Department of Computer Engineering

**Academic Term: July-November 2023**

Rubrics for Lab Experiments

|  |  |  |
| --- | --- | --- |
| **Class** | **: B*.E. Computer*** | **Subject Name :*BDA*** |
| **Semester** | **: VII** | **Subject Code :** |

|  |  |
| --- | --- |
| **Practical No:** | 10 |
| **Title:** | Analyze healthcare data by using R language |
| **Date of Performance:** | 27/09/2023 |
| **Roll No:** | 9427 |
| **Name of the Student:** | Atharva Prashant Pawar |

**Evaluation:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Performance Indicator** | **Below average** | **Average** | **Good** | **Excellent** | **Marks** |
| **On time**  **Submission (2)** | Not submitted(0) | Submitted after deadline (1) | Early or on time submission(2) | --- |  |
| **Test cases and** | Incorrect | The expected | The expected | Expected output is |  |
| **output** | output (1) | output is | output is Verified | obtained for all test |
| **(4)** |  | verified only a | for all test cases | cases. Presentable and |
|  |  | for few test | but is | easy to follow (4) |
|  |  | cases (2) | not presentable (3) |  |
| **Coding** | The code is not | The code is | The code is | - |  |
| **efficiency (2)** | structured at all | structured but | structured |  |
|  | (0) | not efficient (1) | and |  |
|  |  |  | efficient. (2) |  |
| **Knowledge(2)** | Basic concepts not clear  (0) | Understood the basic concepts (1) | Could explain the concept with  suitable example (1.5) | Could relate the theory with real world  application(2) |  |
| **Total** |  | | | | |

**Signature of the Teacher :**

## **|** Libraries

library(tidyverse) library(tidymodels) library(discrim)

library(baguette) library(bonsai)

library(patchwork)

── **Attaching packages ─────────────────────────────────────── tidyverse 1.3.1 ──**

✔ **ggplot2 3.3.6** ✔ **purrr 0.3.4**

✔ **tibble 3.1.7** ✔ **dplyr 1.0.9**

✔ **tidyr 1.2.0** ✔ **stringr 1.4.0**

✔ **readr 2.1.2** ✔ **forcats 0.5.1**

**── Conflicts ────────────────────────────────────────── tidyverse\_conflicts() ── dplyr::filter() masks stats::filter()**



✖

✖

**dplyr::lag() masks stats::lag()**

**── Attaching packages ────────────────────────────────────── tidymodels 1.0.0 ──**

|  |  |  |
| --- | --- | --- |
| ✔ **broom** | **1.0.0** ✔ **rsample** | **1.0.0** |
| ✔ **dials** | **1.0.0** ✔ **tune** | **1.0.0** |
| ✔ **infer** | **1.0.2** ✔ **workflows** | **1.0.0** |
| ✔ **modeldata** | **1.0.0** ✔ **workflowsets** | **1.0.0** |
| ✔ **parsnip** | **1.0.0** ✔ **yardstick** | **1.0.0** |
| ✔ **recipes** | **1.0.1** |  |

**── Conflicts ───────────────────────────────────────── tidymodels\_conflicts() ── scales::discard() masks purrr::discard()**



✖

✖

✖

✖

✖

✖

**dplyr::filter() masks stats::filter() recipes::fixed() masks stringr::fixed() dplyr::lag() masks stats::lag()**

**yardstick::spec() masks readr::spec() recipes::step() masks stats::step()**

* + **Dig deeper into tidy modeling with R at [https://www.tmwr.org](https://www.tmwr.org/)**

**Attaching package: ‘discrim’**

**The following object is masked from ‘package:dials’: smoothness**

## **|** Load and Check Data

diabetes <- read\_csv("../input/pima-indians-diabetes-database/diabetes.csv")

**Rows: 768 Columns: 9**

**── Column specification ──────────────────────────────────────────────────────── Delimiter: ","**

**dbl (9): Pregnancies, Glucose, BloodPressure, SkinThickness, Insulin, BMI, D...**

**ℹ Use `spec()` to retrieve the full column specification for this data.**

**ℹ Specify the column types or set `show\_col\_types = FALSE` to quiet this message.**

diabetes$Outcome <- as.factor(diabetes$Outcome)

diabetes %>% glimpse() Rows: 768

Columns: 9

$ Pregnancies *<dbl>* 6, 1, 8, 1, 0, 5, 3, 10, 2, 8, 4, 10, 10, 1, …

$ Glucose *<dbl>* 148, 85, 183, 89, 137, 116, 78, 115, 197, 125…

$ BloodPressure *<dbl>* 72, 66, 64, 66, 40, 74, 50, 0, 70, 96, 92, 74…

$ SkinThickness *<dbl>* 35, 29, 0, 23, 35, 0, 32, 0, 45, 0, 0, 0, 0, …

$ Insulin *<dbl>* 0, 0, 0, 94, 168, 0, 88, 0, 543, 0, 0, 0, 0, …

$ BMI *<dbl>* 33.6, 26.6, 23.3, 28.1, 43.1, 25.6, 31.0, 35.…

$ DiabetesPedigreeFunction *<dbl>* 0.627, 0.351, 0.672, 0.167, 2.288, 0.201, 0.2…

$ Age *<dbl>* 50, 31, 32, 21, 33, 30, 26, 29, 53, 54, 30, 3…

$ Outcome *<fct>* 1, 0, 1, 0, 1, 0, 1, 0, 1, 1, 0, 1, 0, 1, 1, …

diabetes %>% head()

A tibble: 6 × 9

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Pregnancies**  **<dbl>** | **Glucose**  **<dbl>** | **BloodPressure**  **<dbl>** | **SkinThickness**  **<dbl>** | **Insulin**  **<dbl>** | **BMI**  **<dbl>** | **DiabetesPedigreeFunction**  **<dbl>** | **Age**  **<dbl>** | **Outcome**  **<fct>** |
| 6 | 148 | 72 | 35 | 0 | 33.6 | 0.627 | 50 | 1 |
| 1 | 85 | 66 | 29 | 0 | 26.6 | 0.351 | 31 | 0 |
| 8 | 183 | 64 | 0 | 0 | 23.3 | 0.672 | 32 | 1 |
| 1 | 89 | 66 | 23 | 94 | 28.1 | 0.167 | 21 | 0 |
| 0 | 137 | 40 | 35 | 168 | 43.1 | 2.288 | 33 | 1 |
| 5 | 116 | 74 | 0 | 0 | 25.6 | 0.201 | 30 | 0 |

diabetes %>% summary()

Pregnancies Glucose BloodPressure SkinThickness Min. : 0.000 Min. : 0.0 Min. : 0.00 Min. : 0.00

1st Qu.: 1.000 1st Qu.: 99.0 1st Qu.: 62.00 1st Qu.: 0.00

Median : 3.000 Median :117.0 Median : 72.00 Median :23.00

Mean : 3.845 Mean :120.9 Mean : 69.11 Mean :20.54 3rd Qu.: 6.000 3rd Qu.:140.2 3rd Qu.: 80.00 3rd Qu.:32.00 Max. :17.000 Max. :199.0 Max. :122.00 Max. :99.00

Insulin BMI DiabetesPedigreeFunction Age

Min. : 0.0 Min. : 0.00 Min. :0.0780 Min. :21.00 1st Qu.: 0.0 1st Qu.:27.30 1st Qu.:0.2437 1st Qu.:24.00

Median : 30.5 Median :32.00 Median :0.3725 Median :29.00

Mean : 79.8 Mean :31.99 Mean :0.4719 Mean :33.24 3rd Qu.:127.2 3rd Qu.:36.60 3rd Qu.:0.6262 3rd Qu.:41.00 Max. :846.0 Max. :67.10 Max. :2.4200 Max. :81.00

Outcome 0:500

1:268

1. **|** Exploratory Data Analysis

# 3.1 | Pregnancies Variable

diabetes$Pregnancies %>% summary()

Min. 1st Qu. Median Mean 3rd Qu. Max. 0.000 1.000 3.000 3.845 6.000 17.000

hist\_pregnancies <- ggplot(data = diabetes, aes(x = Pregnancies)) + geom\_histogram(color = "darkblue", fill = "lightblue") +

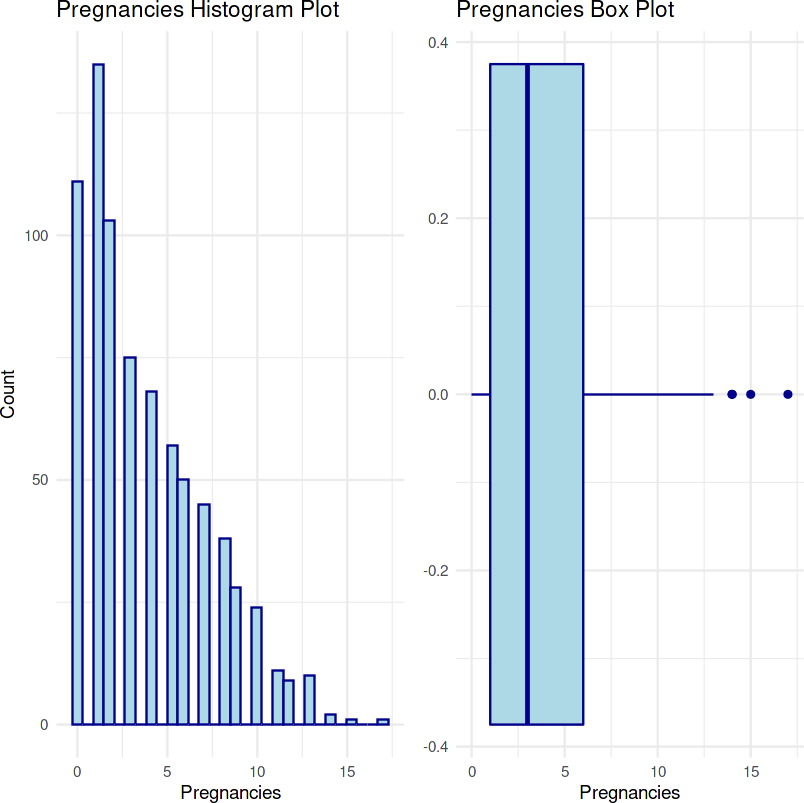
labs(title = "Pregnancies Histogram Plot", x = "Pregnancies", y = "Count") + theme\_minimal()

box\_pregnancies <- ggplot(data = diabetes, aes(x = Pregnancies)) + geom\_boxplot(color = "darkblue", fill = "lightblue") +

labs(title = "Pregnancies Box Plot", x = "Pregnancies") + theme\_minimal()

hist\_pregnancies + box\_pregnancies

`stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



**3.2 | Glucose Variable**

diabetes$Glucose %>% summary()

Min. 1st Qu. Median Mean 3rd Qu. Max. 0.0 99.0 117.0 120.9 140.2 199.0

hist\_glucose <- ggplot(data = diabetes, aes(x = Glucose)) + geom\_histogram(color = "darkblue", fill = "lightblue") +

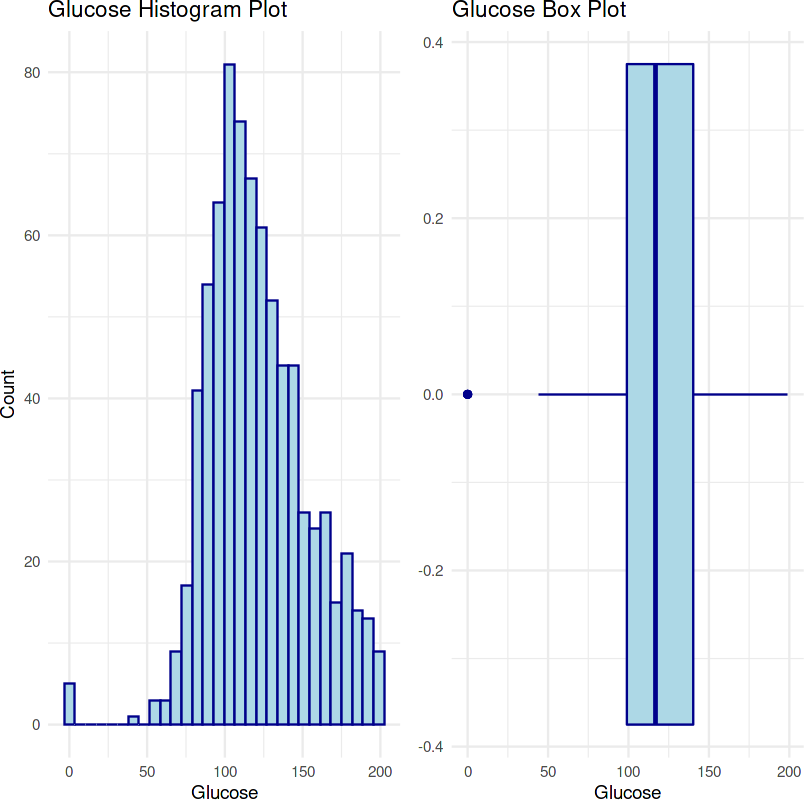
labs(title = "Glucose Histogram Plot", x = "Glucose", y = "Count") + theme\_minimal()

box\_glucose <- ggplot(data = diabetes, aes(x = Glucose)) + geom\_boxplot(color = "darkblue", fill = "lightblue") + labs(title = "Glucose Box Plot", x = "Glucose") +

theme\_minimal()

hist\_glucose + box\_glucose

`stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



# | Blood - Pressure Variable

diabetes$BloodPressure %>% summary()

Min. 1st Qu. Median Mean 3rd Qu. Max. 0.00 62.00 72.00 69.11 80.00 122.00

hist\_blood <- ggplot(data = diabetes, aes(x = BloodPressure)) + geom\_histogram(color = "darkblue", fill = "lightblue") +

labs(title = "BloodPressure Histogram Plot", x = "BloodPressure", y = "Count") + theme\_minimal()

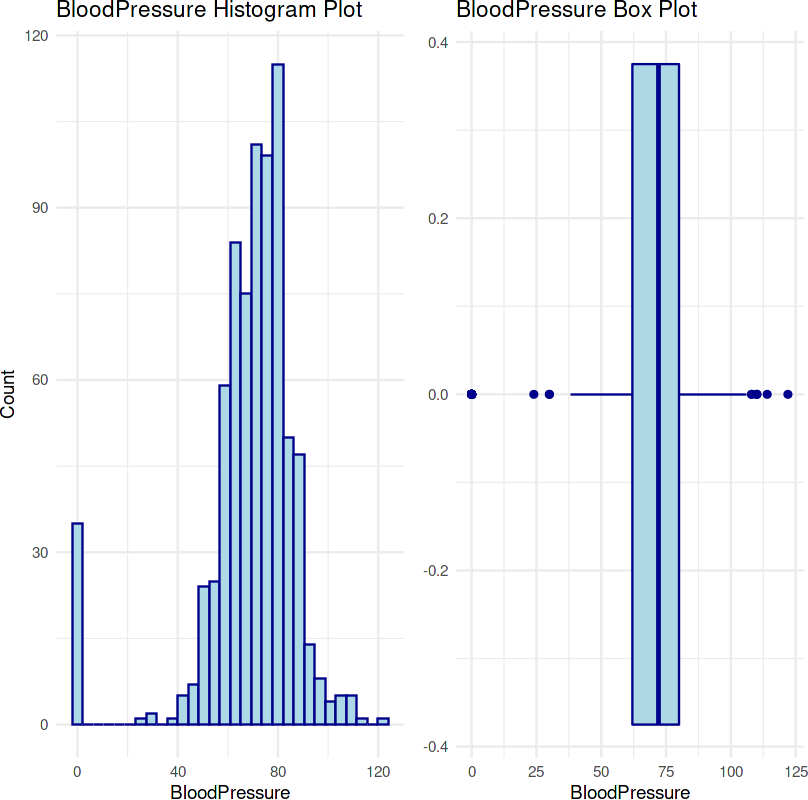
box\_blood <- ggplot(data = diabetes, aes(x = BloodPressure)) +

geom\_boxplot(color = "darkblue", fill = "lightblue") +

labs(title = "BloodPressure Box Plot", x = "BloodPressure") + theme\_minimal()

hist\_blood + box\_blood

`stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



# | SkinThickness Variable

diabetes$SkinThickness %>% summary()

Min. 1st Qu. Median Mean 3rd Qu. Max. 0.00 0.00 23.00 20.54 32.00 99.00

hist\_skin <- ggplot(data = diabetes, aes(x = SkinThickness)) + geom\_histogram(color = "darkblue", fill = "lightblue") +

labs(title = "SkinThickness Histogram Plot", x = "SkinThickness", y = "Count") + theme\_minimal()

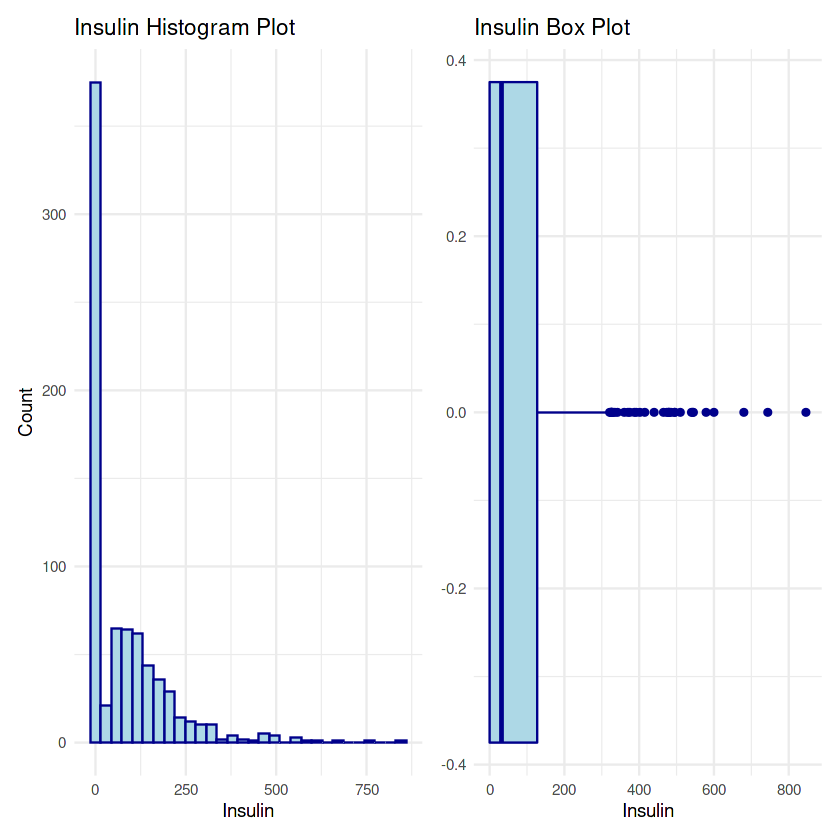
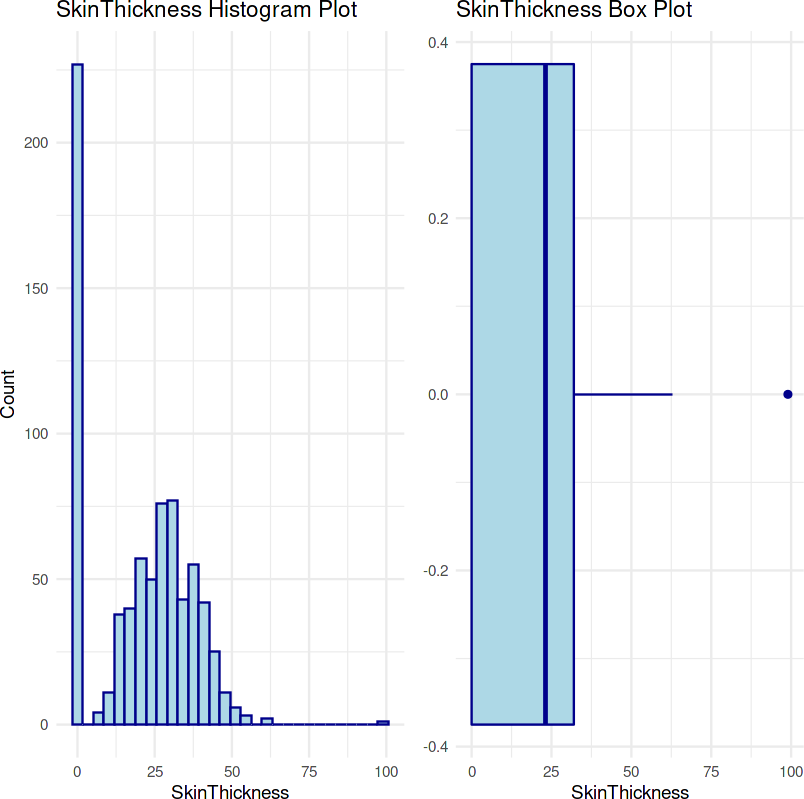
box\_skin <- ggplot(data = diabetes, aes(x = SkinThickness)) + geom\_boxplot(color = "darkblue", fill = "lightblue") +

labs(title = "SkinThickness Box Plot", x = "SkinThickness") + theme\_minimal()

hist\_skin + box\_skin

# | Insulin Variable

`stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



diabetes$Insulin %>% summary()

Min. 1st Qu. Median Mean 3rd Qu. Max. 0.0 0.0 30.5 79.8 127.2 846.0

hist\_insulin <- ggplot(data = diabetes, aes(x = Insulin)) + geom\_histogram(color = "darkblue", fill = "lightblue") +

labs(title = "Insulin Histogram Plot", x = "Insulin", y = "Count") + theme\_minimal()

box\_insulin <- ggplot(data = diabetes, aes(x = Insulin)) + geom\_boxplot(color = "darkblue", fill = "lightblue") + labs(title = "Insulin Box Plot", x = "Insulin") +

theme\_minimal()

hist\_insulin + box\_insulin

`stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.

# | BMI Variable

diabetes$BMI %>% summary()

Min. 1st Qu. Median Mean 3rd Qu. Max. 0.00 27.30 32.00 31.99 36.60 67.10

hist\_bmi <- ggplot(data = diabetes, aes(x = BMI)) +

geom\_histogram(color = "darkblue", fill = "lightblue") +

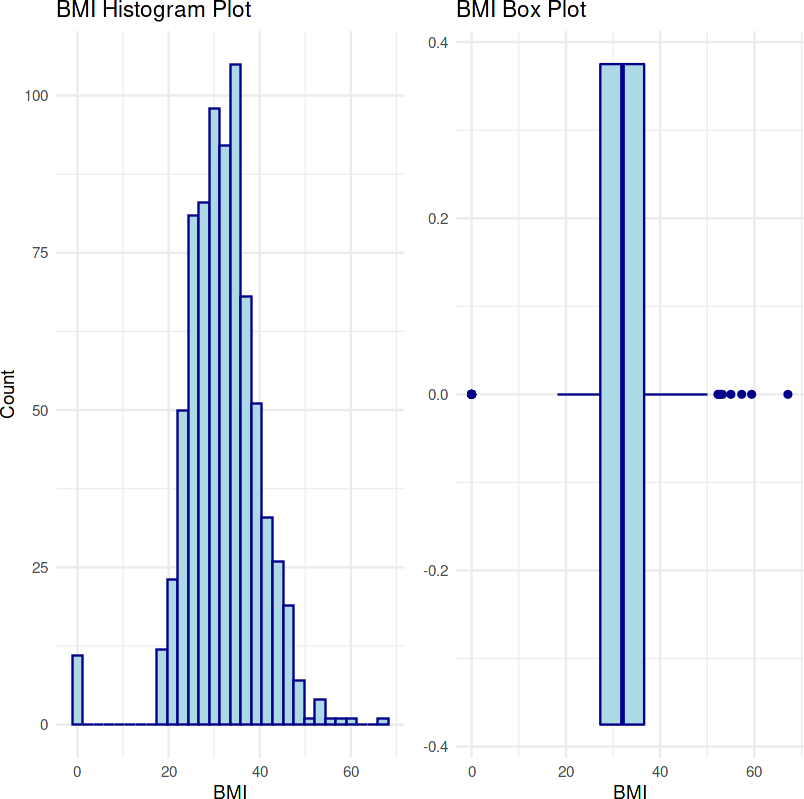
labs(title = "BMI Histogram Plot", x = "BMI", y = "Count") + theme\_minimal()

box\_bmi <- ggplot(data = diabetes, aes(x = BMI)) +

geom\_boxplot(color = "darkblue", fill = "lightblue") + labs(title = "BMI Box Plot", x = "BMI") +

theme\_minimal() hist\_bmi + box\_bmi

`stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



**3.7 | DiabetesPedigreeFunction Variable**

diabetes$DiabetesPedigreeFunction %>% summary()

Min. 1st Qu. Median Mean 3rd Qu. Max. 0.0780 0.2437 0.3725 0.4719 0.6262 2.4200

hist\_pedigree <- ggplot(data = diabetes, aes(x = DiabetesPedigreeFunction)) + geom\_histogram(color = "darkblue", fill = "lightblue") +

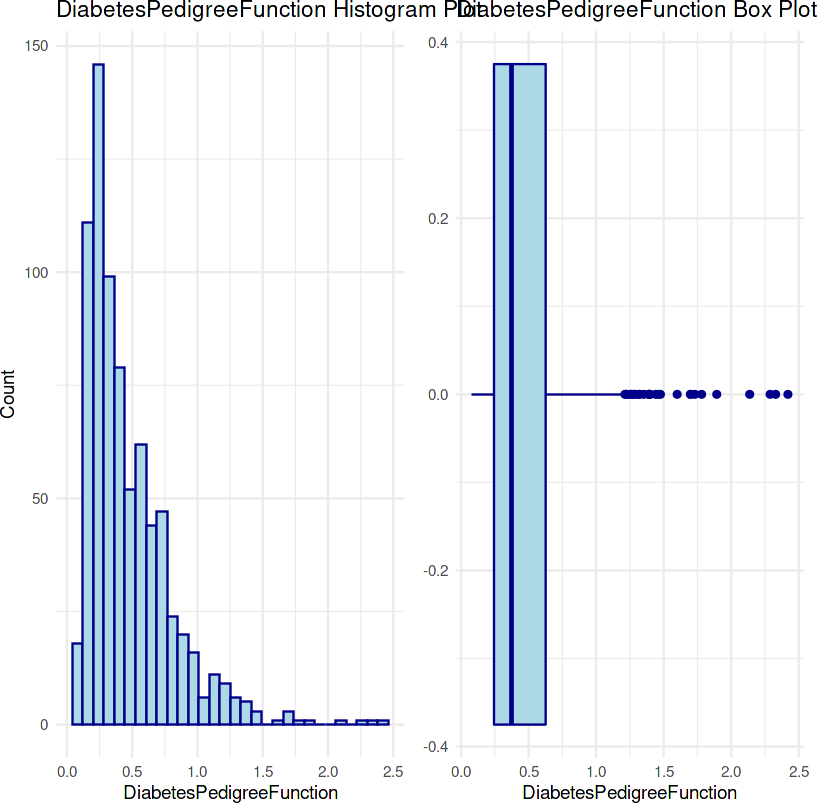
labs(title = "DiabetesPedigreeFunction Histogram Plot", x = "DiabetesPedigreeFunction", y = "Count") + theme\_minimal()

box\_pedigree <- ggplot(data = diabetes, aes(x = DiabetesPedigreeFunction)) + geom\_boxplot(color = "darkblue", fill = "lightblue") +

labs(title = "DiabetesPedigreeFunction Box Plot", x = "DiabetesPedigreeFunction") + theme\_minimal()

hist\_pedigree + box\_pedigree

`stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



# | Age Variable

diabetes$Age %>% summary()

Min. 1st Qu. Median Mean 3rd Qu. Max. 21.00 24.00 29.00 33.24 41.00 81.00

hist\_age <- ggplot(data = diabetes, aes(x = Age)) +

geom\_histogram(color = "darkblue", fill = "lightblue") +

labs(title = "Age Histogram Plot", x = "Age", y = "Count") +

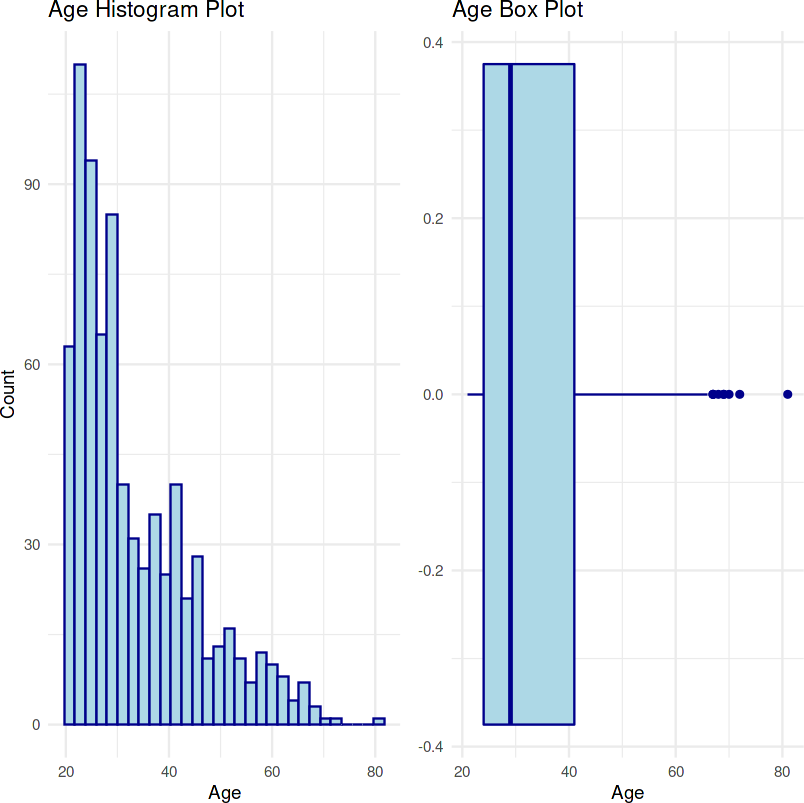
theme\_minimal()

box\_age <- ggplot(data = diabetes, aes(x = Age)) +

geom\_boxplot(color = "darkblue", fill = "lightblue") + labs(title = "Age Box Plot", x = "Age") +

theme\_minimal() hist\_age + box\_age

`stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



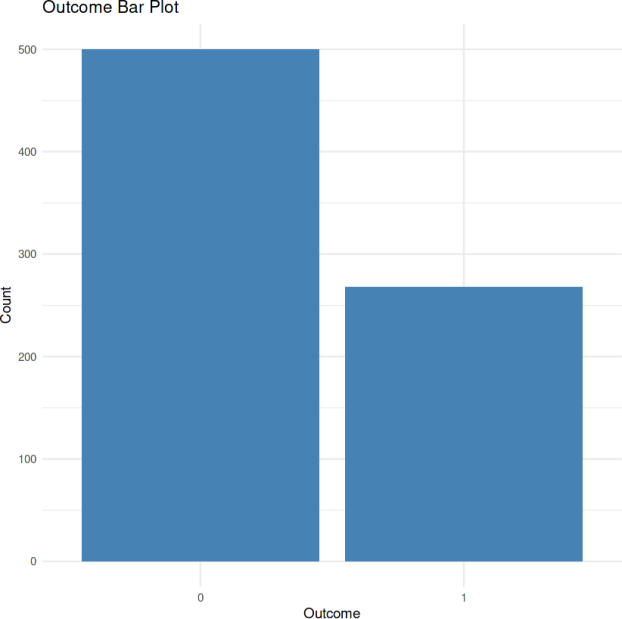
# | Outcome Variable

diabetes$Outcome %>% summary()

0: 500 1: 268

ggplot(data = diabetes, aes(x = Outcome)) + stat\_count(fill = "steelblue") +

labs(title = "Outcome Bar Plot", x = "Outcome", y = "Count") + theme\_minimal()



## **|** Outlier Analysis

Pregnancies\_out <- boxplot(diabetes$Pregnancies, plot = FALSE)

diabetes$Pregnancies[diabetes$Pregnancies <= Pregnancies\_out$stats[1]] <- Pregnancies\_out$stats[1]

diabetes$Pregnancies[diabetes$Pregnancies >= Pregnancies\_out$stats[5]] <- Pregnancies\_out$stats[5]

Glucose\_out <- boxplot(diabetes$Glucose, plot = FALSE)

diabetes$Glucose[diabetes$Glucose <= Glucose\_out$stats[1]] <- Glucose\_out$stats[1] diabetes$Glucose[diabetes$Glucose >= Glucose\_out$stats[5]] <- Glucose\_out$stats[5]

BloodPressure\_out <- boxplot(diabetes$BloodPressure, plot = FALSE)

diabetes$BloodPressure[diabetes$BloodPressure <= BloodPressure\_out$stats[1]] <- BloodPressure\_out$stats[1] diabetes$BloodPressure[diabetes$BloodPressure >= BloodPressure\_out$stats[5]] <- BloodPressure\_out$stats[5]

SkinThickness\_out <- boxplot(diabetes$SkinThickness, plot = FALSE)

diabetes$SkinThickness[diabetes$SkinThickness <= SkinThickness\_out$stats[1]] <- SkinThickness\_out$stats[1] diabetes$SkinThickness[diabetes$SkinThickness >= SkinThickness\_out$stats[5]] <- SkinThickness\_out$stats[5]

Insulin\_out <- boxplot(diabetes$Insulin, plot = FALSE)

diabetes$Insulin[diabetes$Insulin <= Insulin\_out$stats[1]] <- Insulin\_out$stats[1] diabetes$Insulin[diabetes$Insulin >= Insulin\_out$stats[5]] <- Insulin\_out$stats[5]

BMI\_out <- boxplot(diabetes$BMI, plot = FALSE)

diabetes$BMI[diabetes$BMI <= BMI\_out$stats[1]] <- BMI\_out$stats[1] diabetes$BMI[diabetes$BMI >= BMI\_out$stats[5]] <- BMI\_out$stats[5]

DiabetesPedigreeFunction\_out <- boxplot(diabetes$DiabetesPedigreeFunction, plot = FALSE)

diabetes$DiabetesPedigreeFunction[diabetes$DiabetesPedigreeFunction <= DiabetesPedigreeFunction\_out$stats[1]] <- DiabetesPedigreeFunction\_out diabetes$DiabetesPedigreeFunction[diabetes$DiabetesPedigreeFunction >= DiabetesPedigreeFunction\_out$stats[5]] <- DiabetesPedigreeFunction\_out

Age\_out <- boxplot(diabetes$Age, plot = FALSE)

diabetes$Age[diabetes$Age <= Age\_out$stats[1]] <- Age\_out$stats[1] diabetes$Age[diabetes$Age >= Age\_out$stats[5]] <- Age\_out$stats[5]

## **|** Training - Testing - Cross Validation

set.seed(123)

diabetes\_split <- initial\_split(diabetes, prop = 0.80) diabetes\_split

diabetes\_train <- training(diabetes\_split) diabetes\_test <- testing(diabetes\_split)

diabetes\_cv <- vfold\_cv(diabetes\_train, v = 10)

<Training/Testing/Total>

<614/154/768>

## **|** Pre - Processing

model\_recipe <-

recipe(Outcome ~ ., data = diabetes\_train) %>%

step\_mutate(age\_group = ifelse(Age %in% 21:35, 0, 1)) %>% step\_log(Age) %>%

step\_zv(all\_predictors())

## **|** Modelling with Support Vector Machine

svm\_model <-

svm\_rbf(mode = "classification", cost = tune(),

rbf\_sigma = tune(), engine = "kernlab"

)

set.seed(123) svm\_wf <-

workflow() %>%

add\_model(svm\_model) %>%

add\_recipe(model\_recipe) svm\_wf

svm\_results <- svm\_wf %>%

tune\_grid(resamples = diabetes\_cv,

metrics = metric\_set(accuracy)

)

svm\_results %>%

collect\_metrics()

param\_final <- svm\_results %>%

select\_best(metric = "accuracy") param\_final

svm\_wf <- svm\_wf %>%

finalize\_workflow(param\_final) svm\_wf

svm\_fit <- svm\_wf %>%

last\_fit(diabetes\_split)

test\_performance <- svm\_fit %>% collect\_predictions() test\_performance

diabetes\_metrics <- metric\_set(accuracy, f\_meas, precision, recall)

diabetes\_metrics(data = test\_performance, truth = Outcome, estimate = .pred\_class) conf\_mat(test\_performance, Outcome, .pred\_class)

══ Workflow ════════════════════════════════════════════════════════════════════

*Preprocessor:* Recipe

*Model:* svm\_rbf()

── Preprocessor ────────────────────────────────────────────────────────────────

3 Recipe Steps

* step\_mutate()
* step\_log()
* step\_zv()

── Model ─────────────────────────────────────────────────────────────────────── Radial Basis Function Support Vector Machine Model Specification (classification)

Main Arguments: cost = tune()

rbf\_sigma = tune()

Computational engine: kernlab

A tibble: 10 × 8

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **cost**  **<dbl>** | **rbf\_sigma**  **<dbl>** | **.metric**  **<chr>** | **.estimator**  **<chr>** | **mean**  **<dbl>** | **n**  **<int>** | **std\_err**  **<dbl>** | **.config**  **<chr>** |
| 0.012276419 | 2.219732e-03 | accuracy | binary | 0.6480434 | 10 | 0.02852269 | Preprocessor1\_Model01 |
| 12.623649481 | 2.132390e-10 | accuracy | binary | 0.6480434 | 10 | 0.02852269 | Preprocessor1\_Model02 |
| 0.003288151 | 4.829834e-01 | accuracy | binary | 0.6480434 | 10 | 0.02852269 | Preprocessor1\_Model03 |
| 0.063909911 | 4.134013e-04 | accuracy | binary | 0.6480434 | 10 | 0.02852269 | Preprocessor1\_Model04 |
| 9.957288438 | 7.305629e-02 | accuracy | binary | 0.7328133 | 10 | 0.01142637 | Preprocessor1\_Model05 |
| 0.041742599 | 7.065297e-06 | accuracy | binary | 0.6480434 | 10 | 0.02852269 | Preprocessor1\_Model06 |
| 0.207886080 | 5.613432e-05 | accuracy | binary | 0.6480434 | 10 | 0.02852269 | Preprocessor1\_Model07 |
| 3.309192835 | 4.183528e-08 | accuracy | binary | 0.6480434 | 10 | 0.02852269 | Preprocessor1\_Model08 |
| 0.876324914 | 3.298655e-07 | accuracy | binary | 0.6480434 | 10 | 0.02852269 | Preprocessor1\_Model09 |
| 0.002117010 | 1.591156e-09 | accuracy | binary | 0.6480434 | 10 | 0.02852269 | Preprocessor1\_Model10 |

A tibble: 1 × 3

**cost rbf\_sigma .config**

**<dbl> <dbl> <chr>**

9.957288 0.07305629 Preprocessor1\_Model05

══ Workflow ════════════════════════════════════════════════════════════════════

*Preprocessor:* Recipe

*Model:* svm\_rbf()

── Preprocessor ────────────────────────────────────────────────────────────────

3 Recipe Steps

* step\_mutate()
* step\_log()
* step\_zv()

── Model ─────────────────────────────────────────────────────────────────────── Radial Basis Function Support Vector Machine Model Specification (classification)

Main Arguments:

cost = 9.95728843750398

rbf\_sigma = 0.0730562947369229

Computational engine: kernlab

A tibble: 154 × 7

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **id**  **<chr>** | **.pred\_0**  **<dbl>** | **.pred\_1**  **<dbl>** | **.row**  **<int>** | **.pred\_class**  **<fct>** | **Outcome**  **<fct>** | **.config**  **<chr>** |
| train/test split | 0.3059271 | 0.69407293 | 1 | 1 | 1 | Preprocessor1\_Model1 |
| train/test split | 0.4432281 | 0.55677194 | 3 | 1 | 1 | Preprocessor1\_Model1 |
| train/test split | 0.2151097 | 0.78489030 | 9 | 1 | 1 | Preprocessor1\_Model1 |
| train/test split | 0.6908426 | 0.30915742 | 17 | 0 | 1 | Preprocessor1\_Model1 |
| train/test split | 0.7545850 | 0.24541496 | 22 | 0 | 0 | Preprocessor1\_Model1 |
| train/test split | 0.3133315 | 0.68666852 | 27 | 1 | 1 | Preprocessor1\_Model1 |
| train/test split | 0.9509001 | 0.04909991 | 28 | 0 | 0 | Preprocessor1\_Model1 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| train/test split | 0.2159895 | 0.78401052 | 42 | 1 | 0 | Preprocessor1\_Model1 |
| train/test split | 0.9054863 | 0.09451371 | 43 | 0 | 0 | Preprocessor1\_Model1 |
| train/test split | 0.1692705 | 0.83072954 | 44 | 1 | 1 | Preprocessor1\_Model1 |
| train/test split | 0.6854061 | 0.31459386 | 58 | 0 | 0 | Preprocessor1\_Model1 |
| train/test split | 0.8443322 | 0.15566782 | 60 | 0 | 0 | Preprocessor1\_Model1 |
| train/test split | 0.4813811 | 0.51861893 | 62 | 1 | 1 | Preprocessor1\_Model1 |
| train/test split | 0.9481018 | 0.05189818 | 63 | 0 | 0 | Preprocessor1\_Model1 |
| train/test split | 0.6697534 | 0.33024659 | 70 | 0 | 0 | Preprocessor1\_Model1 |
| train/test split | 0.8388929 | 0.16110706 | 77 | 0 | 0 | Preprocessor1\_Model1 |
| train/test split | 0.9392727 | 0.06072734 | 82 | 0 | 0 | Preprocessor1\_Model1 |
| train/test split | 0.7410516 | 0.25894843 | 86 | 0 | 0 | Preprocessor1\_Model1 |
| train/test split | 0.7123598 | 0.28764022 | 92 | 0 | 0 | Preprocessor1\_Model1 |
| train/test split | 0.6904879 | 0.30951215 | 93 | 0 | 0 | Preprocessor1\_Model1 |
| train/test split | 0.8758873 | 0.12411267 | 97 | 0 | 0 | Preprocessor1\_Model1 |
| train/test split | 0.8359555 | 0.16404447 | 99 | 0 | 0 | Preprocessor1\_Model1 |
| train/test split | 0.7429762 | 0.25702383 | 102 | 0 | 0 | Preprocessor1\_Model1 |
| train/test split | 0.9472030 | 0.05279697 | 107 | 0 | 0 | Preprocessor1\_Model1 |
| train/test split | 0.8449814 | 0.15501855 | 109 | 0 | 0 | Preprocessor1\_Model1 |
| train/test split | 0.8620482 | 0.13795184 | 123 | 0 | 0 | Preprocessor1\_Model1 |
| train/test split | 0.7115371 | 0.28846295 | 126 | 0 | 1 | Preprocessor1\_Model1 |
| train/test split | 0.8349618 | 0.16503823 | 140 | 0 | 0 | Preprocessor1\_Model1 |
| train/test split | 0.7261928 | 0.27380718 | 142 | 0 | 0 | Preprocessor1\_Model1 |
| ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ |
| train/test split | 0.7304997 | 0.26950031 | 626 | 0 | 0 | Preprocessor1\_Model1 |
| train/test split | 0.9180201 | 0.08197994 | 630 | 0 | 0 | Preprocessor1\_Model1 |
| train/test split | 0.9292570 | 0.07074302 | 634 | 0 | 0 | Preprocessor1\_Model1 |
| train/test split | 0.7328380 | 0.26716199 | 639 | 0 | 1 | Preprocessor1\_Model1 |
| train/test split | 0.6803634 | 0.31963660 | 642 | 0 | 0 | Preprocessor1\_Model1 |
| train/test split | 0.8168085 | 0.18319155 | 644 | 0 | 0 | Preprocessor1\_Model1 |
| train/test split | 0.5200999 | 0.47990006 | 647 | 1 | 1 | Preprocessor1\_Model1 |
| train/test split | 0.9328268 | 0.06717317 | 651 | 0 | 0 | Preprocessor1\_Model1 |
| train/test split | 0.9007564 | 0.09924363 | 655 | 0 | 0 | Preprocessor1\_Model1 |
| train/test split | 0.7537927 | 0.24620734 | 660 | 0 | 1 | Preprocessor1\_Model1 |
| train/test split | 0.3377125 | 0.66228748 | 676 | 1 | 1 | Preprocessor1\_Model1 |
| train/test split | 0.8036496 | 0.19635037 | 677 | 0 | 1 | Preprocessor1\_Model1 |
| train/test split | 0.9452092 | 0.05479077 | 680 | 0 | 0 | Preprocessor1\_Model1 |
| train/test split | 0.4817004 | 0.51829965 | 692 | 1 | 1 | Preprocessor1\_Model1 |
| train/test split | 0.2582896 | 0.74171038 | 703 | 1 | 1 | Preprocessor1\_Model1 |
| train/test split | 0.8850849 | 0.11491514 | 705 | 0 | 0 | Preprocessor1\_Model1 |
| train/test split | 0.8326606 | 0.16733936 | 715 | 0 | 0 | Preprocessor1\_Model1 |
| train/test split | 0.9182964 | 0.08170361 | 718 | 0 | 0 | Preprocessor1\_Model1 |
| train/test split | 0.7882675 | 0.21173246 | 720 | 0 | 1 | Preprocessor1\_Model1 |
| train/test split | 0.7166334 | 0.28336657 | 724 | 0 | 0 | Preprocessor1\_Model1 |
| train/test split | 0.7313455 | 0.26865447 | 726 | 0 | 0 | Preprocessor1\_Model1 |
| train/test split | 0.8065491 | 0.19345088 | 730 | 0 | 0 | Preprocessor1\_Model1 |
| train/test split | 0.8629751 | 0.13702488 | 732 | 0 | 1 | Preprocessor1\_Model1 |

