assignment_10.2_MunjewarSheetal

Sheetal M

2023-02-18

Install and Load required packages :

```
knitr::opts_chunk$set(echo = TRUE)
knitr::opts_chunk$set(warning = FALSE)
knitr::opts_chunk$set(fig.width = 12, fig.height = 10)
knitr::opts_chunk$set(tidy.opts = list(width.cutoff = 70), tidy = TRUE)

# Package names
# packages <- c("ggplot2","dplyr","tidyr","magrittr","tidyverse","purrr")
packages <- c("broom","dplyr","RWeka","class","ggplot2")

# Install packages not yet installed
installed_packages <- packages %in% rownames(installed.packages())
if (any(installed_packages == FALSE)) {
  install.packages(packages[!installed_packages])
}

# Packages loading
invisible(lapply(packages, library, character.only = TRUE))
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

Problem statement : Predict one year life expectancy of lung cancer patients post surgery.

Set the working directory to the root of your DSC 520 directory

setwd("E:\Data_Science_DSC510\DSC520-Statistics\dsc520")

```
## Set the working directory to the root of your DSC 520 directory
setwd("E:\\Data_Science_DSC510\\DSC520-Statistics\\dsc520")

## Load data from data/binary-classifier-data.csv
bc_data <- read.csv("data/binary-classifier-data.csv")
str(bc_data)
```

```
## 'data.frame':    1498 obs. of  3 variables:
##  $ label: int  0 0 0 0 0 0 0 0 0 0 ...
##  $ x    : num  70.9 75 73.8 66.4 69.1 ...
##  $ y    : num  83.2 87.9 92.2 81.1 84.5 ...
```

```
#nrow(pat_data)
```

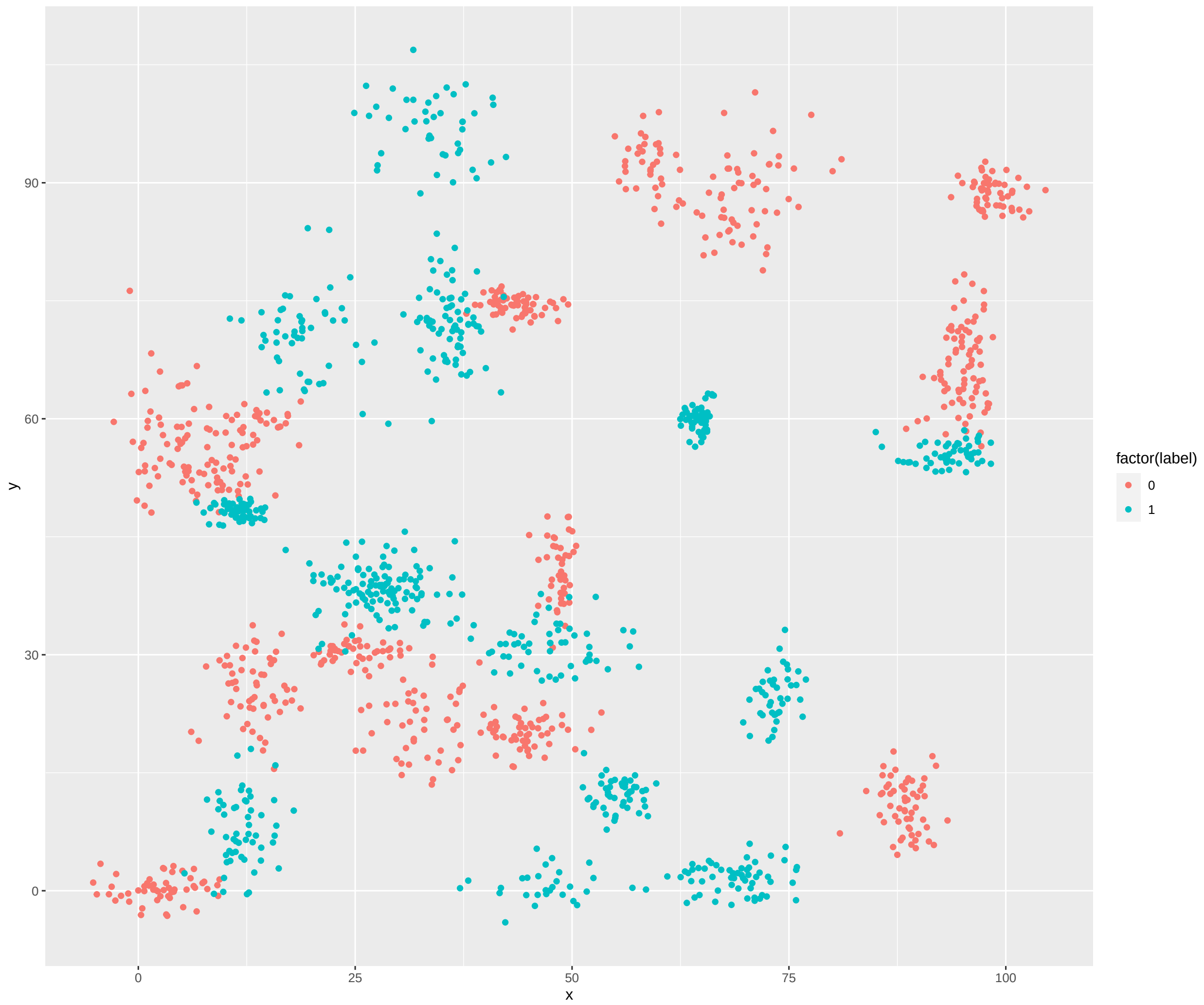
Convert label column data type into factor

```
bc_data$label <- as.factor(bc_data$label)
str(bc_data)
```

```
## 'data.frame':    1498 obs. of  3 variables:
##  $ label: Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
##  $ x    : num  70.9 75 73.8 66.4 69.1 ...
##  $ y    : num  83.2 87.9 92.2 81.1 84.5 ...
```

Visualize data

```
ggplot(data = bc_data, aes(x,y, color=factor(label))) + geom_point()
```



Generalized Linear Model

```
bc_mod01 <- glm(label ~ ., data = bc_data, family = "binomial")
```

Model Summary

```
summary(bc_mod01)
```

```
##
## Call:
## glm(formula = label ~ ., family = "binomial", data = bc_data)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.3728  -1.1697  -0.9575   1.1646   1.3989
##
```

```
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  0.424809   0.117224   3.624  0.00029 ***
## x           -0.002571   0.001823  -1.411  0.15836
## y           -0.007956   0.001869  -4.257 2.07e-05 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 2075.8  on 1497  degrees of freedom
## Residual deviance: 2052.1  on 1495  degrees of freedom
## AIC: 2058.1
##
## Number of Fisher Scoring iterations: 4
```

Variables with significance

1. X is Most Significant

Dataframe with new predicted column predict_Risk

```
knitr::opts_chunk$set(echo = TRUE)
knitr::opts_chunk$set(warning = FALSE)
knitr::opts_chunk$set(fig.width = 12, fig.height = 10)
knitr::opts_chunk$set(tidy.opts = list(width.cutoff = 70), tidy = TRUE)

#mod_plus <- augment(pat_mod01, type type.predict="response")
#class(mod_plus)
bc_mod01_predict <- augment(bc_mod01, type.predict="response") %>% mutate(predict_Risk = round(.fitted)

# Name additional columns and check class.
# class(mod_plus)
# names(mod_plus)
# https://cyberactive.bellevue.edu/ultra/courses/_514803_1/cl/outline
# alternate options using predict function() - predict(bc_mod01, type = "response")
```

Confusion matrix to calculate accurracy

```
bc_mod01_predict %>% select(label, predict_Risk) %>% table()
```

```
##      predict_Risk
## label   0   1
##     0 429 338
##     1 286 445
```

```
# Alternate option :
# predict <- predict(logit, data_test, type = 'response')
# table_mat <- table(data_test$income, predict > 0.5)
```

c. Accuracy of the Model

accuracy = correctly predicted / total Predicted * 100

```
knitr::opts_chunk$set(echo = TRUE)
knitr::opts_chunk$set(warning = FALSE)
knitr::opts_chunk$set(fig.width = 12, fig.height = 10)
knitr::opts_chunk$set(tidy.opts = list(width.cutoff = 70), tidy = TRUE)


# Evaluating accuracy of Regression Models - https://www.youtube.com/watch?v=03FrK8d2QVQ
# Accuracy using confusion matric : https://towardsdatascience.com/confusion-matrix-for-your-multi-clas

accuracy <- (429 + 445) / (429 + 338 + 286 + 445)
accuracy <- accuracy * 100
print(paste(round(accuracy), "%"))
```

```
## [1] "58 %"
```