**LAPORAN UTS**

**MATA KULIAH DATA MINING AND KNOWLEDGE MANAGEMENT**

UTS GANJIL 2020

****

Oleh:

**NAMA :** Nasiya Alifah Utami

**KELAS :** 3SD1

**NIM :** 221810496

**ABSEN :** 26

**POLITEKNIK STATISTIKA STIS**

**JAKARTA**

**2020**

**PAKTA INTEGRITAS**

Yang Bertanda tangan di bawah ini,

**NAMA :** Nasiya Alifah Utami

**NIM :** 221810496

**KELAS :** 3SD1

**NO.ABSEN :** 26

**MATA KULIAH :** Data Mining and Knowledge Management

**HARI/TANGGAL :** Selasa, 27 Oktober 2020

**DOSEN :** Ibnu Santoso SST, MT

Saya menyatakan bahwa ujian ini saya kerjakan dengan jujur sesuai kemampuan sendiri dan tidak mengutip sebagian atau seluruh pekerjaan orang lain. Jika suatu saat ditemukan saya melanggar ketentuan ujian, saya siap menerima konsekuensi yang berlaku.

Saya menggunakan tools RMarkdown untuk menyusun laporan UTS data mining ini.

|  |
| --- |
| Serang, 21 Oktober 2020  Nasiya Alifah Utami  NIM 221810496 |

Klasifikasi Tipe Benih

Nasiya Alifah Utami (221810496)

10/27/2020

# Soal

Sesuai dengan petunjuk pada Soal UTS Data Mining dan Knowledge Management Tahun 2020, Berikut ini adalah metode yang telah ditentukan untuk anda: Regresi Logistik

Berikut adalah Link Dataset yang digunakan: <https://archive.ics.uci.edu/ml/datasets/seeds> Details Dataset Records: 210 , Atribut: 7, Missing: No, Detail Tugas: Klasifikasi tipe benih

# Import Library

Library package yang digunakan untuk melakukan analisis data mining adalah sebagai berikut:

library(nnet) #neural network dan multiple regresi logistik  
 library(rpart) #decision tree id3  
 library(RWeka) #decision tree c45  
 library(e1071) #naive bayes  
 library(randomForest) #random forest

## randomForest 4.6-14

## Type rfNews() to see new features/changes/bug fixes.

library(mltest) #nilai kebaikan model  
 library(maxTPR) #nilai sensitivitas model

# Import Dataset

Dataset perlu di import ke environment di RStudio untuk dapat diolah dan digali informasinya. Dataset yang di import memiliki tipe file text (.txt). Karena dataset yang telah diunduh tidak memiliki nama pada header, maka perlu diberi nama terlebih dahulu sesuai dengan deskripsi atribut yang tercantum. berikut adalah proses import data di RStudio.

dataset <- read.delim("C:/Users/ACER/Downloads/seeds\_dataset (2).txt",  
 header=FALSE,  
 sep="\t",  
 na.strings = '?')  
names(dataset) <- c("Area","Perimeter", "compactness", "length",   
 "width", "as\_coef", "length\_KG", "tipe\_benih")  
lapply(dataset, as.numeric)

dataset$tipe\_benih <- as.factor(dataset$tipe\_benih)  
as.data.frame(dataset)

str(dataset)

## 'data.frame': 210 obs. of 8 variables:  
## $ Area : num 15.3 14.9 14.3 13.8 16.1 ...  
## $ Perimeter : num 14.8 14.6 14.1 13.9 15 ...  
## $ compactness: num 0.871 0.881 0.905 0.895 0.903 ...  
## $ length : num 5.76 5.55 5.29 5.32 5.66 ...  
## $ width : num 3.31 3.33 3.34 3.38 3.56 ...  
## $ as\_coef : num 2.22 1.02 2.7 2.26 1.35 ...  
## $ length\_KG : num 5.22 4.96 4.83 4.8 5.17 ...  
## $ tipe\_benih : Factor w/ 3 levels "1","2","3": 1 1 1 1 1 1 1 1 1 1 ...

head(dataset)

## Area Perimeter compactness length width as\_coef length\_KG tipe\_benih  
## 1 15.26 14.84 0.8710 5.763 3.312 2.221 5.220 1  
## 2 14.88 14.57 0.8811 5.554 3.333 1.018 4.956 1  
## 3 14.29 14.09 0.9050 5.291 3.337 2.699 4.825 1  
## 4 13.84 13.94 0.8955 5.324 3.379 2.259 4.805 1  
## 5 16.14 14.99 0.9034 5.658 3.562 1.355 5.175 1  
## 6 14.38 14.21 0.8951 5.386 3.312 2.462 4.956 1

# Split Training set and Testing Set

Data training dan data testing dibagi kedalam 80% data untuk data training dan 20% data untuk data testing. Pembagian testing set dan training set untuk validasi model dilakukan dengan pendekatan yang dinamakan validation set approach. Berikut adalah pemrosesannya yang dilakukan menggunakan tools Rstudio.

set.seed(123)  
 training\_set <- sample(1:nrow(dataset), floor(.80 \* nrow(dataset)))  
 testing\_set <- (1:nrow(dataset))[1:nrow(dataset) %in% training\_set == FALSE]  
 train <- dataset[training\_set,]  
 tes <- dataset[testing\_set,]  
 head(train)

## Area Perimeter compactness length width as\_coef length\_KG tipe\_benih  
## 159 11.75 13.52 0.8082 5.444 2.678 4.378 5.310 3  
## 207 11.23 12.88 0.8511 5.140 2.795 4.325 5.003 3  
## 179 11.48 13.05 0.8473 5.180 2.758 5.876 5.002 3  
## 14 13.78 14.06 0.8759 5.479 3.156 3.136 4.872 1  
## 195 12.11 13.27 0.8639 5.236 2.975 4.132 5.012 3  
## 170 11.24 13.00 0.8359 5.090 2.715 3.521 5.088 3

head(tes)

## Area Perimeter compactness length width as\_coef length\_KG tipe\_benih  
## 10 16.44 15.25 0.8880 5.884 3.505 1.969 5.533 1  
## 12 14.03 14.16 0.8796 5.438 3.201 1.717 5.001 1  
## 15 13.74 14.05 0.8744 5.482 3.114 2.932 4.825 1  
## 18 15.69 14.75 0.9058 5.527 3.514 1.599 5.046 1  
## 19 14.70 14.21 0.9153 5.205 3.466 1.767 4.649 1  
## 28 12.74 13.67 0.8564 5.395 2.956 2.504 4.869 1

# Training Data (modeling)

Training data dilakukan untuk membangun model yang tepat untuk melakukan prediksi target class data baru. Model yang digunakan adalah model regresi logistik multinomial karena target class memiliki lebih dari 2 faktor, yakni tipe benih 1, tipe benih 2, dan tipe benih 3. berikut adalah modelnya:

mrl\_model <- nnet::multinom(tipe\_benih ~., data = train)

## # weights: 27 (16 variable)  
## initial value 184.566864   
## iter 10 value 24.503158  
## iter 20 value 11.221174  
## iter 30 value 10.685855  
## iter 40 value 10.149665  
## iter 50 value 9.843267  
## iter 60 value 9.554613  
## iter 70 value 8.769631  
## iter 80 value 8.437930  
## iter 90 value 8.313550  
## iter 100 value 8.179113  
## final value 8.179113   
## stopped after 100 iterations

summary(mrl\_model)

## Call:  
## nnet::multinom(formula = tipe\_benih ~ ., data = train)  
##   
## Coefficients:  
## (Intercept) Area Perimeter compactness length width as\_coef  
## 2 -4.442078 22.91969 -13.226977 -7.857921 -105.82688 -24.43629 8.068539  
## 3 -57.926565 -12.50515 7.832224 -70.266175 -32.34159 47.63278 1.979692  
## length\_KG  
## 2 91.39413  
## 3 38.11727  
##   
## Std. Errors:  
## (Intercept) Area Perimeter compactness length width as\_coef  
## 2 8.802951 25.017421 29.86622 11.34033 35.25540 36.82568 12.955928  
## 3 46.494726 8.956063 10.83029 75.20115 24.46875 47.00407 1.002762  
## length\_KG  
## 2 18.00633  
## 3 20.91321  
##   
## Residual Deviance: 16.35823   
## AIC: 48.35823

Untuk mencari model terbaik, dibangun model dengan metode lainnya, yakni menggunakan algoritma SVM (support vector machine), Naive Bayes, Random Forest, Neural Network, dan Decision Tree ID3.

## SVM

Berikut adalah model svm yang dibentuk:

svm\_model <- svm (tipe\_benih ~ ., data = train,   
 kernel="radial", cost=5, probability=TRUE)  
summary(svm\_model)

##   
## Call:  
## svm(formula = tipe\_benih ~ ., data = train, kernel = "radial", cost = 5,   
## probability = TRUE)  
##   
##   
## Parameters:  
## SVM-Type: C-classification   
## SVM-Kernel: radial   
## cost: 5   
##   
## Number of Support Vectors: 44  
##   
## ( 15 20 9 )  
##   
##   
## Number of Classes: 3   
##   
## Levels:   
## 1 2 3

## 

## Naive Bayes

Berikut adalah model naive bayes yang dibentuk:

nb\_model <- naiveBayes(tipe\_benih ~ ., data = train)  
summary(nb\_model)

## Length Class Mode   
## apriori 3 table numeric   
## tables 7 -none- list   
## levels 3 -none- character  
## isnumeric 7 -none- logical   
## call 4 -none- call

## Random Forest

Berikut adalah model svm yang dibentuk:

rf\_model <- randomForest(tipe\_benih ~ ., data = train)  
rf\_model

##   
## Call:  
## randomForest(formula = tipe\_benih ~ ., data = train)   
## Type of random forest: classification  
## Number of trees: 500  
## No. of variables tried at each split: 2  
##   
## OOB estimate of error rate: 8.93%  
## Confusion matrix:  
## 1 2 3 class.error  
## 1 46 2 7 0.16363636  
## 2 1 56 0 0.01754386  
## 3 5 0 51 0.08928571

## Neural Network

Berikut adalah model svm yang dibentuk:

nn\_model <- nnet (tipe\_benih ~ ., data = train, size=5, decay=5e-4,   
 maxit=200)

## # weights: 58  
## initial value 207.984195   
## iter 10 value 176.488629  
## iter 20 value 76.461514  
## iter 30 value 24.326666  
## iter 40 value 11.294202  
## iter 50 value 10.747948  
## iter 60 value 10.589778  
## iter 70 value 10.483068  
## iter 80 value 10.398960  
## iter 90 value 10.340020  
## iter 100 value 10.104767  
## iter 110 value 5.967431  
## iter 120 value 3.342873  
## iter 130 value 2.851634  
## iter 140 value 2.548834  
## iter 150 value 2.371726  
## iter 160 value 2.218172  
## iter 170 value 2.087374  
## iter 180 value 1.839227  
## iter 190 value 1.691219  
## iter 200 value 1.604068  
## final value 1.604068   
## stopped after 200 iterations

nn\_model

## a 7-5-3 network with 58 weights  
## inputs: Area Perimeter compactness length width as\_coef length\_KG   
## output(s): tipe\_benih   
## options were - softmax modelling decay=5e-04

## Decision Tree ID3

Berikut adalah model ID3 yang dibentuk:

id3\_model <- rpart(tipe\_benih ~ ., data = train,   
 control = rpart.control(minsplit = 2))  
id3\_model

## n= 168   
##   
## node), split, n, loss, yval, (yprob)  
## \* denotes terminal node  
##   
## 1) root 168 111 2 (0.327380952 0.339285714 0.333333333)   
## 2) length\_KG>=5.573 57 1 2 (0.017543860 0.982456140 0.000000000) \*  
## 3) length\_KG< 5.573 111 55 3 (0.486486486 0.009009009 0.504504505)   
## 6) Area>=13.41 46 1 1 (0.978260870 0.021739130 0.000000000) \*  
## 7) Area< 13.41 65 9 3 (0.138461538 0.000000000 0.861538462)   
## 14) length\_KG< 4.8265 7 2 1 (0.714285714 0.000000000 0.285714286) \*  
## 15) length\_KG>=4.8265 58 4 3 (0.068965517 0.000000000 0.931034483) \*

# Testing Data (Prediksi)

Setelah dibangun model, data target class dari testing set di prediksi. prediksi dilakukan kepada testing set menggunakan semua model yang telah dibangun.

mrl\_pred <- predict(mrl\_model, tes[,1:7], type = "class")  
 svm\_pred <- predict(svm\_model, tes[,1:7])  
 nb\_pred <- predict(nb\_model , tes[,1:7])  
 rf\_pred <- predict(rf\_model , tes[,1:7])  
 nn\_pred <- predict(nn\_model , tes[,1:7], type = "class")  
 id3\_pred <- predict(id3\_model, tes[,1:7], type = "class")

#Confussion Matrix sebelum membuat confussion matrix, diperlukan perubahan variabel menjadi bentuk faktor sebagai berikut:

mrl\_pred <- as.factor(mrl\_pred)  
 svm\_pred <- as.factor(svm\_pred)  
 nb\_pred <- as.factor(nb\_pred)  
 rf\_pred <- as.factor(rf\_pred)  
 nn\_pred <- as.factor(nn\_pred)  
 id3\_pred <- as.factor(id3\_pred)  
 true\_value<- as.factor(tes$tipe\_benih)

Setelah itu baru dapat dibentuk confussion matrix. Berikut adalah confussion matrix dari setiap model.

1. Regresi Logistik

|  |
| --- |
| Regresi\_Logistik <- table(mrl\_pred, true\_value) Regresi\_Logistik  ## true\_value ## mrl\_pred 1 2 3 ## 1 15 3 0 ## 2 0 10 0 ## 3 0 0 14 |

1. SVM

SVM <- table(svm\_pred, true\_value)  
SVM

## true\_value  
## svm\_pred 1 2 3  
## 1 15 3 0  
## 2 0 10 0  
## 3 0 0 14

1. Naïve Bayes

Naive\_Bayes <- table(nb\_pred, true\_value)  
Naive\_Bayes

## true\_value  
## nb\_pred 1 2 3  
## 1 12 3 1  
## 2 1 10 0  
## 3 2 0 13

1. Random Forest

Random\_Forest <- table(rf\_pred, true\_value)   
Random\_Forest

## true\_value  
## rf\_pred 1 2 3  
## 1 14 1 0  
## 2 0 12 0  
## 3 1 0 14

1. Neural Network

Neural\_Network <- table(nn\_pred, true\_value)  
Neural\_Network

## true\_value  
## nn\_pred 1 2 3  
## 1 15 2 0  
## 2 0 11 0  
## 3 0 0 14

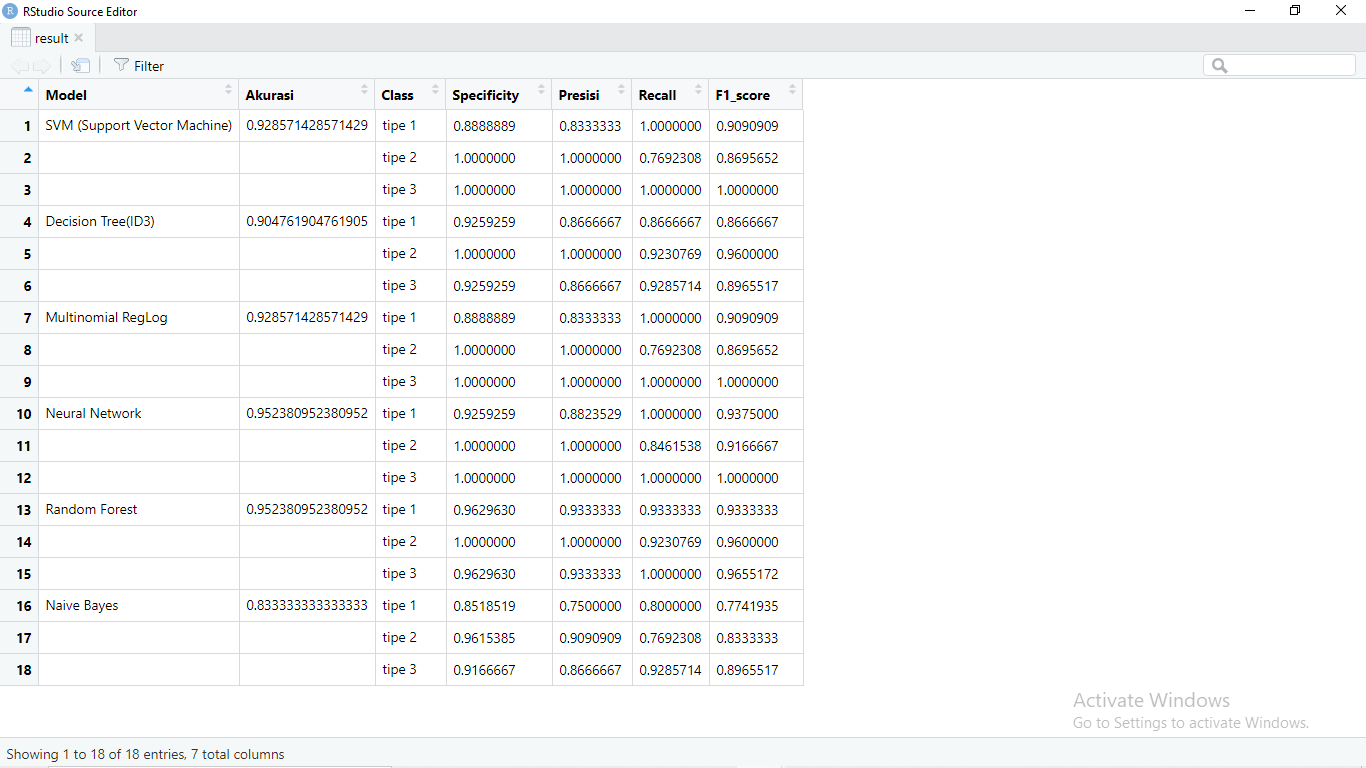
1. ID3

Decision\_Tree <- table(id3\_pred, true\_value)  
Decision\_Tree

## true\_value  
## id3\_pred 1 2 3  
## 1 13 1 1  
## 2 0 12 0  
## 3 2 0 13

# Validasi Model

Validasi model dilakukan dengan mengitung ukuran kebaikan model. ukuran kebaikan model yang akan digunakan kali ini adalah akurasi, sensitivity, specificity presisi, recall, dan F1 score.



# Interpretasi Ukuran Kebaikan Model Regresi Logistik

Berikut adalah interpretasi dari ukuran kebaikan model regresi Logistik Multinomial dia atas sebagai berikut:

1. Akurasi

Model Regresi Logistik Multinomial yang dibuat dapat memprediksikan tipe benih dengan tepat sebesar 92,8%.

1. Presisi

Model Regresi Logistik Multinomial yang dibuat dapat memprediksikan tipe benih yang bertipe 1 dengan tepat sebesar 83%, bertipe 2 adalah 100%, dan bertipe 3 adalah 100%.

1. Recall (Sensitifitas)

Model Regresi Logistik Multinomial yang dibuat dapat memprediksikan tipe benih yang bertipe 2 dan 3 dengan tepat sebesar 100%, bertipe 1 dan 3 adalah 76,9%, dan bertipe 3 adalah 100%.

1. Specificity

Model Regresi Logistik Multinomial yang dibuat dapat memprediksikan tipe benih yang tidak bertipe 1 dengan tepat sebesar 88,9%, tidak bertipe 2 adalah 100%, dan tidak bertipe 3 adalah 100%.

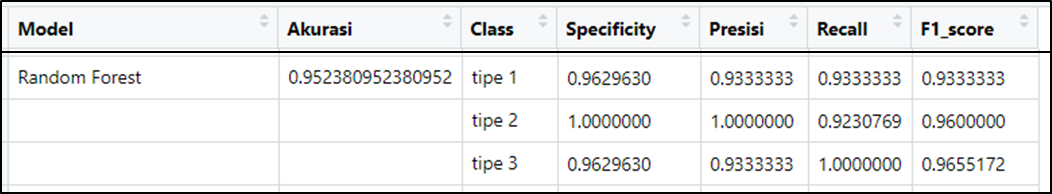
1. F1 Score

Perbandingan rata-rata nilai presisi dan nilai recall dari model regresi logistic multinomial untuk memprediksi data baru dengan tipe benih 1 adalah 90,9%, bertipe 2 adalah 86,9%, dan bertipe 3 adalah 100%.

# Best Model

Setelah menghitung ukuran kebaikan model, bandingkan ukuran kebaikan model tersebut. model terbaik adalah model yang memiliki akurasi, presisi, recall, dan F1 Score yang tinggi.

Pada Penelitian kali ini seharusnya peneliti menggunakan metode regresi logistik multinomial, namun dikarenakan nilai kebaikan model yang lebih bagus jika menggunakan random forest, maka peneliti menyarankan untuk menggunakan model random forest untuk memprediksi tipe benih dengan data baru. Nilai kebaikan model random forest adalah sebagai berikut:



Interpretasi nilai kebaikan model random forest:

1. Akurasi

Model Regresi Logistik Multinomial yang dibuat dapat memprediksikan tipe benih dengan tepat sebesar 95,2%.

1. Presisi

Model Regresi Logistik Multinomial yang dibuat dapat memprediksikan tipe benih yang bertipe 1 dengan tepat sebesar 93,3%, bertipe 2 adalah 100%, dan bertipe 3 adalah 93,3%.

1. Recall (Sensitifitas)

Model Regresi Logistik Multinomial yang dibuat dapat memprediksikan tipe benih yang bertipe 2 dan 3 dengan tepat sebesar 93,3%, bertipe 1 dan 3 adalah 92,3%, dan bertipe 3 adalah 100%.

1. Specificity

Model Regresi Logistik Multinomial yang dibuat dapat memprediksikan tipe benih yang tidak bertipe 1 dengan tepat sebesar 96,2%, tidak bertipe 2 adalah 100%, dan tidak bertipe 3 adalah 96,2%.

1. F1 Score

Perbandingan rata-rata nilai presisi dan nilai recall dari model regresi logistic multinomial untuk memprediksi data baru dengan tipe benih 1 adalah 93,3%, bertipe 2 adalah 96%, dan bertipe 3 adalah 96,5%.