevin AFR***

What is?

Garis resisten merupakan garis lurus y=a+bx, dimana:

- a = intersep/tinggi dari garis ketika x=0
- b = slope/kemiringan garis, dimana perubahan y dipengaruhi oleh kenaikan x satu satuan
- mx1 = nilai median x bagian 1
- my1 = nilai median y bagian 1
- dan seterusnya..

Rumus:

Rumus:

- $a = \frac{(my_1+my_2+my_3)-b(mx_1+mx_2+mx_3)}{3}$
- $b = \frac{(my_3-my_1)}{(mx_3-mx_1)}$

Kegunaan:

- Mengenali tren hubungan linear antara dua variabel, yang tidak banyak terpengaruh oleh adanya data outlier.

Resistant Line

getwd()

Import Data mtcars

```
data(mtcars)
head(mtcars)
```

Split Data

```
head(mtcars[c('mpg', 'disp')])
mtcars[c('mpg', 'disp')]
```

output:

```
                  mpg disp
Mazda RX4         21.0  160
Mazda RX4 Wag     21.0  160
Datsun 710        22.8  108
Hornet 4 Drive    21.4  258
```

```
Hornet Sportabout 18.7  360
Valiant           18.1  225
```

Take the Intercept and Slope Value

> Function = line(x, y, iter = 1)

Fit a line robustly as recommended in Exploratory Data Analysis.

```
x = mtcars$mpg
y = mtcars$disp
```

```
Resline = line(x, y, iter = 4)
Resline
```

Output:

```
Call:
line(x, y, iter = 4)

Coefficients:
[1]  603.92  -19.02
```

```
class(Resline)
```

Output:
[1] "tukeyline"

Methods are available for the generic functions coef, residuals, fitted, and print.

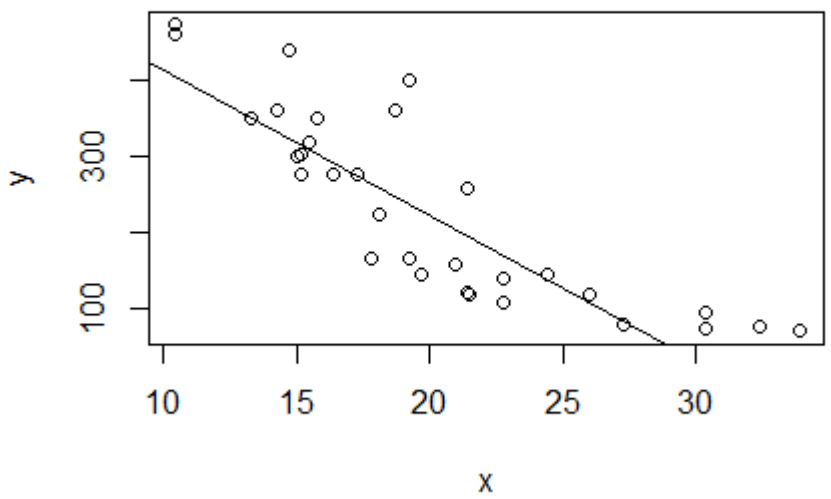
```
residuals(Resline)

coef(Resline)

fitted(Resline)

intersep = Resline$coefficients[1]
slope = Resline$coefficients[2]
```

Making Resistant Line Plot

1

```
plot(x, y)
abline(a = intersep, b = slope)
```

Output:

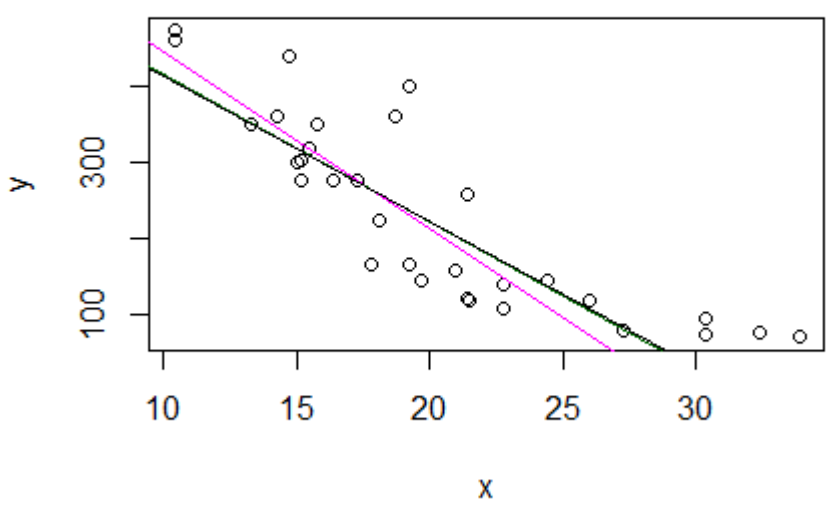


2

```
R1 = line(x, y, iter = 1)
R2 = line(x, y, iter = 2)
R3 = line(x, y, iter = 3)
R4 = line(x, y, iter = 4)

plot(x, y)
abline(a = R1$coefficients[1], b = R1$coefficients[2], col = 'magenta')
abline(a = R2$coefficients[1], b = R2$coefficients[2], col = 'darkgreen')
abline(a = R3$coefficients[1], b = R3$coefficients[2], col = 'lightblue')
abline(a = R4$coefficients[1], b = R4$coefficients[2], col = 'black')
```

---

output:



Data From the PDF Example

```
install.packages('readxl')
library(readxl)
df = read_excel('data temperature mortality.xlsx')
head(df)
```

Output:

```
# A tibble: 6 x 2
   temp mortal
  <dbl>  <dbl>
1  51.3  102.
```

```
2  49.9  104.
3  50    100.
4  49.2   95.9
5  48.5   87
6  47.8   95
```

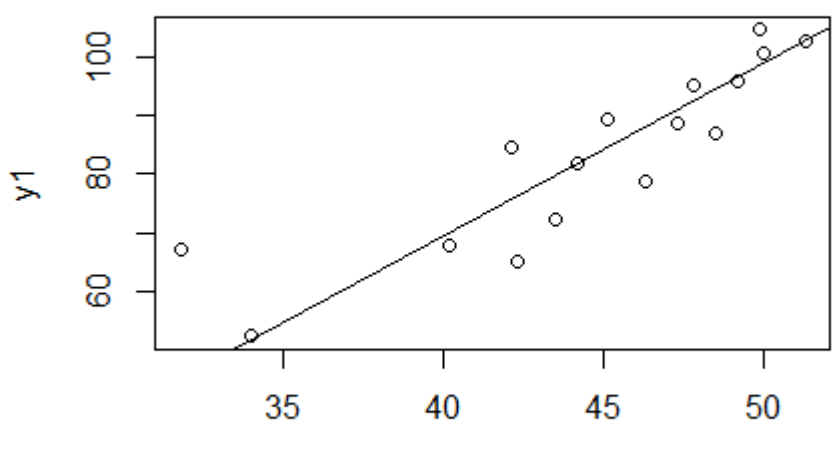
Take the Value of Intercept and Slope

```
x1 = df$temp
y1 = df$mortal

tempmor = line(x1, y1, iter = 4)
tempmor

residuals(tempmor)

coef(tempmor)

fitted(tempmor)

intersep = tempmor$coefficients[1]
slope = tempmor$coefficients[2]
```

Making Plot of PDF Example Data

```
plot(x1, y1)
abline(a = intersep, b = slope)
```

Output:



```
R1 = line(x1, y1, iter = 1)
R2 = line(x1, y1, iter = 2)
R3 = line(x1, y1, iter = 3)
R4 = line(x1, y1, iter = 4)

plot(x1, y1)
abline(a = R1$coefficients[1], b = R1$coefficients[2], col = 'red')
abline(a = R2$coefficients[1], b = R2$coefficients[2], col = 'green')
abline(a = R3$coefficients[1], b = R3$coefficients[2], col = 'blue')
#abline(a = R4$coefficients[1], b = R4$coefficients[2], col = 'black')
```

Output: