Laboration report in Computational Statistics

Computer lab 1 block 1

732A99

Duc Tran
William Wiik
Sigme

Division of Statistics and Machine Learning
Department of Computer Science
Linköping University

14 November 2023

Contents

1 Assignment 3. Logistic regression and basis function expansion

1.1 Data

The data contains information about the onset of diabetes within 5 years in Pima Indians given medical details. The variables are:

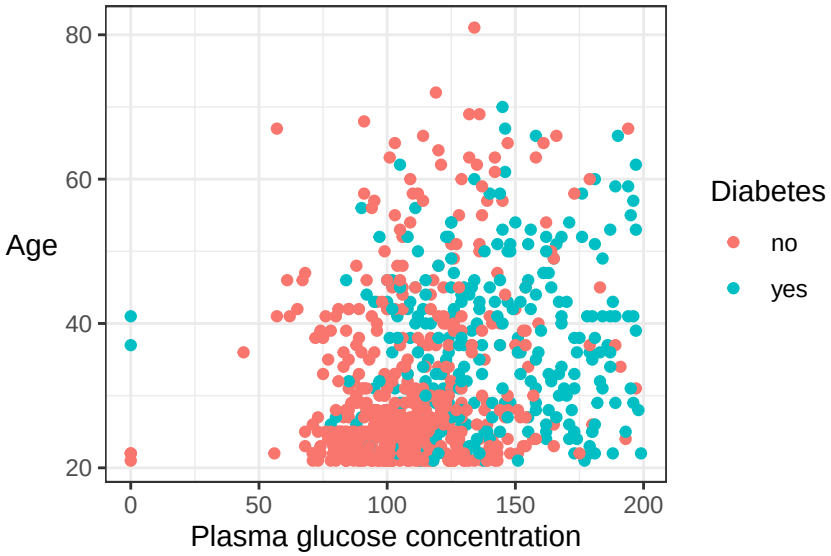
- Number of times pregnant
- Plasma glucose concentration a 2 hours in an oral glucose tolerance test.
- Diastolic blood pressure (mm Hg).
- Triceps skinfold thickness (mm).
- 2-Hour serum insulin (mu U/ml).
- Body mass index (weight in kg/(height in m)^2).
- Diabetes pedigree function.
- Age (years).
- Diabetes (0=no or 1=yes).

```
# Reading data
diabetes_df <- read.csv("pima-indians-diabetes.csv", header=FALSE)

colnames(diabetes_df) <- c("times_pregnant", "plasma_glucose_conc",
                           "diastolic_blood_pressure", "triceps_skinfold_thickness",
                           "serum_insulin","body_mass_index", "diabetes_pedigree",
                           "age", "diabetes")

diabetes_df$diabetes <- ifelse(diabetes_df$diabetes == 0, "no", "yes")
diabetes_df$diabetes <- as.factor(diabetes_df$diabetes)
```

1.2 3.1

Question: Make a scatterplot showing a Plasma glucose concentration on Age where observations are colored by Diabetes levels.

```
library(ggplot2)

ggplot(diabetes_df, aes(x = plasma_glucose_conc, y = age, color = diabetes)) +
  geom_point() +
  theme_bw() +
  theme(axis.title.y = element_text(angle = 0,vjust = 0.5)) +
  labs(colour = "Diabetes",
       x = "Plasma glucose concentration",
       y = "Age")
```



Question: Do you think that Diabetes is easy to classify by a standard logistic regression model that uses these two variables as features? Motivate your answer.

Answer We believe that age and plasma glucose concentration are not suitable variables for classifying diabetes because there's no clear relationship with the disease.

1.3 3.2

Question:

Train a logistic regression model with $y =$ Diabetes as target $x_1 =$ Plasma glucose concentration and $x_2 =$ Age as features and make a prediction for all observations by using $r = 0.5$ as the classification threshold. Report the probabilistic equation of the estimated model and compute also the training misclassification error.

The probabilistic equation:

$$p(y = 1) = \frac{1}{1 + e^{-(\theta_0 + \theta_1 \cdot \text{Plasma glucose} + \theta_2 \cdot \text{Age})}}$$

$$p(y = 1) = \frac{1}{1 + e^{-(-5.91 + 0.04 \cdot \text{Plasma glucose} + 0.02 \cdot \text{Age})}}$$

$$\hat{y} = 1 \text{ if } p(y = 1) > 0.5$$

```r
model <- glm(diabetes ~ plasma_glucose_conc + age, data = diabetes_df,
             family =  "binomial")

pred <- predict(model, newdata = diabetes_df, type = "response")

# Using 0.5 as the classification threshold
pred <- ifelse(pred > 0.5, "yes", "no")

# cunfusion matrix to calculate the misclassification error
```

```r
confusion <- table(diabetes_df$diabetes, pred)
misclass_rate <- (confusion[1,2] + confusion[2,1]) / sum(confusion)
```

Table 1: Misclassification error

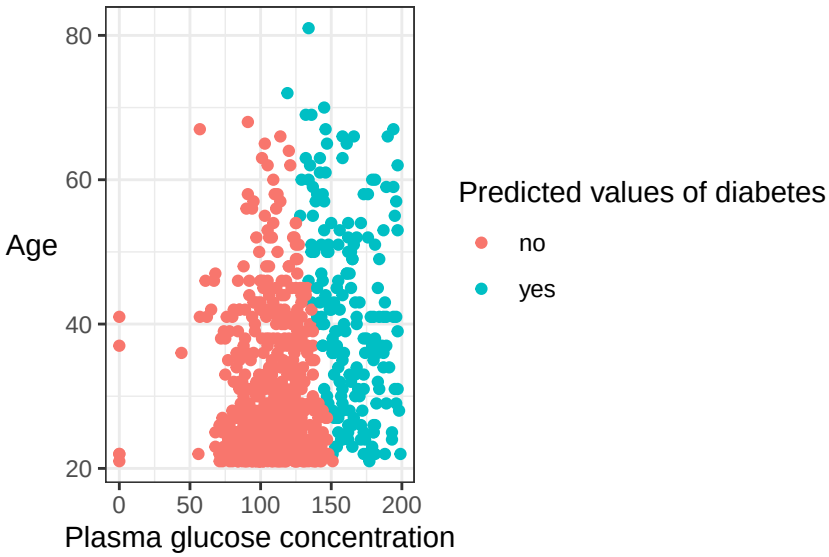
Misclassification error
0.26

The misclassification error is 0.26.

Question:

Make a scatter plot of the same kind as in step 1 but showing the predicted values of Diabetes as a color instead.

```r
diabetes_df_pred <- diabetes_df
diabetes_df_pred$pred <- pred

ggplot(diabetes_df_pred, aes(x = plasma_glucose_conc, y = age, color = pred)) +
  geom_point() +
  theme_bw() +
  theme(axis.title.y = element_text(angle = 0,vjust = 0.5)) +
  labs(colour = "Predicted values of diabetes",
       x = "Plasma glucose concentration",
       y = "Age")
```



Question:

Comment on the quality of the classification by using these results.

Answer

The quality of the classification is not best, because approximately 26% of the observations in our training data are incorrectly classified by the logistic regression model. If we compare this figure X to the figure Y in step 3.1

we can see that the predicted values from figure X do not entirely align with the true values from figure Y.

1.4 3.3

Question:

Use the model estimated in step 2 to a) report the equation of the decision boundary between the two classes b) add a curve showing this boundary to the scatter plot in step 2.

The decision boundary equation:

The decision boundary occurs when the predicted probability is equal to the threshold r.

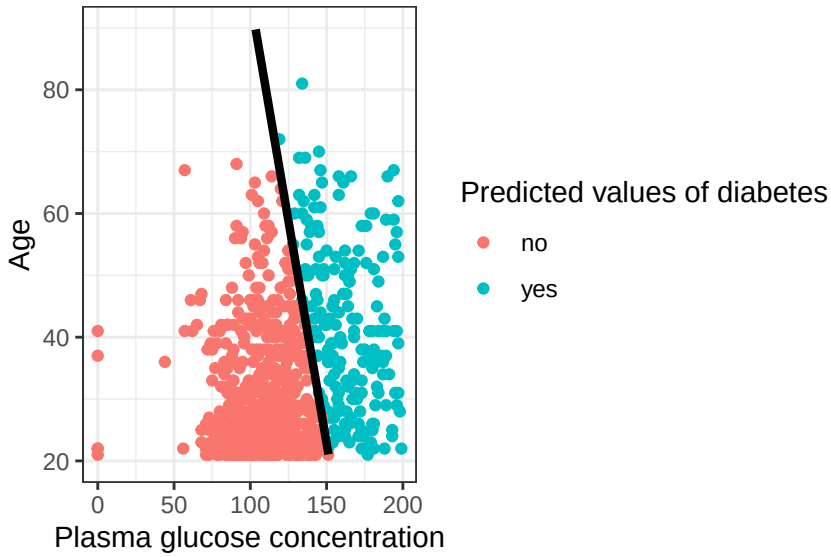
$$\frac{1}{1+e^{-(\theta_0+\theta_1 \cdot \text{Plasma glucose}+\theta_2 \cdot \text{Age})}} = 0.5 \Rightarrow \theta_0 + \theta_1 \cdot \text{Plasma glucose} + \theta_2 \cdot \text{Age} = 0$$

$$-5.91 + 0.04 \cdot \text{Plasma glucose} + 0.02 \cdot \text{Age} = 0$$

```
ggplot(diabetes_df_pred, aes(x = plasma_glucose_conc, y = age, color = pred)) +
  geom_point() +
  theme_bw() +
 stat_function(fun = ({function(x) (-coef(model)[1] - coef(model)[2]*x)/ coef(model)[3] }),
               size=1.5, color = "black") +
  ylim(20,90) +
  labs(colour = "Predicted values of diabetes",
       x = "Plasma glucose concentration",
       y = "Age")
```

```
## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use `linewidth` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

```
## Warning: Removed 76 rows containing missing values (`geom_function()`).
```



Question:

Comment whether the decision boundary seems to catch the data distribution well.

Answer

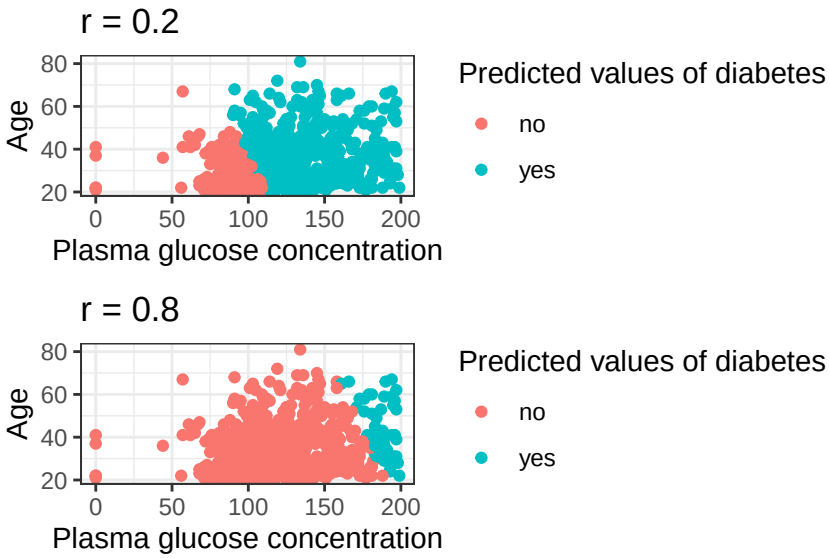
The decision boundary appears to capture the data distribution appropriately. For instance, the cluster with Plasma glucose concentration between 75 to 150 and age 20 to 30 is correctly predicted as class "no" by the decision boundary, aligning with the actual class (figure Y). However, it is notable that the boundary between classes does not show a linear pattern in figure Y, which means that a linear decision boundary is never going to catch the data distribution very well.

1.5 3.4

Question:

Make same kind of plots as in step 2 but use thresholds $r = 0.2$ and $r = 0.8$. By using these plots.

```
library("ggpubr")

# Using 0.2 as the classification threshold
pred <- predict(model, newdata = diabetes_df, type = "response")
pred <- ifelse(pred > 0.2, "yes", "no")
diabetes_df_pred$pred <- pred


p1 <- ggplot(diabetes_df_pred, aes(x = plasma_glucose_conc, y = age, color = pred)) +
      geom_point() +
      theme_bw() +
      labs(colour = "Predicted values of diabetes",
           x = "Plasma glucose concentration",
           y = "Age") +
      ggtitle("r = 0.2")


# Using 0.8 as the classification threshold
pred <- predict(model, newdata = diabetes_df, type = "response")
pred <- ifelse(pred > 0.8, "yes", "no")
diabetes_df_pred$pred <- pred


p2 <- ggplot(diabetes_df_pred, aes(x = plasma_glucose_conc, y = age, color = pred)) +
      geom_point() +
      theme_bw() +
      labs(colour = "Predicted values of diabetes",
           x = "Plasma glucose concentration",
           y = "Age") +
  ggtitle("r = 0.8")


ggarrange(p1, p2,  ncol = 1, nrow = 2)
```



Question:

Comment on what happens with the prediction when r value changes.

Answer

As r increases, the model predict more observations as "no" for diabetes. Opposite, as r decreases, the model predict more observations as "yes" for diabetes.

1.6 3.5

Question:

Perform a basis function expansion trick by computing new features $z_1 = x_1^4$, $z_2 = x_1^3 x^2$, $z_3 = x_1^2 x_2^2$, $z_4 = x_1 x_2^3$, $z_5 = x_2^4$, adding them to the data set and then computing a logistic regression model with y as target and $x_1, x_2, z_1, ..., z_5$ as features. Create a scatterplot of the same kind as in step 2 for this model and compute the training misclassification rate.

```
# new features
diabetes_df$z1  <- diabetes_df$times_pregnant^4
diabetes_df$z2  <- diabetes_df$times_pregnant^3 * diabetes_df$plasma_glucose_conc^2
diabetes_df$z3  <- diabetes_df$times_pregnant^2 * diabetes_df$plasma_glucose_conc^2
diabetes_df$z4  <- diabetes_df$times_pregnant * diabetes_df$plasma_glucose_conc^3
diabetes_df$z5  <- diabetes_df$plasma_glucose_conc^4


model <- glm(diabetes ~ plasma_glucose_conc + age + z1 + z2 + z3 + z4 + z5, data = diabetes_df,
             family =  "binomial")

pred <- predict(model, newdata = diabetes_df, type = "response")

# Using 0.5 as the classification threshold
pred <- ifelse(pred > 0.5, "yes", "no")
```

```r
# cunfusion matrix to calculate the misclassification error
confusion <- table(diabetes_df$diabetes, pred)
misclass_rate <- (confusion[1,2] + confusion[2,1]) / sum(confusion)


diabetes_df_pred <- diabetes_df
diabetes_df_pred$pred <- pred

ggplot(diabetes_df_pred, aes(x = plasma_glucose_conc, y = age, color = pred)) +
  geom_point() +
  theme_bw() +
  theme(axis.title.y = element_text(angle = 0,vjust = 0.5)) +
  labs(colour = "Predicted values of diabetes",
       x = "Plasma glucose concentration",
       y = "Age")
```

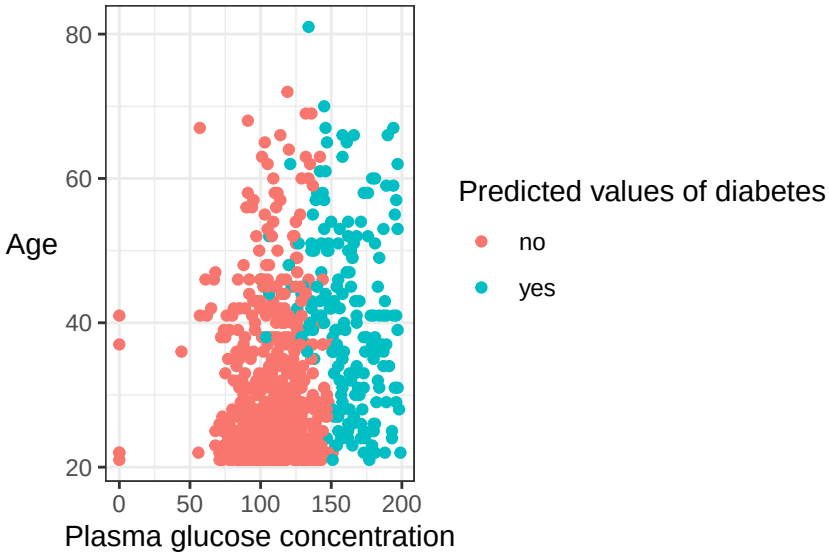


Table 2: Misclassification error

Misclassification error
0.25

The misclassification error is 0.25.

Question:

What can you say about the quality of this model compared to the previous logistic regression model? How have the basis expansion trick affected the shape of the decision boundary and the prediction accuracy?

Answer

The misclassification rate is one percentage point lower than the model in 3.2, indicating a slight improvement.

The decision boundary's shape is not linear, as seen in 3.4, but the data distribution of the predicted values remains very similar.