G64190069_Rizal Mujahiddan_UAS_AED22.pdf

Rizal Mujahiddan

6/10/2022

Import Library

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --

## v tibble  3.1.7     v dplyr   1.0.9
## v tidyr   1.2.0     v stringr 1.4.0
## v readr   2.1.2     v forcats 0.5.1
## v purrr   0.3.4

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()

## Loading required package: MASS

##
## Attaching package: 'MASS'

## The following object is masked from 'package:dplyr':
##
##     select

## Loading required package: survival

##
## Attaching package: 'bestNormalize'

## The following object is masked from 'package:MASS':
##
##     boxcox
```

Buka Data terlebih Dahulu

```
data_uas <- read.csv("DATA UAS AED - 2022 - DATA SET.csv")
head(data_uas)
```

```
##   x  y
## 1 1  0
## 2 1 11
## 3 1 12
## 4 1  3
## 5 2 60
## 6 2 73
```

Summary

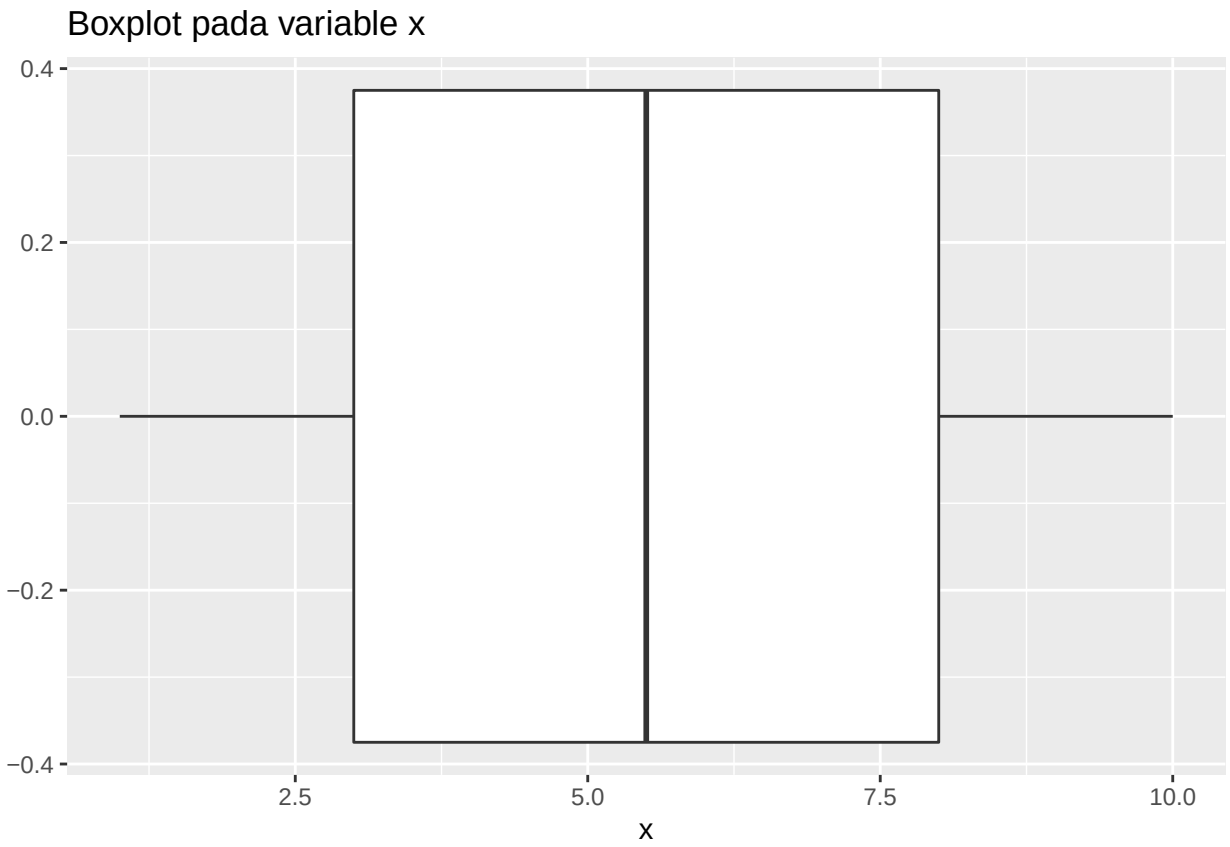
```
summary(data_uas)
```

```
##        x              y
##  Min.   : 1.0   Min.   :    0.0
##  1st Qu.: 3.0   1st Qu.:  187.8
##  Median : 5.5   Median : 1193.5
##  Mean   : 5.5   Mean   : 2189.8
##  3rd Qu.: 8.0   3rd Qu.: 3585.0
##  Max.   :10.0   Max.   :10000.0
```

Check Outlier

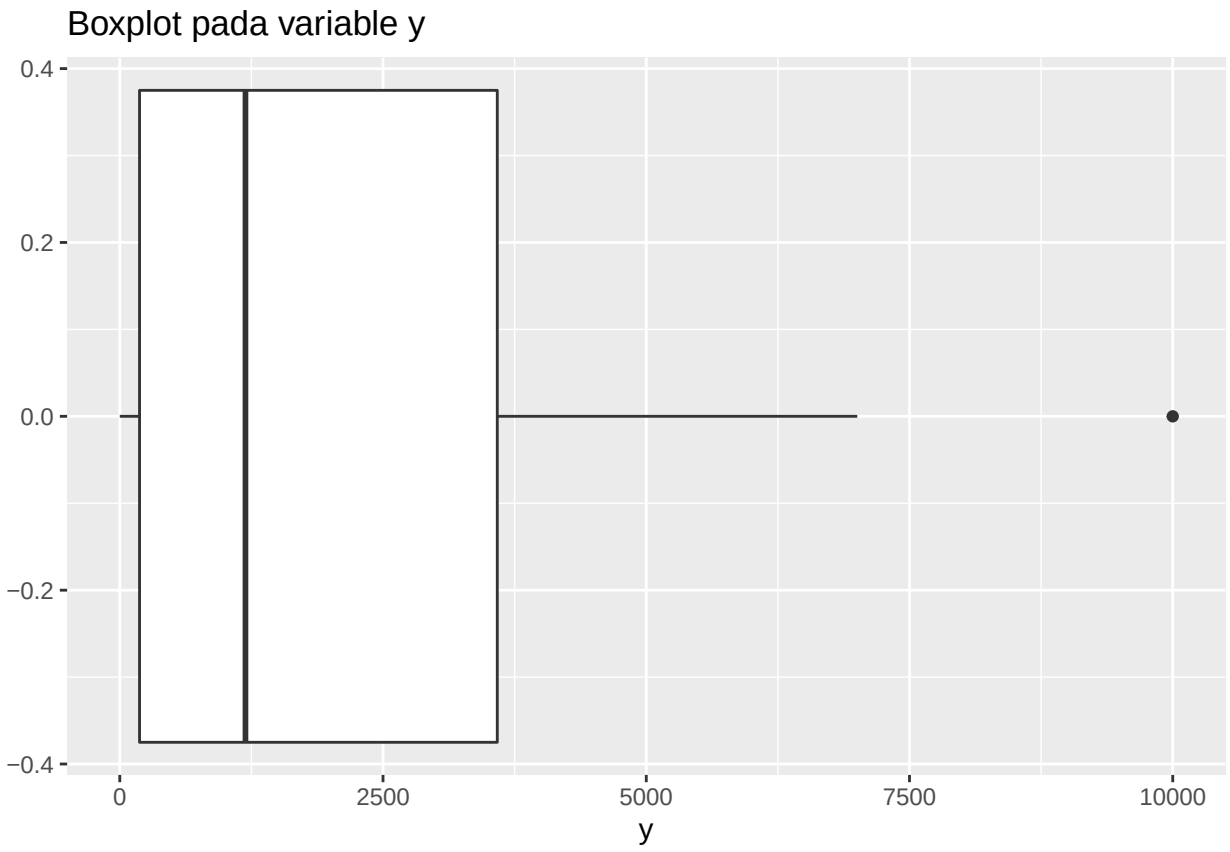
X

```
ggplot(data_uas,aes(x=x)) + geom_boxplot() +
  ggtitle("Boxplot pada variable x")
```



- Tidak ada outlier yang perlu dikhawatirkan
- Cenderung mendekati normal dikarenakan garis mediannya membagi panjang boxplot itu sendiri

```
ggplot(data_uas,aes(x=y)) + geom_boxplot() +
  ggtitle("Boxplot pada variable y")
```

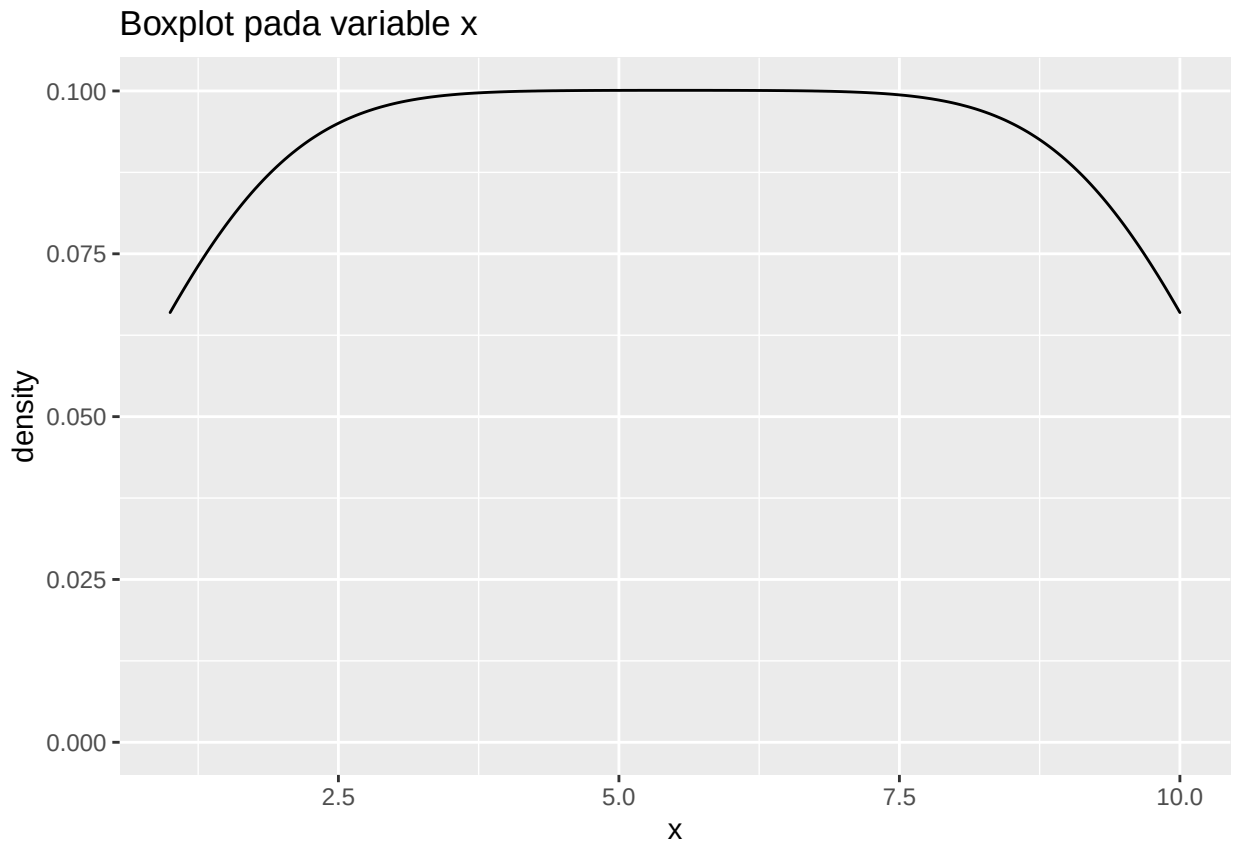


- Ada outlier
- Cenderung tidak normal (Panjang sebelah)

Density Plot

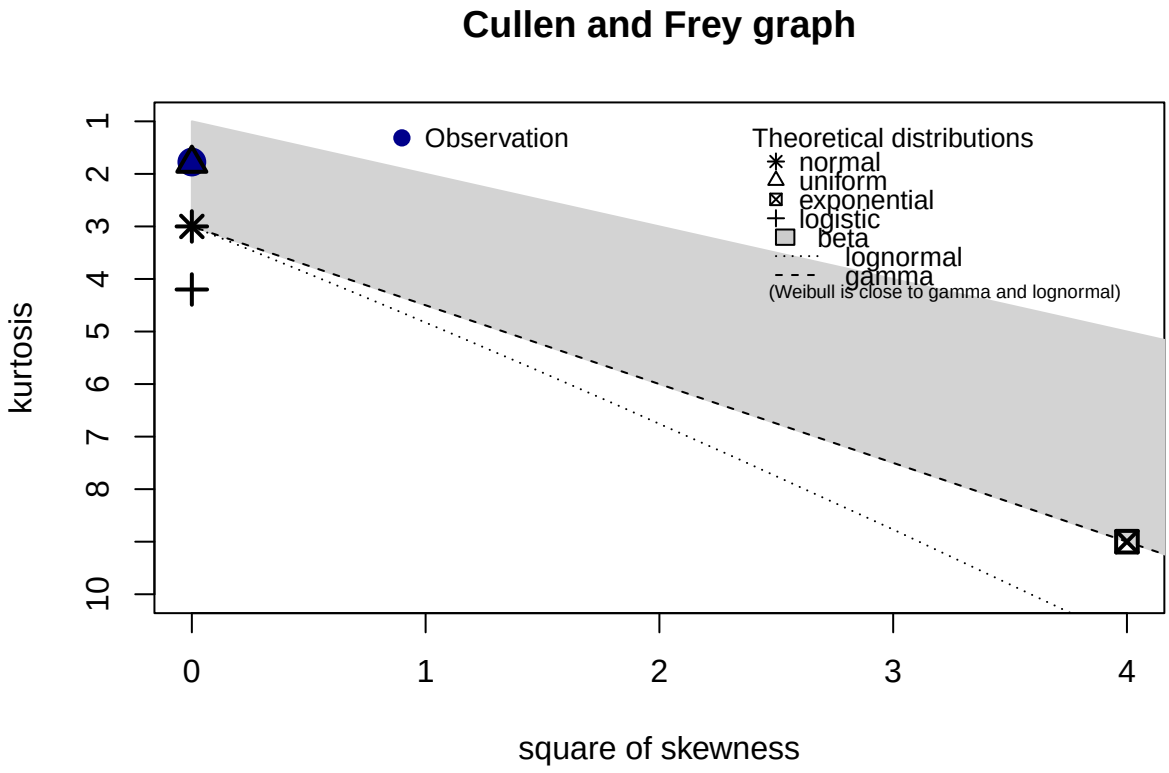
x

```
ggplot(data_uas,aes(x=x)) + geom_density() +
  ggtitle("Boxplot pada variable x")
```



- Bisa Dikatakan mendekati normal saja, dikarenakan masih ada kemungkinan distribusi x itu mendekati distribusi **uniform**
- kemudian ternyata kurang begitu memuncak distribusinya

```
descdist(data_uas$x)
```

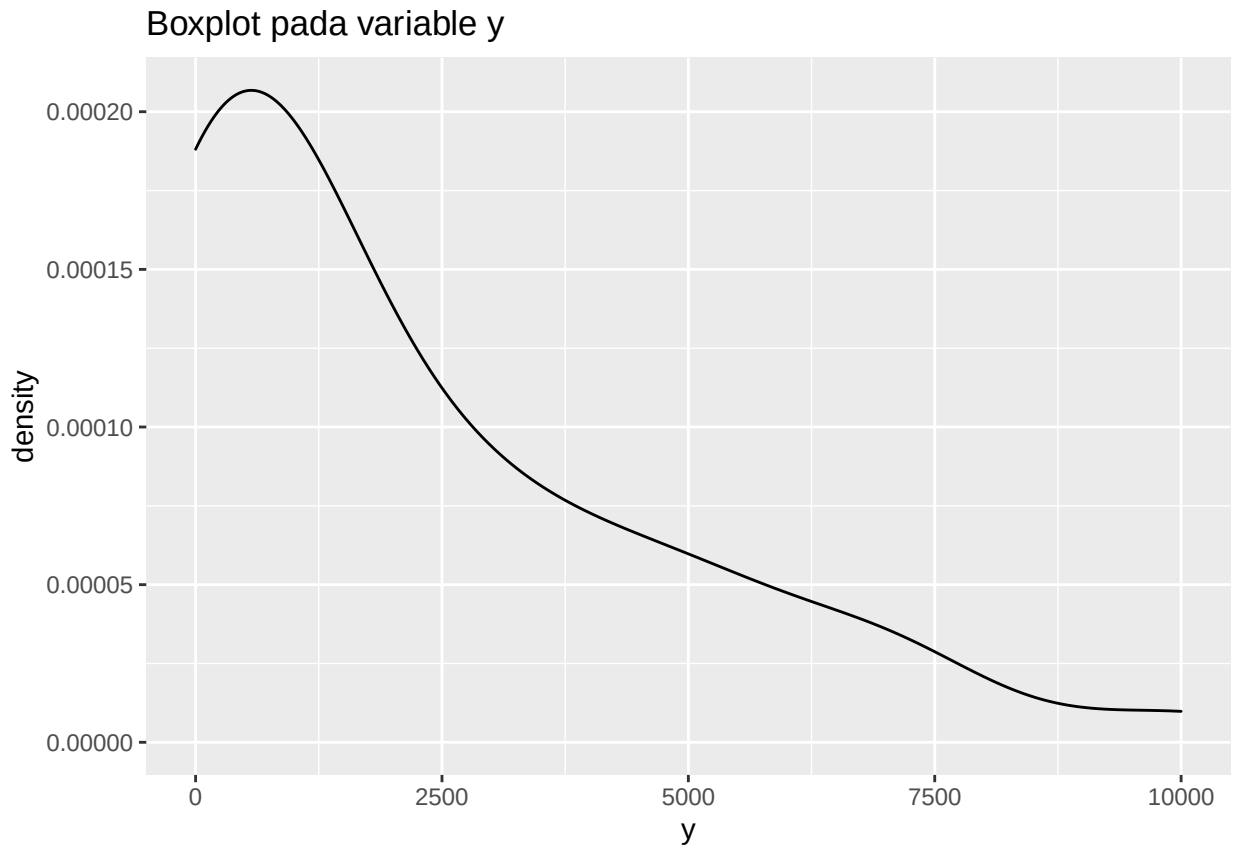



```
## summary statistics
## ------
## min:  1   max:  10
## median:  5.5
## mean:  5.5
## estimated sd:  2.908872
## estimated skewness:  0
## estimated kurtosis:  1.774137
```

- Sudah dipastikan dengan **Cullen and frey graph** , bahwa data x merupakan data berdistribusi uniform bukan normal

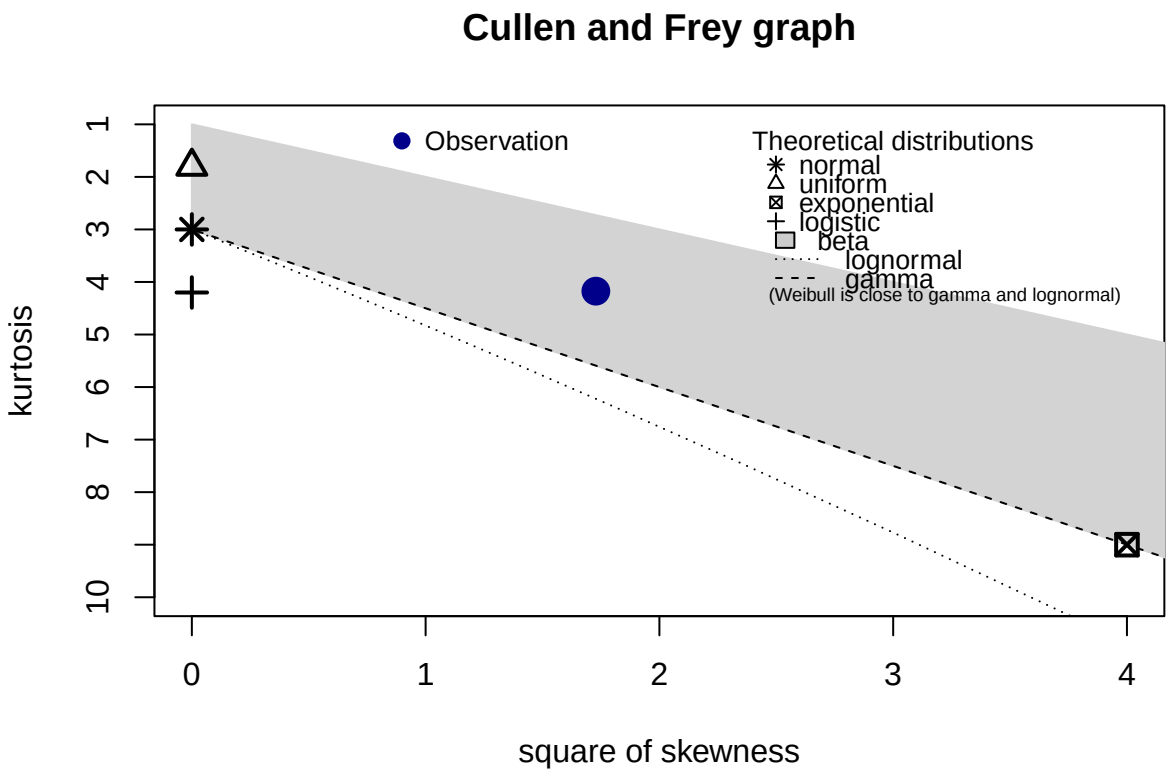
y

```
ggplot(data_uas,aes(x=y)) + geom_density() +
  ggtitle("Boxplot pada variable y")
```



- distribusinya menjulur kekanan
- distribusi masih membingungkan dikarenakan grafiknya tidak mewakili grafik apapun ya

```
descdist(data_uas$y)
```

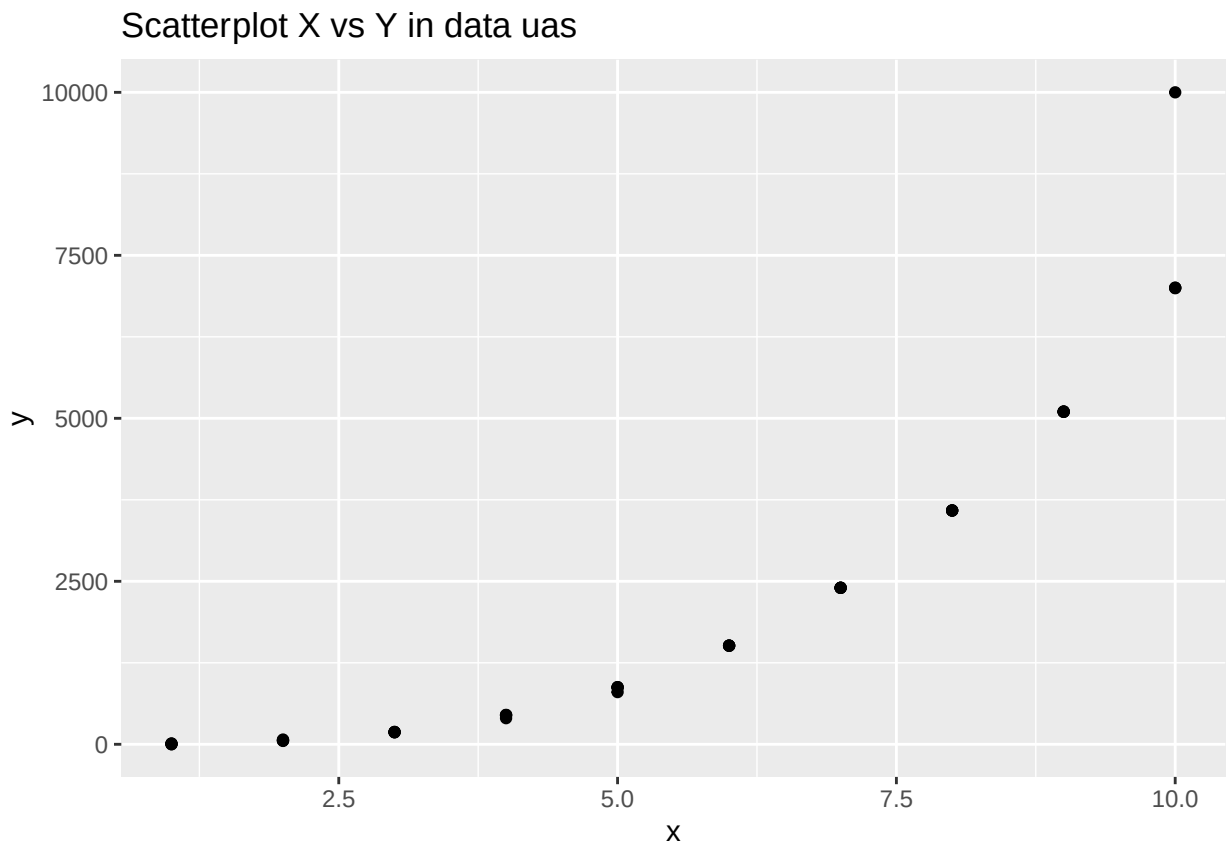



```
## summary statistics
## ------
## min:  0   max:  10000
## median:  1193.5
## mean:  2189.8
## estimated sd:  2515.642
## estimated skewness:  1.314584
## estimated kurtosis:  4.174921
```

- Membingungkan, dikarenakan data tersebut masuk ke distribusi beta, bukan ke distribusi yang lainnya. maka diperlukan **transformasi data**

Scatterplot

```
ggplot(data_uas,aes(x,y)) + geom_point() +
  ggtitle("Scatterplot X vs Y in data uas")
```



- JIka Diperhatikan, hubungan antara x dan y bukanlah linear
- kemungkinan hubungan data tersebut merupakan eksponensial maupun polinomial dikarenakan adanya cekung keatas.

Transformasi data

Berdasarkan dari internet dan artikel, dengan menggunakan bestNormalize. maka akan ditunjukkan dengan metode terbaik untuk menormalisasikan data tersebut

```
set.seed(1234)
ytransform <- bestNormalize(data_uas$y,standardize = FALSE)
ytransform
```

```
## Best Normalizing transformation with 40 Observations
##  Estimated Normality Statistics (Pearson P / df, lower => more normal):
##  - arcsinh(x): 1.72
##  - Log_b(x+a): 1.88
##  - No transform: 2.56
##  - orderNorm (ORQ): 1.56
##  - sqrt(x + a): 1.4
##  - Yeo-Johnson: 1.4
## Estimation method: Out-of-sample via CV with 10 folds and 5 repeats
##
```

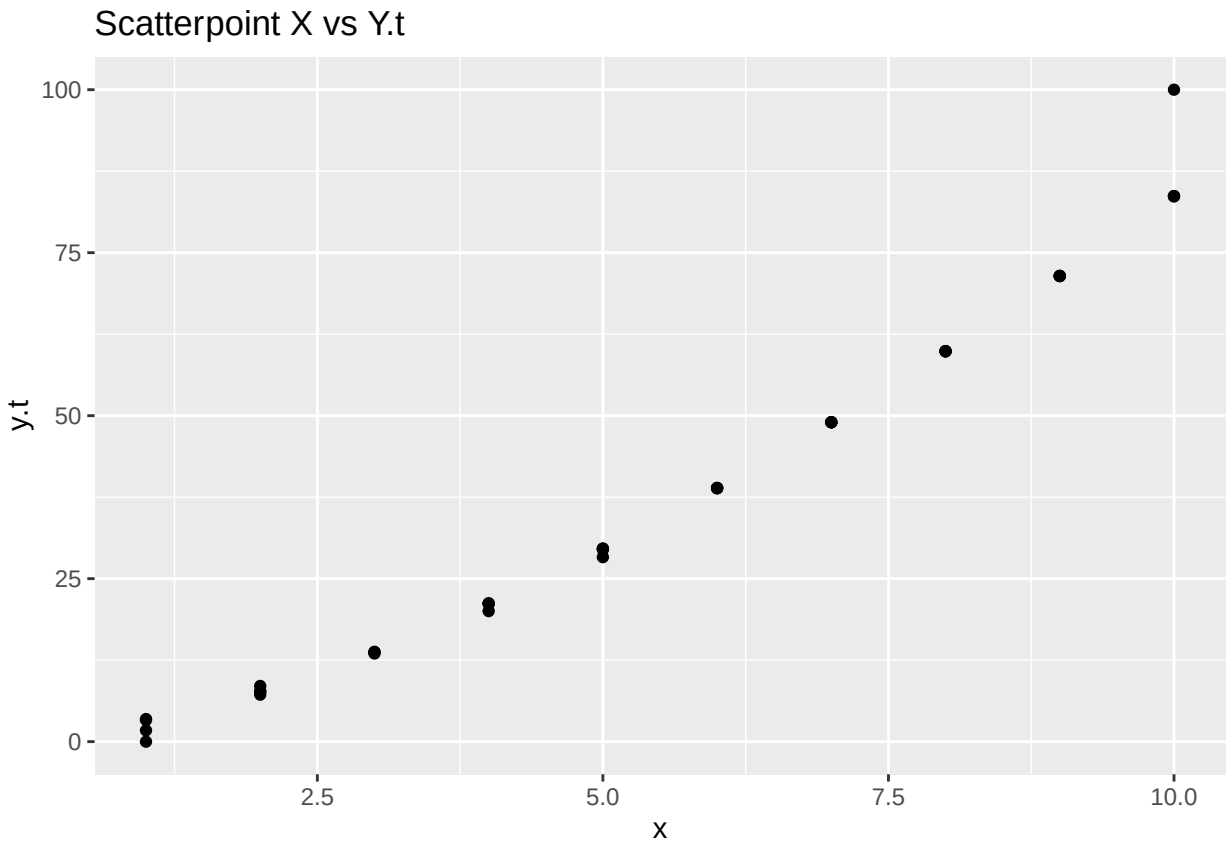
```
## Based off these, bestNormalize chose:
## Non-Standardized sqrt(x + a) Transformation with 40 nonmissing obs.:
##  Relevant statistics:
##  - a = 0
##  - mean (before standardization) = 38.06826
##  - sd (before standardization) = 27.5608
```

- Normalisasi yang bagus adalah dengan mengakarkan nilai y tersebut, sesuai dengan fungsi tersebut

```
data_uas$y.t <- ytransform$x.t
ytransform$x.t
```

```
##  [1]   0.000000   3.316625   3.464102   1.732051   7.745967   8.544004
##  [7]   7.681146   7.211103  13.711309  13.674794  13.784049  13.527749
## [13]  21.142375  21.236761  21.142375  20.024984  29.461840  29.614186
## [19]  29.614186  28.301943  38.923001  38.858718  38.884444  38.910153
## [25]  49.000000  48.989795  48.989795  49.000000  59.908263  59.874870
## [31]  59.874870  59.874870  71.456280  71.407283  71.421285  71.421285
## [37]  83.642095  83.689904  83.671979 100.000000
```

```
ggplot(data_uas,aes(x,y.t)) + geom_point() + ggtitle("Scatterpoint X vs Y.t")
```



* Jika diperhatikan, nilainya lumayan bagus, tetapi permasalahnnya mengapa ada nilai yang menumpuk pada x yang sama, inilah permasalahannya

Duga Parameter

linear regression

```
linear_reg <- lm('y.t ~ x',data=data_uas)
summary(linear_reg)
```

```
##
## Call:
## lm(formula = "y.t ~ x", data = data_uas)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -5.096 -3.088 -1.350  2.318 19.904
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -13.2996     1.6049  -8.287 4.81e-10 ***
## x             9.3396     0.2586  36.109  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.699 on 38 degrees of freedom
## Multiple R-squared:  0.9717, Adjusted R-squared:  0.9709
## F-statistic:  1304 on 1 and 38 DF,  p-value: < 2.2e-16
```

nilai duga dari suatu linear regresi adalah y = 9.3396 * x - 13.2996

```
loessku <- loess('y.t ~ x',data=data_uas)
summary(loessku)
```

```
## Call:
## loess(formula = "y.t ~ x", data = data_uas)
##
## Number of Observations: 40
## Equivalent Number of Parameters: 4.2
## Residual Standard Error: 2.51
## Trace of smoother matrix: 4.58  (exact)
##
## Control settings:
##   span     :  0.75
##   degree   :  2
##   family   :  gaussian
##   surface  :  interpolate      cell = 0.2
##   normalize:  TRUE
##  parametric:  FALSE
## drop.square:  FALSE
```

optional jika diperlukan suatu transformasi pada X pula

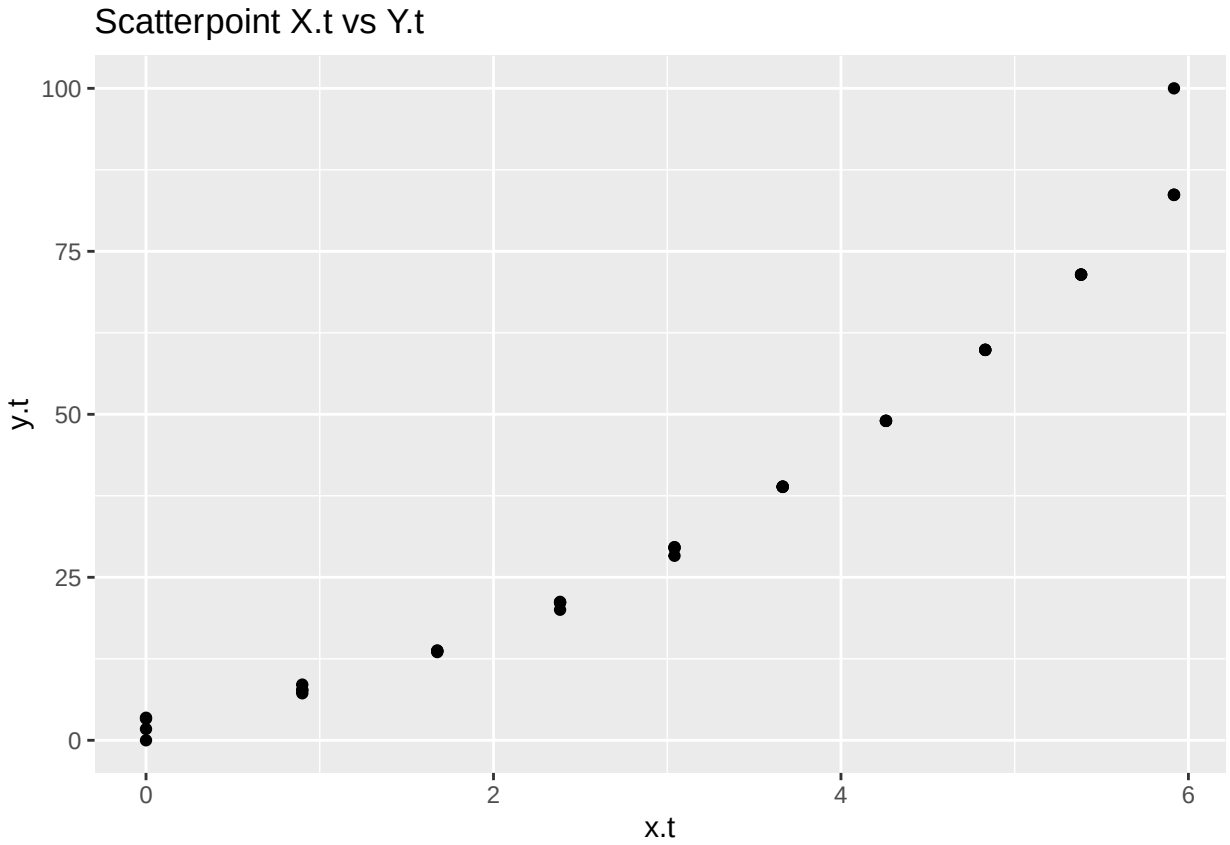
```
set.seed(1234)
xtransform <- bestNormalize(data_uas$x,standardize = FALSE)
xtransform
```

```
## Best Normalizing transformation with 40 Observations
##  Estimated Normality Statistics (Pearson P / df, lower => more normal):
##  - arcsinh(x): 1.72
##  - Box-Cox: 1.52
##  - Exp(x): 4.28
##  - Log_b(x+a): 1.72
##  - No transform: 1.68
##  - orderNorm (ORQ): 1.56
##  - sqrt(x + a): 1.64
##  - Yeo-Johnson: 1.56
## Estimation method: Out-of-sample via CV with 10 folds and 5 repeats
##
## Based off these, bestNormalize chose:
## Non-Standardized Box Cox Transformation with 40 nonmissing obs.:
##  Estimated statistics:
##  - lambda = 0.7219552
##  - mean (before standardization) = 3.205454
##  - sd (before standardization) = 1.886104
```

```
data_uas$x.t <- xtransform$x.t
xtransform$x.t
```

```
##  [1] 0.0000000 0.0000000 0.0000000 0.0000000 0.8995238 0.8995238 0.8995238
##  [8] 0.8995238 1.6764829 1.6764829 1.6764829 1.6764829 2.3832127 2.3832127
## [15] 2.3832127 2.3832127 3.0419262 3.0419262 3.0419262 3.0419262 3.6647414
## [22] 3.6647414 3.6647414 3.6647414 4.2592063 4.2592063 4.2592063 4.2592063
## [29] 4.8304327 4.8304327 4.8304327 4.8304327 5.3820896 5.3820896 5.3820896
## [36] 5.3820896 5.9169252 5.9169252 5.9169252 5.9169252
```

```
ggplot(data_uas,aes(x.t,y.t)) + geom_point() + ggtitle("Scatterpoint X.t vs Y.t")
```



* bagus

duga parameter transformasi X dan Y

```
linear_reg <- lm('y.t ~ x.t',data=data_uas)
summary(linear_reg)
```

```
##
## Call:
## lm(formula = "y.t ~ x.t", data = data_uas)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -7.444 -5.250 -1.921  2.452 23.427
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -7.4512     2.0718  -3.596 0.000916 ***
## x.t          14.2006     0.5589  25.410  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.583 on 38 degrees of freedom
## Multiple R-squared:  0.9444, Adjusted R-squared:  0.943
```

```
## F-statistic: 645.6 on 1 and 38 DF,  p-value: < 2.2e-16
```

nilai duga dari suatu linear regresi adalah y = 14.2006 * x - 7.4512

```
loessku <- loess('y.t ~ x.t',data=data_uas)
summary(loessku)
```

```
## Call:
## loess(formula = "y.t ~ x.t", data = data_uas)
##
## Number of Observations: 40
## Equivalent Number of Parameters: 4.36
## Residual Standard Error: 2.535
## Trace of smoother matrix: 4.76  (exact)
##
## Control settings:
##   span     :  0.75
##   degree   :  2
##   family   :  gaussian
##   surface  :  interpolate      cell = 0.2
##   normalize:  TRUE
##  parametric:  FALSE
## drop.square:  FALSE
```