

# **Praktikum Statistika 2FA**



**Disusun Oleh:**

**Wahyu Ikbal Maulana (3323600056)**

**POLITEKNIK ELEKTRONIKA NEGERI  
SURABAYA**

### #Analisis Faktor Kasus-1

#### #1. Membaca Dataset dan menampilkan

```
data = read.csv("Exh_mvar.csv", header = TRUE)
```

#### #2. Evaluasi Data

##### #Deskripsi Statistik

```
summary(data)
```

```
##           Pop           School           Employ           Health
## Min.      :1.523   Min.      :12.20   Min.      :0.597   Min.      :0.750
## 1st Qu.:2.767   1st Qu.:13.03   1st Qu.:1.267   1st Qu.:1.050
## Median :3.893   Median :13.70   Median :1.884   Median :2.045
## Mean    :4.323   Mean    :14.01   Mean    :1.952   Mean    :2.171
## 3rd Qu.:6.322   3rd Qu.:14.85   3rd Qu.:2.542   3rd Qu.:2.473
## Max.    :8.044   Max.    :17.00   Max.    :3.641   Max.    :5.520
##           Home
## Min.      :1.720
## 1st Qu.:1.880
## Median :2.290
## Mean    :2.454
## 3rd Qu.:2.842
## Max.    :4.250
```

Pada kode di atas menampilkan dataset Exh\_mvar lalu menampilkan summary statistik untuk tiap 5 indikator

### #Evaluasi Matriks Korelasi

```
library(RcmdrMisc)
```

```
## Warning: package 'RcmdrMisc' was built under R version 4.3.2
```

```
## Loading required package: car
```

```
## Warning: package 'car' was built under R version 4.3.2
```

```
## Loading required package: carData
```

```
## Warning: package 'carData' was built under R version 4.3.2
```

```
## Loading required package: sandwich
```

```
## Warning: package 'sandwich' was built under R version 4.3.2
```

```
rcorr.adjust(data) # Fungsi ini dibangun di R Commander.
```

```
##
## Pearson correlations:
##           Pop School   Employ   Health   Home
## Pop      1.0000 0.6102  0.9707  0.7400 -0.1720
## School   0.6102 1.0000  0.4943  0.0954  0.1859
## Employ   0.9707 0.4943  1.0000  0.8480 -0.2492
## Health   0.7400 0.0954  0.8480  1.0000 -0.3580
## Home    -0.1720 0.1859 -0.2492 -0.3580  1.0000
```

```
##
## Number of observations: 14
##
## Pairwise two-sided p-values:
##      Pop      School Employ Health Home
## Pop      0.0205 <.0001 0.0025 0.5566
## School 0.0205      0.0724 0.7456 0.5245
## Employ <.0001 0.0724      0.0001 0.3903
## Health 0.0025 0.7456 0.0001      0.2088
## Home   0.5566 0.5245 0.3903 0.2088
##
## Adjusted p-values (Holm's method)
##      Pop      School Employ Health Home
## Pop      0.1434 <.0001 0.0198 1.0000
## School 0.1434      0.4343 1.0000 1.0000
## Employ <.0001 0.4343      0.0011 1.0000
## Health 0.0198 1.0000 0.0011      1.0000
## Home   1.0000 1.0000 1.0000 1.0000
```

Output di atas adalah contoh hasil evaluasi matriks korelasi menggunakan fungsi `rcorr.adjust()`. Fungsi ini memungkinkan kita untuk menghitung matriks korelasi Pearson dan mendapatkan p-value yang terkait dengan setiap koefisien korelasi. Ada 2 penjelasan terkait korelasi di atas yakni untuk korelasi Pearson dan p-values

```
write.csv(cor(data)>0.8,file="Suspect_Correlations.csv")
write.csv(cor(data),file="Correlation_Values.csv")
#Uji KMO untuk kecukupan sampel,KMO > 0.5 maka sampel cukup
library(psych)

## Warning: package 'psych' was built under R version 4.3.2

##
## Attaching package: 'psych'

## The following object is masked from 'package:RcmdrMisc':
##
##      reliability

## The following object is masked from 'package:car':
##
##      logit

KMO(data)

## Kaiser-Meyer-Olkin factor adequacy
## Call: KMO(r = data)
## Overall MSA = 0.62
## MSA for each item =
##      Pop School Employ Health   Home
##      0.64   0.64   0.59   0.61   0.79
```

Kode di atas menggunakan metode untuk mengukur kecukupan sampel yaitu dengan menggunakan Uji Kebergunaan Sampel atau KMO (Kaiser-Meyer-Olkin). Nilai KMO yang lebih besar dari 0,5 menunjukkan bahwa sampel yang digunakan cukup untuk analisis faktor.

```
round( KMO(data)$MSA, 2 )  
## [1] 0.62
```

Output di atas adalah hasil pembulatan dari KMO data

```
#Uji Bartlett untuk menguji ada tidaknya korelasi antarvariabel  
library(psych)  
cortest.bartlett(data)  
  
## R was not square, finding R from data  
  
## $chisq  
## [1] 62.5523  
##  
## $p.value  
## [1] 1.188108e-09  
##  
## $df  
## [1] 10
```

Uji Bartlett digunakan untuk menguji apakah ada korelasi antara variabel-variabel dalam sebuah dataset. Uji Bartlett membandingkan matriks kovarians antar variabel. Jika varians antar kelompok data tidak homogen, maka uji Bartlett akan menunjukkan adanya korelasi antarvariabel. Hasil uji Bartlett biasanya menghasilkan statistik uji chi-square, nilai p, dan derajat kebebasan (df). Hasil uji Bartlett menunjukkan statistik chi-square yang signifikan, nilai p yang rendah

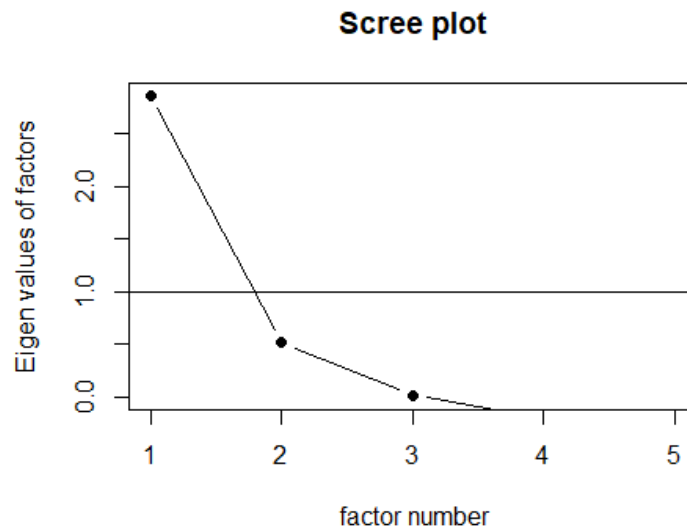
```
#2. Menentukan jumlah faktor yang diekstraksi  
ev <- eigen(cor(data)) # mendapatkan nilai eigen  
ev$values  
  
## [1] 3.02889606 1.29113796 0.57245566 0.09539848 0.01211184
```

Kode R di atas mencari nilai eigen dari korelasi data. Nilai eigen dipakai untuk menentukan jumlah faktor yang diekstraksi dalam analisis faktor.

```
#Scree plot  
scree(data, pc=FALSE)  
  
## Warning in cor.smooth(model): Matrix was not positive definite, smoothing  
was  
## done  
  
## Warning in fa.stats(r = r, f = f, phi = phi, n.obs = n.obs, np.obs =  
np.obs, :
```

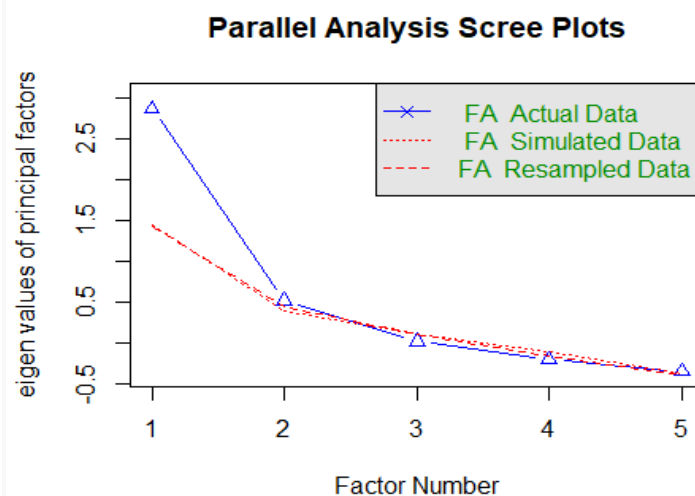
```
## The estimated weights for the factor scores are probably incorrect. Try a
## different factor score estimation method.

## Warning in fac(r = r, nfactors = nfactors, n.obs = n.obs, rotate = rotate,
: An
## ultra-Heywood case was detected. Examine the results carefully
```



Disini kita melakukan visualisasi dari jumlah faktor untuk tiap nilai eigen. Visualisasi ini berguna untuk membuat keputusan dengan menentukan jumlah faktor yang akan kita pilih untuk memilih eigen values.

```
#Cek analisis secara paralel
fa.parallel(data, fa="fa")
```



Disini kita melakukan visualisasi dari jumlah faktor untuk tiap nilai eigen. Visualisasi ini berguna untuk membuat keputusan dengan menentukan jumlah faktor yang akan kita pilih untuk memilih eigen values. Namun bedanya dengan kode yang sebelumnya, di atas merupakan analisis paralel yang menunjukkan perbedaan terkait data simulasi, data resampled dan data sebenarnya.

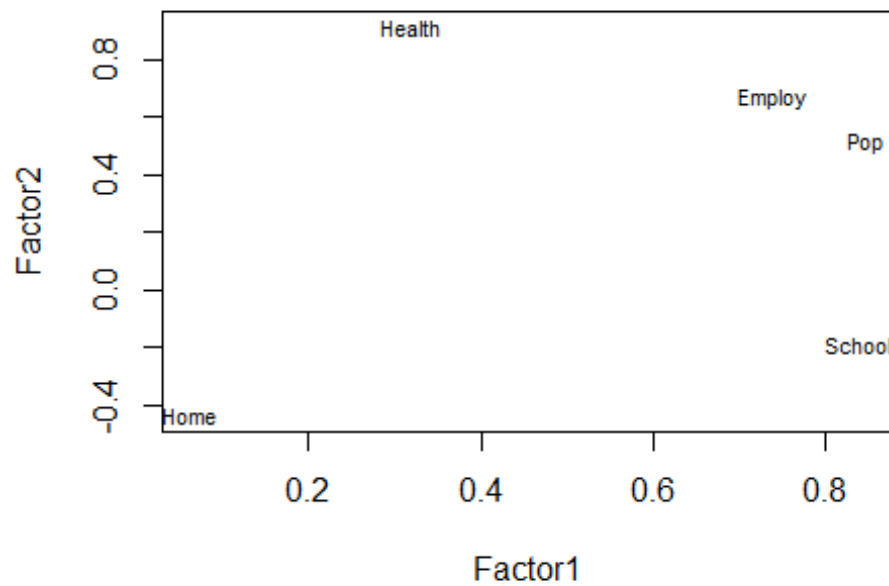
### #3. Ekstrak dan rotasi Faktor

```
Nfacs <- 2 # Misal kita menentukan 3 faktor
fit <- factanal(data, Nfacs, rotation="varimax")
print(fit, digits=2, cutoff=0.3, sort=TRUE)

##
## Call:
## factanal(x = data, factors = Nfacs, rotation = "varimax")
##
## Uniquenesses:
##      Pop School Employ Health   Home
##  0.02  0.26   0.00  0.06   0.81
##
## Loadings:
##           Factor1 Factor2
## Pop       0.85   0.52
## School    0.84
## Employ    0.74   0.67
## Health    0.32   0.91
## Home      -0.44
##
##           Factor1 Factor2
## SS loadings    2.07   1.78
## Proportion Var  0.41   0.36
## Cumulative Var  0.41   0.77
##
## Test of the hypothesis that 2 factors are sufficient.
## The chi square statistic is 0.24 on 1 degree of freedom.
## The p-value is 0.627
```

Kode di atas melakukan metode untuk mengidentifikasi pola dalam data. Disana kita melakukan ekstraksi faktor dan melakukan analisis terkait struktur data yang ada. - Uniquenesses: Menunjukkan seberapa unik variabel masing-masing dalam faktor. - Loadings: Menunjukkan seberapa kuat variabel terkait dengan faktor tertentu. - Proportion Var: Proporsi varians yang dijelaskan oleh setiap faktor. - Cumulative Var: Proporsi varians kumulatif yang dijelaskan oleh faktor-faktor sejauh ini. - Uji Hipotesis: Menguji apakah jumlah faktor yang dipilih sudah cukup untuk menjelaskan data.

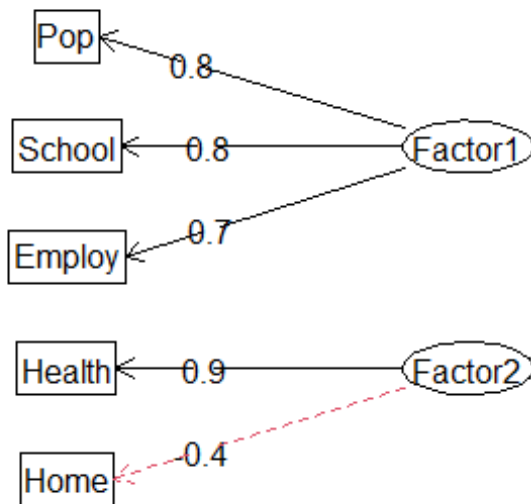
```
#Plot Faktor 1 dan Faktor 2 (Biplot)
load <- fit$loadings[,1:2]
plot(load,type="n") # set up plot
text(load,labels=names(data),cex=.7)
```



Kode di atas melakukan visualisasi terhadap kode yang dijalankan sebelumnya. Posisi tiap variabel menandakan perbandingan dari 2 faktor, yaitu factor 1 dan factor 2.

```
library(psych)
loads <- fit$loadings
fa.diagram(loads)
```

## Factor Analysis



Kode di atas menampilkan diagram analisis faktor yang memberikan informasi hubungan jumlah faktor terhadap variabel. Nilai dan informasi yang diberikan sama dengan hasil kode yang pernah dijalankan sebelumnya, hanya saja disini ditampilkan lebih jelas melalui diagram analisis faktor

```
#Menyimpan hasil
FactorLoadings <- round(fit$loadings[1:5, ], 2)
write.csv(FactorLoadings, file="FacLoads.csv")
```

Kode di atas melakukan pembulatan untuk hasil kode yang telah dijalankan dan telah diambil kesimpulannya. Dan menyimpan hasilnya ke dalam file baru FacLoads csv

```
#Analisis Faktor Kasus-2
#1. Membaca Dataset dan menampilkan
data = read.csv("TimeMan.csv", header = TRUE)
data
```

##	X6	X7	X8	X9	X10	X11	X12	X13	X14	X15	X16	X17
## 1	4	4	5	5	4	5	4	5	5	5	4	4
## 2	1	4	1	4	4	5	4	5	1	2	5	2
## 3	4	5	4	4	4	2	4	5	2	1	1	2
## 4	4	2	4	2	2	2	5	5	1	5	4	4
## 5	5	2	1	5	2	5	5	5	1	4	5	4
## 6	2	1	4	1	2	4	1	2	5	1	5	5
## 7	4	4	4	5	4	5	5	5	2	5	5	1
## 8	2	1	1	2	4	5	2	5	4	2	2	4
## 9	4	2	1	2	2	2	2	4	4	4	4	2



## 10	5	4	4	5	2	5	5	5	4	2	5	1
## 11	5	1	1	5	1	5	5	5	1	5	5	1
## 12	5	2	1	5	5	2	5	4	4	2	5	4
## 13	4	1	1	5	2	4	4	4	2	4	2	4
## 14	2	4	4	2	1	5	2	5	5	5	2	4
## 15	4	4	2	2	1	4	1	5	4	5	2	2
## 16	1	5	4	5	5	1	2	1	4	5	2	1
## 17	5	2	4	2	1	4	4	4	1	4	4	1
## 18	4	2	1	2	1	5	5	2	1	1	5	1
## 19	4	2	4	4	5	4	4	2	4	4	2	1
## 20	2	4	5	2	5	1	2	1	4	2	2	4
## 21	1	4	5	5	2	5	2	4	4	4	5	1
## 22	4	1	1	5	4	2	2	4	4	2	2	4
## 23	4	2	2	2	2	2	2	5	2	2	2	2
## 24	4	2	1	4	1	4	1	5	1	4	4	4
## 25	2	5	1	2	1	5	4	1	1	5	1	2
## 26	5	1	4	4	2	5	5	5	1	5	5	4
## 27	5	4	1	2	2	2	5	5	1	1	5	1
## 28	1	4	2	5	1	5	4	5	4	5	1	1
## 29	5	4	4	5	4	2	2	2	4	2	2	4
## 30	5	1	1	5	1	4	5	5	1	2	5	1
## 31	1	4	4	5	1	5	2	2	1	2	2	5
## 32	5	2	4	4	1	2	4	2	1	1	2	5
## 33	5	4	2	5	4	4	1	2	2	5	2	1
## 34	5	1	1	5	1	5	5	5	2	2	2	4
## 35	4	1	4	4	1	5	5	5	4	4	4	4
## 36	2	2	2	5	2	4	5	5	2	1	4	2
## 37	2	2	1	5	5	1	4	4	1	5	2	4
## 38	4	1	4	4	2	4	4	5	4	4	4	1
## 39	2	1	2	5	5	4	5	5	2	2	5	1
## 40	5	1	1	2	4	5	5	1	1	2	2	1
## 41	4	2	4	2	4	4	2	4	2	2	4	1
## 42	4	4	4	4	1	5	5	5	2	4	2	2
## 43	1	5	5	1	5	1	2	2	5	1	1	5
## 44	2	4	4	2	4	2	2	2	4	2	2	4
## 45	5	1	4	5	1	5	5	4	4	5	4	4
## 46	4	2	4	2	2	4	4	4	2	4	4	4
## 47	2	4	4	5	2	2	5	4	2	2	2	1
## 48	4	2	4	5	1	5	4	4	2	5	4	2
## 49	5	1	1	4	4	5	1	5	2	5	5	2
## 50	5	4	4	2	5	2	2	4	1	5	5	1
## 51	5	4	2	5	4	5	5	2	1	5	2	1
## 52	2	4	1	4	1	5	2	5	2	5	2	4
## 53	2	1	4	2	1	2	5	5	4	5	2	2
## 54	2	4	4	2	1	4	5	5	1	5	1	1
## 55	2	1	4	5	2	4	5	5	4	2	5	4
## 56	4	1	2	4	1	2	4	4	4	5	2	1
## 57	2	1	1	2	2	2	5	4	1	1	1	1
## 58	4	1	5	5	4	4	5	5	2	2	2	2
## 59	4	1	2	2	2	1	2	5	2	1	2	2

## 60	4	2	1	4	2	1	2	5	2	2	2	2
## 61	4	1	4	5	4	4	4	4	4	2	2	1
## 62	5	1	4	5	4	2	4	5	2	4	5	1
## 63	5	1	4	5	4	5	4	5	4	4	2	4
## 64	4	2	4	4	4	4	4	4	4	4	4	4
## 65	4	1	1	5	4	5	2	5	1	1	2	4
## 66	4	4	4	5	4	4	5	5	4	4	4	4
## 67	5	4	1	5	1	5	5	4	1	2	2	4
## 68	5	2	4	5	4	4	1	5	1	2	2	2
## 69	5	4	4	2	4	4	4	5	5	5	4	2
## 70	5	1	4	5	1	2	4	5	1	5	4	1
## 71	2	4	4	5	1	5	2	2	2	4	5	1
## 72	4	2	1	2	2	2	5	5	1	2	2	2
## 73	5	2	1	2	1	5	5	4	2	2	2	1
## 74	4	2	4	2	2	1	2	4	2	4	5	2
## 75	5	2	5	2	2	5	5	4	2	5	2	1
## 76	4	4	1	4	1	2	5	5	4	5	2	1
## 77	1	1	1	5	1	5	1	5	1	1	1	1
## 78	5	1	1	5	1	5	5	5	1	5	5	5
## 79	1	1	4	2	4	5	2	5	1	1	1	4
## 80	2	1	1	5	1	2	2	4	1	5	5	1
## 81	5	4	4	5	4	4	4	5	1	5	2	4
## 82	5	2	4	2	1	2	5	4	1	1	5	2
## 83	5	5	5	2	5	5	5	5	1	2	5	2
## 84	5	1	1	5	4	5	4	4	2	2	4	2
## 85	4	2	4	5	2	4	5	4	4	4	4	1
## 86	2	2	1	2	1	2	5	1	2	1	4	1
## 87	5	2	5	4	2	5	5	5	4	2	1	1
## 88	5	1	1	5	1	5	5	5	1	5	5	1
## 89	5	1	1	5	4	2	5	5	5	5	5	1
## 90	5	4	1	4	1	5	5	5	4	5	5	2
## 91	4	4	4	2	4	5	5	2	4	4	4	4
## 92	2	2	2	2	2	4	2	4	4	2	2	2
## 93	4	2	4	2	1	4	4	4	1	1	2	4
## 94	5	4	4	4	1	5	5	5	4	5	4	4
## 95	4	1	1	5	1	4	5	4	4	5	5	1
## 96	4	4	4	5	4	5	5	4	2	4	5	2
## 97	4	4	5	5	5	5	2	4	1	4	4	4
## 98	4	1	5	4	4	4	4	5	4	2	4	2
## 99	2	5	4	5	4	4	1	4	4	5	5	4
## 100	4	4	4	5	4	4	5	4	1	4	4	4
## 101	5	5	5	5	5	4	5	4	5	5	2	5
## 102	5	2	1	2	4	4	5	5	5	5	2	4
## 103	4	1	1	5	1	5	5	5	1	5	5	1
## 104	4	1	4	5	4	5	2	5	4	4	2	1
## 105	4	1	2	5	4	4	2	4	4	4	2	2
## 106	4	4	2	5	4	2	4	4	1	5	4	4
## 107	4	1	4	2	4	4	4	4	4	2	5	4
## 108	5	2	4	5	4	5	2	5	2	2	2	4
## 109	2	2	4	4	4	2	2	4	1	5	5	4

```
## 110 5 2 4 2 4 2 5 5 2 2 5 2
## 111 5 2 2 5 4 2 5 4 2 4 5 2
## 112 1 2 1 2 1 1 2 5 2 4 4 1
## 113 5 1 1 5 1 5 5 5 1 5 5 1
## 114 4 2 4 4 1 4 4 5 2 2 4 4
## 115 5 4 5 5 4 5 4 5 1 5 5 4
## 116 5 1 5 2 1 4 5 4 2 5 2 4
## 117 4 4 4 5 4 5 5 4 1 4 2 1
## 118 4 2 4 2 4 2 5 5 4 2 1 5
## 119 4 2 4 5 4 2 2 4 1 4 5 4
## 120 5 1 2 2 1 2 4 5 1 5 4 1
## 121 4 2 4 5 2 4 5 4 2 2 5 4
## 122 2 2 1 2 1 1 2 2 2 2 1 4
## 123 5 1 1 5 1 5 5 5 2 1 4 1
## 124 2 2 2 1 2 2 2 5 2 1 4 2
## 125 4 1 4 2 4 5 4 5 4 5 4 5
```

Kode di atas menampilkan data dari dataset TimeMan

## #2. Evaluasi Data

### #Deskripsi Statistik

```
summary(data)
```

```
##           X6           X7           X8           X9           X10
## Min.      :1.000   Min.      :1.000   Min.      :1.000   Min.      :1.000   Min.
## :1.00
## 1st Qu.:2.000   1st Qu.:1.000   1st Qu.:1.000   1st Qu.:2.000   1st
## Qu.:1.00
## Median :4.000   Median :2.000   Median :4.000   Median :4.000   Median
## :2.00
## Mean    :3.776   Mean     :2.384   Mean     :2.904   Mean     :3.752   Mean
## :2.64
## 3rd Qu.:5.000   3rd Qu.:4.000   3rd Qu.:4.000   3rd Qu.:5.000   3rd
## Qu.:4.00
## Max.    :5.000   Max.     :5.000   Max.     :5.000   Max.     :5.000   Max.
## :5.00
##           X11           X12           X13           X14           X15
## Min.      :1.00   Min.      :1.000   Min.      :1.000   Min.      :1.000   Min.
## :1.000
## 1st Qu.:2.00   1st Qu.:2.000   1st Qu.:4.000   1st Qu.:1.000   1st
## Qu.:2.000
## Median :4.00   Median :4.000   Median :5.000   Median :2.000   Median
## :4.000
## Mean    :3.68   Mean     :3.736   Mean     :4.192   Mean     :2.456   Mean
## :3.368
## 3rd Qu.:5.00   3rd Qu.:5.000   3rd Qu.:5.000   3rd Qu.:4.000   3rd
## Qu.:5.000
## Max.    :5.00   Max.     :5.000   Max.     :5.000   Max.     :5.000   Max.
## :5.000
##           X16           X17
```

```
## Min. :1.00 Min. :1.000
## 1st Qu.:2.00 1st Qu.:1.000
## Median :4.00 Median :2.000
## Mean :3.32 Mean :2.544
## 3rd Qu.:5.00 3rd Qu.:4.000
## Max. :5.00 Max. :5.000
```

### #Evaluasi Matriks Korelasi

```
library(RcmdrMisc)
```

```
rcorr.adjust(data) # Fungsi ini dibangun di R Commander.
```

```
##
## Pearson correlations:
##      X6      X7      X8      X9      X10      X11      X12      X13
X14
## X6  1.0000 -0.1788  0.0094  0.1721  0.0498  0.1772  0.4012  0.2431 -
0.1028
## X7 -0.1788  1.0000  0.2901  0.0038  0.2146  0.0186 -0.0732 -0.2601
0.0816
## X8  0.0094  0.2901  1.0000 -0.0418  0.3096  0.0619 -0.0082 -0.0649
0.2393
## X9  0.1721  0.0038 -0.0418  1.0000  0.1124  0.2893  0.1259  0.1684 -
0.0655
## X10 0.0498  0.2146  0.3096  0.1124  1.0000 -0.1072 -0.1583 -0.1277
0.2056
## X11 0.1772  0.0186  0.0619  0.2893 -0.1072  1.0000  0.1987  0.1780 -
0.0197
## X12 0.4012 -0.0732 -0.0082  0.1259 -0.1583  0.1987  1.0000  0.1846 -
0.0778
## X13 0.2431 -0.2601 -0.0649  0.1684 -0.1277  0.1780  0.1846  1.0000 -
0.0459
## X14 -0.1028  0.0816  0.2393 -0.0655  0.2056 -0.0197 -0.0778 -0.0459
1.0000
## X15 0.1664  0.1645  0.0607  0.2345 -0.0443  0.1612  0.0864  0.1316
0.1058
## X16 0.2303 -0.1209 -0.0006  0.1709 -0.0211  0.0992  0.1882  0.1848 -
0.1090
## X17 -0.0334  0.0826  0.2328 -0.0407  0.1921  0.0273 -0.1319 -0.0552
0.1532
##      X15      X16      X17
## X6  0.1664  0.2303 -0.0334
## X7  0.1645 -0.1209  0.0826
## X8  0.0607 -0.0006  0.2328
## X9  0.2345  0.1709 -0.0407
## X10 -0.0443 -0.0211  0.1921
## X11 0.1612  0.0992  0.0273
## X12 0.0864  0.1882 -0.1319
## X13 0.1316  0.1848 -0.0552
## X14 0.1058 -0.1090  0.1532
## X15 1.0000  0.1976 -0.0515
```

```

## X16  0.1976  1.0000 -0.0924
## X17 -0.0515 -0.0924  1.0000
##
## Number of observations: 125
##
## Pairwise two-sided p-values:
##      X6      X7      X8      X9      X10      X11      X12      X13      X14      X15
## X6      0.0460 0.9167 0.0550 0.5814 0.0481 <.0001 0.0063 0.2537 0.0637
## X7 0.0460      0.0010 0.9661 0.0162 0.8368 0.4173 0.0034 0.3656 0.0668
## X8 0.9167 0.0010      0.6432 0.0004 0.4931 0.9276 0.4720 0.0072 0.5013
## X9 0.0550 0.9661 0.6432      0.2122 0.0011 0.1619 0.0605 0.4679 0.0085
## X10 0.5814 0.0162 0.0004 0.2122      0.2340 0.0779 0.1559 0.0214 0.6237
## X11 0.0481 0.8368 0.4931 0.0011 0.2340      0.0263 0.0470 0.8270 0.0725
## X12 <.0001 0.4173 0.9276 0.1619 0.0779 0.0263      0.0394 0.3885 0.3378
## X13 0.0063 0.0034 0.4720 0.0605 0.1559 0.0470 0.0394      0.6111 0.1435
## X14 0.2537 0.3656 0.0072 0.4679 0.0214 0.8270 0.3885 0.6111      0.2402
## X15 0.0637 0.0668 0.5013 0.0085 0.6237 0.0725 0.3378 0.1435 0.2402
## X16 0.0098 0.1793 0.9948 0.0567 0.8150 0.2711 0.0355 0.0391 0.2263 0.0272
## X17 0.7113 0.3600 0.0090 0.6523 0.0319 0.7629 0.1427 0.5410 0.0880 0.5686
##      X16      X17
## X6 0.0098 0.7113
## X7 0.1793 0.3600
## X8 0.9948 0.0090
## X9 0.0567 0.6523
## X10 0.8150 0.0319
## X11 0.2711 0.7629
## X12 0.0355 0.1427
## X13 0.0391 0.5410
## X14 0.2263 0.0880
## X15 0.0272 0.5686
## X16      0.3053
## X17 0.3053
##
## Adjusted p-values (Holm's method)
##      X6      X7      X8      X9      X10      X11      X12      X13      X14      X15
## X6      1.0000 1.0000 1.0000 1.0000 1.0000 1.0000 0.0002 0.3844 1.0000 1.0000
## X7 1.0000      0.0660 1.0000 0.9092 1.0000 1.0000 1.0000 0.2110 1.0000 1.0000
## X8 1.0000 0.0660      1.0000 0.0288 1.0000 1.0000 1.0000 1.0000 0.4320 1.0000
## X9 1.0000 1.0000 1.0000      1.0000 0.0673 1.0000 1.0000 1.0000 1.0000 0.5003
## X10 1.0000 0.9092 0.0288 1.0000      1.0000 1.0000 1.0000 1.0000 1.0000
## X11 1.0000 1.0000 1.0000 0.0673 1.0000      1.0000 1.0000 1.0000 1.0000
## X12 0.0002 1.0000 1.0000 1.0000 1.0000 1.0000      1.0000 1.0000 1.0000
## X13 0.3844 0.2110 1.0000 1.0000 1.0000 1.0000 1.0000      1.0000 1.0000
## X14 1.0000 1.0000 0.4320 1.0000 1.0000 1.0000 1.0000 1.0000      1.0000
## X15 1.0000 1.0000 1.0000 0.5003 1.0000 1.0000 1.0000 1.0000 1.0000
## X16 0.5563 1.0000 1.0000 1.0000 1.0000 1.0000 1.0000 1.0000 1.0000
## X17 1.0000 1.0000 0.5208 1.0000 1.0000 1.0000 1.0000 1.0000 1.0000
##      X16      X17
## X6 0.5563 1.0000
## X7 1.0000 1.0000

```

```
## X8 1.0000 0.5208
## X9 1.0000 1.0000
## X10 1.0000 1.0000
## X11 1.0000 1.0000
## X12 1.0000 1.0000
## X13 1.0000 1.0000
## X14 1.0000 1.0000
## X15 1.0000 1.0000
## X16 1.0000
## X17 1.0000
```

Kode di atas menampilkan evaluasi data yang komprehensif dan analisis statistik deskriptif serta matriks korelasi. Penting untuk memahami karakteristik data sebelum melakukan analisis lebih lanjut.

```
# Write correlations and high correlations to files directly
write.csv(cor(data), file = "Correlation_Values.csv")
write.csv(cor(data) > 0.8, file = "Suspect_Correlations.csv")
#Uji KMO
library(psych)
KMO(data)

## Kaiser-Meyer-Olkin factor adequacy
## Call: KMO(r = data)
## Overall MSA = 0.61
## MSA for each item =
## X6 X7 X8 X9 X10 X11 X12 X13 X14 X15 X16 X17
## 0.61 0.55 0.63 0.59 0.51 0.64 0.63 0.75 0.58 0.52 0.74 0.70

round( KMO(data)$MSA, 2 )

## [1] 0.61

#Uji Bartlett
library(psych)
cortest.bartlett(data)

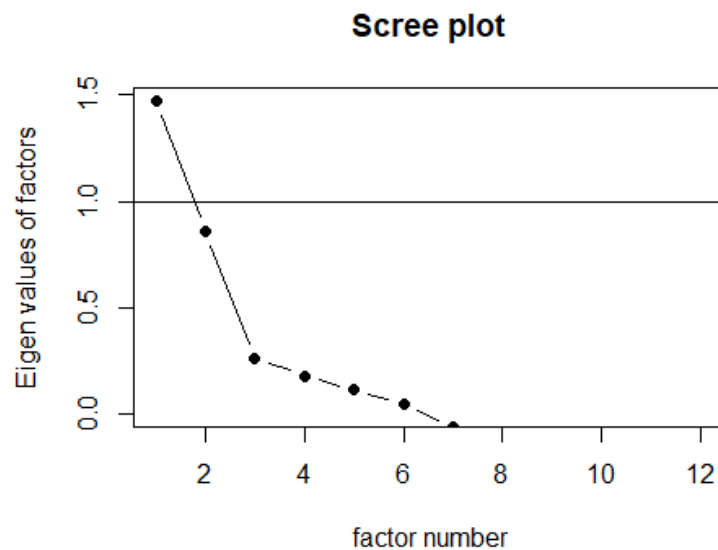
## R was not square, finding R from data

## $chisq
## [1] 161.864
##
## $p.value
## [1] 5.063681e-10
##
## $df
## [1] 66
```

Kode di atas menyimpan hasil korelasi ke dalam file csv baru lalu menampilkan hasil korelasi dari data, serta melakukan uji KMO dan uji Bartlett untuk menguji kelayakan data

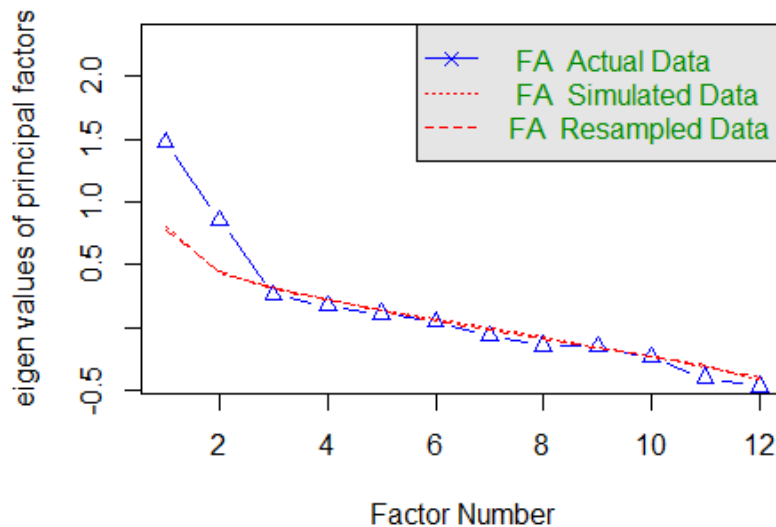
dan homogenitas matriks kovarians. Langkah ini dilakukan untuk mempersiapkan analisis data statistik yang lebih lanjut.

```
#2. Menentukan jumlah faktor yang diekstraksi  
ev <- eigen(cor(data)) # mendapatkan nilai eigen ev$values  
#Scree plot  
scree(data, pc=FALSE)
```



```
#Cek analisis secara paralel  
fa.parallel(data, fa="fa")
```

### Parallel Analysis Scree Plots



Kode R di atas mencari nilai eigen dari korelasi data. Nilai eigen dipakai untuk menentukan jumlah faktor yang diekstraksi dalam analisis faktor. Selanjutnya memakai visualisasi scree plot dan analisis paralel.

#### #3. Ekstrak dan rotasi Faktor

```
Nfacs <- 6 # Misal kita menentukan 6 faktor
fit <- factanal(data, Nfacs, rotation="varimax")
print(fit, digits=2, cutoff=0.3, sort=TRUE)
```

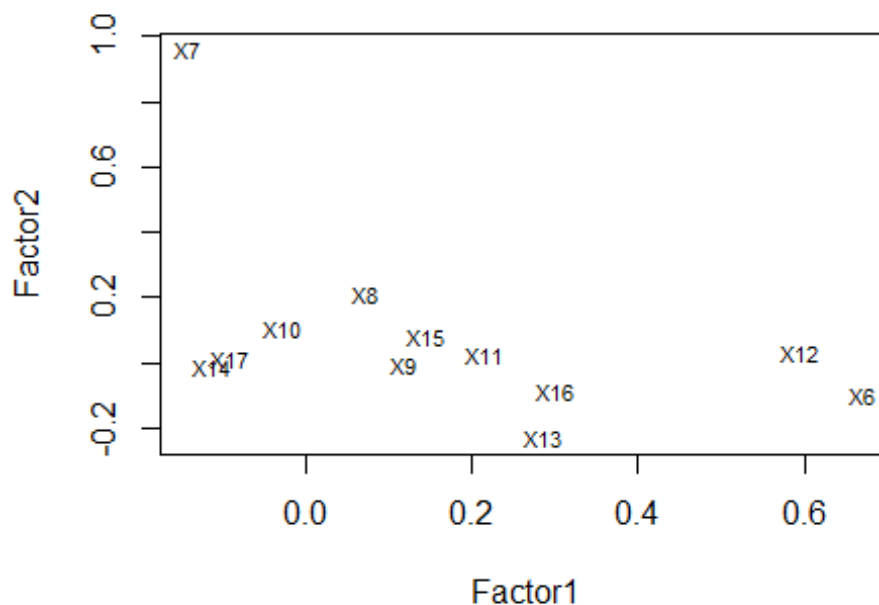
```
##
## Call:
## factanal(x = data, factors = Nfacs, rotation = "varimax")
##
## Uniquenesses:
##   X6   X7   X8   X9  X10  X11  X12  X13  X14  X15  X16  X17
## 0.51 0.02 0.61 0.55 0.00 0.69 0.61 0.80 0.79 0.00 0.85 0.83
##
## Loadings:
##      Factor1 Factor2 Factor3 Factor4 Factor5 Factor6
## X6      0.67
## X12     0.60
## X7              0.96
## X15                  0.96
## X10                  0.93      0.35
## X8                    0.57
## X9                      0.62
## X11                     0.48
## X13
```



```
## X14                                0.42
## X16  0.30
## X17                                0.39
##
##          Factor1 Factor2 Factor3 Factor4 Factor5 Factor6
## SS loadings    1.11    1.06    1.01    0.97    0.85    0.74
## Proportion Var  0.09    0.09    0.08    0.08    0.07    0.06
## Cumulative Var  0.09    0.18    0.26    0.34    0.42    0.48
##
## Test of the hypothesis that 6 factors are sufficient.
## The chi square statistic is 2.99 on 9 degrees of freedom.
## The p-value is 0.965
```

Kode di atas melakukan ekstraksi dan rotasi faktor dalam analisis faktor. Ekstraksi dan rotasi faktor ini membantu kita memahami struktur data dengan lebih baik dan mengidentifikasi faktor yang mendasarinya. Dengan pemahaman yang baik tentang ekstraksi dan rotasi faktor, kita dapat membuat keputusan yang lebih tepat berdasarkan pola hubungan antar variabel yang ada.

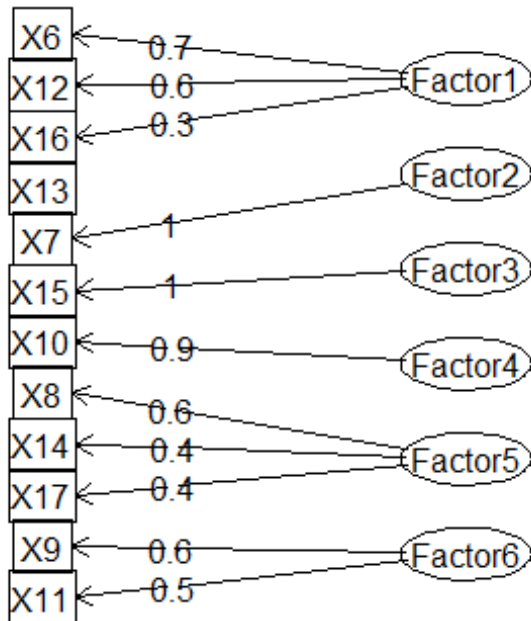
```
# Plot Factor Biplot
load <- fit$loadings[, 1:6]
plot(load, type = "n") # set up plot
text(load, labels = names(data), cex = 0.7)
```



Selanjutnya menampilkan visualisasi biplot faktor untuk untuk memvisualisasikan hubungan antara variabel dan faktor dalam analisis faktor.

```
library(psych)
loads <- fit$loadings
fa.diagram(loads)
```

## Factor Analysis



Dengan menggunakan pustaka psych dan fungsi fa.diagram() dalam R, kita dapat dengan mudah membuat visualisasi diagram faktor analisis untuk menganalisis dan memahami struktur faktor dari data yang kompleks. Proses ini memberikan insight dalam interpretasi data.

### #4. Evaluasi

#### #Ekspor Faktor Loading

```
dim(fit$loadings)
```

```
## [1] 12 6
```

```
round(fit$loadings[ 1:12,], 6)
```

```
##      Factor1  Factor2  Factor3  Factor4  Factor5  Factor6
## X6  0.670984 -0.099103  0.065600  0.097541 -0.034229  0.107607
## X7 -0.141044  0.959417  0.095272  0.063983  0.158026  0.035452
## X8  0.074145  0.212345  0.002681  0.096489  0.571841 -0.000011
## X9  0.119975 -0.006211  0.125592  0.162917 -0.111294  0.618702
## X10 -0.026109  0.103524 -0.036909  0.927083  0.350085  0.014037
## X11  0.216883  0.023911  0.031894 -0.157278  0.103470  0.478638
## X12  0.595848  0.031931 -0.022976 -0.128886 -0.082439  0.105002
## X13  0.287649 -0.230835  0.071710 -0.086927 -0.046025  0.220776
## X14 -0.110032 -0.013708  0.115422  0.067318  0.419038 -0.057312
```

```
## X15  0.146317  0.083675  0.964619 -0.039291  0.058271  0.176524
## X16  0.302347 -0.087905  0.146757  0.035558 -0.096408  0.148219
## X17 -0.088099  0.010424 -0.066537  0.051822  0.394180  0.021525
```

*#Menyimpan hasil*

```
FactorLoadings <- round(fit$loadings[1:12, ], 6)
write.csv(FactorLoadings, file = "FacLoads2b.csv")
```

Dalam analisis faktor, mengevaluasi faktor loading dan menyimpan hasilnya adalah langkah penting untuk memahami model faktor yang telah dibuat. Dengan menggunakan kode R yang sesuai, kita dapat dengan mudah mengakses, menyimpan, dan mengekspor faktor loading untuk analisis lebih lanjut. Semoga penjelasan ini membantu kamu memahami proses evaluasi faktor loading dalam konteks analisis faktor menggunakan R.