TUGAS 4

MATA KULIAH MANAJEMEN DAN ANALISIS DATA DENGAN R

Isni Nurul Aini

NPM. 131520220003

PROGRAM STUDI MAGISTER EPIDEMIOLOGI

FAKULTAS KEDOKTERAN

UNIVERSITAS PADJADJARAN

2023

```
library(readr)
library(readxl)
library(data.table)
library(writexl)
library(ggplot2)
library(lme4)
library(tidyverse)
```

#1. Mengakses dan mendownload dataset ke dalam *global environment* RStudio

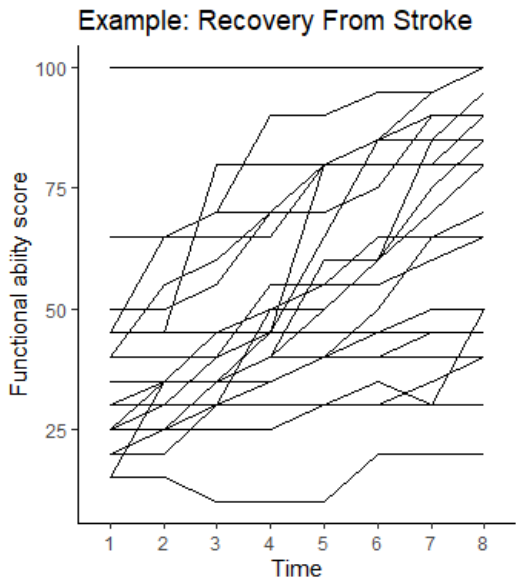
```
stroke <- read_excel("C:/Users/HP/Downloads/stroke.xls")
names(stroke)
library(tidyverse)
#wide to long
stroke_long = stroke %>% select(c(1:6,39:46)) %>%
  pivot_longer(cols=Bart1:Bart8,
               names_to = "time",
               names_prefix = "Bart",
               values_to = "ability")
names(stroke_long)
glimpse(stroke_long)
```

#2.Membuat visualisasi grafik garis dari perkembangan nilai kemampuan motorik (*functional ability score*) dari setiap subyek menggunakan *variable bart*

#A. Grafik keseluruhan

```
ggplot(stroke_long, aes(x = time,
                        y = ability)) +
  geom_line(aes(group = Subject)) +
  theme_classic() + labs( y="Functional abiity score",
```

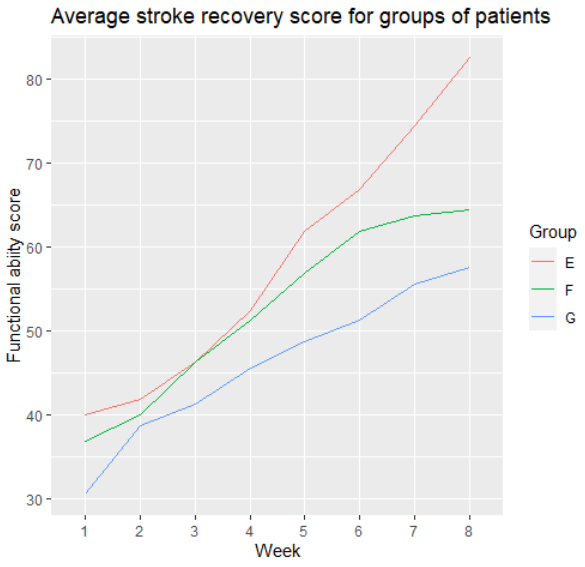
```
x="Time", title="Example: Recovery From Stroke")
```



#3. Membuat grafik nilai rata-rata perkembangan fungsi motorik secara total dan masing-masing yang divisualisasikan pada 1 grafik

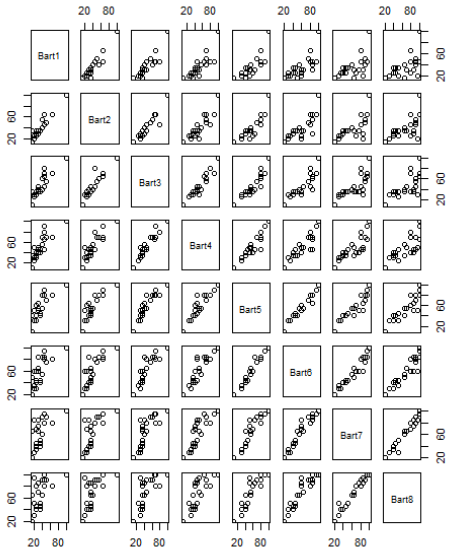
```
stroke_av <- stroke_long %>%
  group_by(Group, time) %>%
  mutate(Average = mean(ability)) %>%
  as.data.frame()

ggplot(stroke_av, aes(x = time, y = Average)) +
  geom_line(aes(group = Group)) +
  theme_classic()+ labs( y="Functional abiity score",
                         x="Week", title="Average stroke recovery score for groups of patients")

stroke_av %>%
  mutate(label = if_else(time == max(time), as.character(Group), NA_character_)) %>%
```

ggplot(aes(x = time, y = Average, group = Group, colour = Group)) + geom_line() + labs(y="Functional abiity score", x="Week", title="Average stroke recovery score for groups of patients")



#4. Membuat Matrix Scatter plot dari nilai fungsi motorik antar waktu/pekan

pairs(~Bart1 + Bart2 + Bart3 + Bart4 + Bart5 + Bart6 + Bart7 + Bart8, data = stroke)



#5&6 Menghitung dan membuat tabel silang koefisien korelasi nilai fungsi motorik antar waktu/pekan, dan menginterpretasikan Matrix Scatter Plot

```
cor(stroke$Bart1, stroke$Bart2)
cor(stroke$Bart1, stroke$Bart3)
cor(stroke$Bart1, stroke$Bart4)
cor(stroke$Bart1, stroke$Bart5)
cor(stroke$Bart1, stroke$Bart6)
cor(stroke$Bart1, stroke$Bart7)
cor(stroke$Bart1, stroke$Bart8)
cor(stroke$Bart2, stroke$Bart3)
cor(stroke$Bart2, stroke$Bart4)
cor(stroke$Bart2, stroke$Bart5)
cor(stroke$Bart2, stroke$Bart6)
cor(stroke$Bart2, stroke$Bart7)
cor(stroke$Bart2, stroke$Bart8)
cor(stroke$Bart3, stroke$Bart4)
cor(stroke$Bart3, stroke$Bart5)
cor(stroke$Bart3, stroke$Bart6)
cor(stroke$Bart3, stroke$Bart7)
cor(stroke$Bart3, stroke$Bart8)
cor(stroke$Bart4, stroke$Bart5)
cor(stroke$Bart4, stroke$Bart6)
cor(stroke$Bart4, stroke$Bart7)
cor(stroke$Bart4, stroke$Bart8)
cor(stroke$Bart5, stroke$Bart6)
cor(stroke$Bart5, stroke$Bart7)
cor(stroke$Bart5, stroke$Bart8)
cor(stroke$Bart6, stroke$Bart7)
cor(stroke$Bart6, stroke$Bart8)
```

```
cor(stroke$Bart7, stroke$Bart8)
```

Correlations coefficient for the stroke recovery scores

	Week						
	1	**2**	**3**	**4**	**5**	**6**	**7**
2	0.928						
3	0.882	0.922					
4	0.830	0.877	0.953				
5	0.793	0.846	0.909	0.921			
6	0.712	0.789	0.854	0.878	0.973		
7	0.616	0.704	0.766	0.831	0.914	0.956	
8	0.554	0.642	0.700	0.771	0.88	0.926	0.977

#7. Menghitung intercept dan slope, beserta standar errornya masing-masing, dari hubungan fungsi motorik dengan waktu/pekan setiap subyek, serta mempresentasikan hasilnya dalam bentuk tabel

```
library(lme4)

stroke_av$time = as.numeric(stroke_av$time)

model <- (lmList(ability ~ time | Subject, data = stroke_av))

summary(model)$coef

intercepts <- sapply(model,coef)[1,]

slope <- sapply(model,coef)[2,]

intercepts

slope
```

Subject	Intercept	(std. error)	Slope	(std. error)
1	38.5714286	4.03722	7.2619048	0.7994887
2	61.9642857	4.03722	2.6190476	0.7994887
3	14.4642857	4.03722	9.7023810	0.7994887
4	26.0714286	4.03722	2.6785714	0.7994887
5	48.7500000	4.03722	5.0000000	0.7994887
6	10.1785714	4.03722	1.0714286	0.7994887
7	31.2500000	4.03722	2.5000000	0.7994887
8	34.1071429	4.03722	3.8095238	0.7994887
A	21.0714286	4.03722	1.4285714	0.7994887
B	34.1071429	4.03722	0.8928571	0.7994887
C	32.1428571	4.03722	1.6071429	0.7994887
D	42.3214286	4.03722	7.2619048	0.7994887

E	48.5714286	4.03722	7.2619048	0.7994887
F	24.8214286	4.03722	2.2619048	0.7994887
G	22.3214286	4.03722	1.8452381	0.7994887
H	13.0357143	4.03722	6.5476190	0.7994887
I	30.0000000	4.03722	7.5000000	0.7994887
II	15.5357143	4.03722	3.2142857	0.7994887
III	39.8214286	4.03722	6.4285714	0.7994887
IV	11.6071429	4.03722	8.3928571	0.7994887
V	100.0000000	4.03722	0.0000000	0.7994887
VI	0.8928571	4.03722	11.1904762	0.7994887
VII	15.3571429	4.03722	7.9761905	0.7994887
VIII	25.3571429	4.03722	5.8928571	0.7994887