

# Pengembangan Media Pembelajaran Analisis Komponen Utama Berbasis Web Menggunakan Shiny R

Asmarianti

Program Studi Matematika, Universitas Islam Negeri Alauddin Makassar, 60600117018@uin-alauddin.ac.id

Wahidah Alwi

Program Studi Matematika, Universitas Islam Negeri Alauddin Makassar, wahidah.alwi@uin-alauddin.ac.id

Adnan Sauddin

Program Studi Matematika, Universitas Islam Negeri Alauddin Makassar, adnan.sauddin@uin-alauddin.ac.id

---

**ABSTRAK**, Penelitian ini membahas mengenai pengembangan media pembelajaran analisis komponen utama berbasis web menggunakan shiny R. Media pembelajaran yang dapat dimanfaatkan untuk meningkatkan efisiensi belajar yaitu media yang berbasis web, salah satunya Shiny R. Shiny R adalah sebuah package bahasa pemrograman R yang dapat membangun aplikasi berbasis website menggunakan paket *Shiny Web Framework* yang merupakan pengembangan dari R studio Server. Salah satu teknik atau metode yang digunakan adalah *Principal Component Analysis* (PCA). Data yang digunakan dalam penelitian menggunakan data mtcars bawaan pada R Studio yang berisi tentang spesifikasi kendaraan tahun 1970. Adapun hasil pada penelitian ini yaitu pembuatan aplikasi berhasil menjalankan analisis menggunakan PCA dengan sampel dataset mtcars.

---

**Kata Kunci:** *Shiny Web, Regression, Principal Component Analysis (PCA)*

---

## 1. PENDAHULUAN

Media pembelajaran sangat penting dalam proses belajar mengajar. Media merupakan alat yang ditujukan untuk memberikan pengetahuan kepada mahasiswa. Penggunaan media pembelajaran dalam proses belajar mengajar dapat membangkitkan keinginan, memotivasi dan menginspirasi kegiatan belajar, bahkan memberikan dampak psikologis bagi mahasiswa. Media pembelajaran merupakan contoh faktor eksternal yang dapat digunakan untuk meningkatkan efisiensi belajar. Hal ini dapat tercapai karena media pembelajaran dapat mengatasi berbagai kendala yang ditemukan.

Oleh karena itu, pengajar atau pembelajar harus mampu menciptakan atau mengembangkan media yang mendukung keberhasilan proses pembelajaran. pengajar atau pembelajar berhak menerima dan memilih apa yang dipelajari, termasuk belajar matematika. Belajar matematika sangat penting karena sangat

berguna dalam kehidupan sehari-hari, namun pada kenyataannya pengajar dan pembelajar mengalami kesulitan atau tidak memahami konsep matematika. Salah satunya dalam data statistik. Dalam matematika, statistika dikenal sebagai “ilmu tentang data” membutuhkan minimal tiga pengetahuan pokok agar dapat diimplementasikan dengan benar. Pertama fenomena yang di gambarkan, kedua teori atau formula matematika, dan ketiga interpretasi dari model matematika atau statistika. Hubungan ketiga keadaan tersebut menjadikan statistika atau matematika menjadi satu basis pengetahuan yang sulit untuk dipahami dalam proses implementasi pada bidang-bidang keilmuan lainnya.

Statistika banyak digunakan dalam berbagai disiplin ilmu, salah satunya dalam bidang komputasi. Ilmu komputasi adalah bidang ilmu yang berhubungan dengan penciptaan model matematika dan solusi numerik. Selain itu, pendekatan ilmu komputasi dapat memberikan berbagai pemahaman baru, melalui penerapan model-model matematika dalam program komputer berdasarkan landasan teori yang telah berkembang, untuk menyelesaikan masalah-masalah nyata dalam ilmu tersebut.

Penggunaan teknologi untuk mendukung pembelajaran statistika (khususnya perangkat lunak komputer) dimulai sejak satu dekade yang lalu. Dalam pemanfaatan simulasi dan visualisasi komputer, program ini harus dirancang dengan semaksimal mungkin (dalam penyusunan perangkat maupun dalam pemanfaatannya di kelas). Ilustrasi, visualisasi dan simulasi dalam pembelajaran statistika, jika dirancang dengan baik diyakini dapat meningkatkan minat, motivasi dan pemahaman dalam belajar statistika baik teori maupun aplikasi yang dalam

pendekatan tradisional dirasa sangat teoritis. Oleh karena itu, pembelajaran dengan model simulasi, visualisasi merupakan hal yang sangat dibutuhkan bagi pengajar atau pembelajar dalam proses belajar mengajar statistika.

Hal tersebut dapat dipermudah dengan memberi satu media pembelajaran yang berbasis visualisasi dan simulasi yakni Shiny R. Shiny R dapat digunakan untuk membangun suatu aplikasi berbasis web, dengan aplikasi tersebut dapat memudahkan dalam menampilkan dan mengekspos melalui browser web sehingga siapa pun dapat menggunakannya.

## 2. TINJAUAN PUSTAKA

### Principal Component Analysis (PCA)

Principal Component Analysis (PCA) merupakan metode statistik yang secara linear mengganti kumpulan variabel asli menjadi kumpulan variabel tidak berkorelasi yang secara substansial lebih kecil yang mewakili sebagian besar data dalam kumpulan variabel asli. Tujuan PCA mirip dengan analisis faktor di mana kedua metode berupaya menguraikan bagian dari variasi dalam satu set variabel yang diamati bersumber pada sebagian ukuran yang mendasarinya. Singkatnya, PCA memiliki model statistik yang mendasari variabel yang diamati berdasarkan sifat varians maksimum dari komponen utama[1].

Banyaknya komponen utama yang terbentuk sama dengan jumlah variabel awal. Apabila komponen utama diturunkan dari populasi multivariat normal dengan random vektor  $X = (X_1, X_2, \dots, X_p)$  dan vektor rata-rata  $\mu = (\mu_1, \mu_2, \dots, \mu_p)$  serta matriks kovarians  $\Sigma$  dengan akar ciri (*eigenvalue*) yaitu  $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_p \geq 0$  didapat kombinasi linear komponen utama yaitu sebagai berikut.

$$\begin{aligned} Y_1 &= e'_1 X = e'_{11} X_1 + e'_{21} X_2 + \dots + e'_{p1} X_p \\ Y_2 &= e'_2 X = e'_{12} X_1 + e'_{22} X_2 + \dots + e'_{p2} X_p \\ &\vdots \\ Y_p &= e'_p X = e'_{1p} X_1 + e'_{2p} X_2 + \dots + e'_{pp} X_p \end{aligned} \quad (2.1)$$

Menurut (Jolliffe, 2002), program PCA bertujuan untuk menyederhanakan dan menghilangkan faktor atau indikator penyaringan yang kurang dominan dan relevan, sekaligus tidak mengurangi arti dan tujuan dari data asli variabel acak  $x$  (matriks

berdimensi  $n \times n$ ), dimana baris yang berisi observasi sebanyak  $n$  dari variabel acak  $x$  adalah sebagai berikut:

- 1) Menghitung Matriks Varians Kovarian

Varians ( $Var(x)$ ) dihitung untuk penyebaran data untuk memastikan penyimpangan data dalam set data sampel. Matriks Kovarian  $Cov(x, y)$  ialah matriks yang nilai-nilai kovariansi pada tiap cellnya diperoleh dari sampel. Misalkan  $x$  dan  $y$  adalah variabel acak.

$$var(x) = \sigma^2 = \frac{1}{n} \sum_{i=1}^n (\bar{z}_{ij} - \mu_j)^2 \quad (2.2)$$

$$cov(x, y) = \frac{1}{n-1} \sum_{i=1}^n (x_{ij} - \mu_{xj})(y_{ij} - \mu_{yj}) \quad (2.3)$$

Dengan  $\mu_x$  dan  $\mu_y$  merupakan rata-rata (*mean*) sampel dari variabel  $x$  dan  $y$  (mean), dimana  $x_i$  dan  $y_i$  adalah nilai observasi ke- $i$  dari variabel  $x$  dan  $y$ . Dari data numerik yang digunakan, dapat diperoleh matriks kovarians berukuran  $n \times n$ .

- 2) Mencari *eigen value* dan *eigen vektor* dari matriks kovarians yang telah diperoleh yaitu nilai eigen dan vektor eigen untuk matriks kovarians dihitung. Nilai eigen yang dikomputasi kemudian ditransformasikan (rotasi orthogonal varimax) menggunakan persamaan berikut:

$$\text{Det}(A - \lambda I) = 0 \quad (2.4)$$

Dimana:

$A$  = matriks  $n \times n$

$\lambda$  = nilai *eigenvalue*

$I$  = matriks identitas (matriks persegi dengan elemen diagonal utama bernilai 1 sedangkan elemen lain bernilai 0)

- 3) Menentukan nilai proporsi komponen utama (proporsi *principal component* (%)) dengan persamaan:

$$PC(\%) = \frac{\text{Nilai Eigen}}{\text{Varians Kovarian}} \times 100 \quad (2.5)$$

- 4) Menghitung bobot faktor (*factor loading*) berdasarkan *eigenvektor*, dengan persamaan:

$$Ax = \lambda x \quad (2.6)$$

Sehingga diperoleh kombinasi linear yaitu:

- a.  $\lambda_1, \lambda_2, \lambda_3, \dots, \lambda_n$  adalah *eigenvalue* matriks  $A$

- b.  $x_1, x_2, x_3, \dots, x_n$  adalah *eigenvector* sesuai *eigenvalue*-nya ( $\lambda_n$ )

Persamaan *eigenvalue* dan *eigenvector* merupakan *Eigen Value Decomposition* (EVD), dengan persamaan sebagai berikut:

$$\begin{aligned} AX &= XD \\ A &= XDX^{-1} \end{aligned} \quad (2.10)$$

Dimana:

$A$  = matrik  $n \times n$  yang memiliki  $n$  nilai eigen ( $\lambda_n$ )

$D$  = nilai eigen dari vektor eigen

$X$  = vektor eigen dari matrik  $A$

$X^{-1}$  = invers dari vektor eigen  $X$ [2].

### Kriteria Pemilihan PCA

Secara umum terdapat 3 kriteria yang digunakan untuk menentukan banyaknya komponen utama adalah sebagai berikut:

- Berdasarkan proporsi kumulatif total keragaman yang mampu dijelaskan. Metode ini diterapkan pada matriks korelasi ataupun matriks kovarian.
- Berdasarkan nilai eigen dari komponen utama. Tetapi hanya dapat diterapkan pada matriks korelasi, yaitu jika nilai eigen lebih atau sama dengan satu.
- Berdasarkan scree plot. Dengan menggunakan metode pemilihan jumlah komponen utama yaitu  $k$ , jika di titik  $k$  grafiknya curam ke kiri tetapi tidak curam ke kanan. Ide di balik metode ini adalah untuk memilih jumlah komponen utama sedemikian rupa sehingga perbedaan antara akar ciri kontinu tidak lagi besar[3].

### Perhitungan PCA

PCA akan diperhitungkan dengan empat fungsi yang akan menjadi perhitungan analisis datanya yaitu :

- Predict PCA*
- Eigenvalue PCA*
- Cos 2 PCA*
- Contrib PCA*

Model PCA menjadi tolak ukur untuk perhitungan algoritma PCA kedepannya. Nantinya ini menjadi perhitungan utama ke dalam algoritma PCA (*predict, eigenvalue, cos2, dan contrib*) hingga ke visualisasi datasetnya. Library yang dibutuhkan untuk mengaplikasikan PCA ke dalam R adalah *factoextra*. Fungsi ini

sebagai syarat utama jika ingin menambahkan algoritma PCA ke dalam R Studio.

Disini perhitungan menggunakan fungsi yang sederhana yang dinamakan "*prcomp*". Di dalam fungsi tersebut terdapat variabel "*hasil\_modelPCA*" yang menggambarkan matriks numerik dari dataset yang digunakan dan *scale* sebagai deskripsi dari nilai variabel bersifat logis atau tidak (bergantung pada nilai *true/false*) yang menunjukkan apakah variabel harus diskalakan agar memiliki varians unit sebelum PCA dilakukan. Varians unit ini juga nanti akan menentukan area dari dimensi di tiap plot yang akan mewakili sebaran instance data lewat nilai dari *eigenvalues*.

Perhitungan yang selanjutnya merupakan *predict PCA* yang didapatkan juga dari nilai model PCA. Pada perhitungan ini akan menganalisa nilai dari tiap satuan individu pada data (mobil) dan tambahan variabel dari informasi perhitungan model sebelumnya.

Terakhir, terdapat perhitungan *eigenvalues* yang menjadi dasar utama dalam perhitungan dimensi varians yang akan merujuk ke sebaran instance data untuk tiap dimensinya, *coord* yang menjelaskan koordinat dari tiap variabel, *cos2* sebagai kualitas representasi nilai untuk variabel di dalam grafik visualisasi (*individuals plot*), dan *contrib* yang menjadikan fungsi yang berisi kontribusi (dalam persen) dari variabel ke PCA (*variables plot*). Untuk nilai *cos2* didapatkan dari hasil perhitungan nilai *coord* dan *contrib* didapatkan dari hasil perhitungan nilai *cos2*. Nilai *cos2* berasal dari :

$$\frac{\text{var.coord}^2}{\text{total.cos2}}$$

Sedangkan nilai *contrib* didapatkan dari :

$$\frac{\text{var.cos2} \times 100}{\text{total.cos2}}$$

### Visualisasi PCA

Visualisasi PCA dilakukan setelah perhitungan, maka visualisasi dalam aplikasi ini juga harus ditampilkan sebagai informasi yang akan didapatkan bagi para pengguna aplikasi. Empat plot yang disajikan akan mewakili informasi dari analisis data menggunakan algoritma PCA dari perhitungan model. Empat plot yang akan divisualisasikan ke dalam aplikasi ini yaitu :

- Scree Plot*
- Individuals Plot*

### c. Variables Plot

### d. Biplot

Langkah terakhir setelah perhitungan analisis PCA selesai maka akan dilakukan klusterisasi data yang akan divisualisasikan menggunakan plot sebagai informasi yang akan didapatkan. Di dalam R, diperlukan dua packages sebagai fungsi menampilkan plot dan visualisasi menggunakan prosedur PCA yaitu *factoextra* dan *ggplot*[4].

## Shiny R

Shiny R adalah sebuah *package* bahasa pemrograman R yang dapat membangun aplikasi berbasis website menggunakan paket Shiny Web Framework yang merupakan pengembangan dari R studio Server[5]. Secara umum, komponen program Shiny dibagi menjadi dua kategori, sebagai berikut:

### 1. User Interface (UI)

Bagian ini yang bermanfaat untuk:

- a. Panel kontrol adalah panel yang mengontrol input berupa data, variabel, dan model sesuai dengan kompleksitas modul. Kontrol tampilan dapat berupa slider, radio button, check-box, dll. Contoh kontrol tampilan (seperti check-box, radio button, slider, dll).
- b. Pemasukan permintaan nilai input (data dengan berbagai jenis variabel yang diperlukan, pemilihan model, jenis dan kriteria uji statistika).
- c. penyajian output terkait hasil analisis/uji. Hasil output dapat berupa:
  - i. Grafik (histogram, diagram pencar, dan lain-lain).
  - ii. Bentuk angka/teks bisa berbentuk asli (verbatim) maupun dalam bentuk tabel. Teks khusus dengan notasi matematika dengan format LaTeX.
  - iii. Untuk mengakomodasi berbagai jenis luaran tadi, shiny juga menyediakan berbagai format output seperti `plotOutput`, `textOutput`, `verbatimTextOutput`, `tableOutput` dan lain-lainnya.
  - iv. Grafik (histogram, diagram pencar, dan lain-lain).
  - v. Bentuk angka/teks bisa berbentuk asli (verbatim) maupun dalam bentuk tabel.

### 2. Server

Bagian ini adalah otak dari program, yang bertanggung jawab untuk simulasi, menganalisis berbagai data sesuai pilihan pengguna, dan kemudian mengirimkan hasilnya ke bagian output. Bagian ini didukung oleh berbagai program dan analisis data, dan biasanya terdapat dalam berbagai paket R. Bagian ini disimpan dalam file dinamakan `server.r`. Bagian *user interface* ini dapat disajikan pada file khusus `ui.r`, dapat juga disajikan penuh melalui file HTML, misalnya `index.html`[6].

## Analisis Regresi

Analisis regresi adalah salah satu teknik statistik yang paling populer yang biasa digunakan untuk tujuan peramalan atau pendugaan tentang nilai variabel dependen (Y). Analisis regresi terbagi menjadi dua bagian yaitu analisis regresi linear sederhana dan analisis regresi linear berganda. Analisis regresi linear sederhana adalah analisis yang menduga satu variabel dependen (Y) oleh satu variabel bebas (X) saja. Sedangkan analisis regresi linear berganda adalah regresi dengan dua atau lebih variabel bebas (X), sehingga merupakan perluasan dari analisis regresi linear sederhana.

Penghitungan dengan metode PCA memerlukan suatu model analisis regresi yaitu Principal Component Regression (PCR). PCR merupakan salah satu metode yang telah dikembangkan untuk mengatasi masalah multikolinearitas.

PCR adalah analisis regresi variabel dependen terhadap komponen-komponen utama yang tidak saling berkorelasi, regresi komponen utama dapat dinyatakan sebagai berikut:

$$Y = w_0 + w_1K_1 + w_2K_2 + \dots + w_mK_m \quad (2.11)$$

$K_1, K_2, \dots, K_m$  merupakan komponen utama yang dilibatkan dalam analisis regresi komponen utama, dimana besaran  $m$  lebih kecil dari pada banyaknya variabel bebas yaitu sejumlah  $p$ ,  $w_0$  merupakan konstanta,  $w_1, w_2, \dots, w_m$  merupakan parameter regresi dan  $Y$  merupakan variabel dependen[7].

### 3. METODOLOGI

Data yang digunakan dalam penelitian ini adalah data sekunder. Data sekunder adalah data yang sudah ada dalam berbagai bentuk. Sumber datanya adalah data set mtcars bawaan pada R Studio yang berisi tentang spesifikasi kendaraan tahun 1970.

#### Langkah-Langkah Pembuatan Aplikasi Shiny Web

Adapun langkah-langkah pembuatan aplikasi shiny web adalah sebagai berikut:

- a. Membuat script R shiny web
  - 1) Membuat user interface
  - 2) Membuat server
- b. Mendeploy script R shiny web
- c. Mengupload ke halaman web
- d. Uji coba

Data yang digunakan dalam uji coba adalah data mtcars dengan urutan uji coba sebagai berikut:

- 1) Upload data
- 2) Menghitung komponen utama
- 3) Menampilkan plot visualisasi pca
- 4) Menentukan regresi pca

### 4. PEMBAHASAN

#### Membuat script R shiny web

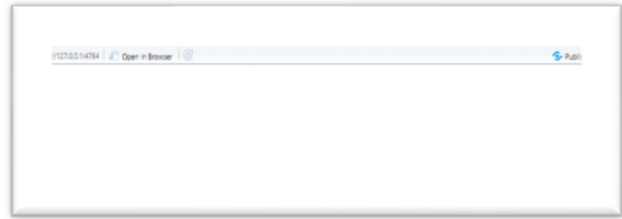
1. *User Interface* (UI) yaitu memuat seluruh input dan output yang akan ditampilkan dalam app. Dengan sintaks/perintah sebagai berikut:

```
#define UI
UI <- fluidPage(
)
```

2. *Server* yaitu menggabungkan antara kekuatan komputasi statistik R dan interaksinya dengan web modern. Dengan sintaks/perintah sebagai berikut:

```
server <- function(input, output, session) {
  par(mfrow = c(2,2))
}
```

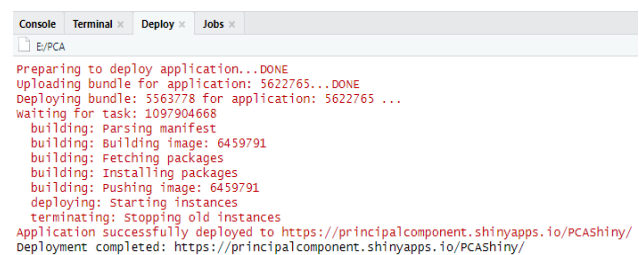
Berikut output mengenai sintaks/perintah yang diatas yaitu:



Gambar 4.1 output syntaks

#### Mendeploy Script R Shiny Web

Mendeploy adalah proses menyalin, mengkonfigurasi, dan mengaktifkan aplikasi tertentu ke URL dasar tertentu disebuah server atau *cluster*. Pada dasarnya bertujuan untuk mendorong perubahan atau pembaruan dari satu lingkungan penerapan ke lingkungan penerapan (*application environment*) lainnya. Adapun tampilan deploy script r shiny web adalah sebagai berikut:



Gambar 4.2 Tampilan deploy script r shiny web

#### Mengupload Ke Halaman Web

Proses upload atau unggah merupakan suatu cara atau proses yang di lakukan untuk melakukan pengiriman File atau data dari perangkat komputer atau perangkat lainnya yang memiliki koneksi internet. Server dan data tersebut akan di publikasikan jaringan internet sehingga sebuah file atau data yang telah di unggah dapat diambil atau bahkan dapat dilihat oleh orang banyak. Berikut linknya:

<https://principalcomponent.shinyapps.io/PCAShiny/>.

#### Melakukan Uji Coba

##### Deskripsi Data

Data yang dianalisis pada penelitian ini adalah data mtcars bawaan dari R- Studio yang berisi tentang spesifikasi kendaraan tahun 1970. Data yang digunakan dalam penelitian ini ditunjukkan pada Gambar 4.3 berikut:

X	mpg	cyl	disp	hp	drat	wt	qsec	vs	am	gear	carb
1 Mazda RX4	21	6	160	110	3.9	2.62	16.46	0	1	4	4
2 Mazda RX4 Wag	21	6	160	110	3.9	2.875	17.02	0	1	4	4
3 Datsun 710	22.8	4	108	93	3.85	2.32	18.61	1	1	4	1
4 Hornet 4 Drive	21.4	6	258	110	3.08	3.215	19.44	1	0	3	1
5 Hornet Sportabout	18.7	8	350	175	3.15	3.44	17.02	0	0	3	2
6 Valiant	18.1	6	225	105	2.76	3.46	20.22	1	0	3	1
7 Duster 360	14.3	8	360	245	3.21	3.57	15.84	0	0	3	4
8 Merc 240D	24.4	4	146.7	62	3.69	3.19	20	1	0	4	2
9 Merc 230	22.8	4	140.8	95	3.92	3.15	22.9	1	0	4	2
10 Merc 280	19.2	6	167.6	123	3.92	3.44	18.3	1	0	4	4

Gambar 4.3 Data penelitian

### Menghitung Komponen Utama

```
Calculate Principal Components

Standard deviations (1, ..., p11):
[1] 2.5706809 1.6280258 0.7919579 0.5192277 0.4727601 0.4599958 0.3677798 0.3505730
[9] 0.2757528 0.2281128 0.1484736

Rotation (n x k) = (11 x 11)

PC1 PC2 PC3 PC4 PC5 PC6 PC7
mpg -0.3625305 0.01612440 0.22574419 -0.02548255 -0.10284468 -0.10879743 -0.367723810
cyl 0.3739160 0.04374371 -0.17531118 -0.002591838 0.05848381 -0.16855369 0.057277736
disp 0.3681852 -0.04932411 -0.06148414 0.256607885 0.39399530 0.33616451 0.214303077
hp 0.3580569 0.24878402 0.14801476 -0.067676157 0.54004744 0.07145363 -0.001495089
drat -0.2941514 0.27469408 0.16118879 0.854828743 0.07732727 0.24449705 0.021119857
wt 0.3461033 -0.14303825 0.34181851 0.248589314 -0.07502912 -0.46493964 -0.020668302
qsec -0.2004563 -0.46337482 0.40316094 0.068076532 0.16466591 -0.33048032 0.050010522
vs -0.3065113 -0.23164699 -0.42881517 -0.214848616 0.59953955 0.19401702 -0.265780836
am -0.2349429 0.42941765 -0.20576657 -0.030462908 0.08978128 -0.57081745 -0.587305101
gear 0.2069162 -0.46234863 0.28977993 0.26469821 -0.04832960 0.24356284 0.605097617
carb 0.2140177 -0.41357186 0.52854459 -0.126789179 -0.36131875 0.18352168 -0.174603192

PC8 PC9 PC10 PC11
mpg 0.754091423 -0.235701617 -0.13928524 0.124895628
cyl 0.23802425 0.054035270 -0.04641949 0.144095441
disp 0.001142134 -0.198427848 0.04937979 0.66066481
hp -0.222358441 -0.575830072 0.24782351 -0.256492062
drat 0.032193501 -0.046901228 0.10149369 0.039530246
wt 0.008571929 0.359492521 -0.09439426 -0.567448697
qsec -0.231840021 -0.528377185 -0.27067295 0.181361780
vs 0.025935128 -0.358582624 0.15903909 0.008414634
am -0.059746952 -0.047403982 0.17778541 0.029823537
gear 0.336150240 -0.001735039 0.21382515 -0.053507885
carb -0.395629107 0.170640677 0.072235958 0.319594676
```

Gambar 4.3 Analisis PCA

Gambar diatas menampilkan hasil perhitungan dari komponen utama yang dijelaskann bahwa *standard deviations* menunjukkan akar dari nilai eigen yang terbentuk. Sedangkan hasil dari *rotation* terdapat 11 variabel yang secara umum akan menghasilkan 11 komponen yang dimana sesuai dengan nilai – nilai disetiap variabelnya yang ada pada tampilan diatas.

### Menghitung Rotasi dari Komponen Utama

```
Display Principal Components (Rotation)

PC1 PC2 PC3 PC4 PC5 PC6 PC7
mpg 0.3625305 -0.01612440 0.22574419 0.02548255 -0.10284468 0.10879743 -0.367723810
cyl 0.3739160 0.04374371 -0.17531118 -0.002591838 0.05848381 -0.16855369 0.057277736
disp 0.3681852 -0.04932411 -0.06148414 0.256607885 0.39399530 0.33616451 0.214303077
hp 0.3580569 0.24878402 0.14801476 -0.067676157 0.54004744 0.07145363 -0.001495089
drat -0.2941514 0.27469408 0.16118879 0.854828743 0.07732727 0.24449705 0.021119857
wt 0.3461033 0.14303825 0.34181851 0.248589314 0.07502912 0.46493964 0.020668302
qsec -0.2004563 -0.46337482 0.40316094 0.068076532 0.16466591 -0.33048032 0.050010522
vs -0.3065113 -0.23164699 -0.42881517 -0.214848616 0.59953955 0.19401702 0.265780836
am -0.2349429 0.42941765 -0.20576657 -0.030462908 0.08978128 -0.57081745 -0.587305101
gear 0.2069162 -0.46234863 0.28977993 0.26469821 -0.04832960 0.24356284 0.605097617
carb 0.2140177 -0.41357186 0.52854459 -0.126789179 0.36131875 0.18352168 0.174603192

PC8 PC9 PC10 PC11
mpg 0.754091423 -0.235701617 -0.13928524 0.124895628
cyl 0.23802425 0.054035270 -0.04641949 0.144095441
disp 0.001142134 -0.198427848 0.04937979 0.66066481
hp -0.222358441 -0.575830072 0.24782351 -0.256492062
drat 0.032193501 -0.046901228 0.10149369 0.039530246
wt 0.008571929 0.359492521 -0.09439426 -0.567448697
qsec -0.231840021 -0.528377185 -0.27067295 0.181361780
vs 0.025935128 -0.358582624 0.15903909 0.008414634
am -0.059746952 -0.047403982 0.17778541 0.029823537
gear 0.336150240 -0.001735039 0.21382515 -0.053507885
carb -0.395629107 0.170640677 0.072235958 0.319594676
```

Gambar 4.4 Rotasi Komponen utama

Gambar 4.4 diatas menampilkan hasil rotasi dari komponen utama yang terdapat 11 variabel dengan menghasilkan 11 komponen, yang dimana 11 komponen tersebut untuk memperlihatkan distribusi variabel yang lebih jelas.

### Menghitung Skor Cadangan dari Komponen Utama

```
Reserve_Sign_Scores

PC1 PC2 PC3 PC4 PC5 PC6 PC7
[1] 0.64686274 -1.7081142 0.5917309 -0.11370221 0.9455234 0.01698737 0.42648652
[2] 0.61948315 -1.5256219 0.3763813 -0.19912121 1.0166807 0.24172464 0.41620046
[3] 2.73562427 0.1441581 0.2374391 0.24521545 -0.3987623 0.34876781 0.60884146
[4] 0.30680683 2.3258038 0.1336213 0.50380035 -0.5492089 -0.01929700 0.04036075
[5] -1.94339268 0.7425211 1.1165366 -0.07446196 -0.2075157 -0.14519276 -0.38350816
[6] 0.05523342 2.7421229 -0.1612456 0.97516743 -0.2116654 0.24383585 0.29464160

PC8 PC9 PC10 PC11
[1] 0.009631217 -0.14642303 0.06670350 -0.17969357
[2] 0.084520213 -0.07452829 0.12692766 -0.08864426
[3] -0.585255765 0.13122859 -0.04573787 0.09463291
[4] 0.049583029 -0.22021812 0.06039981 -0.14761127
[5] -0.160297757 0.02117623 0.05983003 -0.14640690
[6] -0.256612420 0.03222907 0.20165466 -0.01954506
```

Gambar 4.5 Skor cadangan komponen utama

Gambar 4.5 diatas menampilkan hasil skor cadangan dari komponen utama yang terdapat 11 variabel dengan menghasilkan 11 komponen, yang dimana 11 komponen tersebut menampilkan 6 komponen yang dipilih sesuai dengan komponen terbaik.

### Menghitung Varians Total dari Komponen Utama

```
Find Variance Explained by Each Principal Component

[1] 0.600736559 0.240951627 0.057017934 0.024508858 0.020913737 0.019236011 0.012296544
[8] 0.011172858 0.007004241 0.004730495 0.002004037
```

Gambar 4.6 Variansi total dari komponen utama

Gambar 4.6 diatas menampilkan hasil varians total dari komponen utama yang terdapat 11 variabel dengan menghasilkan 11 komponen, yang dimana diperoleh nilai rata-rata di setiap 11 komponen pada gambar diatas.

### Menghitung Nilai Eigen dari Komponen Utama

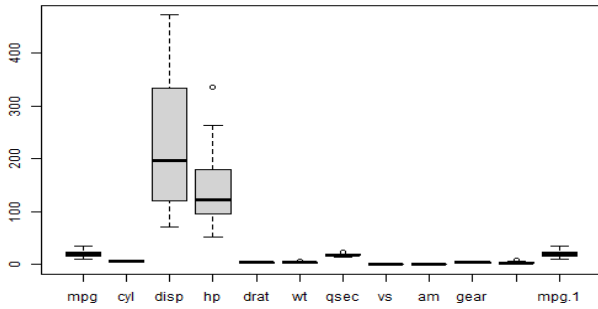
```
Eigen Value

eigenvalue variance.percent cumulative.variance.percent
Dim.1 6.60840025 60.0763659 60.07637
Dim.2 2.65046789 24.0951627 84.17153
Dim.3 0.62719727 5.7017934 89.87332
Dim.4 0.26959744 2.4508858 92.32421
Dim.5 0.22345110 2.0313737 94.35558
Dim.6 0.21159612 1.9236011 96.27918
Dim.7 0.13525159 1.2296544 97.50884
Dim.8 0.12208143 1.1172858 98.62612
Dim.9 0.07704665 0.7004241 99.32655
Dim.10 0.05203544 0.4730495 99.79960
Dim.11 0.02204441 0.2004037 100.00000
```

Gambar 4.7 Nilai eigen dari komponen utama

Gambar 4.7 menampilkan hasil nilai eigen dari komponen utama yang terdapat 11 variabel dengan menghasilkan 11 komponen, yang dimana dari ke 11 komponen nilai eigen yang dihasilkan pada gambar diatas menampilkan nilai varians yang dijelaskan oleh komponen utama.

### Menampilkan Boxplot dari Komponen Utama



Gambar 4.8 Boxplot komponen utama

Berdasarkan gambar 4.8 menampilkan boxplot dari komponen utama yang merupakan sebagai bentuk ringkasan dalam menentukan distribusi sampel yang disajikan secara grafis untuk menggambarkan bentuk distribusi data (*Skewness*) serta ukuran penyebaran (keragaman) dari data pengamatan. Boxplot meliputi beberapa ukuran statistik yang terdiri dari menentukan nilai maksimum, minimum, median, dan nilai Quartil 1 – Quartil 3.

**Menampilkan Variabel PCA**

```
Principal Component Analysis Results for variables
=====
Name      Description
1 "scoord"  "Coordinates for the variables"
2 "scor"   "Correlations between variables and
dimensions"
3 "scos2"  "Cos2 for the variables"
4 "scontrib" "contributions of the variables"
```

Gambar 4.9 Output pada Shiny Web

Output diatas menampilkan variabel PCA dalam aplikasi shiny web dengan tampilan name dan description antara lain coord, cor, cos2, dan contrib.

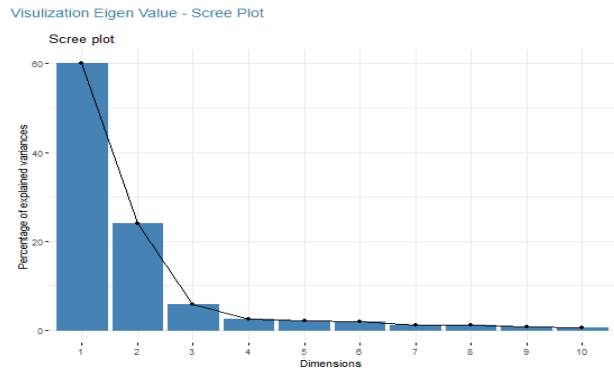
**Menampilkan Prediksi menggunakan PCA**

	PC1	PC2	PC3	PC4	PC5
[1,]	-0.6468627420	1.7081142	-0.5917309	0.113702214	-0.945523363
[2,]	-0.6194831460	1.5256219	-0.3763013	0.199121210	-1.016680740
[3,]	-2.7356242748	-0.1441591	-0.2374391	-0.245215450	0.398762288
[4,]	-0.3068606268	-2.3258038	-0.1336213	-0.503800355	0.549208936
[5,]	1.9433926844	-0.7425211	-1.1165366	0.074461963	0.207515698
[6,]	-0.0552534228	-2.7421229	0.1612456	-0.975167425	0.211665375
[7,]	2.9553851233	0.3296133	-0.3570461	-0.051529216	0.343847875
[8,]	-2.0229593244	-1.4421056	0.9290295	-0.142129082	-0.316651386
[9,]	-2.2513839535	-1.9522879	1.7689364	0.287210957	-0.333682355
[10,]	-0.5180912217	-0.1594610	1.4692603	0.066263362	-0.069624161
[11,]	-0.5011860079	-0.3187934	1.6570701	0.094357222	-0.148803650
[12,]	2.2124096339	-0.6727099	-0.3694707	-0.129797905	-0.378611141
[13,]	2.0155715693	-0.6724606	-0.4768341	-0.210991001	-0.355611763
[14,]	2.1147047372	-0.7891129	-0.2904620	-0.175332868	-0.432140303
[15,]	3.8383725118	-0.8149087	0.6370972	0.290505877	-0.048245223
[16,]	3.8918495626	-0.7218314	0.7092612	0.405336898	0.003899176
[17,]	3.5363862158	-0.4145024	0.5402468	0.665665306	0.208027112
[18,]	-3.7955510831	-0.2920783	-0.4161681	0.055191058	0.219981109
[19,]	-4.1870356784	0.6775721	-0.2035831	1.167526096	0.097674091
[20,]	-4.1675359344	-0.2748890	-0.4589124	0.183313028	0.222152228

Gambar 4.10 Hasil Prediksi menggunakan PCA

Gambar 4.10 menampilkan hasil prediksi menggunakan pca yang terdapat 11 variabel dengan menghasilkan 11 komponen dan 32 jumlah pengamatan, yang dimana 11 komponen tersebut menghasilkan data baru dengan berdasarkan masing – masing data yang dimasukkan dalam kombinasi linear yang terbentuk.

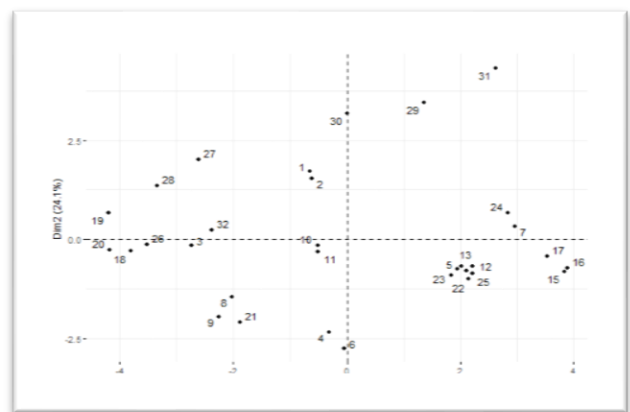
**Scree Plot dari Nilai Eigen**



Gambar 4.11 Plot nilai eigen

Gambar diatas menampilkan scree plot dari nilai eigen yaitu menggambarkan nilai dari eigenvalue tiap variabel ke dalam bentuk dimensi yang dimana dihitung berdasarkan konsep *dimensionality reduction* artinya untuk meminimalisir data menjadi minimum tanpa mengurangi data yang sebenarnya, dengan melihat nilai dari *persentase instances*.

**Scree Plot dari Individu PCA**



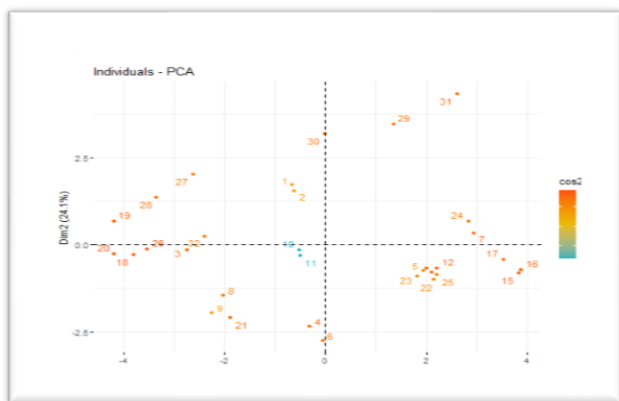
Gambar 4.12 Scree plot dari individu PCA

Berdasarkan gambar 4.10 dan gambar 4.11 scree plot individu pca bertujuan untuk mengidentifikasi salah satu titik, contohnya pada titik 24 plot individu PCA yang memiliki

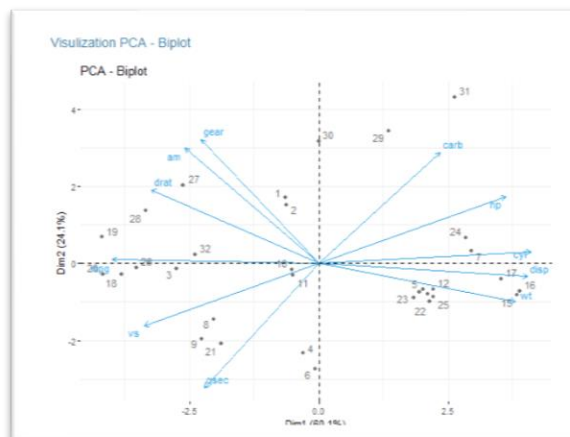


kecenderungan menjauh dari dimensi 1 tetapi mendekati pada dimensi

### Biplot dari PCA



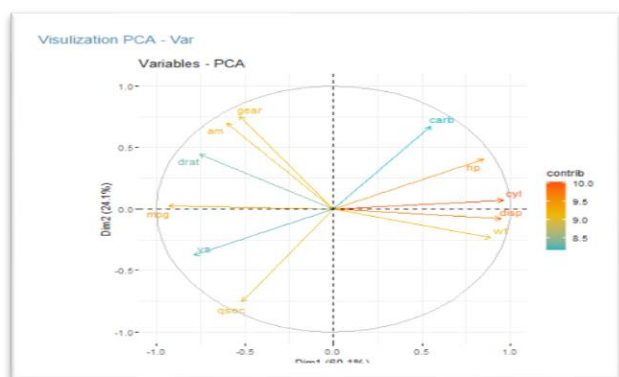
Gambar 4.13 Scree plot dari individu PCA



Gambar 4.14 Biplot dari PCA

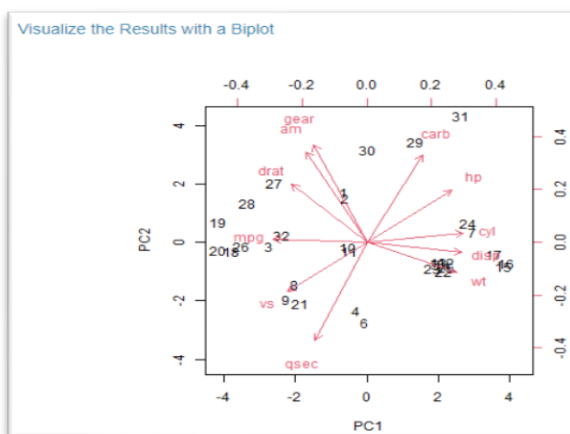
Berdasarkan gambar 4.12 dan gambar 4.13 *scree plot* individu *pca* bertujuan untuk mengidentifikasi salah satu titik, contohnya pada titik 24 plot individu PCA yang memiliki kecenderungan menjauh dari dimensi 1 tetapi mendekati pada dimensi 2. *Scree plot* dari individu *pca* yaitu menggambarkan nilai dari individu yang berdasarkan dari fungsi *cos2* untuk memperkirakan kualitas representasi tiap dimensi plotnya. Hanya saja yang membedakan pada kedua gambar tersebut yaitu gambar 4.10 (Tanpa Warna) sedangkan gambar 4.11 (menampilkan warna yang sesuai di tiap plotnya).

### Scree Plot dari Variabel PCA



Gambar 4.13 Scree plot dari individu PCA

Berdasarkan gambar 4.12 *scree plot* variabel *pca* dijelaskan bahwa menggambarkan nilai dari variabel dengan berdasarkan fungsi *contrib* sebagai kontribusi dengan PCA yang akan diklasterisasi variabelnya (sesuai warna dan arah panah pervariabel).



Gambar 4.15 Scree plot dari PCA

Berdasarkan gambar 4.14 dan 4.15 menampilkan biplot *pca* yang digunakan untuk menafsirkan suatu titik yang menggambarkan variabel dalam plot yang terkait dengan skala pada standar deviasi komponen utama dan akar kuadrat dari jumlah pengamatan. Perbedaan kedua dari gambar diatas terdapat pada tampilannya.

### Fungsi Headdata dari Regresi PCA

	X	mpg	cyl	disp	hp	drat	wt	qsec	vs	am	gear	carb
1	Mazda RX4	21.0	6	160	110	3.90	2.620	16.46	0	1	4	4
2	Mazda RX4 Wag	21.0	6	160	110	3.90	2.875	17.02	0	1	4	4
3	Datsun 710	22.8	4	108	93	3.85	2.320	18.61	1	1	4	1
4	Hornet 4 Drive	21.4	6	258	110	3.08	3.215	19.44	1	0	3	1
5	Hornet Sportabout	18.7	8	360	175	3.15	3.440	17.02	0	0	3	2
6	Valiant	18.1	6	225	105	2.76	3.460	20.22	1	0	3	1

Gambar 4.15 Fungsi headdata

Gambar diatas menampilkan fungsi *headdata* dengan menggunakan model *partial least squares* (PLS) yang dimana PLS yaitu suatu teknik statistik multivariat yang dapat menangani banyak variabel respon serta variabel



eksplanatori dengan *hp* sebagai variabel respon dan variabel *mpg*, *disp*, *drat*, *wt*, *qsec* sebagai variabel prediktor.

### Fungsi *Fitpcr* dari Regresi PCA

```
Data: X dimension: 32 5
      Y dimension: 32 1
Fit method: svdpc
Number of components considered: 5

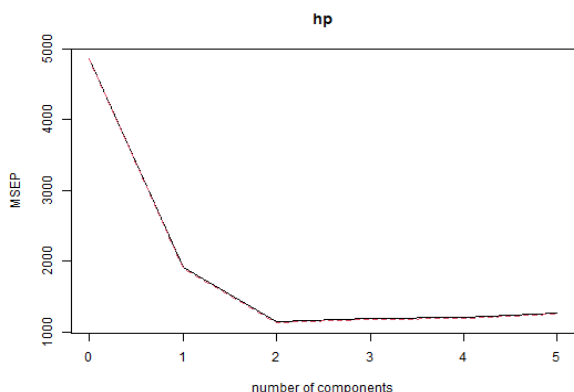
VALIDATION: RMSEP
Cross-validated using 10 random segments.
      (Intercept) 1 comps 2 comps 3 comps 4 comps 5 comps
CV          69.66  44.12  33.73  34.24  35.08  37.17
adjCV       69.66  44.00  33.53  33.96  34.76  36.73

TRAINING: % variance explained
      1 comps 2 comps 3 comps 4 comps 5 comps
X      69.83  89.35  95.88  98.96 100.00
hp     62.98  81.31  81.96  81.98  82.03
NULL
```

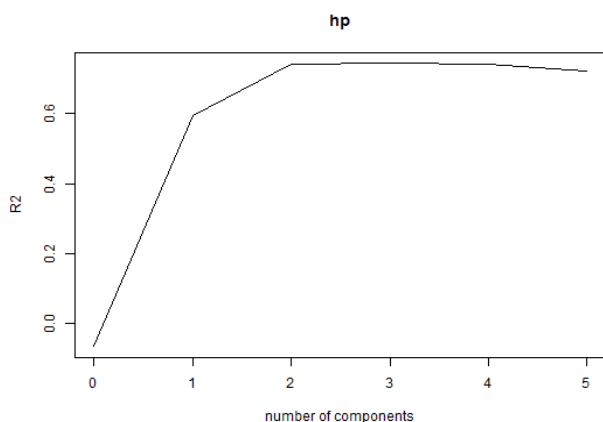
Gambar 4.15 Model Fit PCR

Gambar diatas menampilkan model untuk menentukan jumlah komponen *Partial Least Squares* (PLS) dengan cara melihat *Root Mean Squared Error Test* (*Test RMSE*) yang dihitung dengan *k-fold cross-validation*.

### Fungsi *Modelmse* dan *modelr2* dari Regresi PCA



Gambar 4.16 Model MSEP



Gambar 4.17 ModelR2

Berdasarkan gambar 4.16 dan 4.17 *modelmse* dan *modelr2* dijelaskan bahwa setiap plot dapat

dilihat bahwa model *fit* meningkat dengan menambahkan dua komponen PLS, namun cenderung menjadi lebih buruk ketika menambahkan lebih banyak komponen PLS. Dengan demikian, model optimal hanya mencakup dua komponen PLS pertama.

### Menampilkan Prediksi pada Set Pengujian

```
, , 2 comps
      hp
26 74.36372
27 107.02982
28 94.76093
29 196.32825
30 156.26694
31 204.54085
32 103.87799

[1] 123.5733
```

Gambar 4.18 Model Fit PCR

Gambar 4.18 menampilkan fungsi *predict* dan *predictr2* yang dimana fungsi *predict* untuk menentukan deviasi rata-rata antara nilai yang diprediksi untuk variabel *hp*, sedangkan fungsi *predictr2* untuk menentukan nilai yang diamati untuk variabel *hp* berdasarkan pengamatan pada set pengujian.

## 5. KESIMPULAN

Berdasarkan hasil pembahasan penelitian dalam penelitian ini, maka dapat diambil kesimpulan yaitu pembuatan *User Interface* (UI) dan *server* yang mendefinisikan file upload dan variabel input yang akan ditampilkan pada antar muka user serta pembuatan aplikasi yang berhasil menjalankan analisis menggunakan PCA dengan sampel dataset *mtcars*.

## 6. DAFTAR PUSTAKA

[1] Dunteman, George H. 1989. *Principal Components Analysis*. (Sage University Papers series: Quantitative Applications in the social sciences, No 07-069, Newbury Park, London, New Delhi).

[2] Nasution, Muhammad Zulfahmi. 2019. *Penerapan Principal Component Analysis (Pca) Dalam Penentuan Faktor Dominan Yang Mempengaruhi Prestasi Belajar*

*Siswa (Studi Kasus: Smk Raksana 2 Medan)*. (Jurnal Teknologi Informasi, Vol.3, No.1).

- [3] Nugroho, Sigit. 2008. *Statistik Multivariat Terapan*. Bengkulu: UNIB Press.
- [4] Rhamadhani, Muhd Humam. 2022. *Pengembangan Aplikasi Berbasis Web dengan R Shiny untuk Analisis Data Menggunakan Algoritma PCA*. (Journal Automata, Vol 3, No 1).
- [5] Maisarah dkk. 2021. *Pengembangan Aplikasi Web Interaktif Menggunakan Rshiny Untuk Analisis Statistik Nonparametrik*. (Buletin Ilmiah Math. Stat. dan Terapannya (Bimaster) Volume 10, No.3).
- [6] Tirta, I Made. 2014. *Pengembangan E-Modul Statistika Terintegrasi dan Dinamik dengan R-shiny dan mathJax*. (Prosiding Seminar Nasional Matematika, Universitas Jember, 19 November 2014).
- [7] Pendi. 2021. *Analisis Regresi Dengan Metode Komponen Utama Dalam Mengatasi Masalah Multikolinearitas*. (Buletin Ilmiah Math. Stat. dan Terapannya (Bimaster), Volume 10, No. 1, 2021).
- [8] *Shiny*, Retrieved Juny 8, 2021, from Shiny from R Studio: <https://shiny.rstudio.com/>.