STA 135: Multivariate Data Analysis

Prediction of the onset of diabetes in Female Pima Indians from medical record data

Professor: Xiucai Ding

TA: Rui Hu

Group Member:

Fangyi Su

Zhuoya Wang

I. Introduction

The Pima Indians Diabetes contains information collected from 768 female Pima Indians aged 21 years and above. The data includes 8 medical demographic attributes of Pima Indian women, who are known to have a high prevalence of diabetes. The attributes are the number of times pregnant, Plasma glucose concentration, diastolic blood pressure(mm Hg), triceps skin fold thickness(mm), 2-hour serum insulin (mu U/ml), body mass index(weight in kg/(height in m)^2), diabetes pedigree function, and age (years). The target variable is a binary indicator indicating whether or not the individual developed diabetes within five years of the data collection.

In recent decades, Diabetes has become one of the most prevalent diseases. According to the Centers for Disease Control and Prevention (CDC), in 2020, approximately 34.2 million people in the United States had diabetes, which accounts for around 10.5% of the population. This report will concentrate on the Classification of the onset of diabetes by Quadratic Discriminant Analysis (QDA).

After a series of data analyses, the final classification result helps in diagnosing diabetes, assessing the risk of developing diabetes in individuals based on their demographic and health-related characteristics to take active care aiming at preventing it in the future.

II. Data Preparation

	Pregnant	Glucose	Pressure
Min.	0.00	0.00	0.00
1st quartiles	1.00	99.0	62.00
Mean	3.00	120.9	69.11
3rd quartiles	6.00	140.2	80.00
Max	17.00	199.0	122.00

	Triceps	Insulin	Mass
Min.	0.00	0.00	0.00
1st quartiles	0.00	0.00	27.30
Mean	20.54	79.8	31.99
3rd quartiles	32.00	127.2	36.60
Max	99.00	846.0	67.10

	Pedigree	Age
Min.	0.0780	21.00
1st quartiles	0.2437	24.00
Mean	0.4719	33.24
3rd quartiles	0.6262	41.00
Max	2.4200	81.00

Table 1: Summary of Data (raw data)

The above table presents a summary of 8 attributes in the diabetes dataset. Given that values of zero for attributes such as **Plasma glucose concentration, diastolic blood pressure, triceps skin fold thickness, 2-hour serum insulin, and body mass index** are highly unlikely in real-life scenarios, it is reasonable to assume that these zero values represent potential missing data or data entry errors in the dataset during data analysis and preprocessing. Compared to the mean, the median is a more robust statistic which is less affected by the outliers. Replacing the missing values with medians helps to preserve the shape of the original data, and make it aligned with the real values. Thus, we are supposed to replace those values with each attribute's median to prevent poor decision-making.

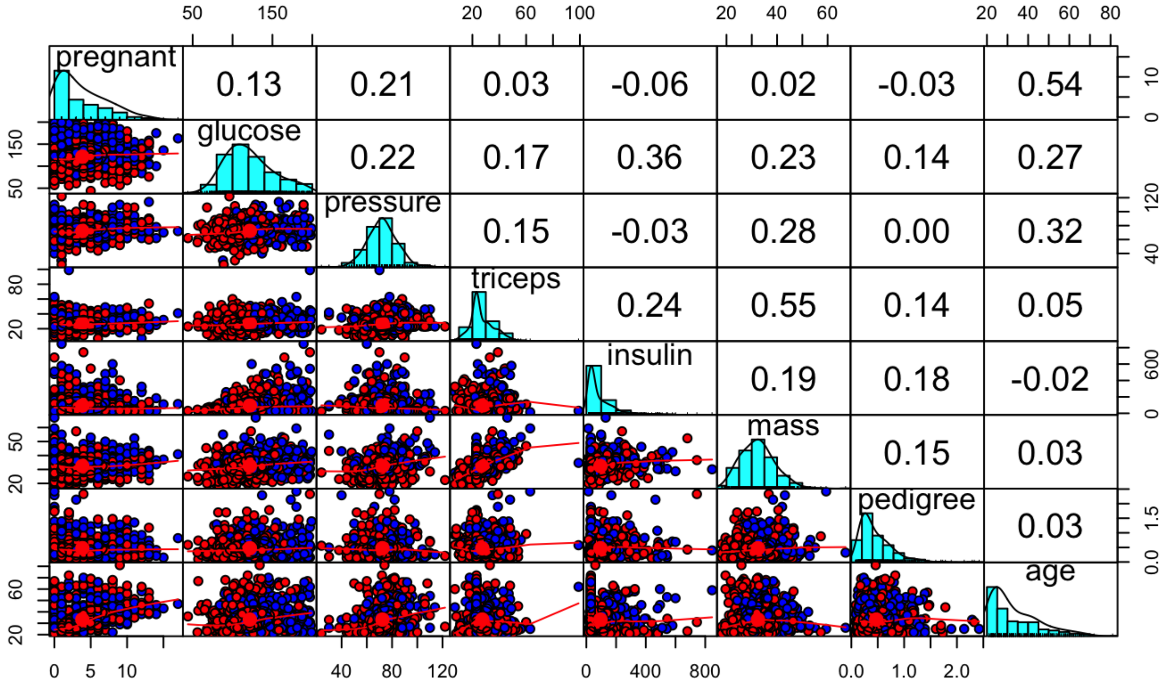


Figure 1: Data Visualization (after replace 0 with median)

The above plot matrix can be divided into three parts. The first part is the scatterplot where each panel represents a scatterplot of two attributes. The points in the scatterplots are coloured based on the onset of diabetes, where 0 indicates the absence of diabetes and 1 indicates the presence of diabetes and will be represented by filled circles red and blue respectively. We can observe that the scatter plots of the paired attributes are mixed and there is no clear distinction between two classes(whether having diabetes or not) based on two attributes. The second part is the diagonal which presents the distribution of each attribute. We can notice the distribution of each variable (attribute) is approximately Gaussian, so these attributes meet the assumption of multivariate normality when using Linear Discriminant Analysis (LDA) or Quadratic Discriminant Analysis (QDA).

The third part is the numbers where each value suggests the correlation between two attributes(variables). It quantifies the strength of the relationship between variables, with values ranging from -1 to 1. A higher absolute value indicates a stronger correlation. A positive correlation suggests that as one attribute increases, the other also tends to increase, while a negative correlation indicates that as one attribute increases, the other tends to decrease.

III. Method Selection

When we have a series of predictors and want to use them to separate them into two classifications, we can use Linear Discriminant Analysis (LDA) to differentiate them. Quadratic Discriminant Analysis (QDA) works in a similar way to LDA but it adapts to the situation that each class has their own covariance matrix.

Since LDA assumes that all population covariance matrices are the same, we must first check the hypothesis ($H_0 : \Sigma_0 = \Sigma_1$) by using Box's M Test to determine whether the dataset is amenable to LDA. Since the p-value (<2.2e-26) is smaller than any significance level, we reject the null hypothesis and conclude that the covariances of the two groups of diabetes are different. Thus, we are supposed to apply QDA, which has no mandatory requirements for covariance matrices, for this dataset.

IV. Quadratic discriminant analysis(QDA)

After dividing the entire dataset into a 70% training dataset and a 30% testing dataset, we utilize the QDA function along with the training data to obtain the following estimators. Then, we have estimators for the prior probabilities $\widehat{\boldsymbol{\pi}_0} = 0.6518$ and $\widehat{\boldsymbol{\pi}_1} = 0.3482$; the estimators for the mean vectors $\widehat{u_0}$ and $\widehat{u_1}$; the estimators for population covariance matrices $\widehat{\boldsymbol{\Sigma}_0}$ and $\widehat{\boldsymbol{\Sigma}_1}$ for each group. (the exact value of mean vectors and population covariance matrices are presented in appendix)

Quadratic discriminant function:

$$\boldsymbol{\delta}_k(x) = -0.5 log|\boldsymbol{\Sigma}_k| - 0.5(x - u_k)^T \boldsymbol{\Sigma}_k^{-1}(x - u_k) + log\pi_k$$

Decision Boundary ($\boldsymbol{\delta}_0(x)$ - $\boldsymbol{\delta}_1(x)$) :

$$f(x) = 0.5x^T(\widehat{\boldsymbol{\Sigma}_1}^{-1} - \widehat{\boldsymbol{\Sigma}_0}^{-1})x + x^T(\widehat{\boldsymbol{\Sigma}_0}^{-1}\widehat{u_0} - \widehat{\boldsymbol{\Sigma}_1}^{-1}\widehat{u_1}) + 7.704483$$

Since the decision boundary is the difference of the quadratic discriminant function for two groups, to evaluate whether a new patient has diabetes based on eight attributes, we can plug the values of those attributes into the decision boundary. If the resulting value is greater than 0, indicating that $\boldsymbol{\delta}_0(x)$ is greater than $\boldsymbol{\delta}_1(x)$, we can confirm that, based on these eight attributes and our model, the patient does not have diabetes. Conversely, If the resulting value is less than 0, indicating that $\boldsymbol{\delta}_1(x)$ is greater than $\boldsymbol{\delta}_0(x)$, we can conclude that the patient does not have diabetes.

Predicted	**Actual**	
	neg	pos
neg	308	76
pos	42	111

The overall error rate for training data: 1-(308+111)/(308+76+42+111)=0.22

Table 2: Confusion Matrix for Training Data

Predicted	**Actual**	
	neg	pos
neg	127	40
pos	23	41

The overall error rate for testing data: 1-(127+41)/(127+40+23+41) = 0.273

Table 3: Confusion Matrix for Testing Data

V. Interpretation

The prior probabilities of groups represent the proportions of each group in the training dataset. 65.18% of all observations in the training set were of the 'neg'(0) group and 34.82% of all observations in the training dataset were of the 'pos'(1) group. 22.0% refers to the error rate calculated on the data that was used to train the model. It measures how well the model fits the training data. 27.3% refers to the error rate calculated on the test dataset, which is not seen by the model during training. The purpose of the test dataset is to evaluate how well the model generalizes to unseen data. Since the difference between the two error rates is in an acceptable range and the error rate for training data is also lower than that of testing data, it indicates good generalization. This suggests that the model is learning the underlying patterns in the data without overfitting.

Varying the data splits may lead to different parameters for the decision boundary, but it has a minimal impact on the error rate for our model. When we fix the ratio of the training dataset to the testing dataset to 7:3, we observe that the error rate remains nearly the same every time we run the data. Additionally, when we further modify the ratio to 6:4 and 8:2, the error rate shows a slight variation but still remains at a similar level. These findings indicate that the performance of QDA is consistently good across different ratios, suggesting its reliability and effectiveness. However, further improvements can still be made to reduce both error rates if necessary.

VI. Conclusion

The above result demonstrates a stable and strong performance of QDA method in binary classification, exhibiting a consistent error rate even when the ratio of training and testing data is altered. Thus, we can conclude that the QDA method is an effective classifier for diabetes classification with the eight indicators. However, the following improvements can be made to help reduce both error rates. Firstly, before applying LDA/QDA, we can use the Principal component analysis (PCA) to reduce the dimensions and extract the number of features of the dataset to figure out the most important attributes related to diabetes. Also, we can improve the accuracy of estimation by collecting more data to train the models which can capture a border range of patterns. Lastly, increasing the size and diversity of the data exposes the model to a wider range of examples, which helps it learn more generalized patterns.

It is strongly recommended that individuals prioritize regular physical health check-ups and place increased emphasis on the aforementioned attributes. This proactive approach to healthcare plays a crucial role in preventing the diagnosis of diabetes and promotes overall well-being.

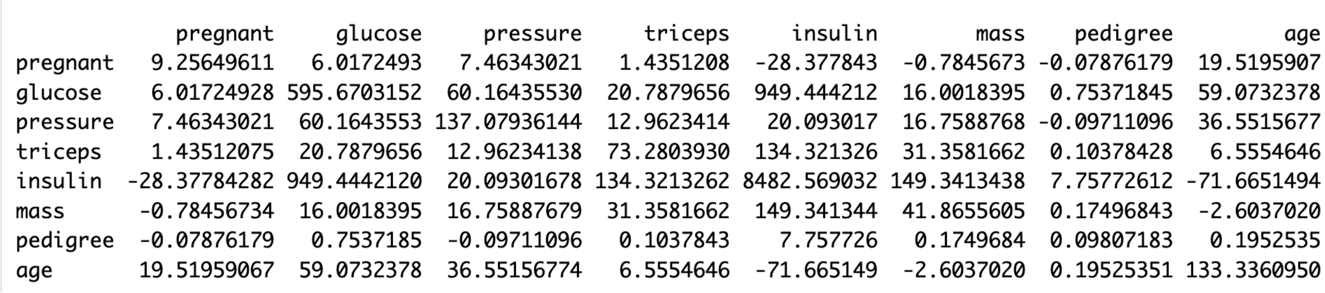
Distribution: Fangyi Su : Coding part + Analysis(equally division +conclusion)
Zhuoya Wang: Coding part + Analysis (equally division +Interpretation)

Appendix

$\widehat{u_0}$: [3.322857 111.460 70.38286 26.14286 83.12714 30.77800 0.4399457 31.18286]^T

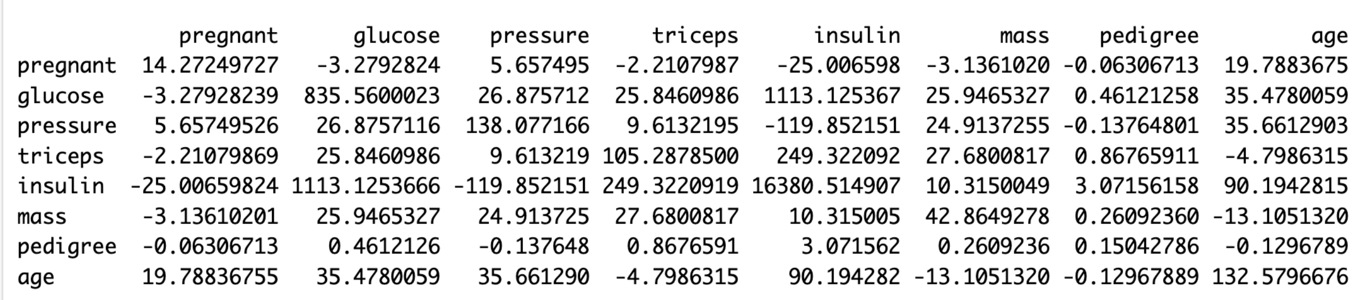
$\widehat{u_1}$: [4.754011 143.262 74.94118 30.27807 117.63636 35.58663 0.5543155 36.81818]^T

$\widehat{\boldsymbol{\Sigma}_0}$:



```
            pregnant     glucose     pressure     triceps     insulin        mass    pedigree         age
pregnant  9.25649611   6.0172493   7.46343021   1.4351208  -28.377843  -0.7845673 -0.07876179  19.5195907
glucose   6.01724928 595.6703152  60.16435530  20.7879656  949.444212  16.0018395  0.75371845  59.0732378
pressure  7.46343021  60.1643553 137.07936144  12.9623414   20.093017  16.7588768 -0.09711096  36.5515677
triceps   1.43512075  20.7879656  12.96234138  73.2803930  134.321326  31.3581662  0.10378428   6.5554646
insulin -28.37784282 949.4442120  20.09301678 134.3213262 8482.569032 149.3413438  7.75772612 -71.6651494
mass     -0.78456734  16.0018395  16.75887679  31.3581662  149.341344  41.8655605  0.17496843  -2.6037020
pedigree -0.07876179   0.7537185  -0.09711096   0.1037843    7.757726   0.1749684  0.09807183   0.1952535
age      19.51959067  59.0732378  36.55156774   6.5554646  -71.665149  -2.6037020  0.19525351 133.3360950
```

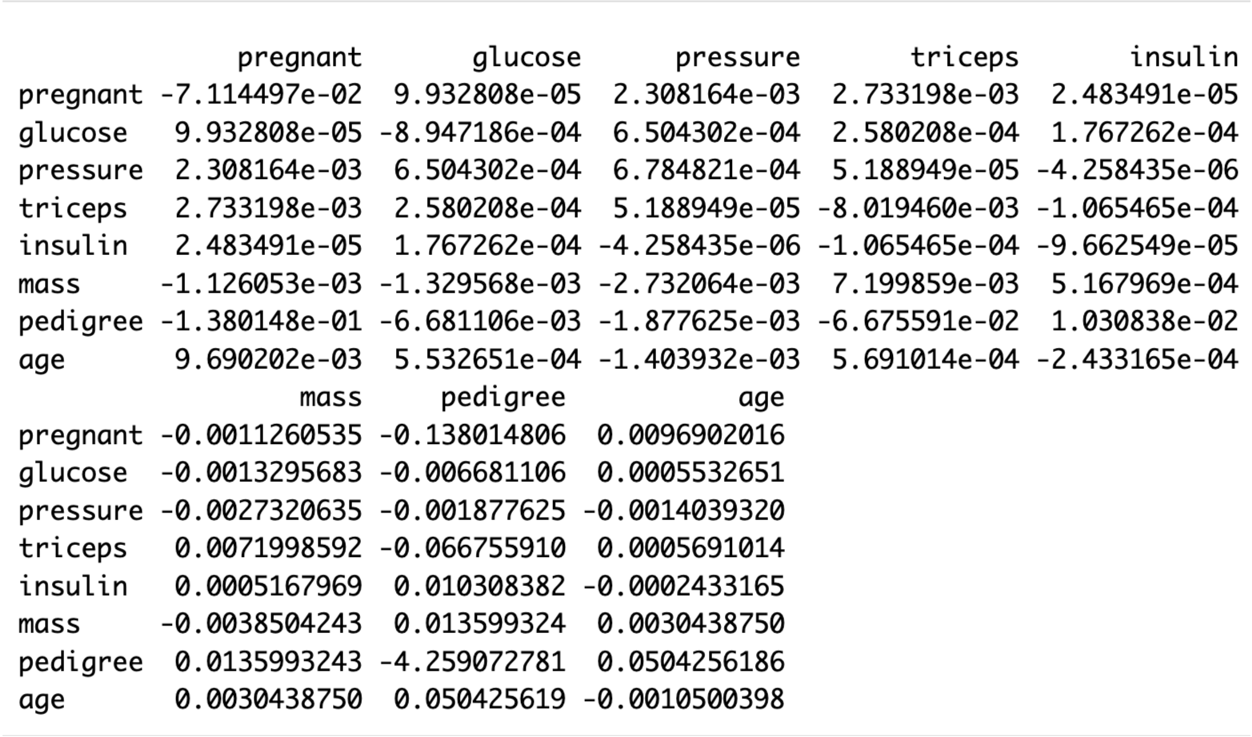
$\widehat{\boldsymbol{\Sigma}_1}$:



```
            pregnant      glucose   pressure     triceps      insulin        mass    pedigree         age
pregnant  14.27249727   -3.2792824   5.657495  -2.2107987   -25.006598  -3.1361020 -0.06306713  19.7883675
glucose   -3.27928239  835.5600023  26.875712  25.8460986  1113.125367  25.9465327  0.46121258  35.4780059
pressure   5.65749526   26.8757116 138.077166   9.6132195  -119.852151  24.9137255 -0.13764801  35.6612903
triceps   -2.21079869   25.8460986   9.613219 105.2878500   249.322092  27.6800817  0.86765911  -4.7986315
insulin  -25.00659824 1113.1253666 -119.852151 249.3220919 16380.514907  10.3150049  3.07156158  90.1942815
mass      -3.13610201   25.9465327  24.913725  27.6800817    10.315005  42.8649278  0.26092360 -13.1051320
pedigree  -0.06306713    0.4612126  -0.137648   0.8676591     3.071562   0.2609236  0.15042786  -0.1296789
age       19.78836755   35.4780059  35.661290  -4.7986315    90.194282 -13.1051320 -0.12967889 132.5796676
```

Some variables in the Decision boundary:

$\widehat{\boldsymbol{\Sigma}_1}^{-1} - \widehat{\boldsymbol{\Sigma}_0}^{-1}$:



```
             pregnant       glucose      pressure      triceps       insulin
pregnant -7.114497e-02  9.932808e-05  2.308164e-03  2.733198e-03  2.483491e-05
glucose   9.932808e-05 -8.947186e-04  6.504302e-04  2.580208e-04  1.767262e-04
pressure  2.308164e-03  6.504302e-04  6.784821e-04  5.188949e-05 -4.258435e-06
triceps   2.733198e-03  2.580208e-04  5.188949e-05 -8.019460e-03 -1.065465e-04
insulin   2.483491e-05  1.767262e-04 -4.258435e-06 -1.065465e-04 -9.662549e-05
mass     -1.126053e-03 -1.329568e-03 -2.732064e-03  7.199859e-03  5.167969e-04
pedigree -1.380148e-01 -6.681106e-03 -1.877625e-03 -6.675591e-02  1.030838e-02
age       9.690202e-03  5.532651e-04 -1.403932e-03  5.691014e-04 -2.433165e-04
                 mass     pedigree           age
pregnant -0.0011260535 -0.138014806  0.0096902016
glucose  -0.0013295683 -0.006681106  0.0005532651
pressure -0.0027320635 -0.001877625 -0.0014039320
triceps   0.0071998592 -0.066755910  0.0005691014
insulin   0.0005167969  0.010308382 -0.0002433165
mass     -0.0038504243  0.013599324  0.0030438750
pedigree  0.0135993243 -4.259072781  0.0504256186
age       0.0030438750  0.050425619 -0.0010500398
```

$\widehat{\boldsymbol{\Sigma}_0}^{-1}\widehat{u_0} - \widehat{\boldsymbol{\Sigma}_1}^{-1}\widehat{u_1}$:

[-0.3073651, 0.02459053, 0.004757732, -0.03690842, -0.02099616, 0.02529609, 1.584619, -0.1055248]

R code:

```
##Packages
#install.packages("mlbench")
#install.packages("broom")
#install.packages("klaR")
#install.packages("psych")
#install.packages("devtools")
#install.packages("usethis")
###Real
library(mlbench)
library(mvtnorm)
library(klaR)
library(psych)
library(MASS)
#library(ggord)
library(devtools)
library(heplots)

data("PimaIndiansDiabetes")

pima = PimaIndiansDiabetes

#pima$diabetes = ifelse(pima$diabetes == 'pos', 1, 0)
#pima$diabetes = as.factor(pima$diabetes)

########statistical summary
#missing_valus = sum(is.na(pima)) ### we dont have missing data

#nearly double the number of observations with class 0 (no onset of diabetes) than there are
with class 1 (onset of diabetes)

### data summary
summary(pima) ####
#install.packages("matrixStats")
library(matrixStats)
attribute_medians <- colMedians(as.matrix(pima[, c("pregnant", "glucose", "pressure",
"triceps", "insulin", "mass", "pedigree", "age")]))
```

```
pima_filled <- pima
##glucose replaced with median
pima_filled$glucose <- ifelse(pima$glucose == 0, attribute_medians["glucose"],
pima$glucose)

#pressure replaced with median
pima_filled$pressure <- ifelse(pima$pressure == 0, attribute_medians["pressure"],
pima$pressure)

###triceps replaced with median
pima_filled$triceps <- ifelse(pima$triceps == 0, attribute_medians["triceps"], pima$triceps)


###insulin replaced with median
pima_filled$insulin <- ifelse(pima$insulin == 0, attribute_medians["insulin"], pima$insulin)

##mass replaced with median
pima_filled$mass<- ifelse(pima$mass== 0, attribute_medians["mass"], pima$mass)

head(pima_filled)
group0 <- subset(pima_filled, diabetes == "neg")
group1 <- subset(pima_filled, diabetes == "pos")

mydata0 = group0[1:8]
mydata1 = group1[1:8]

s0 = round(cov(mydata0),2)
s1 = round(cov(mydata1),2)

##### Data visilization
pairs.panels(pima_filled[1:8],
          gap = 0,
          bg = c("red", "blue")[pima_filled$diabetes],
          pch = 21)

#### Box'M tst.  LDA condition test <--- failed
```

```
res <- boxM(pima_filled[, 1:8], pima_filled[, "diabetes"])
res #### Small pvalue, reject the null hypothesis and we are suppose to use ADQ

summary(res)
library(ggplot2)
library(MASS)
set.seed(123)  # For reproducibility
train_idx <- sample(1:nrow(pima_filled), 0.7 * nrow(pima_filled)) ##### Lower error rate
train_data <- pima_filled[train_idx, ]
test_data <- pima_filled[-train_idx, ]

qda_model <- qda(diabetes ~ ., data = train_data)
qda_model

p1_qda <- predict(qda_model, train_data)$class
tab_qda <- table(Predicted = p1_qda, Actual = train_data$diabetes)
tab_qda

p2_qda <- predict(qda_model, test_data)$class
tab1_qda <- table(Predicted = p2_qda, Actual = test_data$diabetes)
tab1_qda

sum(diag(tab_qda))/sum(tab_qda)
sum(diag(tab1_qda))/sum(tab1_qda)
##### estimator for decision boundary
c0 = subset(train_data, train_data$diabetes == 'neg')
c1 = subset(train_data, train_data$diabetes == 'pos')

c0_data = c0[,1:8]
c1_data = c1[,1:8]

s0 = cov(c0_data)
s1 = cov(c1_data)

colMeans(c0_data)
colMeans(c1_data)
```

```
L1 = 0.5*log(det(cov(c1_data))/det(cov(c0_data)))
L2 = 0.5*(t(colMeans(c1_data))%*%solve(cov(c1_data))%*%(colMeans(c1_data))-t(colMeans(c0_data))%*%solve(cov(c0_data))%*%(colMeans(c0_data)))

L1 + L2
b1 = colMeans(c0_data)%*%solve(cov(c0_data))
b1

b2 = colMeans(c1_data)%*%solve(cov(c1_data))
b2

b1-b2
a = solve(cov(c1_data))-solve(cov(c0_data))
a
#### decision boundary
#f(x) = 0.5*t(x)*a*x + t(x)*(b1-b2)+(L1+L2)
```