KSM PVD + PENGOLAHAN RISET 1

Dasar-dasar Bahasa Pemrograman R

PVD subdivisi Riset 1

APA ITU R?

R merupakan suatu software sekaligus bahasa pemrograman yang dibuat oleh Ross Ihaka dan Robert Gentleman. Bahasa R ini banyak digunakan dalam hal Analisis data.

Pokok Bahasan

BASIC PENGGUNAAN R

STATISTIK DESKRIPTIF

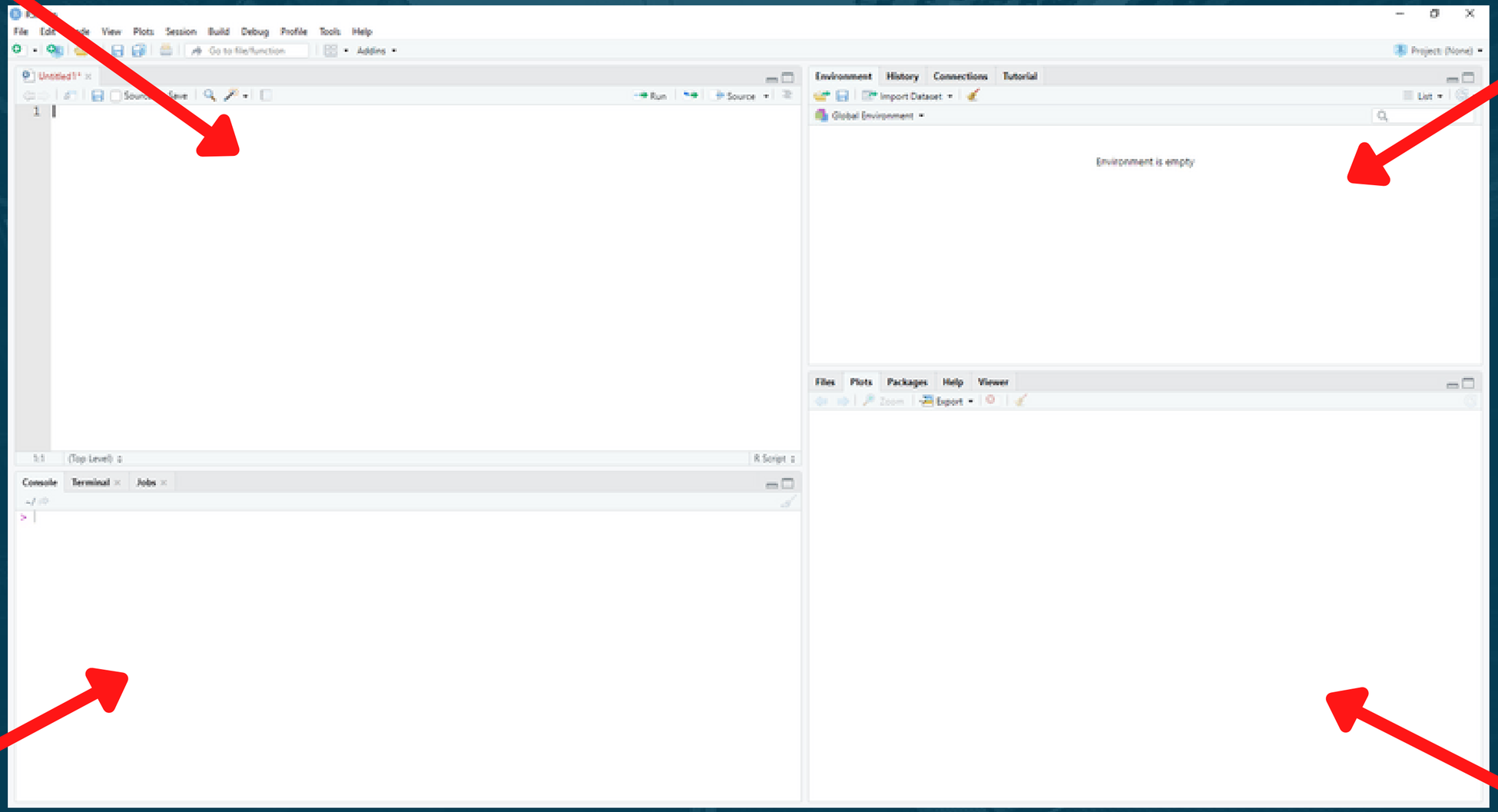
PACKAGE DPLYR

Basic Penggunaan R

Rstudio, Package, Operator, Fungsi-fungsi,
Manajemen Data.

Source

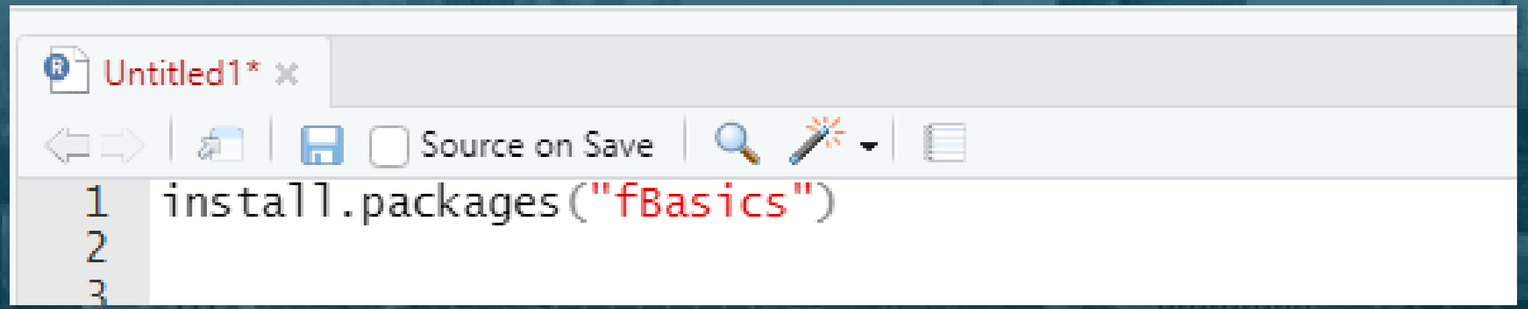
WorkSpace & history



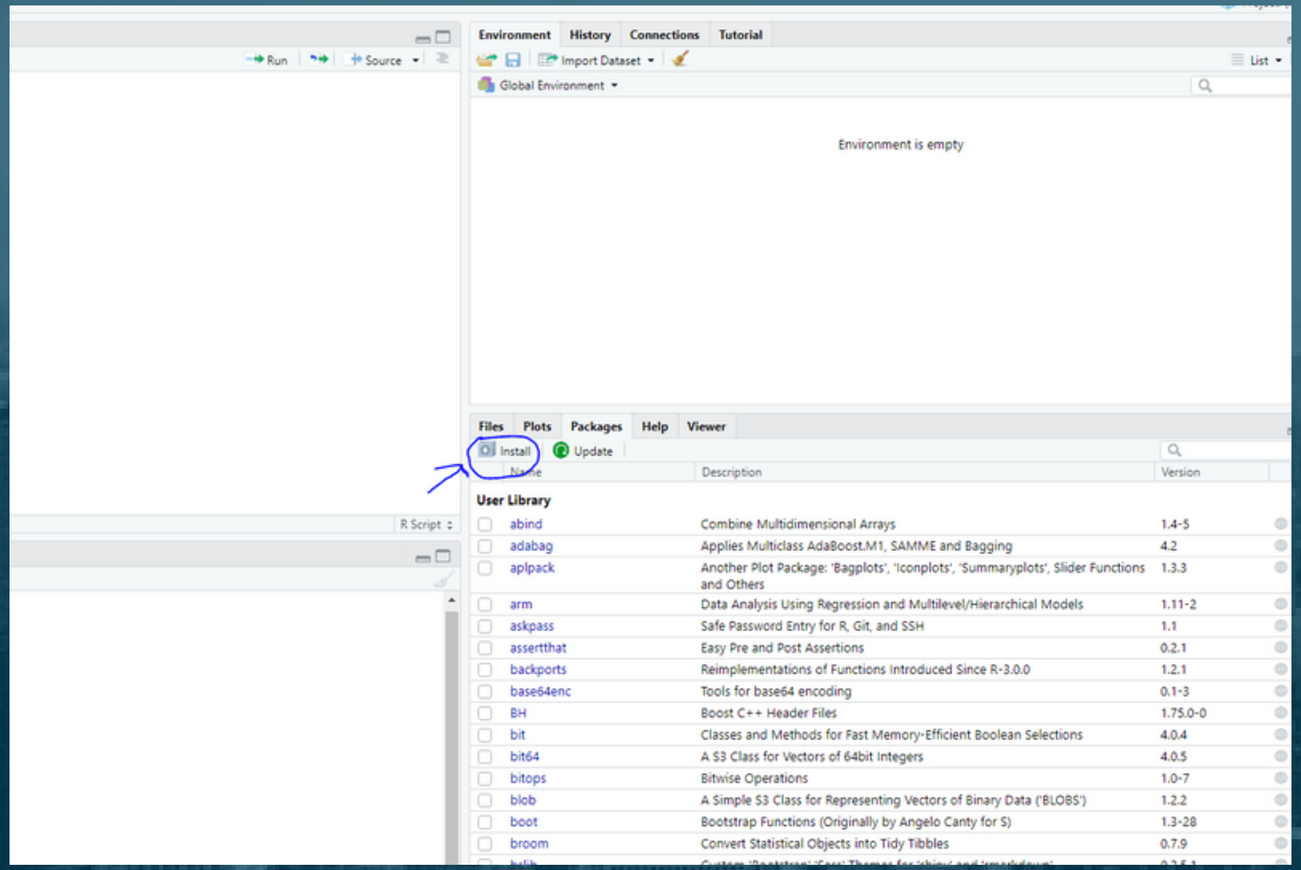
Console

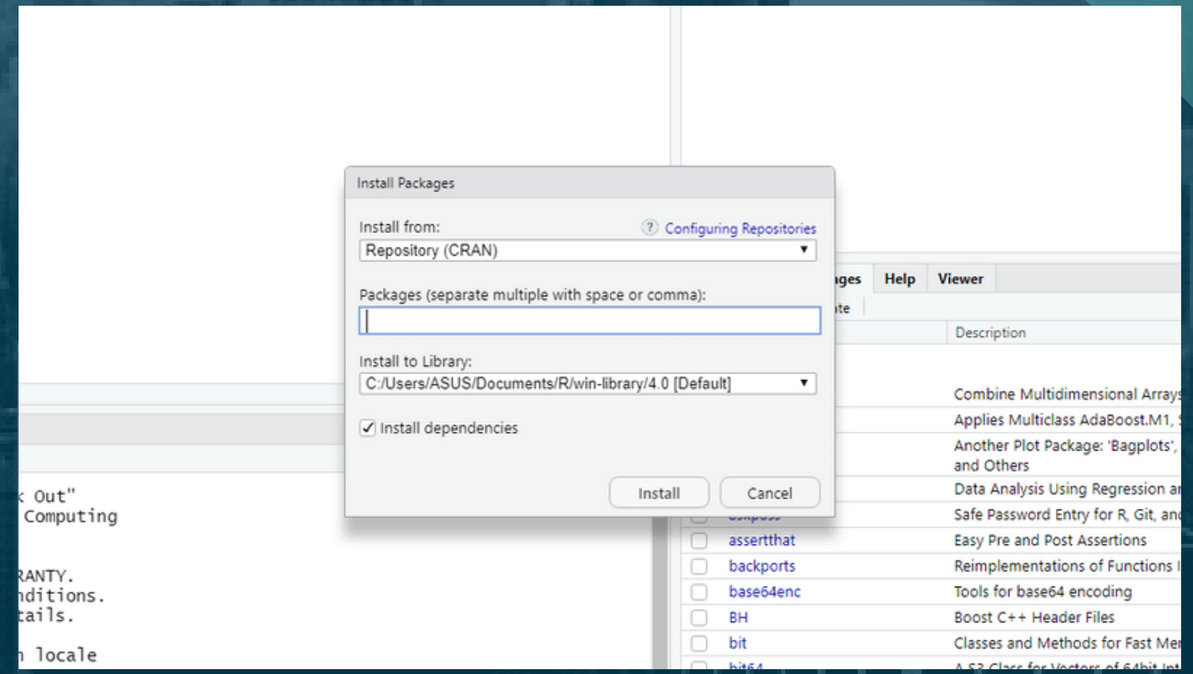
file, plots, package, help, viewer

INSTALL PACKAGE

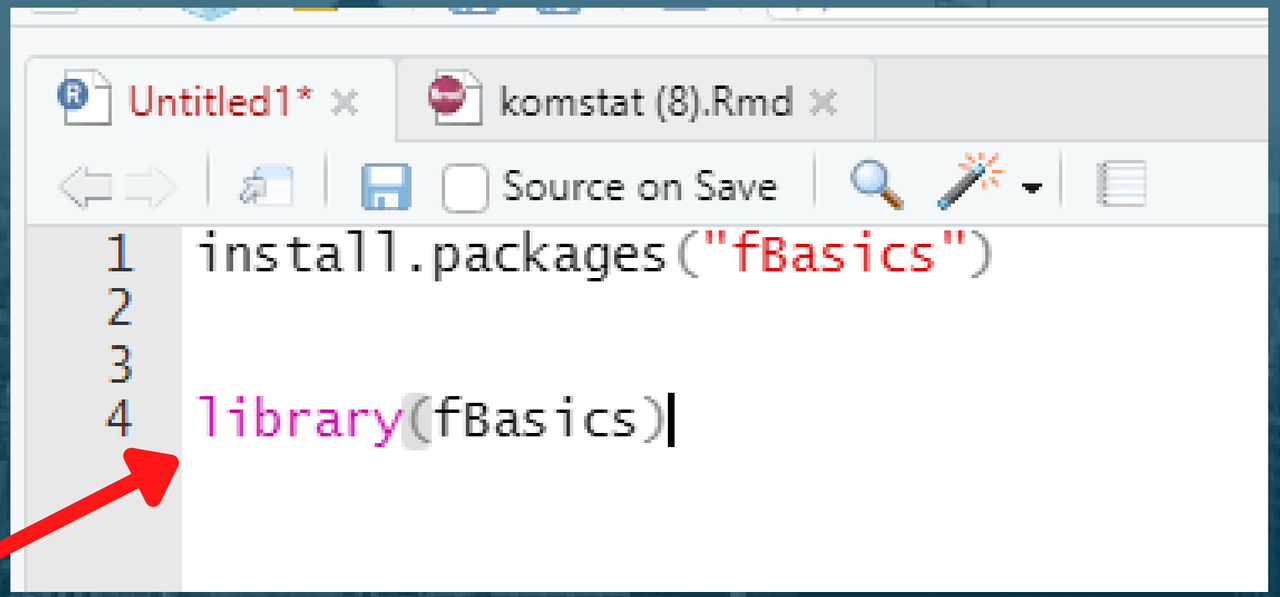


INSTALL PACKAGE





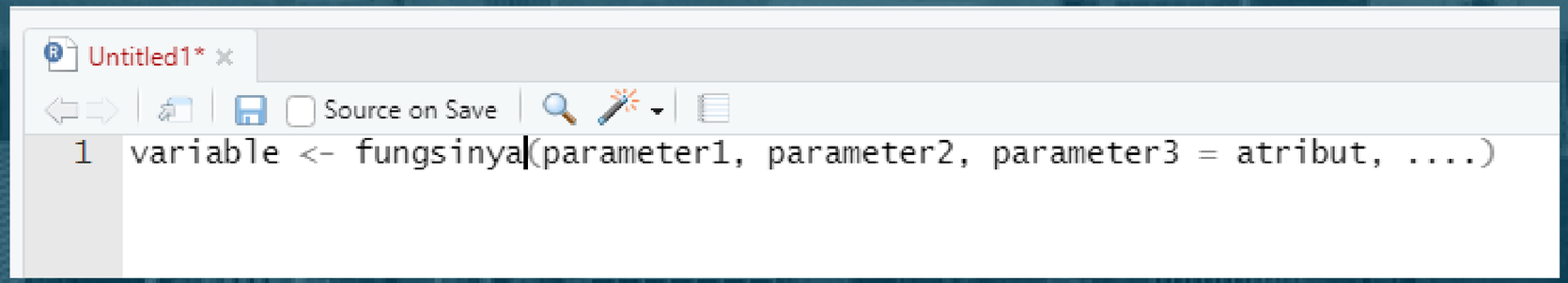
INSTALL PACKAGE



memanggil package

Cobain yuk..

SYNTAX


Untitled1*
Source on Save
1
variable <- fungsinya(parameter1, parameter2, parameter3 = atribut, ...)

FUNGSI-FUNGSI SEDERHANA

NAMA FUNGSI	KEGUNAAN
`sum(x)`	Jumlah dari elemen vektor x
`prod(x)`	Perkalian dari elemen objek x
`max(x)`	Nilai maksimum
`min(x)`	Nilai minimum
`which.min(x)`	Urutan data yang terkecil
`which.max(x)`	Urutan data yang terbesar
`range(x)`	Rentang
`length(x)`	Banyaknya elemen vektor x
`mean(x)`	Rata-rata vektor x
`var(x)`	Variasi dari vektor x
`cor(x,y)`	Korelasi antara x dan y

OPERATOR ARITMATIKA DAN LOGIKA

Operator Aritmatika dalam R menggunakan operator berikut

- Penjumlahan, pengurangan, perkalian, dan pembagian: +, -, *, /
- Pangkat: ^

Sedangkan untuk operator logika, R menggunakan tanda berikut:

- Sama dengan: ==
- Tidak sama dengan: !=
- Lebih besar/kecil: > , <
- Lebih besar/kecil atau sama dengan: >=, <=
- and: &
- or: |

TIPE OBJEK

OBJEK	KEGUNAAN	APAKAH SATU OBJEK BISA MEMILIKI >1 MODE
vector	numeric, character, complex atau logical	Tidak
factor	numeric atau character	Tidak
array	numeric, character, complex atau logical	Tidak
data frame	numeric, character, complex atau logical	Ya
time series	numeric, character, complex atau logical	Tidak
list	numeric, character, complex, logical, function, expression	Ya

VEKTOR

Vektor merupakan suatu himpunan elemen (bilangan, character atau string, logical value) satu dimensi dan merupakan representasi dari suatu variabel.

```
Vektor
```{r}
## vektor ##

v <- 1 : 10 # membuat vektor v dengan elemen 1 s/d 10 #
v

v <- c(1, 5, 3) # (fungsi c() menggabungkan elemen-elemen menjadi vektor)
v

vc1 <- seq(from=1, to=100, by=10) # membuat vektor dengan interval
vc1
```
```

VEKTOR (2)

Vektor juga bisa berisi data logikal (true/false) ataupun berisi karakter (string)

````
Vektor juga bisa berisi data logikal (true/false) ataupun berisi karakter (string).
```{r}
## logical vector

logical_vector <- c(T, TRUE, F, FALSE) # membuat vektor logikal
logical_vector

## vektor string

s <- c("Jakarta", "Malang", "Surabaya")
s
```
````

OPERASI PADA VEKTOR

Fungsi length() digunakan untuk mengetahui panjang suatu vektor atau dengan kata lain banyaknya elemen dalam suatu vektor. Sedangkan sum() digunakan untuk menjumlahkan elemen-elemen di dalam suatu vektor.

````
Operasi pada Vektor
```{r}
vc2 <- seq(from=1, to=100, by=10)
vc2

length(vc2)

sum(vc2)
```
````

```
 [1]  1 11 21 31 41 51 61 71 81 91
[1] 10
[1] 460
```

OPERASI ARITMATIKA PADA VEKTOR

```
Operasi Aritmatika pada vektor
```{r}
x <- 100 : 100
x

x * 2 ## perkalian dengan skalar

x / 2 # pembagian dengan skalar

x - 1

x <- c(10,20,30)
y <- c(2,2,2)
x
y

x*y # perkalian setiap komponen dari vektor

x/y

x %*% y #perkalian vektor
```
```

FAKTOR

Faktor merupakan representasi dari variabel kategorik.
Untuk membuat suatu faktor digunakan fungsi (factor()) dengan opsi sebagai berikut:

- `factor(x, levels = sort(unique(x), na.last = TRUE), labels = levels, exclude = NA, ordered = is.ordered(x))`

x adalah input untuk nilai yang akan dijadikan factor, levels adalah opsi untuk menentukan banyaknya level sedangkan labels untuk memberi nama dari setiap anggota factor tersebut.

```
Faktor
```{r}
factor(1:3) ## default jumlah level akan sama dengan jumlah angka unik

factor(1:3, levels=1:5)

# mendefinisikan level yang ada, dari 5 level,
# yang terdapat pada factor ini hanya 3.

factor(1:3, labels=c('A', 'B', 'C')) # memberikan nama level
```
```

Cobain yuk..

INPUT DATA MANUAL

Untitled1*

Source on Save

```
1  data1<-data.frame(NULL)
2  fix(data1)
3
```

Data Editor

File Edit Help

	var1	var2	var3	var4	var5	var6
1	23					
2	8					
3						
4						
5						
6						
7						
8						
9						
10						
11						
12						
13						
14						
15						
16						
17						
18						
19						

IMPORT DATA

FILE BAWAAN PACKAGE

```
15  library(car)
16  data("cars")
17  head(cars)
```

FILE CSV

```
4  data2<-read.csv(C:\Users\LENOVO\Downloads\archive\Iris.csv)
5  head(data2)
6
```

FILE EXCEL

```
 7  library(readxl)
 8  url<-"D:/Users/LENOVO/Downloads/archive/Iris.xlsx"
 9  dataset<- read_excel(url)
10
```

Cobain yuk..

MENYIMPAN DATA KE DALAM FILE

```
15  ##load data##
16  library(car)
17  data("cars")
18  head(cars)
19  ##menyimpan dalam bentuk file csv##
20  write.csv(cars, file="Cars.csv")
```

MISSING VALUE

```
> ##set x sebagai missing##
> x<-NA
> x+4
[1] NA
> log(x)
[1] NA
> is.na(x)
[1] TRUE
> ##Missing dalam sebuah vector##
> x<-c(1,2,NA,4)
> x
[1]  1  2 NA  4
> ##Mengecek missing values##
> is.na(x)
[1] FALSE FALSE  TRUE FALSE
> ##menghitung jumlah missing value##
> sum(is.na(x))
[1] 1
```

MISSING VALUE(1)

```
> ##missing dalam matrix##
> ##membuat matrix##
> mat1<-matrix(c(1:19, NA),4,5)
> mat1
     [,1] [,2] [,3] [,4] [,5]
[1,]    1    5    9   13   17
[2,]    2    6   10   14   18
[3,]    3    7   11   15   19
[4,]    4    8   12   16   NA
> is.na(mat1)
      [,1]  [,2]  [,3]  [,4]  [,5]
[1,] FALSE FALSE FALSE FALSE FALSE
[2,] FALSE FALSE FALSE FALSE FALSE
[3,] FALSE FALSE FALSE FALSE FALSE
[4,] FALSE FALSE FALSE FALSE  TRUE
> ##complete case memberikan kondisi(TRUE/FALSE)
> ##dari baris yang terdapat missing value
> complete.cases(mat1)
[1]  TRUE  TRUE  TRUE FALSE
> ##memilih baris yang tidak terdapat missing value##
> mat1[complete.cases(mat1),]
     [,1] [,2] [,3] [,4] [,5]
[1,]    1    5    9   13   17
[2,]    2    6   10   14   18
[3,]    3    7   11   15   19
```

Statistik Deskriptif

Ukuran Pemusatan, Visualisasi Data

Bijou Solutions, Inc. | 2020

UKURAN PEMUSATAN DATA

```
> dim(state.x77)#mengecek dimensi data
[1] 50  8
> state.x77<-data.frame(state.x77)#ubah jadi data frame untuk mempermudah
> #ukuran pemusatan untuk variabel income
> mean(state.x77$Income)#menghirung rata-rata
[1] 4435.8
> median(state.x77$Income)
[1] 4519
> ##kuantil (Q1, Q2(median),Q3)
> quantile(state.x77$Income, c(0.25,0.50,0.75))
    25%     50%     75%
3992.75 4519.00 4813.50
> ##desil ke 2
> quantile(state.x77$Income, c(0.20))
   20%
3864.2
> ##persentil ke 99
> quantile(state.x77$Income, c(0.99))
    99%
5841.17
```

UKURAN PENYEBARAN

```
> ##min max
> min(state.x77$Income)
[1] 3098
> max(state.x77$Income)
[1] 6315
> which.max(state.x77$Income)#observasi income terbesar
[1] 2
> ##ukuran penyebaran
> range(state.x77$Income)
[1] 3098 6315
> sd(state.x77$Income)
[1] 614.4699
> var(state.x77$Income)
[1] 377573.3
```

MENAMPILKAN SEMUA STATISTIK

```
> ##menampilkan semua statistik
> summary(state.x77)
   Population        Income       Illiteracy      Life.Exp         Murder
 Min.   :  365   Min.   :3098   Min.   :0.500   Min.   :67.96   Min.   : 1.400
 1st Qu.: 1080   1st Qu.:3993   1st Qu.:0.625   1st Qu.:70.12   1st Qu.: 4.350
 Median : 2838   Median :4519   Median :0.950   Median :70.67   Median : 6.850
 Mean   : 4246   Mean   :4436   Mean   :1.170   Mean   :70.88   Mean   : 7.378
 3rd Qu.: 4968   3rd Qu.:4814   3rd Qu.:1.575   3rd Qu.:71.89   3rd Qu.:10.675
 Max.   :21198   Max.   :6315   Max.   :2.800   Max.   :73.60   Max.   :15.100
    HS.Grad          Frost             Area
 Min.   :37.80   Min.   :  0.00   Min.   :  1049
 1st Qu.:48.05   1st Qu.: 66.25   1st Qu.: 36985
 Median :53.25   Median :114.50   Median : 54277
 Mean   :53.11   Mean   :104.46   Mean   : 70736
 3rd Qu.:59.15   3rd Qu.:139.75   3rd Qu.: 81163
 Max.   :67.30   Max.   :188.00   Max.   :566432
```

STATISTIK DESKRIPTIF DATA KATEGORIK

```
> ##Statistik deskriptif data kategorik
> state.x77<-data.frame(state.x77)#ubah ke data frame
> ##membuat 3 variabel kategorik baru
> state.x77$grpPop<-cut(state.x77$Population, breaks = c(0,3000,10000,Inf), labels = c("low",
 "medium","high"))
> state.x77$grpInc<-ifelse(state.x77$Income<=median(state.x77$Income), "low","high")
> state.x77$grpFrost<-cut(state.x77$Frost, breaks = quantile(state.x77$Frost, c(0, 0.25, 0.5,
 0.75,1)),labels = c("hot","warm","chill","cold"))
> head(state.x77)
           Population Income Illiteracy Life.Exp Murder HS.Grad Frost   Area grpPop grpInc
Alabama          3615   3624        2.1    69.05   15.1    41.3    20  50708 medium    low
Alaska            365   6315        1.5    69.31   11.3    66.7   152 566432    low   high
Arizona          2212   4530        1.8    70.55    7.8    58.1    15 113417    low   high
Arkansas         2110   3378        1.9    70.66   10.1    39.9    65  51945    low    low
California      21198   5114        1.1    71.71   10.3    62.6    20 156361   high   high
Colorado         2541   4884        0.7    72.06    6.8    63.9   166 103766    low   high
           grpFrost
Alabama         hot
Alaska         cold
Arizona         hot
Arkansas        hot
California      hot
Colorado       cold
```

STATISTIK DESKRIPTIF DATA KATEGORIK(2)

```
> ##tabel frekuensi##
> table(state.x77$grpPop)#1-way frequency table

   low medium   high
    26     18      6
> tab1<-table(state.x77$grpFrost,state.x77$grpInc)#2-way frequency table
> tab1

       high low
  hot     5   7
  warm    6   6
  chill   6   6
  cold    7   6
> tab2<-table(state.x77$grpFrost,state.x77$grpPop, state.x77$grpInc)#3-way frequency table
> tab2
, ,  = high


       low medium high
  hot    2      2    1
  warm   2      3    1
  chill  1      3    2
  cold   6      1    0

, ,  = low


       low medium high
  hot    3      3    1
  warm   2      4    0
  chill  4      1    1
  cold   5      1    0
```

STATISTIK DESKRIPTIF DATA KATEGORIK(3)

```
> margin.table(tab1,1)#jumlah per baris

  hot  warm chill  cold
   12    12    12    13
> margin.table(tab1,2)#jumlah per kolom

high  low
  24   25
> prop.table(tab1)#presentase total

             high       low
  hot   0.1020408 0.1428571
  warm  0.1224490 0.1224490
  chill 0.1224490 0.1224490
  cold  0.1428571 0.1224490
> prop.table(tab1, 1)#presentase per baris

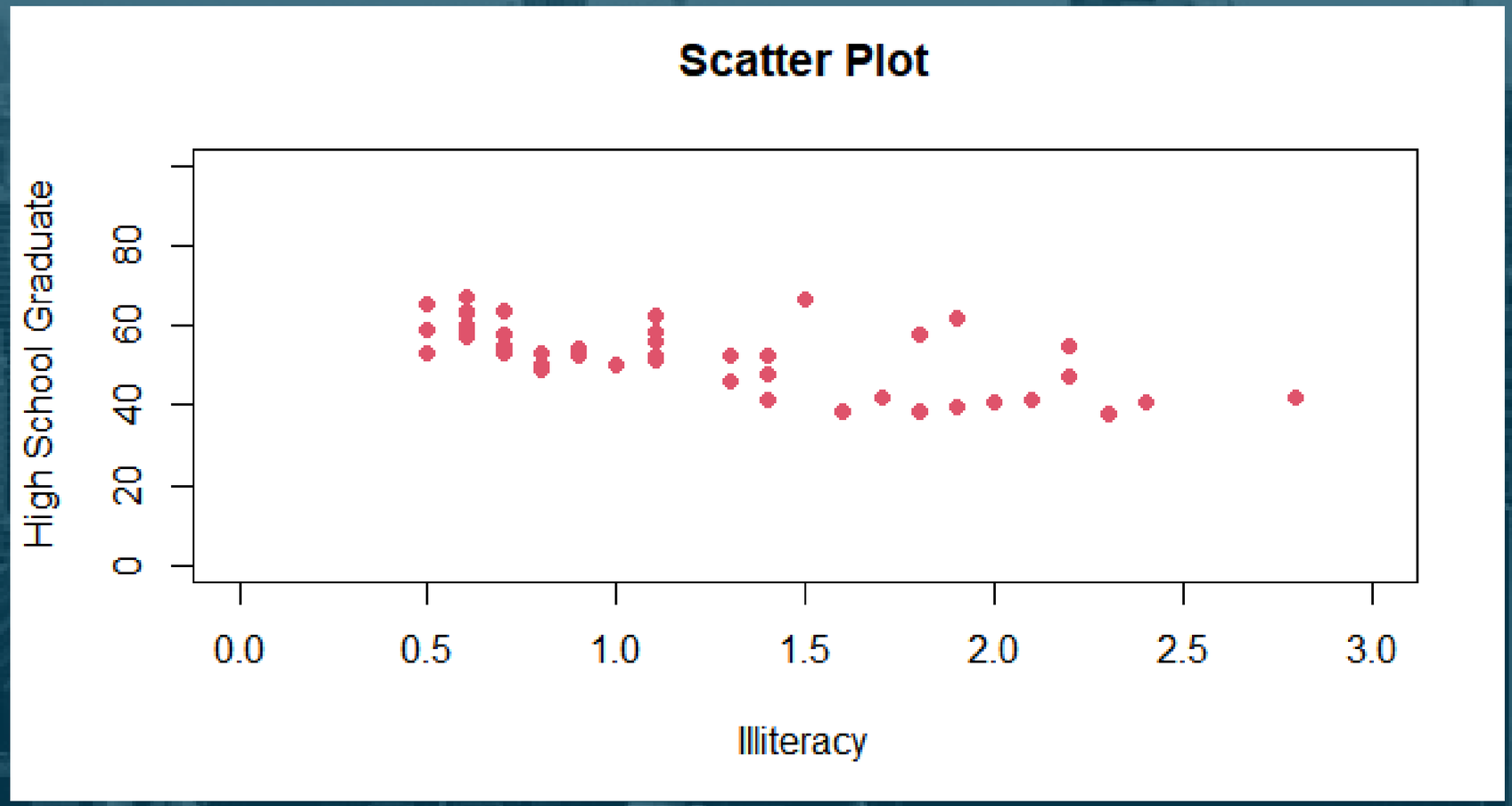
             high       low
  hot   0.4166667 0.5833333
  warm  0.5000000 0.5000000
  chill 0.5000000 0.5000000
  cold  0.5384615 0.4615385
> prop.table(tab1, 2)#presentase per kolom

             high       low
  hot   0.2083333 0.2800000
  warm  0.2500000 0.2400000
  chill 0.2500000 0.2400000
  cold  0.2916667 0.2400000
```

Cobain yuk..

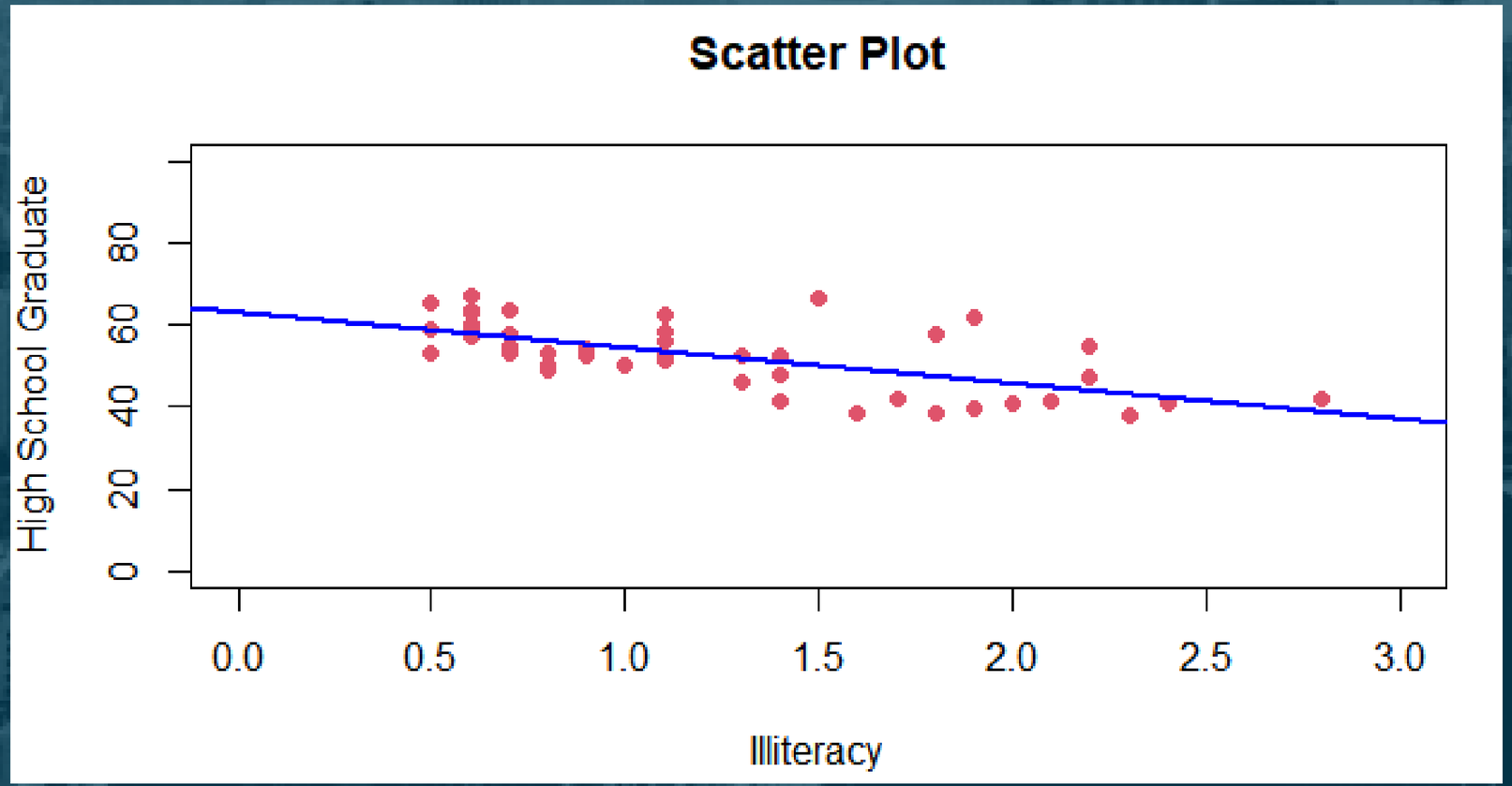
VISUALISASI DATA:SCATTER PLOT

```
45  ##scatterplot dengan mendifinisikan nilai pada sumbu x
46  ## dan sumbu y(xlim dan ylim)
47  plot(state.x77[,3],state.x77[,6], xlab = "Illiteracy", ylab = "High School
48       Graduate", col=2, pch=16, main = "Scatter Plot", xlim = c(0,3),
49       ylim = c(0,100))
50
```



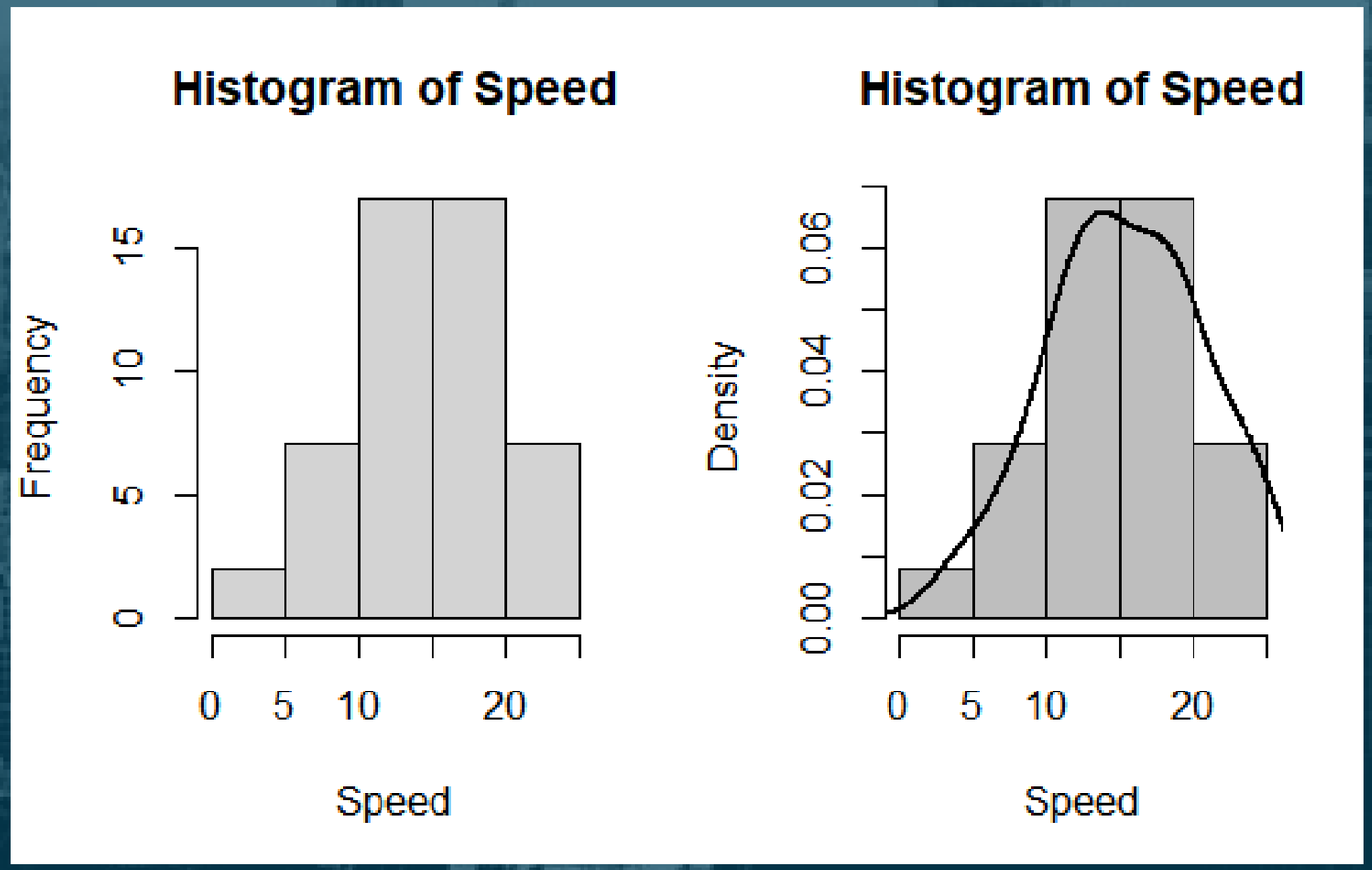
VISUALISASI DATA: MENAMBAH GARIS REGRESI

```
45  ##scatterplot dengan mendifinisikan nilai pada sumbu x
46  ## dan sumbu y(xlim dan ylim)
47  plot(state.x77[,3],state.x77[,6], xlab = "Illiteracy", ylab = "High School
48       Graduate", col=2, pch=16, main = "Scatter Plot", xlim = c(0,3),
49       ylim = c(0,100))
50  ##menambah garis regresi
51  abline(lm(state.x77[,6]~state.x77[,3]), lwd=2, col="blue")
```



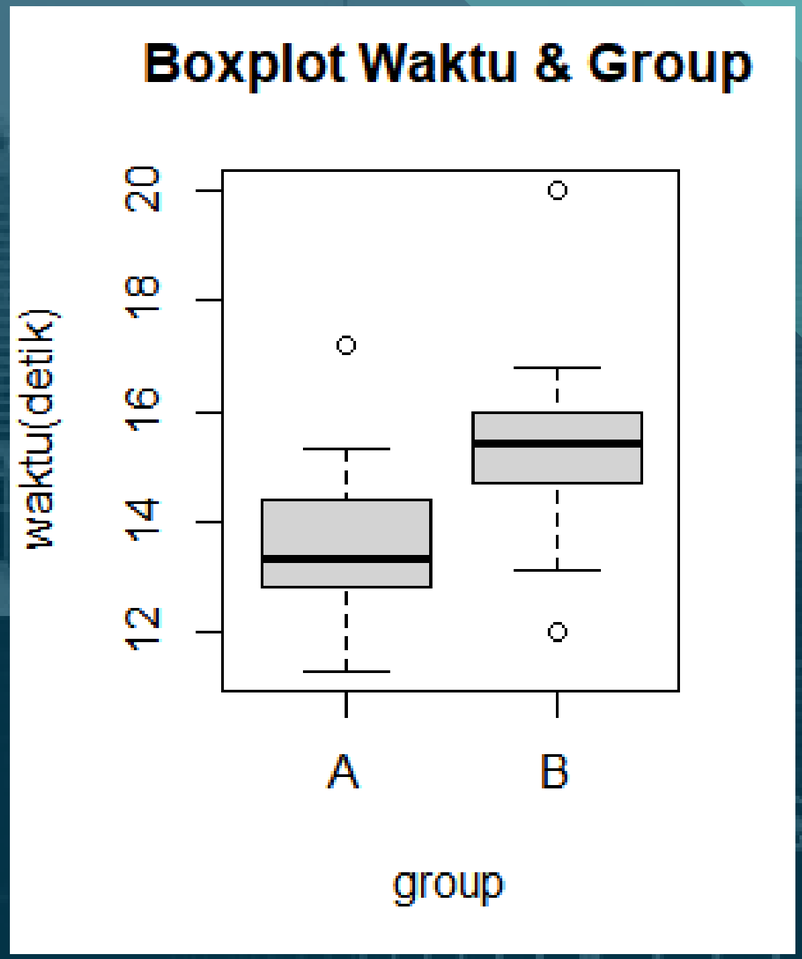
VISUALISASI DATA: HISTOGRAM

```
par(mfrow=c(1,2))#set gambar menjadi 2 kolom
##Histogram frekuensi
hist(cars$speed, xlab="Speed", main="Histogram of Speed")
##Histogram dengan density plot
hist(cars$speed, prob=TRUE, xlab="Speed", main="Histogram of Speed", col="grey")
lines(density(cars$speed), lwd=2)#menggambar garis density
```



VISUALISASI DATA: BOXPLOT

```r
##boxplot
##input data
A<-c(12.9, 13.5, 12.8, 13.6, 17.2, 13.2, 12.6, 15.3, 14.4, 11.3)
B<-c(14.7, 15.6, 15.0, 15.2, 16.8, 20.0, 12.0, 15.9, 16.0, 13.1)
##digabung dalam matrix
dt<-cbind(A,B)
boxplot(dt, xlab="group", ylab="waktu(detik)", main="Boxplot Waktu & Group")
```



VISUALISASI DATA:STEM & LEAF PLOT

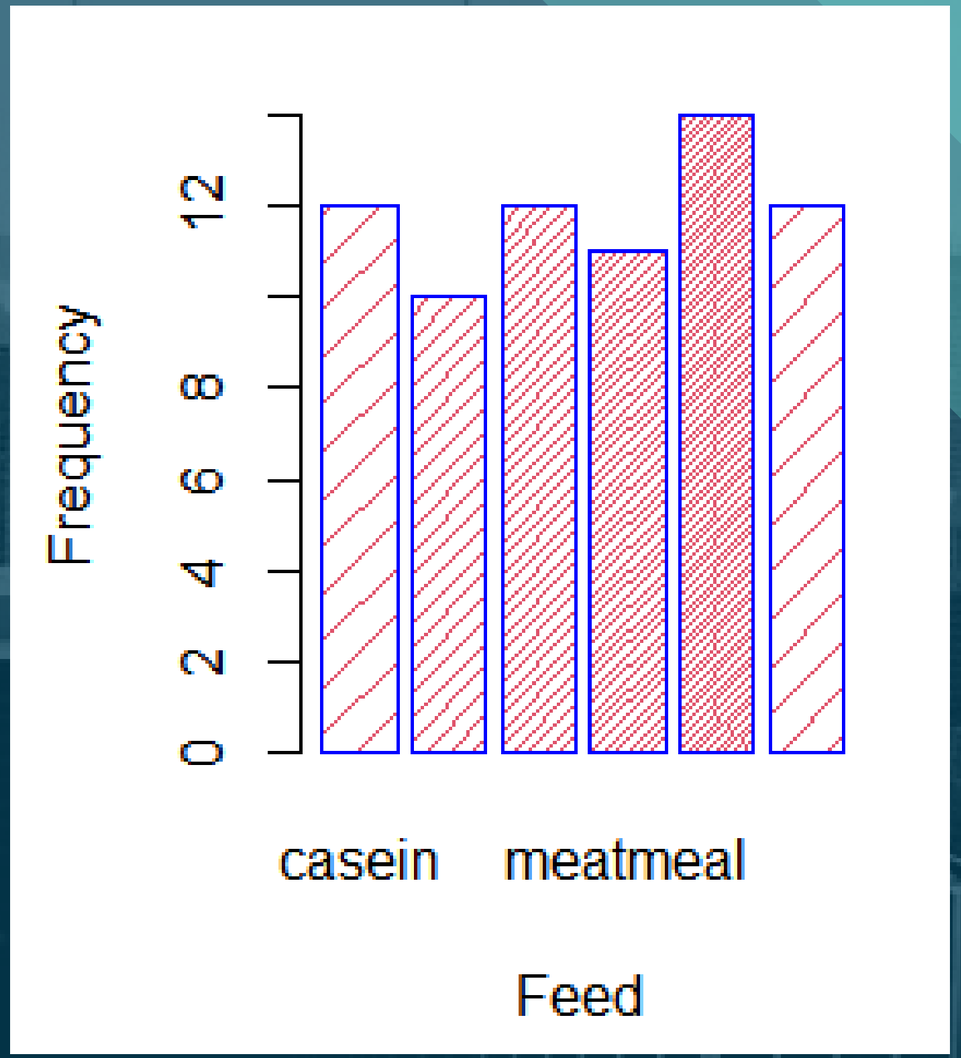
```
##stem leaf plot
stem(cars$speed)
```

```
The decimal point is at the |

  4 | 00
  6 | 00
  8 | 00
 10 | 00000
 12 | 00000000
 14 | 0000000
 16 | 00000
 18 | 0000000
 20 | 00000
 22 | 00
 24 | 00000
```

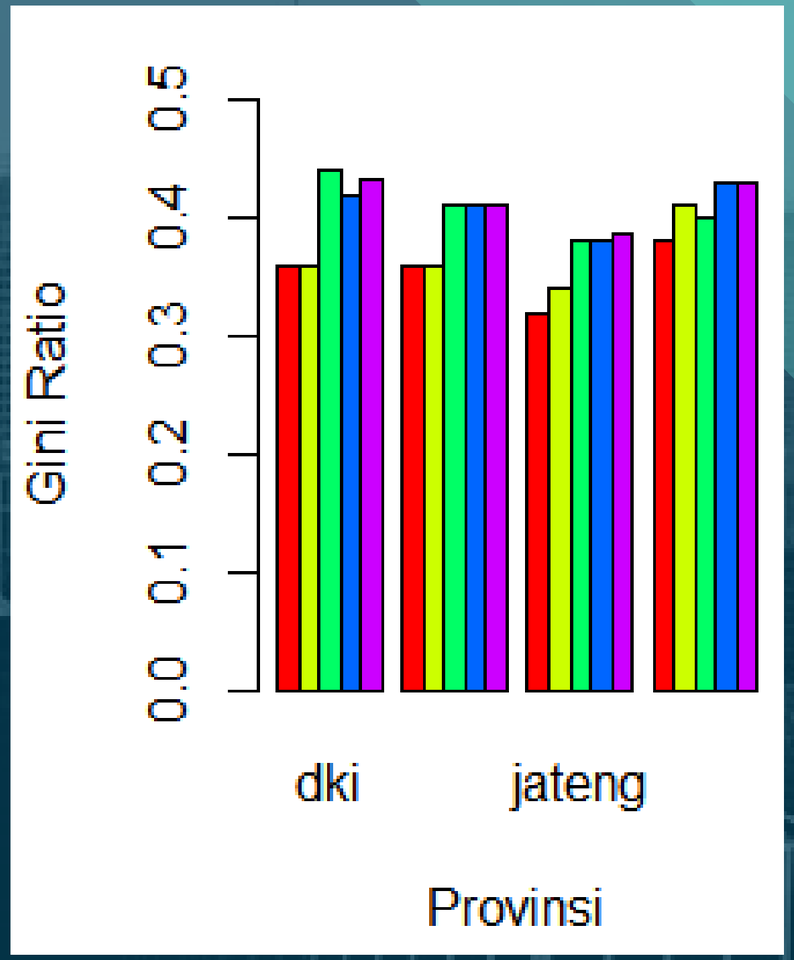
VISUALISASI DATA: BARCHART

```r
##bar chart
barplot(table(chickwts$feed), xlab = "Feed", ylab = "Frequency",
        density = c(10,20,30,40,50), border = "blue", col = 2)
```



VISUALISASI DATA: BARCHART(1)

```r
##Input Data
tahun<-c(2009, 2010, 2011, 2012, 2013)
dki<-c(0.36, 0.36, 0.44, 0.42,0.433)
jabar<-c(0.36,0.36,0.41,0.41,0.411)
jateng<-c(0.32,0.34,0.38,0.38,0.387)
yogya<-c(0.38,0.41,0.40,0.43,0.43)
##Menggabungkan variabel
dt1<-cbind(dki, jabar, jateng , yogya)
##membuat barchart
barplot(dt1, xlab="Provinsi", col=rainbow(5), ylab="Gini Ratio", beside=TRUE,
        ylim=c(0,0.5))
legend("topleft",c("2009","2010","2011","2012","2013"), cex = 0.7, bty = "n",
       fill = rainbow(5))
```



Cobain yuk..

Package DPLYR

select, filter, group_by, summarise, arrange, join, mutate

```
select(flights, year,month,day, arr_delay, dep_delay)

filter(flights, dep_delay>1000)

arrange(flights, desc(dep_delay))

summarise(flights, mean_dep_delay= mean(dep_delay, na.rm = T))

new_flight <- mutate(flights, air_time_hours = air_time/60)


summarise(group_by(flights, month), mean_dep_delay = mean(dep_delay, na.rm = T))
```

Cobain yuk..