Analisis Statistika dan Pemodelan

Materi 4 – STK571 Komputasi Statistik

Outline

- Analisis Statistik
- Pengujian Hipotesis
  - T-test
  - Non-parametric tests
- Pemodelan Linier
  - Analisis Regresi
    - Diagnostics
- Pendugaan Parameter
  - Optimasi

Analisis statistik

- Beberapa fungsi untuk meringkas data:

mean	var	length
cor	min	max
median	quantile	summary

Pengujian Hipotesis

- Fungsi t.test dapat digunakan untuk melakukan uji satu dan dua populasi
  - `t.test(x, y = NULL, alternative = c("two.sided", "less", "greater"),`
  - `mu = 0, paired = FALSE, var.equal = FALSE, conf.level = 0.95, ...)`
- Satu populasi : t.test(x)
- Dua populasi:

  Ragam sama: t.test(x,y,var.equal=TRUE)

  Ragam beda: t.test(x,y,var.equal=FALSE)

  Data berpasangan : t.test(x,y,paired=TRUE)

Pengujian Hipotesis Non-parametrik

- R memberikan fungsi untuk Mann-Whitney U, Wilcoxon Signed Rank, Kruskal Wallis, and Friedman tests.
- # independent 2-group Mann-Whitney U Test
  wilcox.test(y~A)
  # where y is numeric and A is A binary factor
- # independent 2-group Mann-Whitney U Test
  wilcox.test(y,x) # where y and x are numeric
- # dependent 2-group Wilcoxon Signed Rank Test
  wilcox.test(y1,y2,paired=TRUE) # where y1 and y2 are numeric

Pengujian Hipotesis Non-parametrik

- # Kruskal Wallis Test One Way Anova by Ranks
  kruskal.test(y~A) # where y1 is numeric and A is a factor
- # Randomized Block Design - Friedman Test
  friedman.test(y~A|B)
  # where y are the data values, A is a grouping factor
  # and B is a blocking factor

Pemodelan linier

- Fungsi lm dapat digunakan untuk melakukan pemodelan linier diantaranya adalah regresi
- Beberapa fungsi untuk mengekstrak objek lm adalah : coef, effects, residuals, fitted, vcov, predict, confint, summary
- Formula model menggunakan operator ~

Symbol	Example	Meaning
+	+X	include this variable
−	−X	delete this variable
:	X:Z	include the interaction between these variables
*	X*Y	include these variables and the interactions between them
|	X|Z	conditioning: include x given z
^	(X + Z + W)^3	include these variables and all interactions up to three way
I	I(X*Z)	as is: include a new variable consisting of these variables multiplied
1	X - 1	intercept: delete the intercept (regress through the origin)

Ilustrasi Regresi

Percobaan dalam bidang lingkungan

Apakah semakin tua mobil semakin besar juga emisi HC yang dihasilkan?

Diambil contoh 10 mobil secara acak, kemudian dicatat jarak tempuh yang sudah dijalani mobil (dalam ribu kilometer) dan diukur Emisi HC-nya (dalam ppm)

Jarak	Emisi
31	553
38	590
48	608
52	682
63	752
67	725
75	834
84	752
89	845
99	960

$$\text{Emisi} = 382 + 5.39\ \text{Jarak}$$

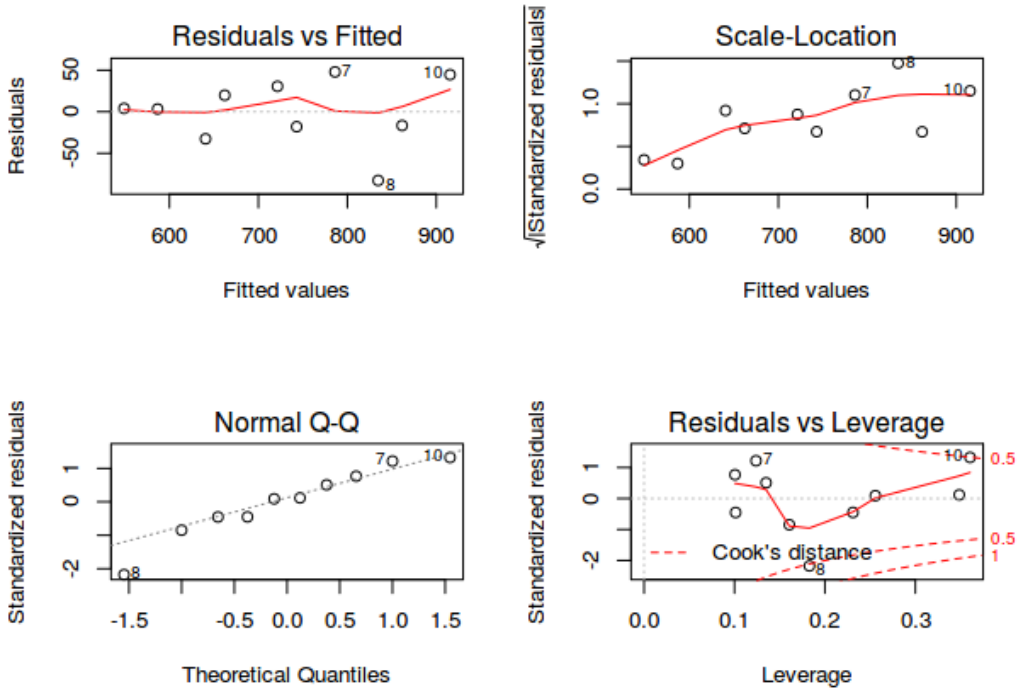
```
> jarak <- c(31,38,48,52,63,67,75,84,89,99)
> emisi <- c(553,590,608,682,752,725,834,752,845,960)
> hasil <- lm(emisi~jarak)
> coef(hasil)
(Intercept)       jarak
  381.95060     5.38931
> anova(hasil)
Analysis of Variance Table

Response: emisi
          Df Sum Sq Mean Sq F value    Pr(>F)
jarak      1 131932  131932  74.757 2.486e-05 ***
Residuals  8  14118    1765
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
>
> resid(hasil)
         1          2          3          4          5          6          7          8
  3.980803   3.255636 -32.637460  19.805301  30.522895 -18.034343  47.851180 -82.652607
         9         10
-16.599155  44.507749
> fitted(hasil)
       1        2        3        4        5        6        7        8        9       10
549.0192 586.7444 640.6375 662.1947 721.4771 743.0343 786.1488 834.6526 861.5992 915.4923
>
```

Analisis Regresi Ganda

- fit <- lm(y ~ x1 + x2 + x3, data=mydata)
  summary(fit) # show results
- # Other useful functions
  coefficients(fit) # model coefficients
  confint(fit, level=0.95) # CIs for model parameters
  fitted(fit) # predicted values
  residuals(fit) # residuals
  anova(fit) # anova table
  vcov(fit) # covariance matrix for model parameters
  influence(fit) # regression diagnostics

Plot Diagnostik

- memberikan pemeriksaan untuk: heteroscedasticity, normality, dan influential observerations.
- # diagnostic plots
  layout(matrix(c(1,2,3,4),2,2)) # optional 4 graphs/page
  plot(fit)



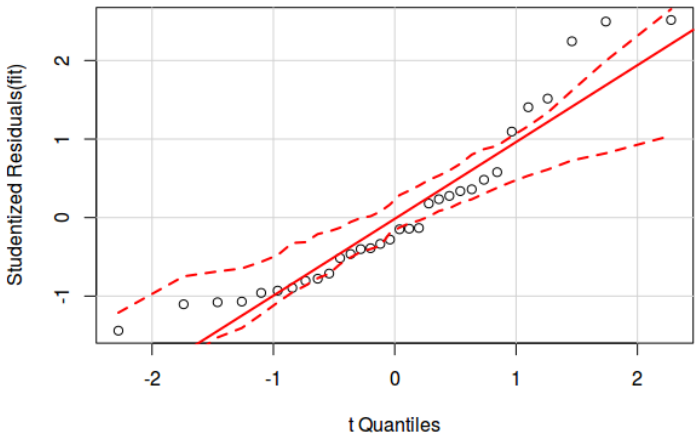
Plot Diagnostik

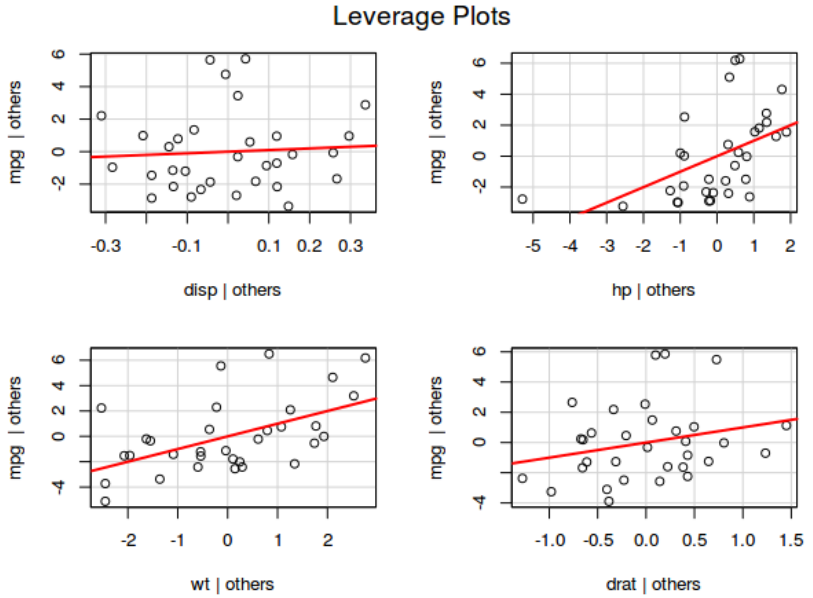
- Paket "car" memberikan fungsi-fungsi yang lebih powerfull dalam analisis regresi

```
library(car)
fit <- lm(mpg~disp+hp+wt+drat, data=mtcars)
```

Pemeriksaan Pencilan

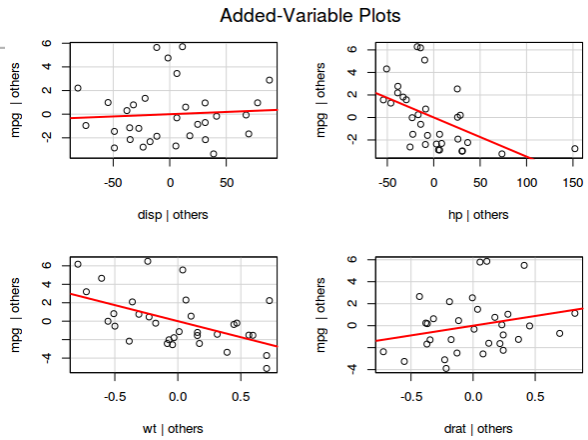
```
# Assessing Outliers
outlierTest(fit) # Bonferonni p-value for most extreme obs
qqPlot(fit, main="QQ Plot") #qq plot for studentized resid
layout(matrix(c(1,2,3,4,5,6),2,3)) # optional layout
leveragePlots(fit, ask=FALSE)
```

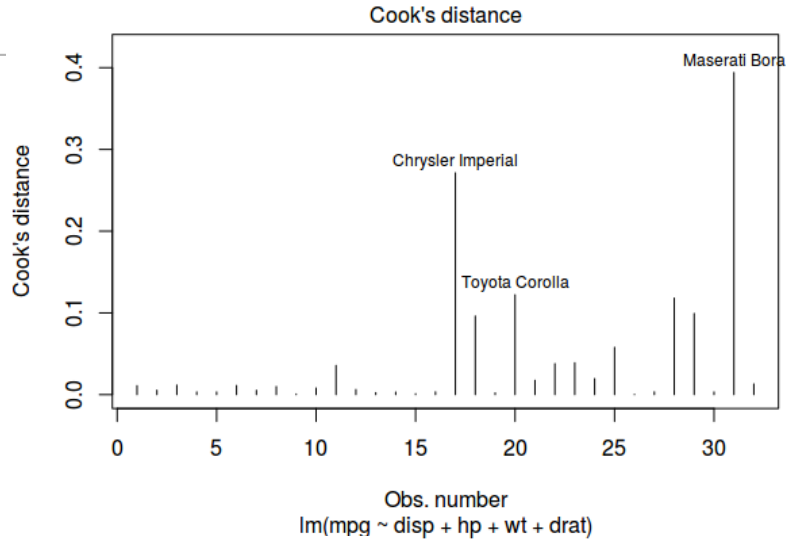


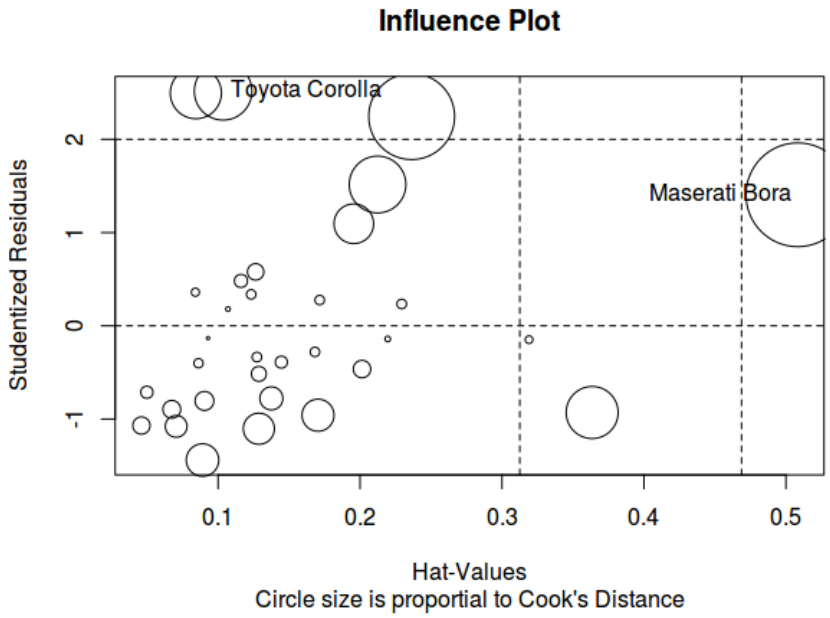


Influential Observations

```
# Influential Observations
# added variable plots
avPlots(fit, one.page=TRUE, ask=FALSE)
# Cook's D plot
# identify D values > 4/(n-k-1)
cutoff <- 4/((nrow(mtcars)-length(fit$coefficients)-2))
plot(fit, which=4, cook.levels=cutoff)
# Influence Plot
influencePlot(fit, main="Influence Plot",
  sub="Circle size is proportial to Cook's Distance" )
```

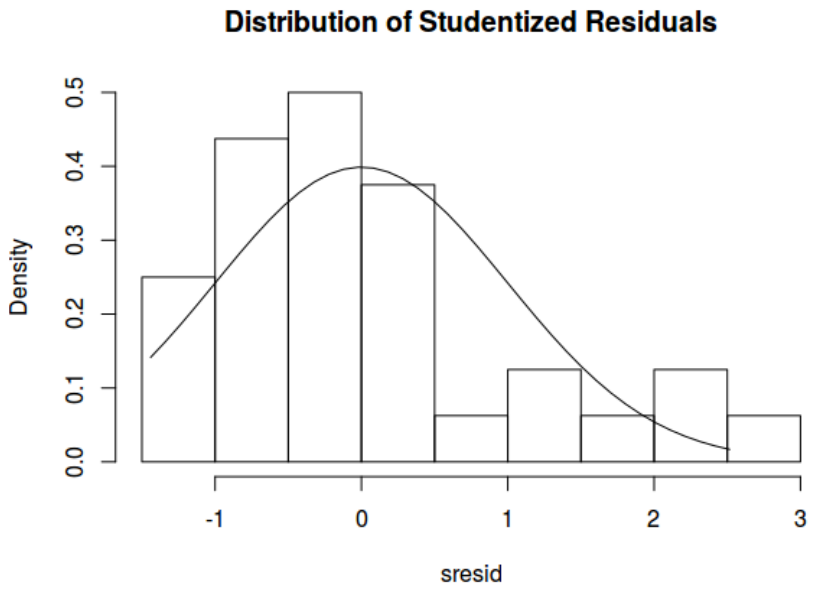

Added-Variable Plots
mpg | others
disp | others
hp | others
wt | others
drat | others


Cook's distance
Cook's distance
Maserati Bora
Chrysler Imperial
Toyota Corolla
Obs. number
lm(mpg ~ disp + hp + wt + drat)


Influence Plot
Toyota Corolla
Maserati Bora
Studentized Residuals
Hat-Values
Circle size is proportial to Cook's Distance

Non-normality

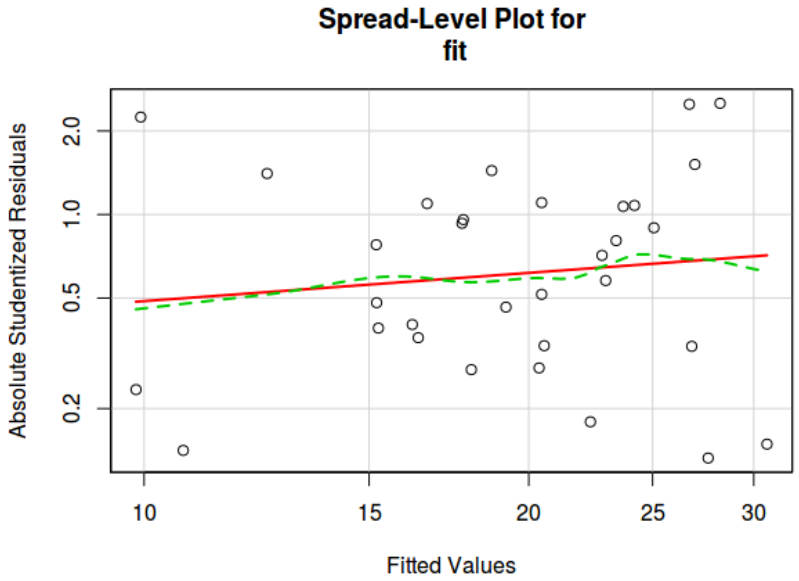
```
# Normality of Residuals
# qq plot for studentized resid
qqPlot(fit, main="QQ Plot")
# distribution of studentized residuals
library(MASS)
sresid <- studres(fit)
hist(sresid, freq=FALSE,
   main="Distribution of Studentized Residuals")
xfit<-seq(min(sresid),max(sresid),length=40)
yfit<-dnorm(xfit)
lines(xfit, yfit)
```



Non-constant Error Variance

```
# Evaluate homoscedasticity
# non-constant error variance test
ncvTest(fit)
# plot studentized residuals vs. fitted values
spreadLevelPlot(fit)
```

```
> ncvTest(fit)
Non-constant Variance Score Test
Variance formula: ~ fitted.values
Chisquare = 1.429672    Df = 1     p = 0.231818
```



Multi-collinearity

Evaluate Collinearity
vif(fit) # variance inflation factors
sqrt(vif(fit)) > 2 # problem?

```
> vif(fit)
    disp       hp       wt     drat
8.209402 2.894373 5.096601 2.279547
> sqrt(vif(fit)) > 2
 disp    hp    wt  drat
 TRUE FALSE  TRUE FALSE
```

Fungsi optimisasi lain

- R memiliki banyak fungsi untuk optimisasi dari suatu fungsi, diantaranya:

  optimize : variabel tunggal

  optim : variabel lebih dari satu

Pendugaan Parameter

- Secara umum pendugaan parameter dilakukan melalui pendekatan optimasi
- Pendekatan optimasi: max/min fungsi tujuan (objective function)
- Metode yang sering digunakan dalam statistic adalah Metode Kemungkinan Maksimum (maximum likelihood/ML)
- Metode ini berbasis sebaran (fkp/fmp)
- Apa definisi fungsi Kemungkinan?

Optimasi

- R memiliki banyak fungsi untuk optimisasi dari suatu fungsi, diantaranya:

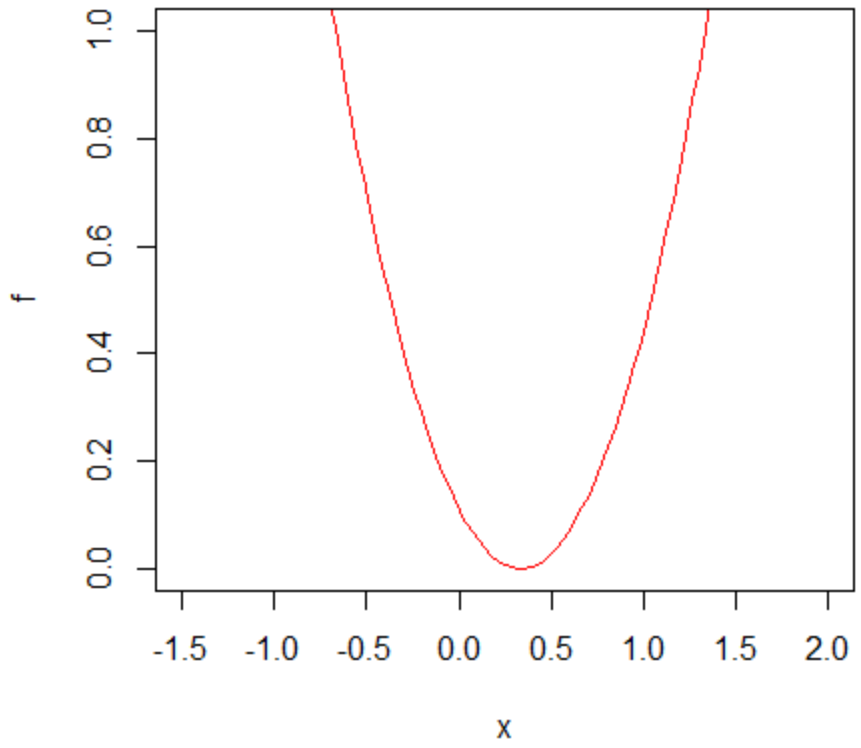
  optimize : variabel tunggal

  optim : variabel lebih dari satu

- Selain paket standar terdapat fungsi mle pada paket stats4 untuk menduga parameter melalui metode kemungkinan maksimum

optimize / optimise

```
> f <- function (x, a) (x - a)^2
> xmin <- optimize(f, c(0, 1), tol = 0.0001, a = 1/3)
> xmin
$minimum
[1] 0.3333333

$objective
[1] 0
```



optim

```
fr <- function(x) {
  x1 <- x[1]
  x2 <- x[2]
  100 * (x2 - x1 * x1)^2 + (1 -
x1)^2
}
optim(c(-1.2,1), fr)
```

```
$par
[1] 1.000260 1.000506

$value
[1] 8.825241e-08

$counts
function gradient
     195       NA

$convergence
[1] 0

$message
NULL
```

Ilustrasi pendugaan ML

```
> x <- rnorm(N, mean = 3, sd = 2)
>
> mean(x)
[1] 2.998305
> sd(x)
[1] 2.288979
```

Dengan Metode ML:

```
> LL <- function(mu, sigma) {
+     R = dnorm(x, mu, sigma)
+     #
+     -sum(log(R))
+ }
> library(stats4)
>
> mle(LL, start = list(mu = 1, sigma=1))

Call:
mle(minuslogl = LL, start = list(mu = 1, sigma = 1))

Coefficients:
      mu    sigma
2.998305 2.277506
Warning messages:
1: In dnorm(x, mu, sigma) : NaNs produced
2: In dnorm(x, mu, sigma) : NaNs produced
3: In dnorm(x, mu, sigma) : NaNs produced
```

Selesai...