Project 1: SEMMA with Regularized Logistic Regression

DS 5494 - Statistical Machine Learning II

Willliam Ofosu Agyapong[*] University of Texas at El Paso (UTEP)

September 06, 2022

Contents

[*] woagyapong@miners.utep.edu

1 Importing the diabetes dataset

```
# importing data
diabetes <- readr::read_csv("diabetes_data_upload.csv")

dim(diabetes)
```

```
## [1] 520  17
```

The diabetes data set consists of **17** variables with **520** observations. Below is a snapshot of the data revealing the first 10 observations.

```
kable(head(diabetes, 10), booktabs=T, linesep="", align = "c",
      caption = "First 10 observations from the data") %>%
kable_styling(latex_options = c("scale_down", "HOLD_position"))
```

Table 1: First 10 observations from the data

Age	Gender	Polyuria	Polydipsia	sudden weight loss	weakness	Polyphagia	Genital thrush	visual blurring	Itching	Irritability	delayed healing	partial paresis	muscle stiffness	Alopecia	Obesity	class
40	Male	No	Yes	No	Yes	No	No	No	Yes	No	Yes	No	Yes	Yes	Yes	Positive
58	Male	No	No	No	Yes	No	No	Yes	No	No	No	Yes	No	Yes	No	Positive
41	Male	Yes	No	No	Yes	Yes	No	No	Yes	No	Yes	No	Yes	Yes	No	Positive
45	Male	No	No	Yes	Yes	Yes	Yes	No	Yes	No	Yes	No	No	No	No	Positive
60	Male	Yes	Yes	Yes	Yes	Yes	No	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Positive
55	Male	Yes	Yes	No	Yes	Yes	No	Yes	Yes	No	Yes	No	Yes	Yes	Yes	Positive
57	Male	Yes	Yes	No	Yes	Yes	Yes	No	No	No	Yes	Yes	No	No	No	Positive
66	Male	Yes	Yes	Yes	Yes	No	No	Yes	Yes	Yes	No	Yes	Yes	No	No	Positive
67	Male	Yes	Yes	No	Yes	Yes	Yes	No	Yes	Yes	No	Yes	Yes	No	Yes	Positive
70	Male	No	Yes	Yes	Yes	Yes	No	Yes	Yes	Yes	No	No	No	Yes	No	Positive

2 Exploratory Data Analysis

In this step, we explore the data by inspecting the variable types, outlying and possibly wrong records, and other issues.

2.1 Variable types

The table below shows the variable types and unique values for each of the 17 variables. We observe that all the variables, except **Age** being numeric (continuous), are dichotomous qualitative or categorical variables. The ages of the patients ranges between 16 and 90 years.

```
output <- NULL
for(i in seq_along(diabetes)) {
  output <- rbind(output, c(names(diabetes)[i],
                  class(diabetes[[i]]),
                  paste(sort(unique(diabetes[[i]])), collapse = ", "))
                  )
}

as.data.frame(output) %>%
  kable(booktabs=T, linesep="",
        col.names = c("Variable Name", "Type", "Unique values"))%>%
  column_spec(1, '10em') %>%
  column_spec(2, '5em') %>%
  column_spec(3, '20em') %>%
kable_styling(latex_options = c("HOLD_position"))
```

Variable Name	Type	Unique values
Age	numeric	16, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 72, 79, 85, 90
Gender	character	Female, Male
Polyuria	character	No, Yes
Polydipsia	character	No, Yes
sudden weight loss	character	No, Yes
weakness	character	No, Yes
Polyphagia	character	No, Yes
Genital thrush	character	No, Yes
visual blurring	character	No, Yes
Itching	character	No, Yes
Irritability	character	No, Yes
delayed healing	character	No, Yes
partial paresis	character	No, Yes
muscle stiffness	character	No, Yes
Alopecia	character	No, Yes
Obesity	character	No, Yes
class	character	Negative, Positive

2.2 Inspecting distinct values of each variable

In this subsection, I investigated the distinct values of each variable as an attempt to identifying any unusual values or errors. The outputs showed nothing concerning. For brevity in reporting, the outputs were suppressed but the codes used are presented as follows.

```
cols <- 1:NCOL(diabetes)
for (j in cols){
  x <- diabetes[,j]
  print(names(diabetes)[j])
  print(sort(unique(x, incomparables=T)))
  print(table(x, useNA="ifany"))
}
```

2.3 Distribution of target variable

```
library(gtsummary)
diabetes %>%
  dplyr::select(class) %>%
  tbl_summary() %>%
  modify_caption("Frequency distribution of the target variable class") %>%
  modify_footnote(c(all_stat_cols()) ~ NA) %>%
  bold_labels() %>%
  modify_header(label="**Target Variable**") %>%
  as_kable_extra(booktabs=T) %>%
  kable_classic()%>%
kable_styling(latex_options = c("HOLD_position"))
```

Table 2: Frequency distribution of the target variable class

Target Variable	**N = 520**
class	
Negative	200 (38%)
Positive	320 (62%)

With 62% and 38% observations representing the positive and negative class, respectively, there imbalance. However, I do not consider this to be a serious unbalanced classification problem.

2.4 Checking for missing values

```
# inspecting missing values using the "naniar" package
naniar::miss_var_summary(diabetes) %>%
  kable(booktabs = T, linesep="", align = "lcc",
        col.names = c("Variable", "Number missing", "Percent missing"),
        cap = "Amount of missing values in the diabetes dataset") %>%
kable_styling(latex_options = c("HOLD_position"))
```

Table 3: Amount of missing values in the diabetes dataset

Variable	Number missing	Percent missing
Age	0	0
Gender	0	0
Polyuria	0	0
Polydipsia	0	0
sudden weight loss	0	0
weakness	0	0
Polyphagia	0	0
Genital thrush	0	0
visual blurring	0	0
Itching	0	0
Irritability	0	0
delayed healing	0	0
partial paresis	0	0
muscle stiffness	0	0
Alopecia	0	0
Obesity	0	0
class	0	0

Clearly, the data set has no missing values.

3 Variable Screening

3.1 Association between class and continuous predictors

Treating Age as a continuous predictor, the figure below shows how the distribution of Age varies between each level of the class variable. The p-value associated with the nonparametric Wilcoxon rank-sum test is displayed in red. A nonparametric testing procedure was adopted because Age does not appear to be symmetrically (normally) distributed between the two groups. We observe two outliers for the positive class.

The small p-value ($0.012 < 0.25$) shows that there exists statistically significant association between the Age and class of the patients.

```
# diabetes %>%
#   ggplot(aes(Age, fill=class)) +
#     geom_density(alpha=0.4) +
#     scale_fill_manual(values = c("pink", "dodgerblue")) +
#     # scale_color_manual(values = c("pink", "dodgerblue"), guide='none') +
#     ggpubr::stat_compare_means(label.sep = " | ", vjust = 1, color = "red",
#                                method = "t.test", paired = F)



ggplot(diabetes, aes(class, Age, fill = class)) +
      geom_boxplot(alpha = 0.3) + xlab("") +
      scale_fill_manual(values = c("pink", "dodgerblue")) +
      theme(legend.position = "none") +
      ggpubr::stat_compare_means(label.sep = " | ", vjust = 1, color = "red",
                                 method = "wilcox", paired = F)
```

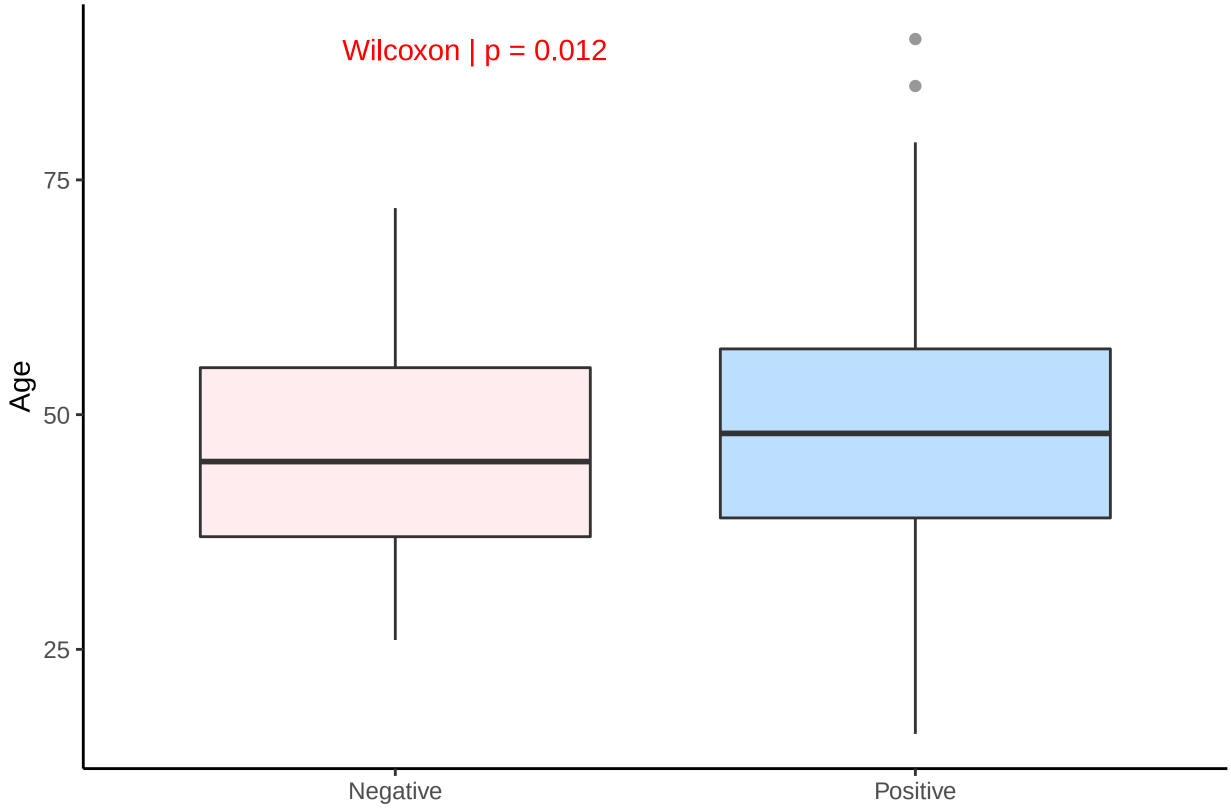


Figure 1: This figure assesses the association between Age and the target variable class.

3.2 Association between class and categorical predictors

```
diabetes %>%
  dplyr::select(-Age) %>%
  tbl_summary(by=class,
              type = all_dichotomous() ~ "categorical") %>%
  add_p() %>%
  bold_labels() %>%
  modify_caption("Associations between target variable and each categorical predictor") %>%
  modify_header(label="**Patient Characteristic**") %>%
  modify_footnote(c(all_stat_cols()) ~ NA)  %>%
  modify_spanning_header(paste0("stat_",1:2) ~ "**Target variable (class) **") %>%
  as_kable_extra(booktabs = TRUE, linesep="") %>%
  kable_styling(latex_options = c("HOLD_position", "repeat_header")) %>%
  kable_classic()
```

Table 4: Associations between target variable and each categorical predictor

	Target variable (class)		
Patient Characteristic	**Negative**, N = 200	**Positive**, N = 320	**p-value**
Gender			<0.001
Female	19 (9.5%)	173 (54%)	
Male	181 (90%)	147 (46%)	
Polyuria			<0.001
No	185 (92%)	77 (24%)	
Yes	15 (7.5%)	243 (76%)	
Polydipsia			<0.001
No	192 (96%)	95 (30%)	
Yes	8 (4.0%)	225 (70%)	
sudden weight loss			<0.001
No	171 (86%)	132 (41%)	
Yes	29 (14%)	188 (59%)	
weakness			<0.001
No	113 (56%)	102 (32%)	
Yes	87 (44%)	218 (68%)	
Polyphagia			<0.001
No	152 (76%)	131 (41%)	
Yes	48 (24%)	189 (59%)	
Genital thrush			0.012
No	167 (84%)	237 (74%)	
Yes	33 (16%)	83 (26%)	
visual blurring			<0.001
No	142 (71%)	145 (45%)	
Yes	58 (29%)	175 (55%)	
Itching			0.8
No	101 (50%)	166 (52%)	
Yes	99 (50%)	154 (48%)	
Irritability			<0.001
No	184 (92%)	210 (66%)	
Yes	16 (8.0%)	110 (34%)	
delayed healing			0.3
No	114 (57%)	167 (52%)	
Yes	86 (43%)	153 (48%)	
partial paresis			<0.001
No	168 (84%)	128 (40%)	
Yes	32 (16%)	192 (60%)	
muscle stiffness			0.005
No	140 (70%)	185 (58%)	
Yes	60 (30%)	135 (42%)	
Alopecia			<0.001
No	99 (50%)	242 (76%)	
Yes	101 (50%)	78 (24%)	
Obesity			0.10
No	173 (86%)	259 (81%)	
Yes	27 (14%)	61 (19%)	

[1] Pearson's Chi-squared test

Table 4 above presents the contingency table for assessing associations between the target variable and each categorical predictor. The cells include the number and proportion of observations in each group for all categorical predictors and the target variable. The cell counts are within reasonable levels, so the χ^2 test of independence method was employed throughout. All the corresponding p-values except those for **Itching** and **delayed healing** suggest enough evidence of association at the significance level $\alpha = 0.25$.

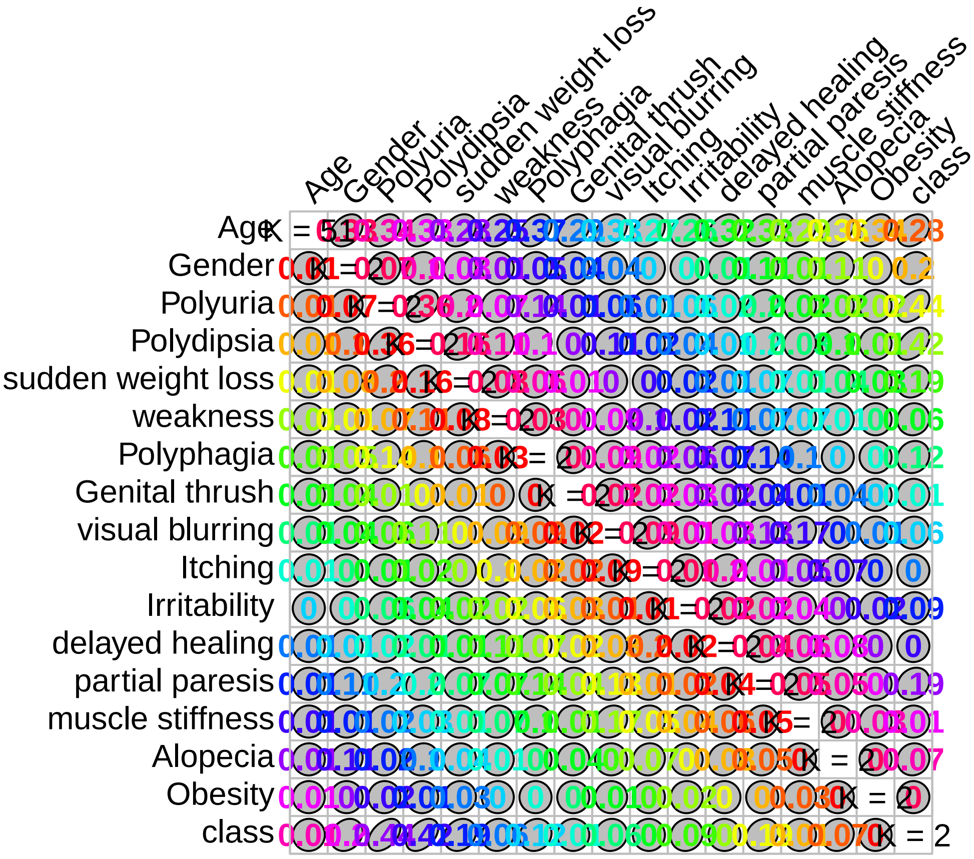
3.3 Reporting unimportant predictors

From the foregoing results, **Itching** and **delayed healing** turn out to be the unimportant predictors when the liberal threshold significance level of $\alpha = 0.25$ is used. These two predictors will be removed from the predictor set in the model building phase to be performed later.

3.4 Correlation plot among the variables

Since almost all the variables are categorical, the Goodman and Kruskal tau measure was used to investigate the association among the variables.

```r
# install.packages("GoodmanKruskal")
library(GoodmanKruskal)
# data1<- diabetes %>% select(class)
dat<- GKtauDataframe(diabetes)
plot(dat, colorPlot=T)
```



Generally, there is no high correlation among the predictors. This suggests that multicollinearity is not an issue here.

4 Data Partition

For the purpose of model building, the target variable class was recoded to 0 and 1 for the negative and positive levels, respectively. Also, the unimportant predictors identified in Step 3 were removed, leaving behind 14 potential predictors. The resulting data is therefore partitioned as follows. A ***123*** seed was used throughout to ensure reproducibility of results affected by random generation.

```
set.seed(123) # set seed for reproducibility
ratio <- 2/3
n <- nrow(diabetes)
train_index <- sample(1:n, size=trunc(n*ratio), replace=FALSE)

# recode the levels of the target  variable and remove unimportant predictors
# class = factor(class, levels = c("Negative", "Positive"), labels = c(0, 1))
diabetes_new <- diabetes %>%
  mutate(class = ifelse(class == "Negative", 0,1)) %>%
  dplyr::select(-Itching, -`delayed healing`)

names(diabetes_new) <- snakecase::to_snake_case(names(diabetes_new)) # join variable names hav
D1 <- diabetes_new[train_index, ] # training set
D2 <- diabetes_new[-train_index, ] # test set
dim(D1); dim(D2)
```

```
## [1] 346  15
```

```
## [1] 174  15
```

Using the ratio 2:1, the diabetes data set was partitioned into 346 training observations and 174 test observations, respectively, both with 15 variables (14 candidate predictors, 1 target variable).

5 Logistic Regression Modeling

We now build a logistic regression model for this medical diagnosis task.

5.1 Part (a): Fitting the regularized logistic regression model

A 5-fold cross validation regularized logistic regression model with LASSO penalty was fitted to the training set D_1.

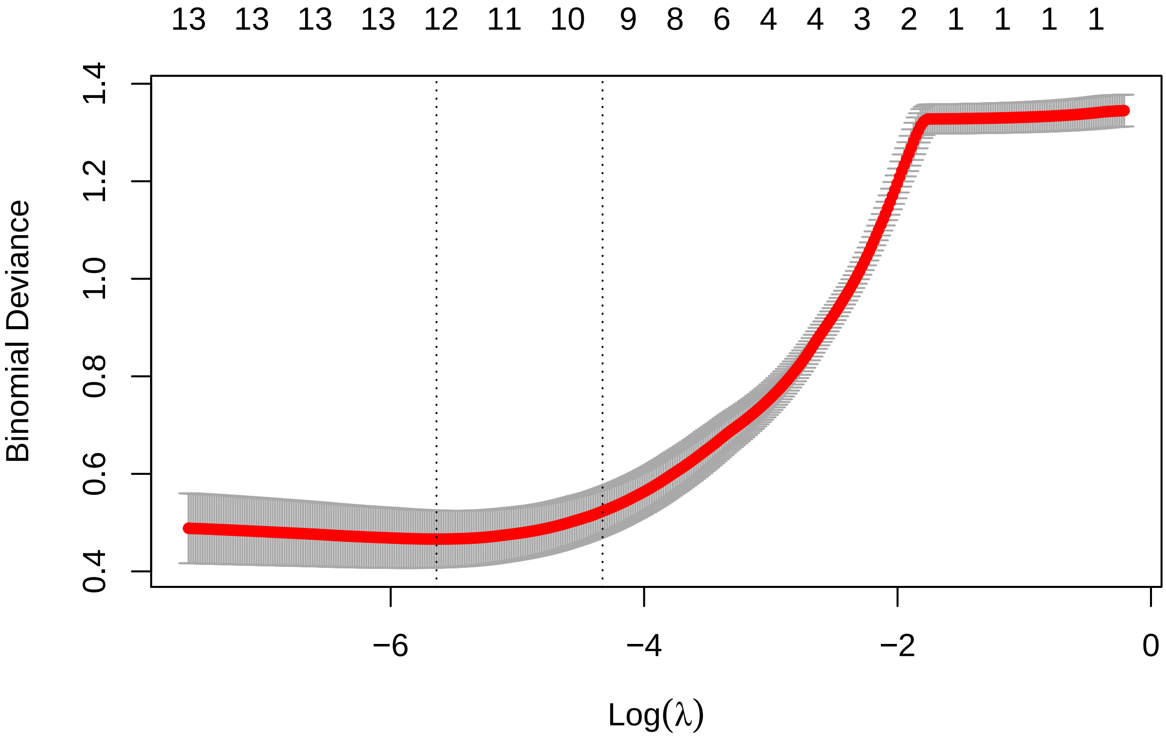
```
library(glmnet)
# select target and create the design matrix
y <- D1$class
X <- model.matrix(~ . -class, data = D1)

# fit the model
set.seed(123)
cv.lasso <- cv.glmnet(X, y, nfolds = 5, family="binomial", alpha=1, lambda.min=.0001,
                      thresh = 1e-07, nlambda=500, standardize=F, maxit=3000, type.measure = "
```

```
# best tuning parameter
best.lambda <- cv.lasso$lambda.min; best.lambda
```

```
## [1] 0.003558
```

```
plot(cv.lasso)
```



- The best tuning parameter λ is obtained as 0.0036 based on the minimum cross-validated deviance.
- The plot suggests two possible models; one with 12 and the other with 10 variables. However, for the purpose of obtaining a simple model, the one with fewer variables is chosen.

5.2 Part (b): Presenting fnal 'best' model fit

A final model containing 10 predictors whose coefficients in absolute terms are greater than 0 is selected as the 'best' model. The selected variables include ***Age, Gender, Polyuria, Polydipsia, Sudden weight loss, Polyphagia, Genital thrush, Irritability, partial paresis, and Alopecia.***.

```
# beta.hat <-coef(cv.lasso, s="lambda.1se")
beta.hat <-as.vector(coef(cv.lasso))
# beta.hat <-as.vector(coef(cvfit.SCAD))
cutoff <- 0
terms <- colnames(X)[abs(beta.hat[-1]) > cutoff]; terms
```

```
##  [1] "age"                 "genderMale"          "polyuriaYes"
##  [4] "polydipsiaYes"       "sudden_weight_lossYes" "polyphagiaYes"
##  [7] "genital_thrushYes"   "irritabilityYes"     "partial_paresisYes"
## [10] "alopeciaYes"
```

```
# Get the actual variables names for model building.
vars_selected <- stringr::str_remove(terms, "Yes|Male")
formula.lasso <- as.formula(paste(c("class ~ 1", vars_selected),collapse = "  + "))
D1 <- D1 %>% mutate(across(-age, as.factor))
best.fit <- glm(formula.lasso, family = "binomial", data = D1)
summary(best.fit)
```

```
##
## Call:
## glm(formula = formula.lasso, family = "binomial", data = D1)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.5999  -0.2446   0.0127   0.0960   3.0422
##
## Coefficients:
##                       Estimate Std. Error z value Pr(>|z|)
## (Intercept)             1.8752     1.1093    1.69   0.0910 .
## age                    -0.0506     0.0255   -1.99   0.0469 *
## genderMale             -3.6357     0.6482   -5.61  2.0e-08 ***
## polyuriaYes             3.1485     0.6357    4.95  7.3e-07 ***
## polydipsiaYes           3.1975     0.7607    4.20  2.6e-05 ***
## sudden_weight_lossYes   0.8432     0.5288    1.59   0.1108
## polyphagiaYes           1.4776     0.5832    2.53   0.0113 *
## genital_thrushYes       1.6548     0.5989    2.76   0.0057 **
## irritabilityYes         2.5360     0.6440    3.94  8.2e-05 ***
## partial_paresisYes      1.4103     0.5799    2.43   0.0150 *
## alopeciaYes            -0.9838     0.5989   -1.64   0.1005
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 461.92  on 345  degrees of freedom
## Residual deviance: 122.09  on 335  degrees of freedom
## AIC: 144.1
##
## Number of Fisher Scoring iterations: 7
```

- At 5% significance level, the predictors ***Age, Gender, Polyuria, Polydipsia, Polyphagia, Genital thrush, Irritability, partial paresis*** remain important predictors since their p-values are less than 0.05.
- Age, Gender, and Alopecia appear to have a negative effect on the target class, while the remaining predictors show positive effect. For instance, the coefficient associated with Gender for males is -*3.6357* which means that the odds of being diagnosed diabetic positive is approximately 2.64% (exp(-3.6357)=0.0264) lower for male patients.

- The residual deviance of 122.09 on 335 degrees of freedom compared to the null deviance of 461.92 on 345 degrees of freedom signifies that the chosen model is better than a null model containing no predictors.

6 Model Assessment

6.1 Applying the fnal best model to the test data D_2

```
# MAKING PREDICTION
# =====================
phat <- predict(best.fit, newdata = D2, type="response") # predicted probabilities
cutoff <- 0.5
yhat <- ifelse(phat <= cutoff, 1, 0)
yobs <- D2$class
table(yobs, yhat) # confusion matrix
```

```
##     yhat
## yobs  0  1
##    0  7 59
##    1 98 10
```

- The confusion table shows that the predictions made on the test set resulted in high missclassifications.

6.2 Presenting ROC and AUC

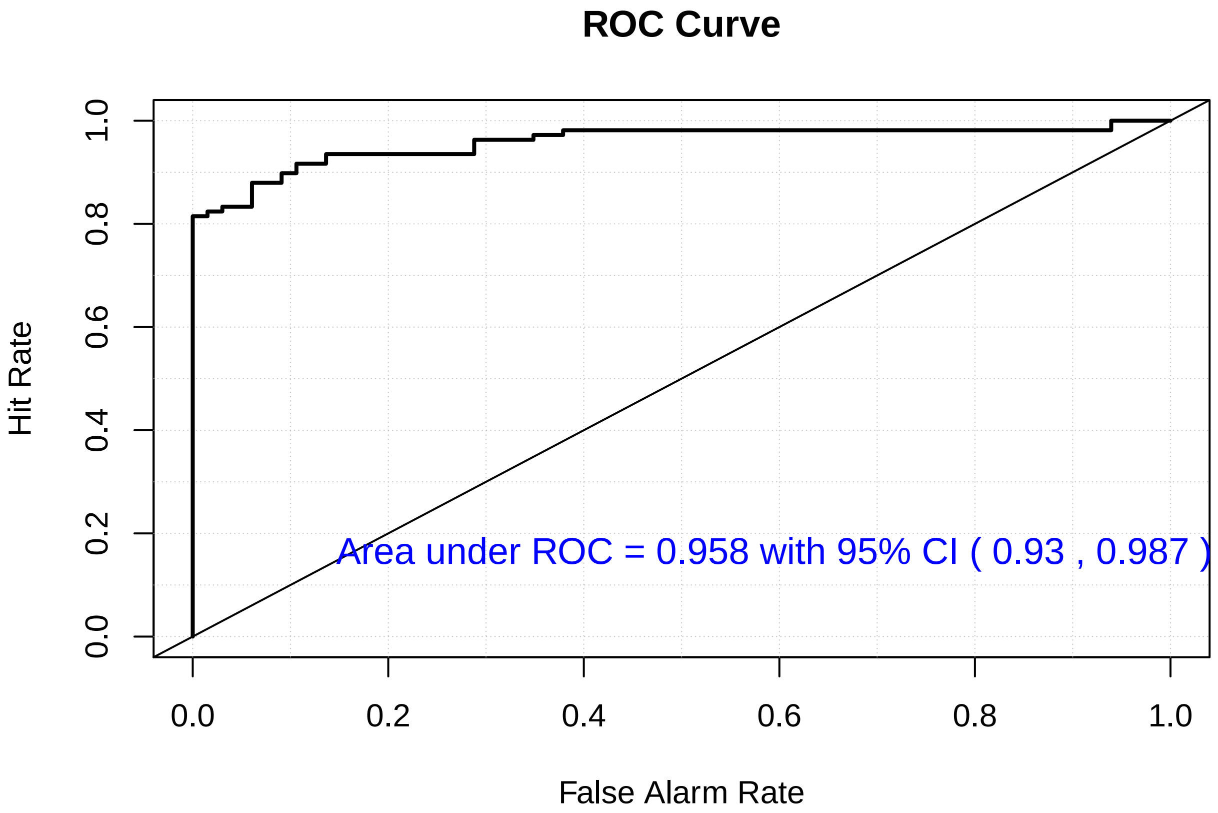
```
library(verification)
a.ROC <- roc.area(obs=yobs, pred=phat)$A
print(a.ROC)
```

```
## [1] 0.9585
```

```
library(cvAUC)
AUC <- ci.cvAUC(predictions=phat, labels=yobs, folds=1:NROW(D2), confidence=0.95)
auc.ci <- round(AUC$ci, digits=3) # confidence interval for cross-validated Area Under the ROC
mod.glm <- verify(obs=yobs, pred=phat)
```

```
## If baseline is not included, baseline values  will be calculated from the  sample obs.
```

```
roc.plot(mod.glm, plot.thres = NULL)
text(x=0.6, y=0.16, paste("Area under ROC =", round(AUC$cvAUC, digits=3),
    "with 95% CI (", auc.ci[1], ",", auc.ci[2], ").",
    sep=" "), col="blue", cex=1.2)
```



The Area under the ROC curve is obtained as **0.958**. With 95% confidence level, the ROC is estimated to lie between **0.93** and **0.987**.