KLASIFIKASI BUNGA IRIS MENGGUNAKAN *SUPPORT VECTOR MACHINE* (*S*VM)

Disusun oleh :

Glandy Hizkia Arvenar Mundung (19101106054)

Dosen Pengampu :

Dr. Winsy Christo Deilan Weku, M.Cs

PROGRAM STUDI SISTEM INFORMASI

FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM

UNIVERSITAS SAM RATULANGI

MANADO

2021

1. PENDAHULUAN

1.1 Latar Belakang

Data Mining adalah proses yang menggunakan teknik statistik, matematika, kecerdasan buatan dan machine learning untuk mengekstraksi dan mengidentifikasi informasi yang bermanfaat dan pengetahuan yang terkait dari basis data yang besar. Data Mining merupakan serangkaian proses mendapatkan pengetahuan atau pola dari kumpulan data, data mining memecahkan masalah dengan menganalisis data yang telah ada pada basis data.

Support Vector Machine (SVM) adalah salah satu metode *Supervised Learning* untuk menganalisis data dan mengenali pola, digunakan untuk klasifikasi dan analisis regresi.

Dataset iris digunakan dalam paper Fisher tahun 1936 yang berjudul "The Use of Multiple Measurements in Taxonomic Problems". Dataset ini memiliki 50 contoh dari tiga spesies Iris (Iris *Setosa*, Iris *Virginica*, dan Iris *Versicolor*). Ketiga spesies tersebut memiliki Sepal Length, Sepal Width, Petal Length, dan Petal Width dalam satuan centimeter yang berbeda-beda.

Dalam riset ini akan dilakukan klasifikasi untuk mengetahui spesies dari Iris menggunakan metode *Support Vector Machine.*

1.2. Rumusan Masalah

1. Bagaimana implementasi metode *Support Vector Machine* menggunakan Bahasa Pemrograman R?
2. Berapa tingkat akurasi menggunakan metode *Support Vector Machine*.

1.3. Tujuan Penelitian

1. Mengimplementasikan metode *Support Vector Machine* menggunakan Bahasa Pemrograman R.
2. Mengetahui tingkat akurasi klasifikasi menggunakan metode *Support Vector Machine*.

2. METODOLOGI PENELITIAN

2.1 Pengambilan Data

Data yang digunakan dalam penelitian ini adalah data iris yang merupakan sekunder yang diambil dari R. Dataset iris adalah dataset built-in dalam R yang berisi pengukuran pada 4 atribut yang berbeda (dalam sentimeter) untuk 50 bunga dari 3 spesies yang berbeda.

2.2 Lokasi Penelitian

Penelitian ini berlangsung di rumah peneliti yang berlokasi di Desa Tombasian Atas Satu, kec. Kawangkoan Barat, kab. Minahasa, prov. Sulawesi Utara. Penelitian ini dilakukan selama 7 hari dimulai dari tanggal 20 September 2021.

2.3 Metode Penelitian

1. Pengumpulan Data

   Pada tahap awal dilakukan pengumpulan data sebagai bahan masukan untuk penelitian ini. Data yang diambil adalah dataset iris yang berisi pengukuran pada empat atribut yang berbeda (dalam sentimeter) untuk 50 bunga dari 3 spesies yang berbeda.
2. *Data Preprocessing*

   Proses ini merupakan tahapan awal sebelum melakukan pengujian metode, dimana dataset yang akan digunakan diperiksa terlebih dahulu apakah terdapat missing value.
3. *Data Splitting*

   Tahapan selanjutnya pada arsitektur penelitian ini adalah melakukan pemisahan data yaitu antara data training dan data testing. Dimana, pada penelitian ini akan digunakan data latih atau data *training* sebanyak 80%, sedangkan data uji atau data *testing* sebanyak 20% dari total data keseluruhan.

4. Pembuatan Model SVM

   Setelah data terbagi, peneliti akan melakukan proses klasifikasi dengan membuat model menggunakan metode SVM dengan variabel dependennya yaitu species. Data-data yang digunakan adalah data training.

5. Pengujian Model SVM

Setelah membuat model SVM, peneliti melakukan pengujian model SVM yang diperoleh menggunakan data *training* dan data *testing*.\

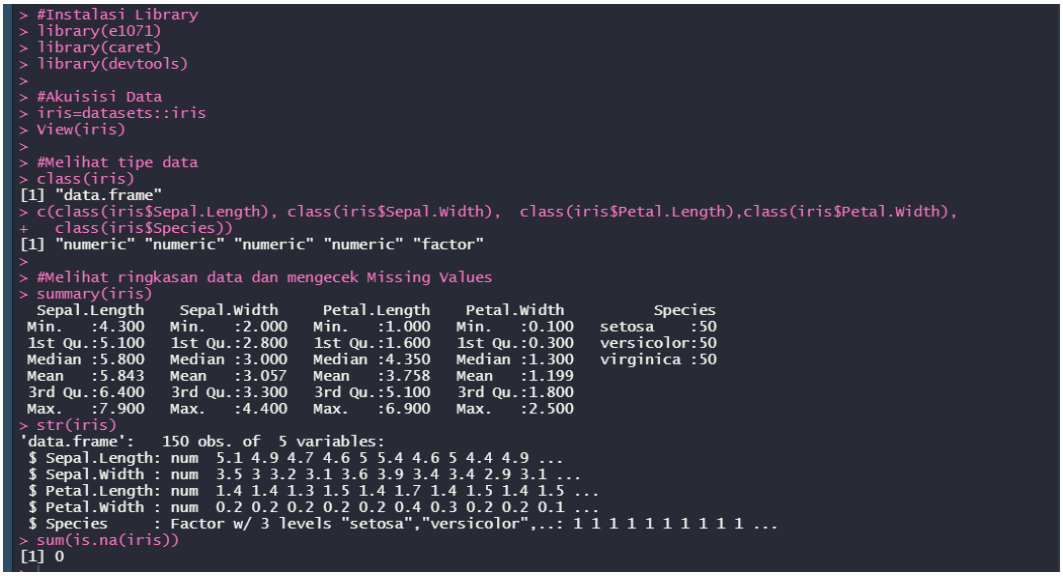
3. HASIL DAN PEMBAHASAN

3.1 *Data Preprocessing*

Tahapan ini merupakan tahapan dimana dataset yang akan digunakan diperiksa terlebih dahulu apakah terdapat missing value.

```
#Data Preprocessing

#Instalasi Library

library(e1071)

library(caret)

library(devtools)

#Akuisisi Data

iris=datasets::iris

View(iris)

#Melihat tipe data

class(iris)

c(class(iris$Sepal.Length),class(iris$Sepal.Width),class(iris$Petal.Length),

class(iris$Petal.Width),class(iris$Species))

#Melihat ringkasan data dan mengecek Missing Values

summary(iris)

str(iris)

sum(is.na(iris))
```

Berikut merupakan hasil *running* dari kode program diatas :



```
> #Instalasi Library
> library(e1071)
> library(caret)
> library(devtools)
>
> #Akuisisi Data
> iris=datasets::iris
> View(iris)
>
> #Melihat tipe data
> class(iris)
[1] "data.frame"
> c(class(iris$Sepal.Length), class(iris$Sepal.Width),  class(iris$Petal.Length),class(iris$Petal.Width),
+    class(iris$Species))
[1] "numeric" "numeric" "numeric" "numeric" "factor"
>
> #Melihat ringkasan data dan mengecek Missing Values
> summary(iris)
  Sepal.Length    Sepal.Width     Petal.Length    Petal.Width          Species
 Min.   :4.300   Min.   :2.000   Min.   :1.000   Min.   :0.100   setosa     :50
 1st Qu.:5.100   1st Qu.:2.800   1st Qu.:1.600   1st Qu.:0.300   versicolor:50
 Median :5.800   Median :3.000   Median :4.350   Median :1.300   virginica :50
 Mean   :5.843   Mean   :3.057   Mean   :3.758   Mean   :1.199
 3rd Qu.:6.400   3rd Qu.:3.300   3rd Qu.:5.100   3rd Qu.:1.800
 Max.   :7.900   Max.   :4.400   Max.   :6.900   Max.   :2.500
> str(iris)
'data.frame':   150 obs. of  5 variables:
 $ Sepal.Length: num  5.1 4.9 4.7 4.6 5 5.4 4.6 5 4.4 4.9 ...
 $ Sepal.Width : num  3.5 3 3.2 3.1 3.6 3.9 3.4 3.4 2.9 3.1 ...
 $ Petal.Length: num  1.4 1.4 1.3 1.5 1.4 1.7 1.4 1.5 1.4 1.5 ...
 $ Petal.Width : num  0.2 0.2 0.2 0.2 0.2 0.4 0.3 0.2 0.2 0.1 ...
 $ Species     : Factor w/ 3 levels "setosa","versicolor",..: 1 1 1 1 1 1 1 1 1 1 ...
> sum(is.na(iris))
[1] 0
```

Gambar 1. Hasil *Preprocessing* Data

Berdasarkan hasil pada **gambar 1**, dapat dilihat bahwa data iris yang digunakan tergolong ke dalam *data frame*. Sedangkan *class* untuk masing-masing variabel antara lain :

1. *Sepal Length* : Numerik
2. *Sepal Width* : Numerik
3. *Petal Length* : Numerik
4. *Petal Length* : Numerik
5. *Species* : Faktor

3.2 *Data Splitting*

Setelah tahap *preprocessing* data. Dataset iris yang berisi 150 data dibagi dengan proporsi data *training* sebesar 80% dan data *testing* sebesar 20% dengan pengambilan sampel secara acak, sehingga dilakukan perintah set.seed() dengan menggunakan sintaks sebagai berikut:

```
#80% data sebagai data training
n = round(nrow(iris)*0.80)
n
```

Maka, akan muncul hasil sebagai berikut:

```
> #Data Splitting
> #80% data sebagai data training
> n = round(nrow(iris)*0.8)
> n
[1] 120
```

Gambar 2. Tampilan Hasil Jumlah Data Training

Berdasarkan hasil pada **gambar 2,** diketahui bahwa dari 150 dataset iris, diambil 120 untuk dijadikan data training. Kemudian dilakukan penarikan data sampel sebanyak n=120. Data sampel yang berjumlah 120 tersebut disimpan dalam variabel yang bernama "data.train", sedangkan sisanya disimpan dalam variabel yang bernama "data.test" dengan menggunakan sintaks sebagai berikut:

```
set.seed(12345)

sample_iris = sample(seq_len(nrow(iris)), size = n)

data.train = iris[sample_iris,]

data.test = iris[-sample_iris,]

sample_iris

head(data.train,5); head(data.test,5)
```

Maka, akan muncul hasil sebagai berikut:

```
> sample_iris
  [1] 142  51  58  93  75  96   2  86 146  38 103  94  10  40 141  30   1  72  12   3  14 106 119  16  80  62
 [27] 145  98 116  60 105  32  25  36 104 150 136   9   5  79  83  17 109  37 148  67 110  13  95 120  49  74
 [53]  56  91  34  44  35 143 149  55  23  26  97   7  15  46 132 112  68  11 100 101  99 147  87  42  88   4
 [79]  73 128  57  65  48  78 108 131  18 122 138  54  76  31  70 125  20  84 124 133 139   6  90  47  53 135
[105]  59  52 137  29 118 115  28 111 102 117   8 130  64 123  21  71
> head(data.train,5); head(data.test,5)
    Sepal.Length Sepal.Width Petal.Length Petal.Width    Species
142          6.9         3.1          5.1         2.3  virginica
51           7.0         3.2          4.7         1.4 versicolor
58           4.9         2.4          3.3         1.0 versicolor
93           5.8         2.6          4.0         1.2 versicolor
75           6.4         2.9          4.3         1.3 versicolor
   Sepal.Length Sepal.Width Petal.Length Petal.Width Species
19          5.7         3.8          1.7         0.3  setosa
22          5.1         3.7          1.5         0.4  setosa
24          5.1         3.3          1.7         0.5  setosa
27          5.0         3.4          1.6         0.4  setosa
33          5.2         4.1          1.5         0.1  setosa
>
```

Gambar 3. Tampilan Hasil Jumlah Sampel Iris dan Lima Data *Training* dan Data *Testing*

Setelah data terbagi, proses klasifikasi dilakukan dengan membuat model menggunakan metode SVM dengan species sebagai variabel dependen. Data-data yang digunakan dalam klasifikasi ini berupa data training.

```
#Model SVM & Prediksi Data Training

svm.iris <- svm(factor(Species) ~., data = data.train)

svm.iris
```

Maka, hasil *running* kode program diatas dapat dilihat pada gambar berikut:

```
> #Model SVM & Prediksi Data Training
> svm.iris <- svm(factor(Species) ~., data = data.train)
> svm.iris

Call:
svm(formula = factor(Species) ~ ., data = data.train)


Parameters:
   SVM-Type:  C-classification
 SVM-Kernel:  radial
       cost:  1

Number of Support Vectors:  45
```

Gambar 4. Hasil Klasifikasi SVM dari Data *Training*

Berdasarkan hasil pada **gambar 4**, dapat diketahui tipe dari metode SVM yang dipakai adalah C-classification dengan kernel radial. Nilai parameter dari kernel (cost) adalah sebesar 1 dan *support vector* sebanyak 45. Langkah selanjutnya dalam melakukan analisis SVM adalah melakukan pengujian model SVM yang diperoleh menggunakan data training. Prediksi dilakukan terhadap data training dengan menggunakan sintaks sebagai berikut:

```
#Pengujian Model SVM Data Training
prediksi1 <- predict(svm.iris, data.train)
prediksi1
head(prediksi1,10); tail(prediksi1,10)
```

Maka hasilnya dapat dilihat pada gambar berikut:

```
> #Pengujian Model SVM Data Training
> prediksi1 <- predict(svm.iris, data.train)
> head(prediksi1,10); tail(prediksi1,10)
       142         51         58         93         75         96          2         86        146         38
 virginica versicolor versicolor versicolor versicolor versicolor     setosa versicolor  virginica     setosa
Levels: setosa versicolor virginica
        28        111        102        117          8        130         64        123         21         71
    setosa  virginica  virginica  virginica     setosa  virginica versicolor  virginica     setosa versicolor
Levels: setosa versicolor virginica
```

Gambar 5. Tampilan Hasil Pengujian Data *Training*

Berdasarkan hasil pada **gambar 5**, Bisa dilihat sepuluh data training teratas dan sepuluh data training terbawah pada data prediksi. *Virginica*, *Versicolor*, dan *Setosa* merupakan jenis spesies dari bunga iris, sedangkan angka yang letaknya tepat di atas jenis spesies merupakan urutan dari data yang digunakan. Sebagai contoh, data ke-96 diprediksi sebagai spesies *Versicolor*, data ke-28 diprediksi sebagai *Setosa*. Kemudian tingkat akurasi akan dilihat dari data training yang telah diprediksi dengan menggunakan sintaks sebagai berikut:

```
#Training Accuracy

confusionMatrix(prediksi1, factor(data.train$Species))
```

Maka, hasilnya dapat dilihat pada gambar berikut:

```
> #Training Accuracy
> confusionMatrix(prediksi1, factor(data.train$Species))
Confusion Matrix and Statistics

            Reference
Prediction   setosa versicolor virginica
  setosa         40          0         0
  versicolor      0         38         0
  virginica       0          2        40

Overall Statistics

               Accuracy : 0.9833
                 95% CI : (0.9411, 0.998)
    No Information Rate : 0.3333
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.975

 Mcnemar's Test P-Value : NA

Statistics by Class:

                     Class: setosa Class: versicolor Class: virginica
Sensitivity                 1.0000            0.9500           1.0000
Specificity                 1.0000            1.0000           0.9750
Pos Pred Value              1.0000            1.0000           0.9524
Neg Pred Value              1.0000            0.9756           1.0000
Prevalence                  0.3333            0.3333           0.3333
Detection Rate              0.3333            0.3167           0.3333
Detection Prevalence        0.3333            0.3167           0.3500
Balanced Accuracy           1.0000            0.9750           0.9875
```

Gambar 6. Hasil Tingkat Akurasi Model SVM dari Data *Training*

Berdasarkan hasil pada **gambar 6**, diketahui bahwa tingkat akurasi dari hasil prediksi untuk data training sebesar 0.9833 atau 98.33% dengan jumlah spesies *Setosa* yang diprediksi *Setosa* adalah 40 bunga, *Versicolor* yang diprediksi *Versicolor* sebanyak 38 bunga, *Virginica* yang diprediksi sebagai *Versicolor* sebanyak 2 bunga, dan *Virginica* yang diprediksi benar sebagai *Virginica* sebanyak 40 bunga. Nilai *sensitivity* pada hasil yang diperoleh digunakan untuk mengukur proporsi jumlah observasi positif yang tepat diprediksi. Nilai specificity pada hasil yang diperoleh digunakan untuk mengukur proporsi jumlah observasi negatif yang tepat prediksi. Nilai *balance accuracy* digunakan untuk mengukur akurasi proporsi jumlah

observasi kelas positif yang tepat diprediksi. Nilai-nilai tersebut pada masing-masing spesies dapat dilihat pada gambar berikut:

	Setosa	*Versicolor*	Virginica
Sensivity	100%	95%	100%
Specifity	100%	100%	97.50%
Balance Accuracy	100%	97.50%	98.75%

Gambar 7. Hasil Tingkat Akurasi Klasifikasi dari Data *Training*

Kemudian, dilakukan pengujian model SVM untuk data testing dengan menggunakan sintaks seperti berikut:

```
#Pengujian Model SVM Data Testing
prediksi2 <- predict(svm.iris, data.test)
head(prediksi2,10); tail(prediksi2,10)
```

Maka, hasilnya dapat dilihat sebagai berikut:

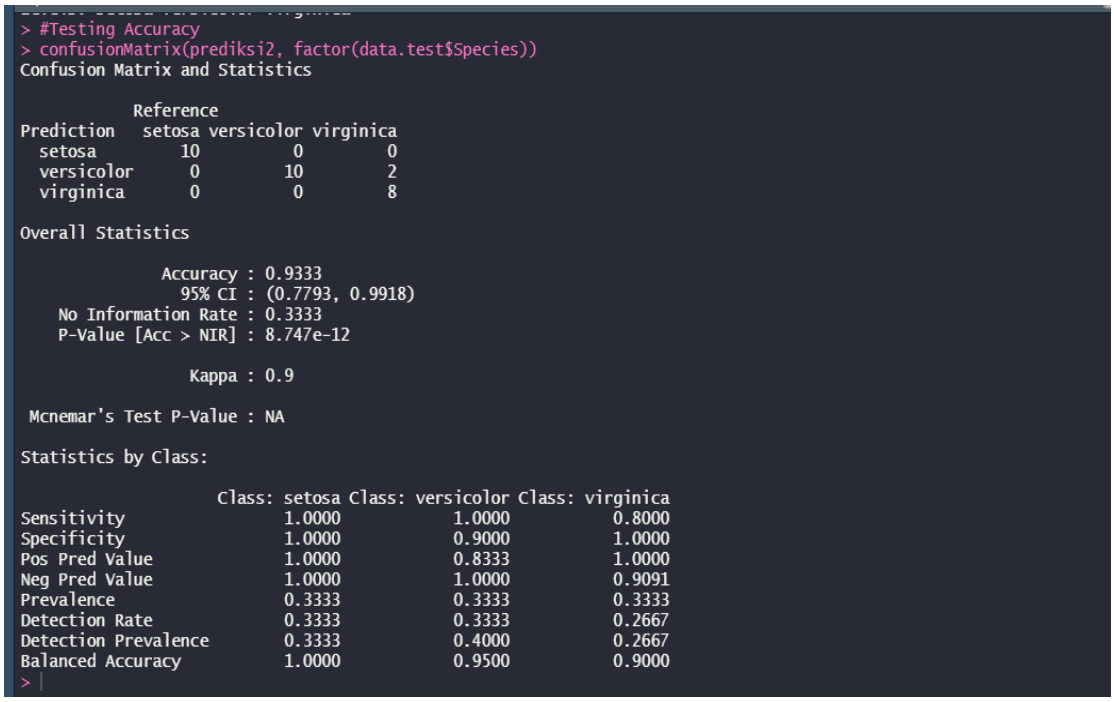
```
> #Pengujian Model SVM Data Testing
> prediksi2 <- predict(svm.iris, data.test)
> head(prediksi2,10); tail(prediksi2,10)
    19     22     24     27     33     39     41     43     45     50
setosa setosa setosa setosa setosa setosa setosa setosa setosa setosa
Levels: setosa versicolor virginica
       107        113        114        121        126        127        129        134        140        144
versicolor  virginica  virginica  virginica  virginica  virginica  virginica versicolor  virginica  virginica
Levels: setosa versicolor virginica
>
```

Gambar 8. Tampilan Hasil Pengujian Data *Testing*

Berdasarkan hasil pada **gambar 12**, dapat diketahui bahwa urutan data ke-19, ke-22, ke-22, ke-24, ke-27, ke-33, ke-39, ke-41, ke-43, ke-45, dan ke-50 diprediksi sebagi *Setosa*. Untuk data ke-107, ke-113, ke-114, ke-121, ke-126, ke-127, ke-129, ke-134, ke-140, dan ke-144 diprediksi sebagai *Virginica*. Kemudian tingkat akurasi dari data yang telah diprediksi dapat dilihat dengan menggunakan sintaks sebagai berikut:

```
#Testing Accuracy
confusionMatrix(prediksi2, factor(data.test$Species))
```

Maka, hasilnya dapat dilihat sebagai berikut:



```
> #Testing Accuracy
> confusionMatrix(prediksi2, factor(data.test$Species))
Confusion Matrix and Statistics

            Reference
Prediction   setosa versicolor virginica
  setosa         10          0         0
  versicolor      0         10         2
  virginica       0          0         8

Overall Statistics

               Accuracy : 0.9333
                 95% CI : (0.7793, 0.9918)
    No Information Rate : 0.3333
    P-Value [Acc > NIR] : 8.747e-12

                  Kappa : 0.9

 Mcnemar's Test P-Value : NA

Statistics by Class:

                     Class: setosa Class: versicolor Class: virginica
Sensitivity                 1.0000            1.0000           0.8000
Specificity                 1.0000            0.9000           1.0000
Pos Pred Value              1.0000            0.8333           1.0000
Neg Pred Value              1.0000            1.0000           0.9091
Prevalence                  0.3333            0.3333           0.3333
Detection Rate              0.3333            0.3333           0.2667
Detection Prevalence        0.3333            0.4000           0.2667
Balanced Accuracy           1.0000            0.9500           0.9000
>
```

Gambar 9. Hasil Tingkat Akurasi Model SVM dari Data *Testing*

Berdasarkan hasil pada **gambar 9**, dapat diketahui bahwa tingkat akurasi dari hasil prediksi untuk data testing sebesar 0.9333 atau 93.33% dengan jumlah spesies *Setosa* yang diprediksi *Setosa* adalah 10 bunga, *Versicolor* yang diprediksi *Versicolor* sebanyak 10 bunga, *Versicolor* yang diprediksi sebagai *Virginica* sebanyak 2 bunga, dan *Virginica* yang diprediksi benar *Virginica* sebanyak 8 bunga. Nilai *sensitivity*, *specificity*, dan *balance accuracy* pada masing-masing spesies dapat dilihat pada gambar berikut:

	Setosa	*Versicolor*	Virginica
Sensivity	100%	100%	80%
Specifity	100%	90%	100%
Balance Accuracy	100%	95%	90%

Gambar 10. Hasil Tingkat Akurasi Klasifikasi dari Data *Testing*

4. KESIMPULAN

1. Dengan proporsi 80% data *training* dan 20% data *testing*, jenis dari metode SVM adalah *C-classification* dengan kernel yang digunakan adalah radial, nilai parameter dari kernel adalah sebesar 1 dan banyak *support vector* pada klasifikasi SVM sebesar 45.
2. Berdasarkan pengujian model SVM data *training*, hasil prediksi menunjukan tingkat akurasi dari prediksi untuk data training sebesar 0.9833 atau 98.33% dengan jumlah spesies *Setosa* yang diprediksi *Setosa* adalah 40 bunga, *Versicolor* yang diprediksi *Versicolor* sebanyak 38 bunga, *Virginica* yang diprediksi sebagai *Versicolor* sebanyak 2 bunga, dan *Virginica* yang diprediksi benar sebagai *Virginica* sebanyak 40 bunga.
3. Berdasarkan pengujian model SVM data *testing*, hasil prediksi menunjukan tingkat akurasi dari prediksi untuk data training sebesar 0.9333 atau 93.33% dengan jumlah spesies *Setosa* yang diprediksi *Setosa* adalah 10 bunga, *Versicolor* yang diprediksi *Versicolor* sebanyak 10 bunga, *Versicolor* yang diprediksi sebagai *Virginica* sebanyak 2 bunga, dan *Virginica* yang diprediksi benar *Virginica* sebanyak 8 bunga.

DAFTAR PUSTAKA

Adi, S., Wida, P.M. 2018. Komparasi Algoritma Naive bayes dan SVM Untuk Memprediksi Keberhasilan Imunoterapi Pada Penyakit Kutil. *Jurnal Sains Komputer & Informatika (J-SAKTI), 2(2), pp. 152–161.*

Dhea, P. 2021 Classification of Iris with Support Vector Machine (SVM) in R. https://18611006.medium.com/classification-of-iris-with-support-vector-machine-svm-in-r-9f22e09f571a. (Diakses tanggal 24 September 2021.)

Haydar. 2017. Learning Data Science: Day 21 - Decision Tree on Iris Dataset https://haydar-ai.medium.com/learning-data-science-day-21-decision-tree-on-iris-dataset-267f3219a7fa. (Diakses tanggal 26 September 2021.)

Jonie, H. 2021. Klasifikasi Teks Humor Bahasa Indonesia Memanfaatkan SVM. *Journal of Information System, Graphics, Hospitality and Technology*, 03(1), 39-48

Mohit, K. 2021. Classification Problem: Relation between Sensitivity, Specificity and Accuracy. https://www.analyticsvidhya.com/blog/2021/06/classification-problem-relation-between-sensitivity-specificity-and-accuracy. (Diakses tanggal 26 September 2021.)

Nawawi H. M., Purnama J. J., dan Hikmah A. B. Komparasi Algoritma Neural Network Dan Naïve Bayes Untuk Memprediksi Penyakit Jantung. *J. Pilar Nusa Mandiri*, 15(2), 189-194.