Data Mining Project

Dhruv Upadhyay – RA1811003010320

Vikash Arya – RA1811003010332

Shreyash Singh – RA1811003010342

21/08/2020

**Theoretical Analysis:**

1. NO. OF INSTANCE ARE 768 WITH 9 ROWS

The attributes and types are

preg float64

plas float64

pres float64

skin float64

insu float64

mass float64

pedi float64

age float64

class object

dtype: object

The Regression Analysis:

SUMMARY OUTPUT

**Regression Statistics:**

Multiple R 0.550684207

R Square 0.303253096

Adjusted R Square 0.295909255

Standard Error 0.400210451

Observations 768

**ANOVA:**

df SS MS F Significance F

Regression 8 52.91134742 6.613918427 41.29352749 7.35557E-55

Residual 759 121.5678192 0.160168405

Total 767 174.4791667

Coefficient STD. Error t-statistics P- value

Intercept -0.853894266 0.085484958 -9.988824781 3.70747E-22

preg 0.020591872 0.00512998 4.014026061 6.56146E-05

plas 0.005920273 0.000515123 11.49293763 2.69119E-28

pres -0.002331879 0.000811639 -2.873048753 0.004178788

skin 0.00015452 0.001112215 0.138929839 0.889542445

insu -0.000180535 0.000149819 -1.205020272 0.228571053

mass 0.013244031 0.00208776 6.34365595 3.85348E-10

pedi 0.147237439 0.045053885 3.268029785 0.001131733

age 0.002621394 0.00154864 1.69270703 0.09092163

1. Every data is related with the age. and the central tendency which we related is

according to the above regression analysis, the parameters that affect the patient for

Diabetes are Pregnancy, Plasma glucose concentration, Diastolic blood pressure, Body mass index, Diabetes pedigree function.

1. Yes there were data quality issue . at many place in plasma , insulin, Etc there were null or “0” value were placed . we have rectified that problem with replacing that null or “0” with the mean of

The whole col and replace it. The command we used is

df.loc[df['insu']<1,'insu']=np.mean(df['insu'])

1. Supervised Learning
2. [Supervised machine learning](https://in.mathworks.com/discovery/supervised-learning.html) builds a model that makes predictions based on evidence in the presence of uncertainty. A supervised learning algorithm takes a known set of input data and known responses to the data (output) and trains a model to generate reasonable predictions for the response to new data. Use supervised learning if you have known data for the output you are trying to predict.
3. Supervised learning uses classification and regression techniques to develop predictive models.
4. **Classification techniques** predict discrete responses—for example, whether an email is genuine or spam, or whether a tumor is cancerous or benign. Classification models classify input data into categories. Typical applications include medical imaging, speech recognition, and credit scoring.
5. Use classification if your data can be tagged, categorized, or separated into specific groups or classes. For example, applications for hand-writing recognition use classification to recognize letters and numbers. In image processing and computer vision, [unsupervised pattern recognition](https://in.mathworks.com/discovery/pattern-recognition.html) techniques are used for object detection and image segmentation.
6. Common algorithms for performing classification include [support vector machine (SVM)](https://in.mathworks.com/help/stats/support-vector-machine-classification.html), [boosted](https://in.mathworks.com/help/stats/classification-ensembles.html) and [bagged](https://in.mathworks.com/help/stats/classification-ensembles.html) [decision trees](https://in.mathworks.com/help/stats/classification-trees.html), [*k*-nearest neighbor](https://in.mathworks.com/help/stats/classification-nearest-neighbors.html), [Naïve Bayes](https://in.mathworks.com/help/stats/classification-naive-bayes.html), [discriminant analysis](https://in.mathworks.com/help/stats/classification-discriminant-analysis.html), [logistic regression](https://in.mathworks.com/help/stats/generalized-linear-regression-1.html), and [neural networks](https://in.mathworks.com/help/deeplearning/pattern-recognition-and-classification.html).

**Data Analysis Is As Below:**

**qwert <-** read.csv**("~/qwert.txt")**library**(dslabs)**library**(tidyverse)**

## -- Attaching packages --------------------------------------------------------------------------------- tidyverse 1.3.0 --

## v ggplot2 3.3.2 v purrr 0.3.4  
## v tibble 3.0.1 v dplyr 1.0.1  
## v tidyr 1.1.1 v stringr 1.4.0  
## v readr 1.3.1 v forcats 0.5.0

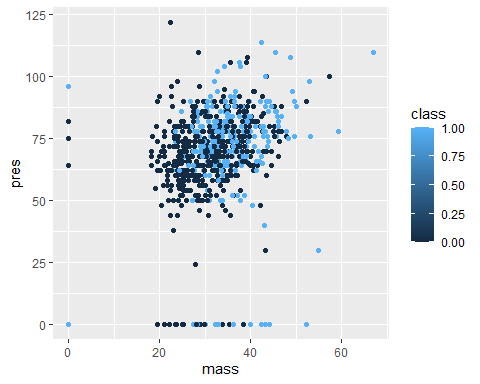
## -- Conflicts ------------------------------------------------------------------------------------ tidyverse\_conflicts() --  
## x dplyr::filter() masks stats::filter()  
## x dplyr::lag() masks stats::lag()

library**(dplyr)**library**(ggplot2)**summary**(qwert)**

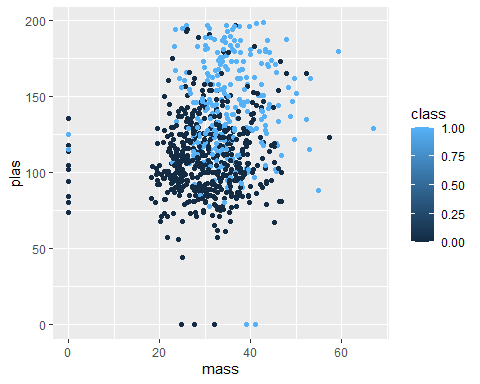
## preg plas pres skin   
## Min. : 0.000 Min. : 0.0 Min. : 0.00 Min. : 0.00   
## 1st Qu.: 1.000 1st Qu.: 99.0 1st Qu.: 62.00 1st Qu.: 0.00   
## Median : 3.000 Median :117.0 Median : 72.00 Median :23.00   
## Mean : 3.845 Mean :120.9 Mean : 69.11 Mean :20.54   
## 3rd Qu.: 6.000 3rd Qu.:140.2 3rd Qu.: 80.00 3rd Qu.:32.00   
## Max. :17.000 Max. :199.0 Max. :122.00 Max. :99.00   
## insu mass pedi 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   
## class   
## Length:768   
## Class :character   
## Mode :character   
##   
##   
##

**qwert**$**class <-** if\_else**(qwert**$**class** == **"tested\_positive",1,0,missing=NULL)  
positive <-** filter**(qwert, class**==**'1')**

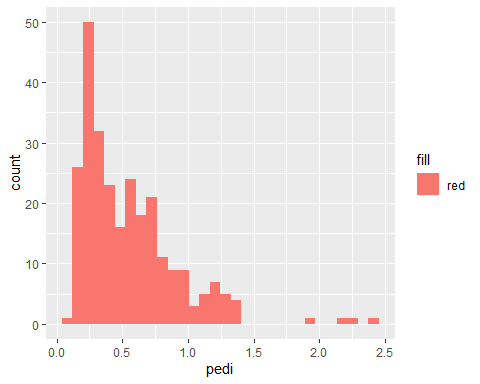
ggplot**(qwert,**aes**(x = mass,y = pres))** +geom\_point**(**aes**(col = class))**



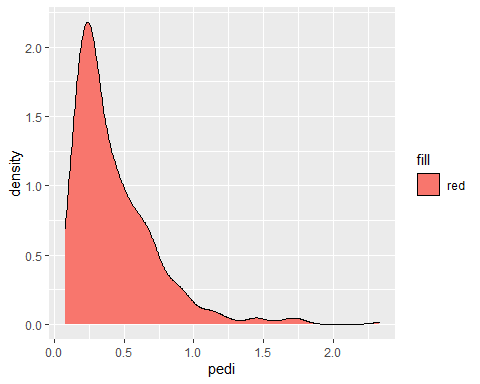
ggplot**(qwert,**aes**(x = mass,y = plas))** +geom\_point**(**aes**(col = class))**



**positive** %>%ggplot**(**aes**(x = pedi,fill = "red"))** +geom\_histogram**()**



**negative <-** filter**(qwert, class** == **'0')  
negative** %>%ggplo t**(**aes**(pedi,fill = "red"))** +geom\_density**()**



**m\_x <-** mean**(qwert**$**preg)  
m\_y <-** mean**(qwert**$**plas)  
r <-** cor**(qwert**$**preg,qwert**$**plas)  
s\_x <-** sd**(qwert**$**preg)  
s\_y <-** sd**(qwert**$**plas)**

**m1 <- r** \* **s\_y**/**s\_x  
n <- m\_y** -**m1**\***m\_x  
m2 <- r** \* **s\_x**/**s\_y  
o <- m\_x** -**m2**\***m\_y  
qwert** %>%ggplot**(**aes**(preg,plas))** +geom\_point**(alpha = 0.5)** +geom\_abline**(intercept = n, slope = m1, col = "blue")**

