CAPSTONE PROJECT – ADVANCED STATISTICAL MODELLING

Predicting The Onset of Diabetes – An investigation into the optimal statistical model for predicting the onset of gestational diabetes

William Belcher

Gestational diabetes mellitus (GDM) is a subtype of diabetes diagnosed during the second part of pregnancy, gestation, and remains until the baby is born. Currently, all women are screened for GDM during their routine 24-to-28-week check-up ("Gestational diabetes", 2021). This test is quite lengthy, taking a couple of hours. With the rapid increase in computing power and use of machine learning models in a clinical environment, it now seems plausible for a time-efficient screening process to be implemented for GDM. In this paper, the method of Logistic Regression is utilized in modelling the occurrence of GDM in women of Pima Indian heritage. Best subset selection indicated the of the 9 medical predictors present, only pregnancies, glucose, BMI, diabetes pedigree function and age had statistical significance in predicting GDM. The proposed model has a cross-validated accuracy estimate of 80.82% and a test accuracy of 78.48%. It can therefore be argued that the proposed model can assist in the early detection and diagnosis of gestational diabetes, reducing the physiological impact of the condition on the mother.

Introduction

Gestation diabetes mellitus (GDM) is a form of diabetes that occurs during pregnancy and persists until after the birth of the child ("About Gestational Diabetes", 2021). Diagnosis typically happens in around the 24th to 28th week of the pregnancy. GDM can be attributed to the increased insulin resistance caused by the hormone blocking nature of the placenta. During pregnancy, the need for insulin can be as high as 2 to 3 times higher than normal. If a woman already suffered from insulin resistance, the pancreas may be unable to cope with the heightened demand, leading to higher blood glucose levels and a diagnosis of GDM. Currently, women are referred to an oral glucose tolerance test at a pathology lab to determine if they have GDM. This test requires fasting from the previous night and can take between one to two hours. With the swift increase in computation and of machine learning models in the clinical setting, it now seems plausible for a statistical model to assist in the early detection and diagnosis of gestational diabetes mellitus. The early detection of GDM enables a management plan to be developed and enacted earlier, minimising the impact of the condition on the expecting mother.

Data

The Diabetes data set, originally from the National Institute of Diabetes and Digestive and Kidney Diseases, contains several medical predictors and one target variable ("Pima Indians Diabetes Database", 2021). The response variable classified the patients into two classes: diabetic and non-diabetic. The medical predictors were eight different risk factors associated with GDM, being: number of pregnancies the patient has had, the plasma glucose concentration of two hours in an oral glucose tolerance test, diastolic blood pressure, triceps skin fold thickness, two-hour serum insulin, body mass index, diabetes pedigree function and age as in Table 1. These variables had many types, being either numerically discrete or continuous and the response variable being a binary 1 or 0, with a 1 representing the patient being diabetic. The dataset set is a subset of a larger database, with each observation being taken from a female patient over the age of 20 and of Pima Indian heritage. The following table details all 9 variables:

Variable	Description
Pregnancies	Number of pregnancies (discrete)
Glucose	Plasma glucose concentration (mg/Dl)
BloodPressure	Diastolic blood pressure (mm Hg)
SkinThickness	Triceps skin fold thickness (mm)
Insulin	2-Hour serum insulin (mu U/ml)
BMI	Body mass index (kg / m^2)
DiabetesPedigreeFunction	Diabetes pedigree function
Age	Age (years)
Outcome	Response (binary)

Table 1: List of variables and their associated types

In total, there were 768 observations. It is worth noting not much information about the

DiabetesPedigreeFunction field was supplied. However, for this report it is assumed that this function returns some information based on the family's history of gestational diabetes.

Data Pre-processing

The pre-processing and following methods were completed using the software package RStudio, an integrated development environment for the R programming language (RStudio: Integrated Development Environment for R, 2021). Data pre-processing is a technique in which the raw, imported data is transformed into a more meaningful and usable format. A summary of the dataset showed that minimum value for: Glucose, BloodPressure, SkinThickness, Insulin and BMI, was zero. It is unreasonable to have a zero value for these fields so the rows containing zeros in these columns were removed. Missing value imputation was not utilised in order to not introduce unnecessary variance into the models. The size of the cleaned data that was then utilised in the construction of the models contained 9 columns, 8 being predictors and 1 being a response with a total of 392 observations.

Methods

Binary Logistic Regression

Binary Logistic Regression models the relationship between a binary response and its predictors. More specifically, there exists a linear combination of variables that predicts the log-odds of the probability of an event from a logistic model. Let $\boldsymbol{Y}$ be the binary outcome where $Y_i = 1$ if the patient is diabetic and $Y_i = 0$ if the patient does not suffer from GDM, with all observations being independent. Additionally, let $\boldsymbol{X} = [x_1, \dots, x_p]$ be the set of explanatory variables, which can be of any form (Cheng Hua, 2021). In this case, π_i is the probability of the patient being diabetic with the following logistic function

$$logit(\pi_i) = log\left(\frac{\pi_i}{1 - \pi_i}\right) = \beta_0 + \sum_{j=1}^{p} \beta_j \, x_{ij}$$

Thus,

$$\pi_i = \frac{e^{\,\beta_0 + \sum_{j=1}^{p} \beta_j x_{ij}}}{1 - e^{\,\beta_0 + \sum_{j=1}^{p} \beta_j x_{ij}}}$$

From this function, $0 \leq \pi_i \leq 1$.

Maximum Likelihood Estimation

Due to the nature of generalised linear models, and more specifically binary logistic regression which assumes the response comes from a binomial distribution, least squares regression cannot be employed and is instead substituted for maximum likelihood estimation (MLE). MLE attempts to maximize the value of the following equation using iterative methods

$$L(\pi, n; y) = \prod_i \binom{n_i}{n_i y_i} \pi^{n_i y_i} (1 - \pi_i)^{(n_i - n_i y_i)}$$

Likelihood Ratio Test

The likelihood ratio test assesses the goodness of fit between two statistical models, based on the ratio of their likelihoods. The effect of β_j can be assessed by setting $\beta_j = 0$ in the first model and letting $\beta_j = \widehat{\beta_j}$ in the second. The likelihood ratio statistic can then be calculated with the following formula

$$\Lambda^* = -2(\ell_0 - \ell_1)$$

Where ℓ_0 and ℓ_1 are the log-likelihoods of model 0 and model 1, respectively.

Model Selection Criteria

Akaike's Information Criteria (AIC) was selected as the measure of model fit. AIC estimates the distance between the true likelihood function of the data and the fitted likelihood function of the model plus a constant which penalises model complexity. AIC is defined by

$$AIC = -2\,\ell(\hat{\beta}_M) + 2p$$

Where p is the number of parameters in model M.

Cross-fold Validation

Cross-fold validation is employed as a method for estimating the model test error rate. This approach divides the data into k groups of equal size in a random manner. To start, the first group is removed from the training data, leaving the model to be trained on the remaining $k - 1$ groups. The Mean Square Error (MSE) is then calculated using the observations from the left-out group. This process is repeated until all groups have been utilised as the testing set, leaving a vector of MSEs,

$[MSE_1, \dots, MSE_k]$. The k-fold cross validation error is then calculated by finding the mean of these values:

$$CV_k = \frac{1}{k}\sum_{i=1}^{k} MSE_i$$

Performance metrics

The result of a model is measured in terms of its accuracy. The values are obtained by using the generated models to predict the outcome on the test data set and making note of the resulting confusion matrix.

	Predicted Outcome	
Actual Outcome	True Positive	False Positive
	False Negative	True Negative

Table 2: Confusion Matrix

With accuracy being calculated in the following way:

$$Accuracy = \frac{TP + TN}{TN + TP + FN + FP}$$

Where TP, TN, FP and FN are true positive, true negative, false positive and false negative respectively.

Results

Figure 1 is the Pearson's Correlation heat map, which visually illustrates the correlation between pairs of predictors.

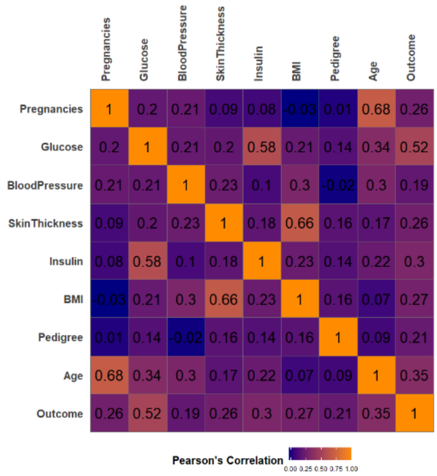

Figure 1: Pearson's Correlation Matrix

The correlation between pregnancy and age (0.68), BMI and skin thickness (0.66), insulin and glucose (0.58) are considered high as they have a correlation over 0.5 and are significantly larger than other pairwise correlations. This suggest these three pairs of predictors are correlated.

Further investigation into possible interaction between predictors was completed with interaction plots. Figure 2 illustrates how the variation in one predictor changes the value of the response when the value for the interacting variable is held constant.

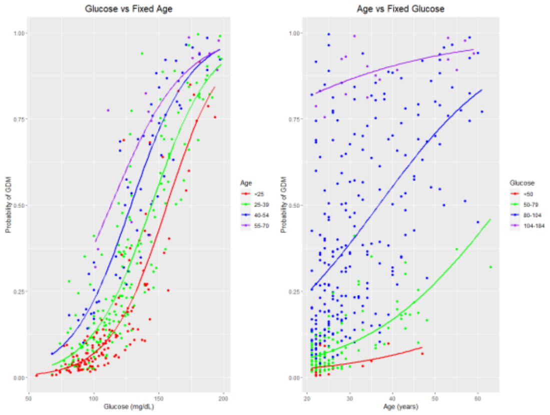

Figure 2: Interaction Plots

As seen in Figure 2, the probability of a patient having gestational diabetes is greater among the older age groups. This increase in probability reduces as the glucose levels rise. The second plot in Figure 2 indicates that probability of diabetes is highest when the glucose level is high, with a lower glucose level resulting in a lower chance of GDM. Utilising the *glmFSA* from the *rFSA* package, all feasible solutions to a specified generalized linear model that could include *m*-th order interactions can be found. In this case, *m* was limited to two. When comparing models that included the interaction terms with those that did not, these interactions were found to be insignificant.

In total, nine models were fit utilising best subset selection. Best subset selection was employed due to the low number of predictors in the data set. Appendix A illustrates the predictors found to be significant for each subset of the saturated model. From the AIC value, it can be seen that a model with five predictors proved to fit best, with these predictors being: Pregnancies, Glucose, BMI, DiabetesPedigreeFunction and Age.

Proposed Model

An 80/20 training-test split was utilised on the pre-processed data. 5-fold cross validation using the training data was run on the model and returned from the best-subset selection. This model had a test accuracy estimate of 80.82% and therefore a misclassification rate of 19.18%.

The remaining test set was then used to test the model trained on all the training data. Table 3 shows the confusion matrix produced from the test set:

		Predicted	
		ASD	Not ASD
Actual	ASD	47	10
	Not ASD	7	15

Figure 3: Predicted vs Actual Classifications

Thus, the following model is proposed:

$$\begin{aligned} logit(\pi_i) = -10.49 &+ 0.0898 \ast Pregnancies \\ &+ 0.0378 \ast Glucose + 0.084 \\ &\ast BMI + 1.60 \\ &\ast DiabetesPedigreeFunction \\ &+ 0.43 \ast Age \end{aligned}$$

Plotting the receiver operating characteristic curve of the model assists in determining the threshold value to be used in classifying the patients. Figure 4 shows the generated ROC curve.

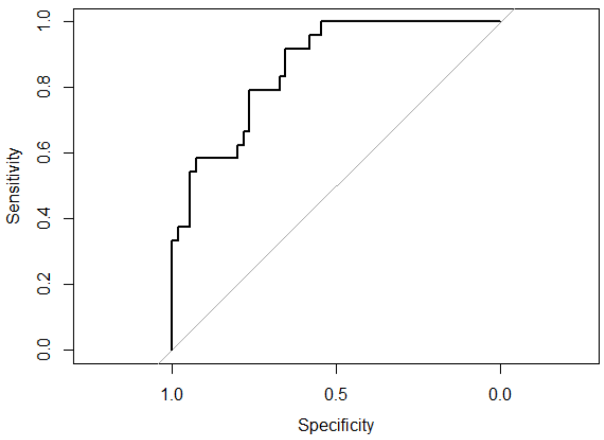

Figure 4: ROC Curve

From the figure, it can be seen that a specificity of 50% provides a sensitivity of 100%, meaning that 100% of the diabetics in the data set were classified as diabetics. This does come with the draw back that 50% of women without diabetes are classified as diabetics, leading to the reduced accuracy. This is imperative as misclassifying a diabetic as non-diabetic will lead to significant health issues for both the to-be mother and the unborn child.

Discussion

From the model, it can be seen that a family history of GDM will significantly increase the log odds of developing gestation diabetes. As each of the predictors are quantitative, they have a linear relationship with the log odds of a patient having diabetes. It is important, however, to acknowledge the limitations of this model. Firstly, the data only contained a small number of predictors, of which, an even smaller subset were selected for inclusion in the model. The data was also restricted to women of Pima Indian heritage, meaning that the model may not be suitable for data sets with more predictors, or for data from patients of differing ethnicities. Future research into this problem may be able to incorporate more lifestyle factors such as whether the patient smokes, alcohol consumption, physical activity frequency and a larger array of genetic traits.

Conclusion

The early detection of gestational diabetes can significantly minimise the impact of the condition on expectant mothers. This paper proposes a binary logistic regression-based model that can assist medical professionals in rapidly diagnosing GDM. In total, five predictors were found to be significant, being Pregnancies, Glucose, BMI, DiabetesPedigreeFunction and Age. The proposed model had a test accuracy of 78.48%, with a 5-fold cross validation accuracy estimate of 80.82%. These results imply that if patients were able to exercise some control over these predictors, it could be possible to reduce the likelihood of them suffering from gestation diabetes. Additionally, the early detection and diagnosis will enable the enactment of a management plan, reducing the impact of the condition on the expectant mother.

References

- About Gestational Diabetes. (2021). Retrieved 15 October 2021, from https://www.diabetesaustralia.com.au/about-diabetes/gestational-diabetes/

- Cheng Hua, Q. (2021). Chapter 10 Binary Logistic Regression | Companion to BER 642: Advanced Regression Methods. Retrieved 15 October 2021, from https://bookdown.org/chua/ber642_advanced_regression/binary-logistic-regression.html

- Gestational diabetes. (2021). Retrieved 14 October 2021, from https://www.pregnancybirthbaby.org.au/gestational-diabetes

- Pima Indians Diabetes Database. (2021). Retrieved 12 October 2021, from https://www.kaggle.com/uciml/pima-indians-diabetes-database

- PBC. (2021). RStudio: Integrated Development Environment for R (Version 1.4.1717) [Windows]. Boston, MA.

Appendix A – Significant Predictors per Subset

# Of parameters / predictors	0	1	2	3	4	5*	6	7	8
Intercept	■	■	■	■	■	■	■	■	■
Pregnancies									■
Glucose		■	■	■	■	■	■	■	■
BloodPressure								■	■
SkinThickness			■	■	■	■	■	■	■
Insulin							■	■	■
BMI						■	■	■	■
DiabetesPedigreeFunction					■	■	■	■	■
Age				■	■	■	■	■	■
loglikelihood	-200	-156	-147	-143	-140	-139.6	-139.0	-138.8	-138.6
AIC	400	314	299	293	289.9	289.2	290	291	293

Appendix B - R Code

```r
library(dplyr)
library(reshape2)
library(ggplot2)
library(faraway)
library(tidyverse)
library(bestglm)
library(caret)
library(rFSA)
library(regclass)
library(cowplot)
library(pROC)
set.seed(1)
rm(list = ls())
importedData <- read.csv(file = "diabetes.csv")
names(importedData)[names(importedData) == "DiabetesPedigreeFunction"] <- "Pedigree"
summary(importedData)

cleanData <- filter(importedData, Glucose > 0, BloodPressure > 0, SkinThickness > 0, Insulin > 0, BMI >

summary(cleanData)

# Pairs
pairs(cleanData)


# Correlation heat map
cc = cor(cleanData, method = "pearson")
cc_df <- as.data.frame(cc)
cc_df$Vars = row.names(cc_df)
ccm = melt(cc_df, id = "Vars")
ccm$Vars <- factor(ccm$Vars, levels = row.names(cc_df))
ggplot(ccm, aes(x = variable, y = Vars)) +
  geom_tile(aes(fill = value), colour = "grey45") +
  coord_equal() +
  geom_text(size = 7, aes(label = round(value,2))) +
  scale_fill_gradient(low = "navy", high = "darkorange") +
  theme(axis.text.y = element_text(size = 15, face = "bold", colour = "grey25"),
        legend.title = element_text(size = 15, face = "bold"),legend.position = "bottom",
        axis.text.x = element_text(size = 15, angle = 90, face = "bold",colour = "grey25", vjust = 0.5,
        panel.background = element_blank(), panel.border = element_rect(fill = NA, colour = NA),
        axis.ticks = element_blank()) +
  labs(x= "", y = "", fill = "Pearson's Correlation") +
  scale_x_discrete(position = "top") +
  scale_y_discrete(limits = rev(levels(ccm$Vars)))
```

```
###############################
# CV Data
###############################
names(cleanData)[names(cleanData) == "Pedigree"] <- "DiabetesPedigreeFunction"
n <- nrow(cleanData)
index <- sample(1:n, n*0.8, replace=FALSE)
trainDat <- cleanData[index, ]
testDat <- cleanData[-index, ]

# Cross fold validation
nFolds <- 5
folds <- createFolds(trainDat$Outcome, k = nFolds)


########################
# Best subset selection
########################
best.logit <- bestglm(trainDat, family = binomial("logit"), IC = "AIC", method = "exhaustive")
summary(best.logit$BestModel)
best.logit$Subsets


x <- glmFSA(Outcome ~ ., data = trainDat, interactions = TRUE, return.models = TRUE)
x$solutions
x

# Logit with 5 predictors (Pregnancies,   Glucose + BMI + PedigreeFunction + Age) has the lowest AIC of
########################
# 5 Interactions
########################
LR.prob <- predict(best.logit$BestModel, newdata = cleanData, type = "response")
plotData <- cbind(cleanData, LR.prob)
plotData$AgeGroup <- cut(plotData$Age, breaks=c(20,25,40,55,70), right = FALSE)
plotData$logAge <- log(plotData$Age)
plotData$GlucoseGroup <- cut(plotData$Glucose, breaks=c(50,80,105,185,200), right = FALSE)
plotData <- plotData[complete.cases(plotData), ]

par(mfrow = c(1,2))
fixedage <- ggplot(data = plotData) +
  aes(x = Glucose, colour = AgeGroup, group = AgeGroup, y = LR.prob) +
  geom_point() +
  stat_smooth(method = "glm", se = FALSE, method.args = list(family=binomial)) +
  ylab("Probability of GDM") +
  xlab("Glucose (mg/dL)") +
  scale_color_manual(name="Age",
                     labels=c("<25","25-39","40-54", "55-70"),
                     values=c("red","green","blue","purple")) +
  ggtitle("Glucose vs Fixed Age") +
  theme(plot.title = element_text(lineheight=2, hjust=0.5, size = 15))

fixedglucose <- ggplot(data = plotData) +
  aes(x = Age, colour = GlucoseGroup, group = GlucoseGroup, y = LR.prob) +
  geom_point() +
```

```r
  stat_smooth(method = "glm", se = FALSE, method.args = list(family=binomial)) +
  ylab("Probability of GDM") +
  xlab("Age (years)") +
  ggtitle("Age vs Fixed Glucose") +
  theme(plot.title = element_text(lineheight=2, hjust=0.5, size = 15)) +
  scale_color_manual(name="Glucose",
                     labels=c("<50","50-79","80-104", "104-184"),
                     values=c("red","green","blue","purple"))

plot_grid(fixedage, fixedglucose, labels="")


########################
# ROC on test data
########################
par(mfrow = c(1,1))
LR.model <- glm(Outcome ~ Glucose + BMI + DiabetesPedigreeFunction + Age, family = binomial("logit"), tr
LR.prob <- predict(LR.model, testDat, type = "response")
g <- roc(Outcome ~ LR.prob, data = testDat)
plot(g)


########################
# 5 Fold Cross Validation
########################
train.acc <- c()
test.acc <- c()
train.acc.temp <- c()
test.acc.temp <- c()
for(i in 1:nFolds) {
  LR.model <- glm(Outcome ~ Glucose + BMI + DiabetesPedigreeFunction + Age, family = binomial("logit"),
  trainPred <- predict(LR.model, trainDat[-folds[[i]], ], type = "response")
  LR.pred <- rep(0, dim(trainDat[-folds[[i]], ])[1])
  LR.pred[trainPred > .5] <- 1
  trainTable <- table(LR.pred, trainDat[-folds[[i]], ]$Outcome)


  testPred <- predict(LR.model, trainDat[folds[[i]], ], type = "response")
  LR.pred <- rep(0, dim(trainDat[folds[[i]], ])[1])
  LR.pred[testPred > .5] <- 1
  testTable <- table(LR.pred, trainDat[folds[[i]], ]$Outcome)

  train.acc.temp <- c(train.acc.temp, (trainTable[1,1]+trainTable[2,2])/sum(trainTable))
  test.acc.temp <- c(test.acc.temp, (testTable[1,1]+testTable[2,2])/sum(testTable))
}
train.acc <- rbind(train.acc, train.acc.temp)
test.acc <- rbind(test.acc, test.acc.temp)
rowMeans(train.acc)
rowMeans(test.acc)

########################
# Test accuracy
########################
```

```
LR.model <- glm(Outcome ~ Glucose + BMI + DiabetesPedigreeFunction + Age, family = binomial("logit"), tr
LR.prob <- predict(LR.model, testDat, type = "response")
LR.pred <- rep(0, dim(testDat)[1])
LR.pred[LR.prob > .5] <- 1
LR.test.table <- table(LR.pred, testDat$Outcome)
LR.test.acc <- (LR.test.table[1,1]+LR.test.table[2,2])/sum(LR.test.table)
LR.test.spec <- LR.test.table[1,1]/(LR.test.table[1,1] + LR.test.table[2,2])
LR.test.sens <- LR.test.table[1,1]/(LR.test.table[1,1] + LR.test.table[2,1])
LR.test.acc
LR.test.spec
LR.test.sens
```