Pengembangan Aplikasi Berbasis Web dengan R Shiny untuk Analisis Data Menggunakan Algoritma PCA

Muhd Humam Rhamadhani
Program Studi Informatika
Universitas Islam Indonesia
D.I. Yogyakarta, Indonesia
muhd.rhamadhani@students.uii.ac.id

Lizda Iswari
Program Studi Informatika
Universitas Islam Indonesia
D.I. Yogyakarta, Indonesia
lizda.iswari@uii.ac.id

***Abstrak*—Data menjadi kebutuhan yang sangat penting di zaman sekarang ini. Manusia pasti selalu membutuhkan data untuk memproses pekerjaannya menjadi lebih informatif dan deskriptif. Salah satu ilmu yang dapat memproses data tersebut adalah sains data. Tetapi yang terutama di dalam sains data, penggunaan algoritma *machine learning* menjadi pembeda dengan ilmu data yang lainnya. Disini penelitian akan menggunakan konsep sains data dalam membangun sebuah aplikasi berbasis *web* menggunakan R Shiny dimana penggunaan algoritma akan berfokus kepada salah satu *machine learning* yang disebut PCA. Penggunaan algoritma PCA ini akan diimplementasikan ke dalam aplikasi *web* sebagai bentuk analisis data dengan metode *dimensionality reduction*. Sehingga nantinya aplikasi ini dapat berguna bagi para pengguna yang ingin menganalisis sebuah data menggunakan *machine learning* PCA yang ingin mengetahui sebaran dari suatu data dengan algoritma tersebut menjadi informasi yang berguna.**

Kata Kunci*—*sains data, machine learning, PCA, analisis data, dimensionality reduction, R Shiny

I. PENDAHULUAN

Di era teknologi yang selalu berkembang setiap harinya, tidak henti-hentinya dunia selalu membutuhkan data dalam proses kerja kehidupan manusia di dalamnya. Data disini akan berperan sebagai hasil yang didapatkan untuk mendefinisikan suatu deskripsi dari sebuah objek atau kejadian [1]. Menurut Nuzulla Agustina, data adalah keterangan mengenai sesuatu hal yang sudah sering terjadi dan berupa himpunan fakta, angka, grafik, tabel, gambar, lambang, kata, huruf-huruf yang menyatakan sesuatu pemikiran, objek, serta kondisi dan situasi. Dengan adanya data, manusia diberikan kemudahan. Data-data ini akan berperan penting dan memiliki sebuah nilai yang bersifat faktual dikarenakan ini sudah dikumpulkan sebelumnya dengan berbagai metode yang sudah dilakukan oleh para pencari atau pengumpul data sesuai dengan kejadian, objek, dan peristiwa yang didapatkannya. Ketika sebuah data telah terkumpul dari berbagai macam objek ataupun kejadian yang telah diriset maka akan membentuk sebuah dataset. Dataset inilah yang akan menjadi cikal bakal dari pemrosesan suatu kesimpulan dari data yang telah diambil. Munculnya data ke dalam dunia teknologi dan informasi tentunya akan membuat pengolahan/pengumpulan data maupun dataset menjadi berkembang sesuai dengan beriringnya waktu. Salah satu yang menjadi faktor berkembangnya suatu data adalah munculnya sains data, dimana kategori data disini berfungsi kegiatan yang bersifat ilmiah.

Sains data adalah istilah dari gabungan dua kata yaitu "sains" dan "data" dimana dari dua kata tersebut menjelaskan suatu kegiatan ilmiah di sekitar dan berhubungan dengan data, mulai dari pengumpulan, pengolahan, hingga informasi yang dapat berguna nantinya sebagai pengambil keputusan dan dapat berguna pagi pihak yang membutuhkannya dengan data tersebut terkhusus bagi *data scientist* atau *data analyst* [2]. Mereka kemudian dapat menggunakan data-data hasil dari sains data tersebut menjadi ke dalam sebuah bentuk *machine learning*. *Machine learning* merupakan suatu bidang studi yang bisa dikatakan baru dan bersumber dari algoritma kuantum yang dinamakan *supervised* (terpandu) dan *unsupervised* (mandiri) [3]. Algoritma *supervised* adalah suatu pencarian algoritma yang bersumber dari contoh yang sudah tersedia (diambil secara eksternal) untuk menghasilkan suatu hipotesa yang mana hipotesis tersebut nantinya dapat membuat prediksi untuk kejadian masa depan yang biasanya akan dianalisis ke dalam bentuk klasifikasi sedangkan algoritma *unsupervised* adalah algoritma yang menyimpulkan fungsi dari beberapa contoh data *training* dan algoritma mencoba menemukan struktur yang tersembunyi di dalam data[4],[5].

Salah satu metode algoritma *machine learning* yang digunakan dalam penelitian ini adalah *Principal Component Analysis* (PCA) dimana ini merupakan salah satu bagian dari algoritma *unsupervised learning*. PCA merupakan algoritma dengan teknik multivariat yang menganalisis sebuah tabel data yang mana nantinya data tersebut akan menghasilkan beberapa sebaran variabel yang bersifat independen dengan sebaran data lainnya yang saling berkorelasi [6]. Dengan adanya PCA ini, penelitian ini akan mencoba membangun sebuah sistem aplikasi berbasis *web* dengan sebuah *tool* yang digunakan R Shiny. R Shiny merupakan aplikasi *web development* dengan basis bahasa pemrograman R dimana sangat sesuai untuk menganalisis dan memproses sebuah dataset dengan algoritma *machine learning* yang akan digunakan. Nantinya sistem dari aplikasi tersebut dapat memvisualisasikan dan memperhitungan dari hasil model dataset yang diinput oleh pengguna aplikasi menggunakan metode algoritma PCA.

Dari seluruh rincian yang sudah dijelaskan, penelitian ini akan berfokus untuk membangun sebuah aplikasi web untuk analisis data dengan algoritma PCA dengan alasan bahwa algoritma ini sangat mudah dalam memproses sebuah perhitungan data. Harapannya aplikasi dapat digunakan pengguna yang berasal dari kalangan apapun mulai dari yang masih belum mengetahui apa itu sains data hingga professional seperti *data scientist* dan *data analyst*. Aplikasi ini layak digunakan para pengguna ke dalam bidang seperti bisnis, sosial, manufaktur, dan lainnya dalam upaya

mengetahui penyebaran *instance* dari suatu data yang diinput seperti apa dan dapat menghasilkan suatu nilai ataupun kesimpulan yang didapatkan dengan menggunakan perhitungan metode algoritma PCA ini.

II. TINJAUAN PUSTAKA

A. *Principal Component Analysis*

Principal Component Analysis atau disebut PCA adalah salah satu machine learning yang dikategorikan ke dalam algoritma unsupervised learning. PCA sendiri merupakan algoritma yang memiliki banyak cara membentuk dasar untuk analisis data yang bersifat multivariat [7]. Algoritma PCA juga memiliki beberapa proses di dalam menganalisis sebuah data antara lain :

1) *Modeling*
2) *Simplification*
3) *Dimensionality Reduction*

Modelling menjadi perhitungan dasar pada penggunaan algoritma ini. Analisis data akan dimulai dari perhitungan *modelling* dari data yang akan diambil. Lalu langkah simplifikasi ini akan menganalisis dari komponen tiap *instance* suatu data. Terakhir metode *dimensionality reduction* menjadi metode akhir dan kunci dari analisis PCA ini yang akan disajikan ke dalam bentuk *plot* dalam aplikasi ini.

B. *Bahasa Pemrograman R*

R merupakan salah satu bahasa pemrograman yang menjadi sarana pengolahan analisis data yang interaktif. Terlebih bahasa ini berkembang dengan cepat dan telah memiliki banyak *packages*. Namun R merupakan bahasa yang rata-rata berumur pendek dan dikembangkan untuk satu tujuan analisis yang digunakan seperti dengan penelitian ini [8].

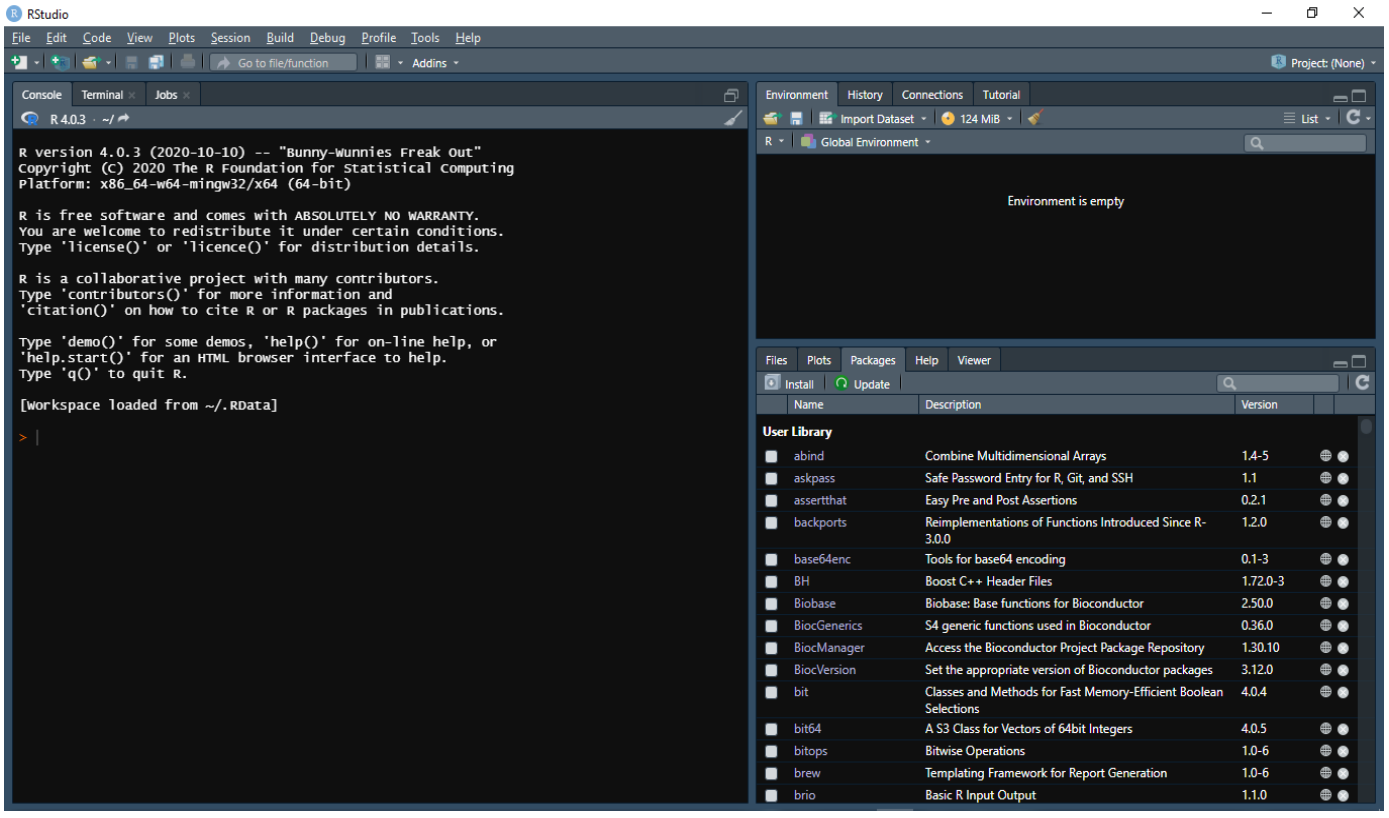
C. *R Shiny*

Shiny merupakan suatu *framework* aplikasi pengembangan *web* yang mana kebutuhan *environment* aplikasi tersebut menggunakan bahasa pemrograman R yang dikembangkan oleh RStudio [9]. Shiny merupakan aplikasi yang mudah dan reliabel penggunaanya sebagai pengembangan *web* karena langsung terintegrasi dengan bahasa R. Dengan aplikasi ini, sistem yang dibangun diharapkan akan menjadi lebih dinamis dan mudah digunakan kedepannya bagi para *user*.

D. *Plot*

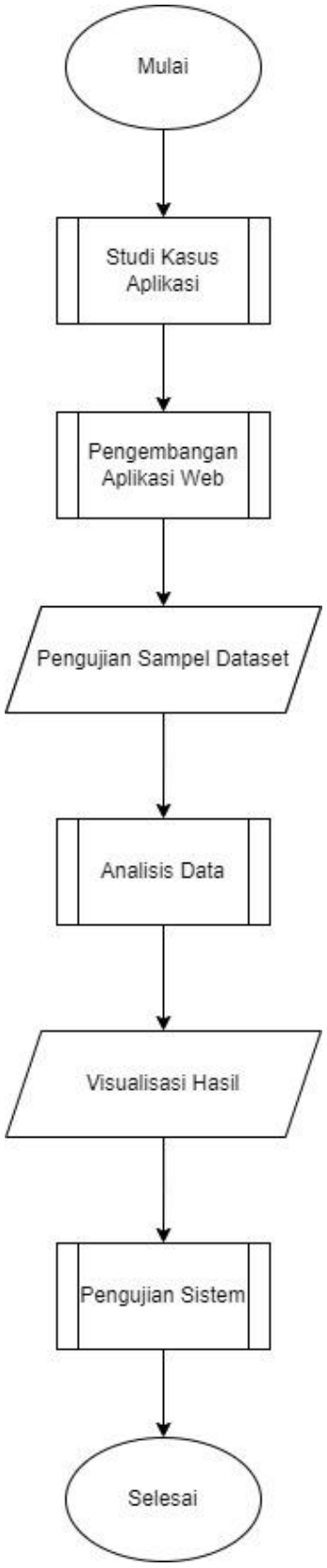
Visualisasi pada aplikasi ini akan disajikan dalam bentuk *plotting*. *Plot* ini berasal dari *packages* pada R Studio yang telah tersedia yaitu “factoextra”. Factoextra merupakan *packages* yang menyediakan beberapa fungsi yang mudah digunakan dalam memvisualisasikan data yang bersifat multivariat [10]. PCA sendiri merupakan algoritma yang bersifat multivariat seperti yang sudah dijelaskan pada bab pendahuluan.

Penggunaan plot pada *packages* ini akan menggunakan 6 plot yang nantinya akan disajikan sebagai informasi visualisasi dari hasil analisis data menggunakan algoritma PCA. Enam plot tersebut antara lain *scree plot*, *cos2 plot*, *individuals plot*, *contrib plot*, *variables plot*, dan *biplot*.



Gambar 1. Aplikasi RStudio

III. METODE PERANCANGAN SISTEM



Gambar 2. *Flowchart* perancangan sistem

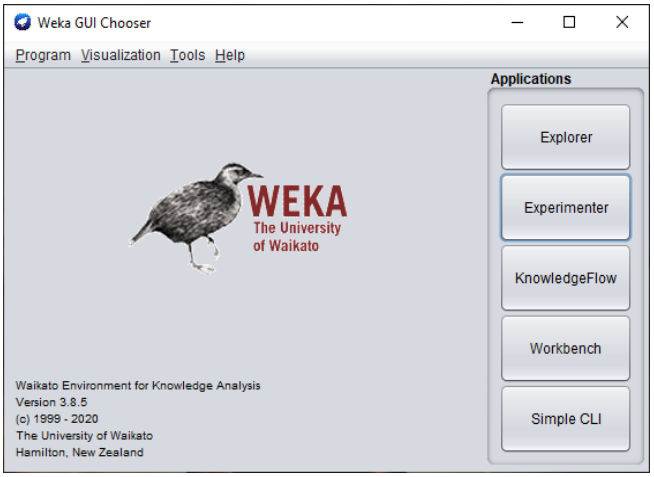
Pada gambar 2 merepresentasikan langkah-langkah yang akan dibentuk dalam merancang sebuah sistem hingga menjadi produk sebuah aplikasi *web* menggunakan R Shiny.

Langkah-langkah dari *flowchart* yang bersifat kredensial berawal dari langkah pengujian sampel dataset hingga pengujian sistem. Jika proses tersebut berjalan sesuai alur maka secara otomatis pengembangan aplikasi juga akan bekerja secara baik nantinya.

A. Studi Kasus Aplikasi

Disini perancangan sistem akan dimulai dengan mempelajari dan membandingkan aplikasi sejenis dengan basis sains data dan *machine learning* yang sudah ada untuk digunakan secara publik.

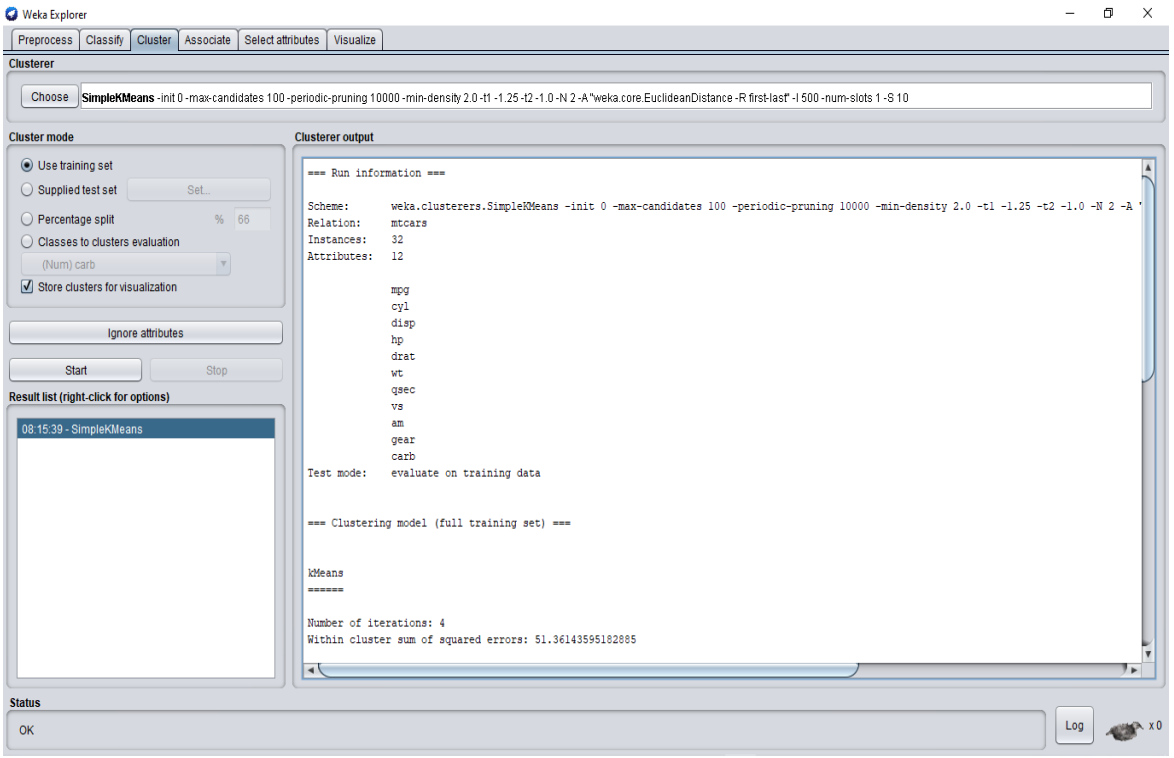
Perbandingan untuk studi kasus aplikasi ini hanya menggunakan salah satu aplikasi dengan fungsi aplikasi yang berhubungan dengan data, yaitu WEKA. Waikato Environment for Knowledge Analysis atau yang disingkat WEKA adalah aplikasi analisis untuk kebutuhan *data mining* tentunya dengan menggunakan algoritma *machine learning* yang juga terdapat di dalamnya. Para pengguna yang menggunakan aplikasi ini bertujuan untuk menggali informasi terhadap dataset yang dimilikinya. Banyak pilihan algoritma model yang ingin dijalankan baik dengan algoritma *supervised* maupun *unsupervised*. Tetapi sesuai dengan kebutuhan penelitian disini, studi kasus akan menggali algoritma *machine learning* menggunakan di aplikasi ini yang bertujuan sebagai *data clustering*.



Gambar 3. Aplikasi Weka

Gambar 3 menjelaskan aplikasi WEKA yang masih dalam bentuk GUI. Studi kasus untuk aplikasi WEKA disini memilih pilihan "explorer" karena hanya disini pilihan yang bisa memilih metode klasterisasi pada suatu dataset.

Dalam aplikasi ini tidak ada algoritma PCA sebagai algoritma untuk analisis datanya. Namun disini pilihan yang mendekati untuk klasterisasi data menggunakan algoritma K-Means. PCA dan K-Means merupakan algoritma yang identik karena berada pada tipe *unsupervised learning* [11]. Hal inilah yang menjadi alasan studi kasus aplikasi untuk perbandingan klasterisasi bisa dibandingkan dengan algoritma PCA. Dan berikut hasil dari klasterisasi dari dataset yang digunakan yaitu mtcars dimana ini merupakan kumpulan 32 *data entry* dari merk mobil dengan format dataset csv menggunakan algoritma K-Means.



Gambar 4. WEKA Explorer

B. Pengembangan Aplikasi Web

Proses selanjutnya setelah melakukan studi kasus pada aplikasi yang sejenis, maka penelitian dilanjutkan dengan membangun sebuah aplikasi *web* menggunakan R Shiny. Pada pembuatan *web* ini, sistem membutuhkan *script code* dengan format R sebanyak 3 jenis *script*. Penjelasan tiap *script* akan dijelaskan sebagai berikut :

1. *UI*

   Sesuai dengan namanya, pada *script* ini menjadi penunjang utama *user interface* (UI) untuk tampilan *web* dari sistem aplikasi ini. Penggunaan bahasa dalam *script* ini tidak hanya menggunakan bahasa R tapi juga dengan HTML. Di dalam *script* ini terdapat tema untuk kustomisasi tampilan yang diinginkan, gambar, dan judul sebagai profil tampilan di aplikasi nantinya.
2. *Shiny*

   Untuk shiny sendiri merupakan isi seluruh *pseudocode* PCA dari awal hingga akhir termasuk *page* yang digunakan dan *library* yang harus disediakan. Sehingga isi dari *script code* disini menjadi penentu untuk isi tampilan dari aplikasi ini.
3. *Server*

   Server disini berfungsi sebagai konektor antara UI dan shiny untuk aplikasi *web*-nya. *Script* ini membuat UI menjadi *source* yang harus ditampilkan.

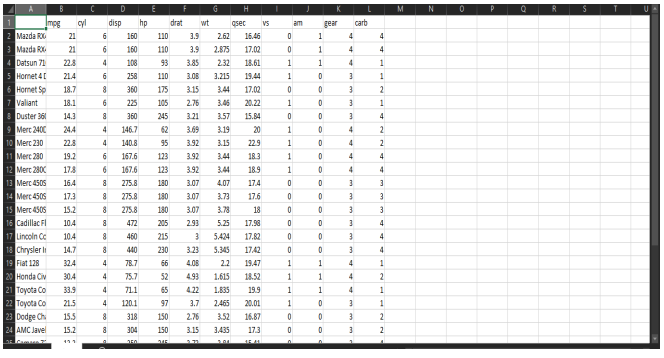
Untuk *pseudocode* shiny bukan menjadi opsi utama untuk penamannya melainkan bebas dengan pilihan masing-masing. Di dalam shiny terdapat isi *pseudocode* dari perhitungan algoritma PCA, analisis, hingga visualisasi data yang menjadi *script* utama dalam pengembangan aplikasi ini.

C. Pengujian Sampel Dataset

Sampel dataset yang digunakan disini bersumber dari *R dataset packages factoextra* yang sudah tersedia di RStudio. Namun dikarenakan dataset yang digunakan disini harus diinput secara manual ke aplikasi *web* dengan *soft file* yang tersedia (menggunakan format .csv) maka sampel yang digunakan disini di-*download* terlebih dahulu yang didapatkan dari *public code* pada Github. Untuk sampelnya sendiri masih sama digunakan seperti pada studi kasus aplikasi sebelumnya yaitu dataset dari mtcars yang berisikan 32 tipe merk mobil dengan 11 variabel di dalamnya.

D. Analisis Data

Tahap ini menjadi kunci utama dalam pengembangan aplikasi *web* ini. Dalam proses ini akan melakukan pembuatan *code* untuk perhitungan dari algoritma PCA di dalam aplikasi menggunakan bahasa R. Untuk perhitungannya akan menggunakan *data model*, *summary,* dan *predict* PCA dari datasetnya.



	mpg	cyl	disp	hp	drat	wt	qsec	vs	am	gear	carb
Mazda RX4	21	6	160	110	3.9	2.62	16.46	0	1	4	4
Mazda RX4	21	6	160	110	3.9	2.875	17.02	0	1	4	4
Datsun 71	22.8	4	108	93	3.85	2.32	18.61	1	1	4	1
Hornet 4 D	21.4	6	258	110	3.08	3.215	19.44	1	0	3	1
Hornet Sp	18.7	8	360	175	3.15	3.44	17.02	0	0	3	2
Valiant	18.1	6	225	105	2.76	3.46	20.22	1	0	3	1
Duster 36(	14.3	8	360	245	3.21	3.57	15.84	0	0	3	4
Merc 240(	24.4	4	146.7	62	3.69	3.19	20	1	0	4	2
Merc 230	22.8	4	140.8	95	3.92	3.15	22.9	1	0	4	2
Merc 280	19.2	6	167.6	123	3.92	3.44	18.3	1	0	4	4
Merc 280(	17.8	6	167.6	123	3.92	3.44	18.9	1	0	4	4
Merc 450S	16.4	8	275.8	180	3.07	4.07	17.4	0	0	3	3
Merc 450S	17.3	8	275.8	180	3.07	3.73	17.6	0	0	3	3
Merc 450S	15.2	8	275.8	180	3.07	3.78	18	0	0	3	3
Cadillac Fl	10.4	8	472	205	2.93	5.25	17.98	0	0	3	4
Lincoln Cc	10.4	8	460	215	3	5.424	17.82	0	0	3	4
Chrysler Ir	14.7	8	440	230	3.23	5.345	17.42	0	0	3	4
Fiat 128	32.4	4	78.7	66	4.08	2.2	19.47	1	1	4	1
Honda Civ	30.4	4	75.7	52	4.93	1.615	18.52	1	1	4	2
Toyota Co	33.9	4	71.1	65	4.22	1.835	19.9	1	1	4	1
Toyota Co	21.5	4	120.1	97	3.7	2.465	20.01	1	0	3	1
Dodge Ch	15.5	8	318	150	2.76	3.52	16.87	0	0	3	2
AMC Javel	15.2	8	304	150	3.15	3.435	17.3	0	0	3	2

Gambar 5. Dataset mtcars

Gambar 5 menunjukkan bahwa nama menjadi baris utama yang akan ditampilkan nantinya ke dalam plot visualisasi. Sebelas variabel lainnya menjadi perhitungan untuk analisis datanya ke dalam *data model* di dalam datasetnya.

E. Visualisasi Hasil

Setelah melakukan perhitungan dan analisis, maka akan dilakukan penyebaran datanya dengan cara memvisualisasikan hasilnya menggunakan teknik *plotting*. Disini visualisasi menggunakan enam plot yaitu *scree plot*, *cos2 plot, individuals plot, contrib plot, variables plot,* dan *biplot* yang dihasilkan dari *data model*.

F. Pengujian Sistem

Proses terakhir dari rancangan penelitian ini adalah pengujian sistem. Pengujian sistem aplikasi ini harus dilakukan berulang kali hingga sempurna dari langkah pengujian sampel dataset hingga visualisasi hasilnya. Jika semua proses tersebut berjalan dengan lancar tanpa maka pengujian sistem aplikasi dinyatakan berhasil dan dapat berlanjut ke arah pembuatan tahapan implementasi aplikasinya.

IV. IMPLEMENTASI

A. Tampilan Aplikasi

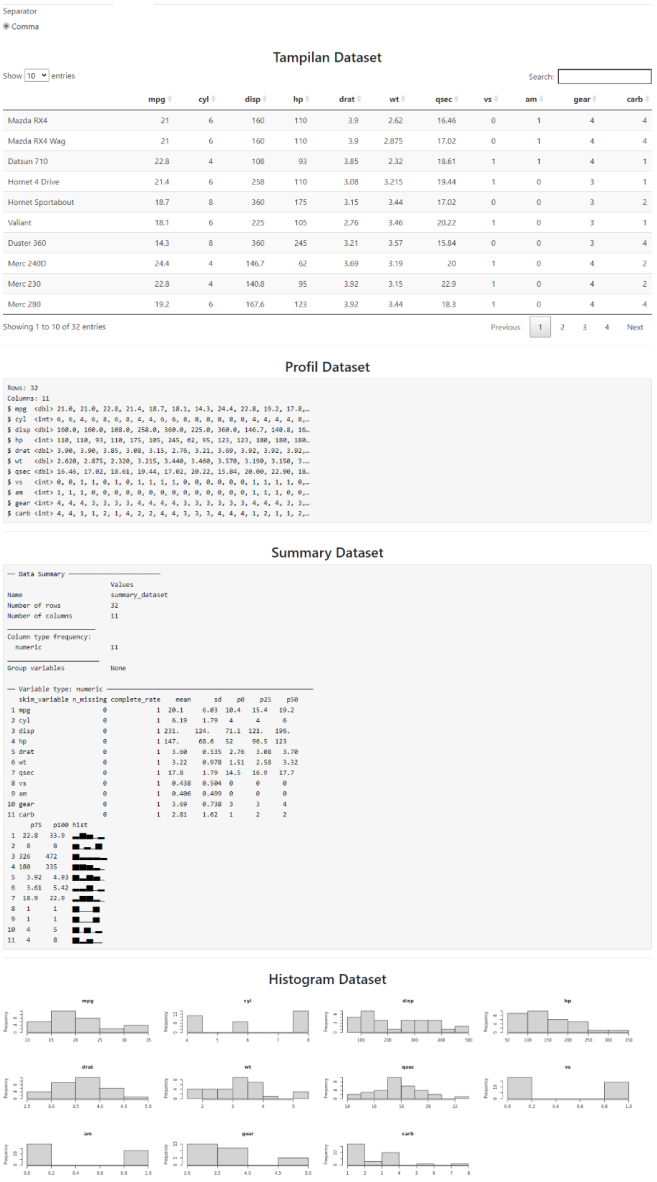
Tampilan aplikasi ini akan menjelaskan keseluruhan fitur yang bisa digunakan oleh *user* dan judul dari aplikasi sebagai penjelasan fungsi dari aplikasi *web* ini. Aplikasi ini juga sudah di-*hosting* menggunakan shinyapps. Tidak lupa pilihan tema yang disajikan juga beragam sehingga dapat menyesuaikan pilihan tampilan tema yang diinginkan oleh pengguna aplikasi menggunakan *theme customizer*.

Gambar 6. Tampilan dasar aplikasi

B. Dataset

Penyajian dataset akan disajikan dengan pemisah *comma* sebagai fungsi penyajian dataset dalam bentuk tabel (*dataframe*). Dari bagian pada aplikasi ini akan disuguhkan tampilan dari dataset, profil dari atribut tiap dataset, *summary* dataset yang menjelaskan nilai statistik dari dataset, dan terakhir histogram dataset yang menggambarkan nilai dari atribut ke dalam visual bentuk histogram (diagram batang).

Profil Dataset akan merincikan bentuk datasetnya apakah numerik atau string. Statistik dari dataset ini juga menjadi profil dataset. Pada gambar 7 terlihat terdapat tampilan dataset, profil dataset, hingga histogram dataset untuk nilai tiap variabelnya.



Gambar 7. Deskripsi dataset mtcars

Sebelum menggunakan profil dataset ini, di dalam R dikhususkan untuk membutuhkan tiga *library* yang harus tersedia *dplyr*, *skimr*, dan *visdat*.

Library "dplyr" akan menjabarkan profil dataset yang (bentuk tipe data tiap variabel). Lalu *library "skimr"* menjelaskan nilai statistik dari variabel dataset yang digunakan. Terakhir, *"visdat"* sebagai *library* yang membuat fungsi histogram pada variabel dataset.

Terdapat syarat untuk dataset yang dapat digunakan untuk perhitungan algoritma PCA sebagai berikut :

1. Dataset memiliki nilai yang bersifat multivariat.
2. Isi dari kolom data yang diperhitungkan tidak membentuk suatu grup kecil atau bisa dikatakan bervariasi satu sama lain (tidak duplikat). Seperti contoh di atas menggunakan dataset mtcars, nama mobil yang akan dianalisis tidak boleh ada yang sama.
3. Kolom variabel tidak boleh ada yang bersifat string dikarenakan perhitungan ini merupakan perhitungan analisis yang bersifat numerik (*unsupervised*).

4. Nilai dari baris utama tidak boleh bernilai *null*. Dengan contoh dataset di atas, nama mobil tidak boleh ada yang *null* walaupun tetap memiliki nilai tiap variabelnya.

C. Perhitungan PCA

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

1. *Data Model* PCA
2. *Summary* PCA
3. *Predict* PCA
4. *Eigenvalue* PCA
5. *Coord* PCA
6. *Cos 2* PCA
7. *Contrib* PCA

Model PCA menjadi tolak ukur perhitungan dari algoritma PCA kedepannya. Nantinya ini menjadi perhitungan utama ke dalam algoritma PCA (*summary, 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 RStudio.

Model PCA juga menjadi dasar dari algoritma PCA dimana teknik yang digunakan dalam perhitungannya berdasarkan metode dari *dimensionality reduction* pada tiap nilai komponen variabel yang nantinya dalam dimensi tersebut akan mewakilkan dari daerah dari hasil analisisnya.

```
output$hasil_modelPCA <- DT::renderDT(
  {
    hasil_modelPCA <- dataset()

    pcaModel <- prcomp(hasil_modelPCA, scale. = TRUE,
center = TRUE)
    pcaModel$rotation
  })
```

Gambar 8. *Pseudocode* model PCA

Pada gambar di atas menunjukkan *pseudocode* dari penggunaan perhitungan model PCA di dalam R. 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 mewakilkan sebaran *instance* data lewat nilai dari *eigenvalues*.

Selanjutnya akan dilakukan proses *summary* dari *model* PCA yang sudah diperhitungkan. Disini *summary* PCA menjelaskan nilai statistik varians yang didapatkan dari nilai *model* seperti *standard deviation*, *proportion of variance*, dan *cumulative proportion*.

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.

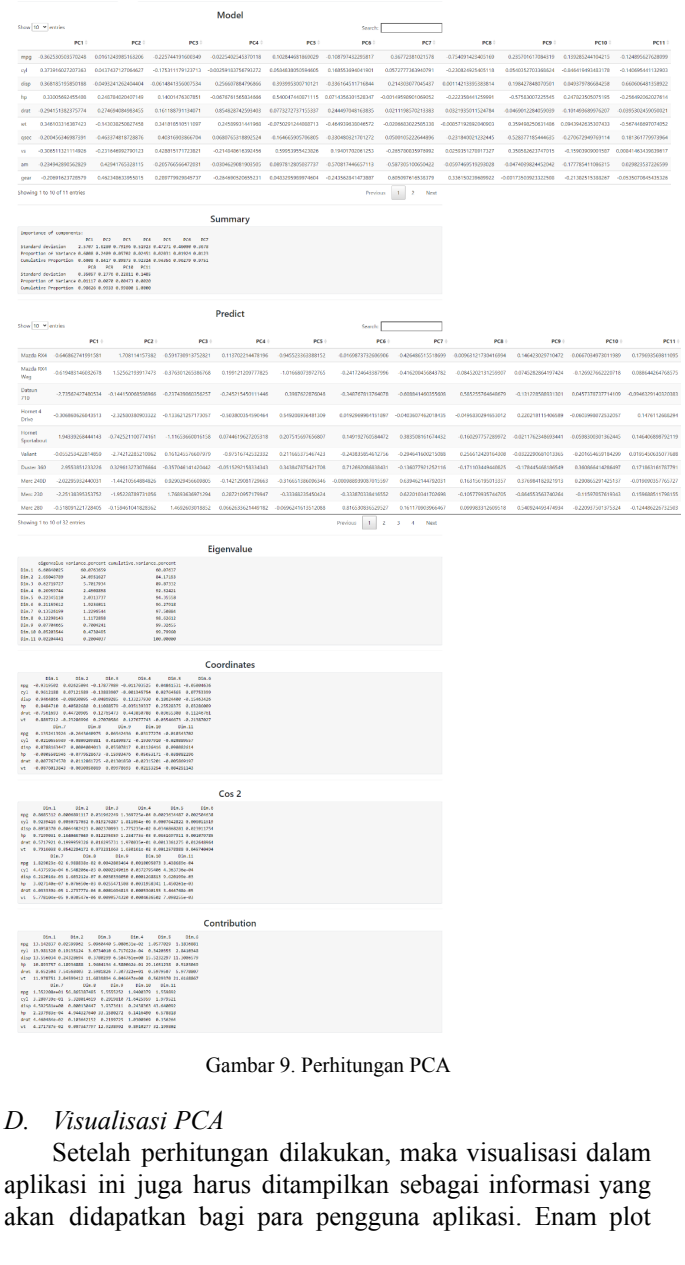
Terakhir, terdapat perhitungan *eigenvalues* yang menjadi dasar utama dalam perhitungan dimensi varians yang akan merujuk ke sebaran *instance* data untuk tiap dimensinya, *coord* yang menjelaskan koordinat dari tiap variabel, *cos 2* sebagai kualitas representasi nilai untuk variabel di dalam grafik 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 :

$$var.coord^2$$

sedangkan nilai *contrib* didapatkan dari :

$$\frac{var.cos2\ x\ 100}{total.cos2}$$



Gambar 9. Perhitungan PCA

D. Visualisasi PCA

Setelah perhitungan dilakukan, maka visualisasi dalam aplikasi ini juga harus ditampilkan sebagai informasi yang akan didapatkan bagi para pengguna aplikasi. Enam plot

yang disajikan akan mewakilkan informasi dari analisis data menggunakan algoritma PCA dari perhitungan *model*.

Enam plot yang akan divisualisasikan ke dalam aplikasi ini yaitu :

1. *Scree Plot*
2. *Cos2 Plot*
3. *Individuals Plot*
4. *Contrib Plot*
5. *Variables Plot*
6. *Biplot*

Plot-plot ini akan mewakilkan klasterisasi setelah perhitungan PCA yang kompleks. Para pengguna aplikasi ini akan disuguhkan visualisasi ini tanpa harus mengetahui perhitungan tersebut secara keseluruhan. Sehingga para pengguna akan tetap dapat menarik informasi ini menjadi sebuah kesimpulan apa yang didapatkan dari dataset yang digunakan.

Langkah terakhir setelah perhitungan analisis PCA selesai maka akan dilakukan klasterisasi data yang akan divisualisasikan menggunakan plot sebagai informasi yang akan didapatkan. Di dalam R, dibutuhkan dua *packages* sebagai fungsi menampilkan plot dan visualisasi dengan algoritma PCA yaitu *factoextra, corrplot,* dan *ggplot*. Untuk pembahasan tiap plot akan dijelaskan sebagai berikut.

1. *Scree plot* : menggambarkan nilai dari *eigenvalue* tiap variabel ke dalam bentuk dimensi. Dimensi ini dihitung berdasarkan konsep *dimensionality reduction*. Nantinya akan diteruskan menjadi jumlah dari persentase *instances* yang akan teranalisis.
2. *Cos2 plot* : menggambarkan nilai dari *cos2.*
3. *Individuals plot* : menggambarkan nilai dari individu yang dijelaskan dari fungsi *cos2* digunakan untuk memperkirakan kualitas representasi tiap dimensinya plotnya. (dijelaskan sesuai warna dari tiap plot).
4. *Contrib plot* : menggambarkan nilai dari *contrib*
5. *Variables plot* : menggambarkan nilai dari variabel yang dijelaskan dari fungsi *contrib* sebagai kontribusi dengan PCA yang akan diklasterisasi variabelnya. (dijelaskan sesuai warna dan arah panah).
6. *Biplot* : menggambarkan gabungan dari *individuals plot* dan *variables plot* (dijelaskan dengan arah panah yang mengarah ke plot). Hasil klasterisasi data dapat terlihat jelas disini karena sudah kompleks dimana tiap data sudah diklasterisasi berdasarkan dimensi pada plot.

Biplot menjadi informasi utama dalam visualisasi ini sebagai bentuk klasterisasi datanya dari tiap individu dan variabel membentuk sebuah klaster-klaster. Informasi dari visualisasi data dijelaskan secara terurut dan informasi singkat juga dijelaskan di bawah plot sehingga para pengguna dapat menarik kesimpulan secara baik dan benar dari informasi *plot* yang didapatkan.

Pada gambar 10 di bawah, akan dipaparkan isi dari *pseudocode* bentuk plot utama yaitu Biplot yang menggambarkan bentuk akhir sebagai analisis pada aplikasi ini menggunakan algoritma PCA.

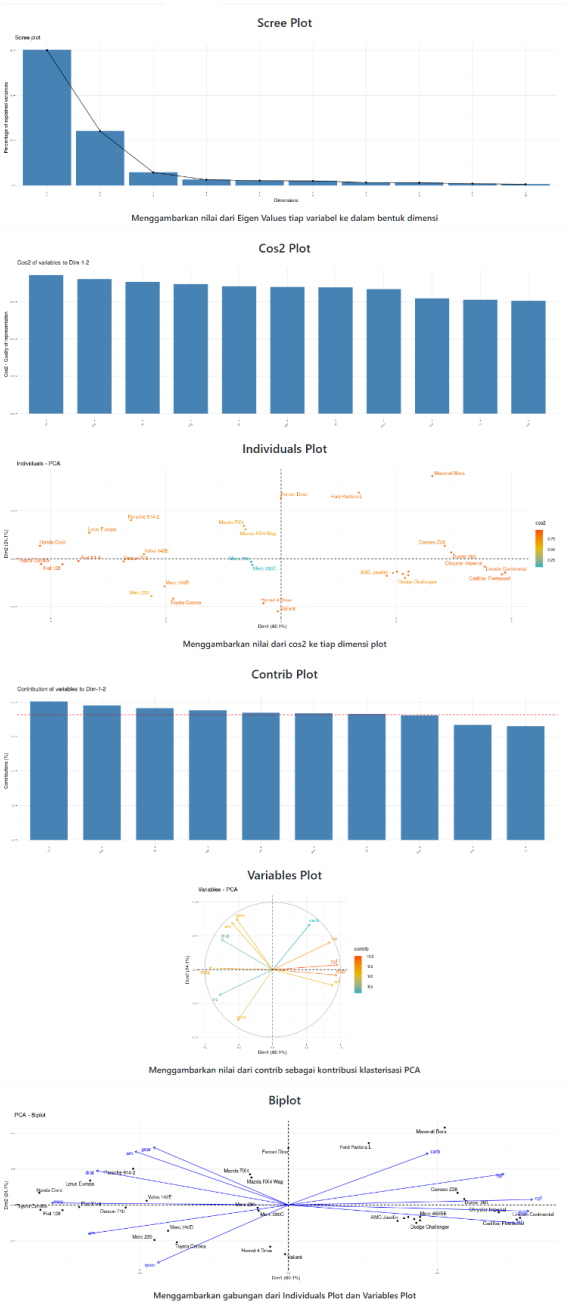
```
output$biplot_PCA <- renderPlot(
  {
    biplot_PCA <- dataset()

    modelPCA <- prcomp(biplot_PCA, scale. = TRUE, center
= TRUE)
    modelPCA$rotation

    fviz_eig(modelPCA)

    fviz_pca_biplot(modelPCA, repel = TRUE,
                    col.var = "blue",
                    col.ind = "#000000")
  })
```

Gambar 10. *Pseudocode* Biplot



Gambar 11. Visualisasi PCA

V. KOMPARASI HASIL

Insight yang didapatkan dari aplikasi *web* ini akan dibandingkan dari studi kasus sebelumnya menggunakan aplikasi WEKA menggunakan algoritma K-Means dan aplikasi dari penelitian ini yang dikembangkan menggunakan algoritma PCA. Perbandingan akan disajikan dalam bentuk tabel 1 di bawah ini.

TABEL 1 PERBANDINGAN APLIKASI

	Penelitian ini	WEKA
Bentuk Aplikasi	*Web*	*Desktop*
Machine Learning	PCA	K-Means
Metode	*Dimensionality Reduction*	*Clustering*
Preprocessing Dataset	*Modelling*	*Training*
Instance Dataset	*Variance*	*Cluster*
Format Dataset	.txt, .csv	.arff, .arff.gz, .names, .data, .csv, .json, .json.gz

```
=== Model and evaluation on training set ===

Clustered Instances

0       12 ( 38%)
1        7 ( 22%)
2        8 ( 25%)
3        5 ( 16%)
```

Gambar 12. Klasterisasi K-Means menggunakan aplikasi WEKA

	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	0.13526199	1.2296544	97.50884
Dim.8	0.12290143	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 13. Analisis PCA menggunakan aplikasi dari penelitian ini

Dari hasil komparasi yang telah dilakukan seperti pada gambar 12 dan 13 dapat didapatkan informasi dari dataset yang telah diinput bahwa hasil klasterisasi dari aplikasi WEKA menggunakan algoritma K-Means dan algoritma PCA untuk aplikasi dalam penelitian ini mendapatkan hasil yang identik karena memaparkan jumlah persentase yang telah diperhitungkan dengan metode masing-masing pada nilai tiap atribut pada *instance* data yang dimasukkan. Selain itu bentuk dataset yang dapat digunakan baik pada aplikasi WEKA untuk algoritma K-Means dan aplikasi *web* ini untuk algoritma harus menggunakan dataset yang bersifat numerik. Namun pada aplikasi WEKA jika terdapat dataset yang bersifat *string* atau kategorikal maka akan secara otomatis di filter atribut dari dataset tersebut [12].

VI. KESIMPULAN

Dari penelitian dan sistem aplikasi *web* yang telah dibangun, maka terdapat kesimpulan sebagai berikut :

1. Aplikasi dapat menjalankan analisis menggunakan algoritma PCA dari sampel dataset yang sesuai untuk digunakan.
2. Hasil perhitungan dari algoritma PCA sangat berpengaruh untuk bisa atau tidaknya diproses pada jenis dataset yang akan digunakan (numerik).
3. Visualisasi data merupakan hasil informasi dari analisis data yang diproses menggunakan algoritma PCA.

DAFTAR PUSTAKA

[1] F. Irmansyah, “Pengantar Database”, 2003, bll 1–13.

[2] M. K. M. Nasution, “Sains Data”, University of Sumatera Utara, 2019.

[3] A. B. Mutiara, R. Reanti, en D, “MACHINE LEARNING KUANTUM UNTUK SAINS DATA Penerbit Gunadarma, amutiara”.

[4] S. B. Kotsiantis, “Supervised Machine Learning: A Review of Classification Techniques”, Informatica (Ljubljana) Oct, vol 2007, bll 249–268.

[5] Lloyd, S., Mohseni, M., & Rebentrost, P. (n.d.). “Quantum algorithms for supervised and unsupervised machine learning”, In arxiv.org. Retrieved June 27, 2021, from https://arxiv.org/abs/1307.0411.

[6] H. Abdi en L. J. Williams, “Principal component analysis”, Wiley Interdiscip. Rev. Comput. Stat., vol 2, no 4, bll 433–459, Jul 2010.

[7] A. Directions, “Principal Component Analysis (PCA)”, 2007, bll 1–12.

[8] Drs. A.P. Hardhono, M.Ed., Ph.D, and M.Kom. Dr. Imas Sukaesih Sitanggang, S.Si. n.d. “Pengenalan Dan Instalasi Perangkat Lunak Dan Lingkungan Pemrograman R”, 2014, 1–29.

[9] J. Doi, G. Potter, J. Wong, I. Alcaraz, en P. Chi, “Web application teaching tools for statistics using R and shiny”, Technology Innovations in Statistics Education, vol 9, no 1, 2016.

[10] Kassambara, A., & Mundt, F. (2017). Package ‘factoextra’. Extract and visualize the results of multivariate data analyses, 76.

[11] Y. Liang, M. Balcan, en V. Kanchanapally, “Distributed PCA and k -Means Clustering”, 2013, bll 1–8.

[12] R. Sharma en A. Rani, “K-Means Clustering in Spatial Data Mining using Weka Interface”, in International Conference on Advances in Communication and Computing Technologies (ICACACT), 2012, bll 2012–2026.