|  |  |
| --- | --- |
| A close up of a logo  Description automatically generated  Assignment 02  CLASSIFICATION MODELS | Abstract  **This Assignments will give the idea of classification models to the selected dataset “seeds”.**  Kamaldeep Kaur  MBA-6693 |

**Introduction**In this assignment, we are going to discuss few classification models on the selected data “seeds”. I will describe that which sort of classification be useful or interesting according to the dataset. I have selected the data of seeds with the columns “Area”, “Perimeter”, “Compactness”, “Kernal.length”, “Kernal.width”, “Asymmetry.Coeff”, “Kernel.Groove” and “Type”. To test our data, I will be using Pareto Principle, generally called 80/20 split which states that 80% of consequences come from 20% of the causes.

**Objective**The main objective of this assignment to classify the seeds data according to their Area, Perimeter, Kernel Length, Kernel Width etc. I will try to draw some conclusions at the end after doing the classification models and interpreting the data.

**Import the data**I have called the library readxl to import the excel dataset in R.

library(readxl)

seeds <- read\_excel("Docs/Business Analytics/Classification/seeds.xlsx")  
View(seeds)

**Install the packages**I have installed the following packages which are quite helpful in this model:

install.packages("caret")  
library(caret)  
install.packages("ggthemes")

library(ggthemes)

install.packages("ggplot2")

library(ggplot2)  
install.packages("car")

library("car")  
install.packages("MASS")

library(MASS)

install.packages("tidyverse")

library("tidyverse")

library(class)

**Splitting the dataset into Training and Testing sets**I have splitted the dataset into two sets, training set (seen data) which usually used to train or build the model and Testing set (Unseen data).

# Loading the Caret package which is used for data partition  
library(caret)

# Create a partition on dataset (80% training 20% testing)

|  |
| --- |
| data\_split <- createDataPartition(seeds$Type, p = 0.8, list = FALSE)  # select 20% data for training testset <- seeds[-data\_split,]  # select 80% data to build or train the models  trainset <- seeds[data\_split,]  I have attached the screenshot below which shows the partition of dataset, trainset and test set. |

A screenshot of a social media post

Description automatically generated

**Understand the data**

* We will check the attributes, dimensions and size of the data.

Dim(trainset)  
[1] 160 8

* Understanding the structure of the data.

str(trainset)

tibble [160 x 8] (S3: tbl\_df/tbl/data.frame)

$ Area : num [1:160] 14.9 14.3 13.8 16.1 14.4 ...

$ Perimeter : num [1:160] 14.6 14.1 13.9 15 14.2 ...

$ Compactness : num [1:160] 0.881 0.905 0.895 0.903 0.895 ...

$ Kernel.Length : num [1:160] 5.55 5.29 5.32 5.66 5.39 ...

$ Kernel.Width : num [1:160] 3.33 3.34 3.38 3.56 3.31 ...

$ Asymmetry.Coeff: num [1:160] 1.02 2.7 2.26 1.35 2.46 ...

$ Kernel.Groove : num [1:160] 4.96 4.83 4.8 5.17 4.96 ...

$ Type : chr [1:160] "Type 1" "Type 1" "Type 1" "Type 1" ...

* Summary of the data

Summary(trainset)  
  
 Area Perimeter Compactness Kernel.Length Kernel.Width

Min. :10.59 Min. :12.41 Min. :0.8081 Min. :4.899 Min. :2.630

1st Qu.:12.35 1st Qu.:13.46 1st Qu.:0.8580 1st Qu.:5.259 1st Qu.:2.933

Median :14.47 Median :14.38 Median :0.8731 Median :5.550 Median :3.279

Mean :14.93 Mean :14.60 Mean :0.8710 Mean :5.645 Mean :3.266

3rd Qu.:17.57 3rd Qu.:15.85 3rd Qu.:0.8865 3rd Qu.:6.021 3rd Qu.:3.568

Max. :21.18 Max. :17.21 Max. :0.9183 Max. :6.675 Max. :4.033

Asymmetry.Coeff Kernel.Groove Type

Min. :0.7651 Min. :4.607 Length:160

1st Qu.:2.4202 1st Qu.:5.044 Class :character

Median :3.5285 Median :5.221 Mode :character

Mean :3.6389 Mean :5.423

3rd Qu.:4.8610 3rd Qu.:5.879

Max. :7.5240 Max. :6.550

1. **Histogram**

hist(trainset$Area)  
hist(trainset$Perimeter)

I have selected two parameters to check the frequency distribution of data i.e. Area and Perimeter.

A screenshot of a cell phone

Description automatically generated  
A picture containing screenshot

Description automatically generated

1. **Box Plot**

Box plot will display the distribution of data based on a five number summary (“min”, Q1, Median, Q3 and “Max”)  
I have selected five columns to check the data distribution i.e. “Area”, “Perimeter”, “Compactness”, “Kernel.Length”, “Kernel.Width”.

par(mfrow=c(1,5))

for(i in 1:5) {

boxplot(trainset[,i], main=names(trainset)[i])

}

#### A screenshot of a video game Description automatically generated

#### Visualization To understand the data more precisely and aesthetically, I have created the Box Plot and Histogram using more variables. I have created the scatter plot. g <- ggplot(data=trainset, aes(x = Area, y = Perimeter)) print(g)

g <-g +

geom\_point(aes(color=Type, shape=Type)) +

xlab("Area") +

ylab("Perimeter") +

ggtitle("Area and Perimeter")+

geom\_smooth(method="lm")

print(g)

Now, We can check how our data is segregated throughout. We can see that Type 1 is segregates where Perimeter is between 13 to 15.5 and Area is between 11.25 to 17.5, Type 2 data is segregated where Perimeter is between 15 to 17.5 and Area is between 16.25 to 22, Type 3 where Perimeter is from 12 to 14 and Area from 11 to 13.5 respectively.

**A close up of a map

Description automatically generated**

**Box Plot**

box <- ggplot(data=trainset, aes(x=Type, y=Area)) +

+ geom\_boxplot(aes(fill=Type)) +

+ ylab("Area") +

+ ggtitle("Seeds Boxplot") +

+ stat\_summary(fun=mean, geom="point", shape=5, size=4)

> print(box)

In the screenshot below, We can see that how the Type 1, Type 2 and Type 3 is segregated with respect to Area.

**A picture containing screenshot

Description automatically generated**

**Histogram**histogram <- ggplot(data=seeds, aes(x=Area)) +

geom\_histogram(binwidth=0.2, color="black", aes(fill=Type)) +

xlab("Area") +

ylab("Frequency") +

ggtitle("Histogram of Sepal Width")

print(histogram)  
  
We can check the frequency of data distribution of Type1, Type2, and Type3. We can also able to see that area of Type 1 and Type 2 are overlapping from 11 to 13 and area of Type 2 and Type 3 are overlapping from 15.25 to 17.

**A picture containing implement, pencil

Description automatically generated**

**Correlation of data**correlation\_data <- cor(seeds[,1:5])

ggcorrplot(correlation\_data, method = "circle")  
  
We can see that all the other parameters are highly correlated with each other except Compactness.

A picture containing food

Description automatically generated

**Q-Q Plot**

qqPlot(seeds$Area)

|  |
| --- |
| [1] 85 110  A close up of a map  Description automatically generated |
|  |
| |  | | --- | |  | |

The Q-Q plots help us to provide quick comparisons between probability distributions and can tell us how closely a data sample is to normally distributed. We can observe that the trad is increasing from –3 to 2.

**Classification Models**

1. **Linear Discriminant Analysis**

Linear Discriminant Analysis can be performed by using lda function and MASS package. Before performing the LDA, we will split the data into 80% and 20%.   
  
# Loading the Caret package which is used for data partition  
library(caret)

# Create a partition on dataset (80% training 20% testing)

|  |
| --- |
| data\_split <- createDataPartition(seeds$Type, p = 0.8, list = FALSE)  # select 20% data for training testset <- seeds[-data\_split,]  # select 80% data to build or train the models  trainset <- seeds[data\_split,]  Normalizing the data  # Estimating preprocessing parameters preproc.param <- trainset %>%  + preProcess(method = c("center", "scale")) # Transforming the data  traindata <- preproc.param %>% predict(trainset)  testdata <- preproc.param %>% predict(testset)  library(MASS) #Fit lda Model model <- lda(Type~., data = traindata) #Make some predictions predictions <- model %>% predict(testdata) #Accuracy of the model  mean(predictions$class==testdata$Type) #Compute LDA  library(MASS)  model <- lda(Type~., data = traindata)  model  Call:  lda(Type ~ ., data = traindata)  Prior probabilities of groups:  Type 1 Type 2 Type 3  0.33125 0.34375 0.32500  Group means:  Area Perimeter Compactness Kernel.Length Kernel.Width Asymmetry.Coeff  Type 1 -0.2244533 -0.2526908 0.3616753 -0.3158593 -0.09222299 -0.687503862  Type 2 1.1849267 1.1942226 0.5322406 1.1553494 1.10303249 -0.009770821  Type 3 -1.0245182 -1.0055698 -0.9315773 -0.9000706 -1.07267248 0.711059613  Kernel.Groove  Type 1 -0.6818740  Type 2 1.2186340  Type 3 -0.5939529  Coefficients of linear discriminants:  LD1 LD2  Area 1.75974715 -12.0623673  Perimeter -5.79658588 11.5063640  Compactness -0.36751322 2.1088666  Kernel.Length 2.58479087 3.2149276  Kernel.Width 0.36805553 -0.6340659  Asymmetry.Coeff 0.06869209 -0.4678836  Kernel.Groove -1.44294028 -3.4027528  Proportion of trace:  LD1 LD2  0.7029 0.2971  **Analysis:** Linear Discriminant Analysis provide the Group mean, probabilities and coefficient for each individual parameter.  1. Probabilities:We can see that there are 33.1%, 34.3%, 32.5% of Type 1, Type 2 and Type 3 in the Group. 2. Group Mean: It shows the group center of gravity of all the variables.  3. Coefficients: It provide us the combination of all the variable that are required to form the LDA decision. For eg: **LD1** = Area\*1.75974715 +  Perimeter\*-5.79658588 + Compactness\*-0.36751322 + Kernel.Length\*2.58479087 +  Kernel.Width\*0.36805553 + Asymmetry.Coeff\*0.06869209 +  Kernel.Groove\*-1.44294028  and **LDA2** = Area\*-12.0623673 + Perimeter\*11.5063640 +  Compactness\*2.1088666 + Kernel.Length\*3.2149276 + Kernel.Width\*-0.6340659 + Asymmetry.Coeff\*-0.4678836 + Kernel.Groove\*-3.4027528  **Plotting LDA**  plot(model)  A screenshot of a cell phone  Description automatically generated  **Plotting LDA using GGPLOT**  lda.data <- cbind(traindata, predict(model)$x)  > ggplot(lda.data, aes(LD1, LD2)) +  + geom\_point(aes(color = Type))    **Checking the model accuracy** mean(predictions$class==testdata$Type)  [1] 1 Now, We can see that our model classified 100% of the observation of the selected data that means our model is Excellent. |

1. **Logistic Regression**

|  |
| --- |
| glm(Type ~ Area, data= seeds) **[Using one variable]**  Call: glm(formula = Type ~ Area, data = seeds)  Coefficients:  (Intercept) Area  3.41069 -0.09489  Degrees of Freedom: 198 Total (i.e. Null); 197 Residual  Null Deviance: 131  Residual Deviance: 115.8 AIC: 463 |
|  |
| |  | | --- | |  | |

glm(Type ~ Area+ Perimeter, data= seeds) **[Using Two variable]**

Call: glm(formula = Type ~ Area + Perimeter, data = seeds)

Coefficients:

(Intercept) Area Perimeter

-3.9371 -0.5080 0.9257

Degrees of Freedom: 198 Total (i.e. Null); 196 Residual

Null Deviance: 131

Residual Deviance: 112.6 AIC: 459.3

> glm(Type ~ Area+ Perimeter+ Compactness+ Kernel.Length+ Kernel.Width+ Asymmetry.Coeff+ Kernel.Groove, data= seeds) **[Using All variable]**

Call: glm(formula = Type ~ Area + Perimeter + Compactness + Kernel.Length +

Kernel.Width + Asymmetry.Coeff + Kernel.Groove, data = seeds)

Coefficients:

(Intercept) Area Perimeter Compactness

55.6479 1.5649 -3.4187 -31.9797

Kernel.Length Kernel.Width Asymmetry.Coeff Kernel.Groove

-2.1474 0.2469 0.1084 2.1509

Degrees of Freedom: 198 Total (i.e. Null); 191 Residual

Null Deviance: 131

Residual Deviance: 31.72 AIC: 217.3

**Plotting the Logistic Regression**

plot(seeds$Type, seeds$Area, pch = 16, xlab = "Type", ylab = "Area")

A picture containing traffic

Description automatically generated

**Conclusion**

In the conclusion, I want to include that Type 1 of seeds are of smaller area and Perimeter than Type 2 and Type 3. As Type 2 and Type 3 are more wider in Length and Width. Also We can include that data of the seeds is strongly correlated except one variable i.e. Compactness. we can say that Linear Descriminant Analysis is excellent model because it is providing 100% accuracy of our all the observations.