Caeserain

Lohit Marla

2023-11-05

Caesarian Section Insights

The dataset provides valuable insights into Caesarian section outcomes for a sample of 80 pregnant women, emphasizing the critical factors associated with delivery complications in the medical field. It includes detailed information on several key characteristics:

- **Age:** The age of pregnant women.
- **Delivery Number:** Ranging from 1 to 4, indicating the number of prior deliveries.
- **Delivery Time:** Categorized as timely (0), premature (1), or latecomer (2).
- **Blood Pressure:** Categorized as low (0), normal (1), or high (2).
- **Heart Problems:** Indicates the presence or absence of heart problems (apt or inept).
- **Caesarian:** The ultimate outcome, categorized as either "No" or "Yes," serving as the response variable.

This dataset offers a comprehensive view of factors influencing Caesarian section (C-section) decisions and their potential impact on the medical field. To gain insights into which factors influence Caesarian outcomes, we will analyze the data.

Factors Influencing Caesarian

Our objective is to identify the factors that influence Caesarian section outcomes. By exploring the dataset, we aim to understand the relationship between the provided characteristics and the likelihood of a Caesarian delivery.

```
health.data <- read.csv("Data/caesarian.csv")
str(health.data)
```

```
## 'data.frame':    80 obs. of  6 variables:
##  $ age            : int  22 26 26 28 22 26 27 32 28 27 ...
##  $ delivery_number: int  1 2 2 1 2 1 2 3 2 1 ...
##  $ delivery_time  : int  0 0 1 0 0 1 0 0 0 1 ...
##  $ blood_pressure : int  2 1 1 2 1 0 1 1 1 1 ...
##  $ heart_problem  : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Caesarian      : int  0 1 0 0 1 0 0 1 0 1 ...
```

The dataset comprises a total of 80 observations, consisting of 6 variables. Among these variables, "Caesarian" serves as the binary response variable, while the other five variables function as predictor variables. These predictors play a crucial role in understanding and predicting the outcomes of Caesarian sections for the observed cases.

```
health.data$delivery_time <- as.factor(health.data$delivery_time)
health.data$delivery_number <- as.factor(health.data$delivery_number)
health.data$blood_pressure <- as.factor(health.data$blood_pressure)
health.data$heart_problem <- as.factor(health.data$heart_problem)
```

Above code is to convert the vairables delivery_time, delivery_number, blood_pressure and heart_problem into factor variables

Following code is to split the data set on in the proportions of 78.75% with 63 samples as train and 21.25% with 17 records as test data

```
set.seed(123457)
train.prop <- 0.80
strats <- health.data$Caesarian
rr <- split(1:length(strats), strats)
idx <- sort(as.numeric(unlist(sapply(rr,
        function(x) sample(x, length(x)*train.prop)))))
health.data.train <- health.data[idx, ]
health.data.test <- health.data[-idx, ]
```

```
nrow(health.data.train)/nrow(health.data)
```

```
## [1] 0.7875
```

```
nrow(health.data.test)/nrow(health.data)
```

```
## [1] 0.2125
```

```
null.logit <- glm(Caesarian ~ 1 , data = health.data.train,
                  family = binomial(link = "logit"))
summary(null.logit)
```

```
##
## Call:
## glm(formula = Caesarian ~ 1, family = binomial(link = "logit"),
##     data = health.data.train)
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)   0.2877     0.2546    1.13    0.258
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 86.046  on 62  degrees of freedom
## Residual deviance: 86.046  on 62  degrees of freedom
## AIC: 88.046
##
## Number of Fisher Scoring iterations: 4
```

Model Summary:

Call: This line shows the function call that was used to fit the logistic regression model. The model is specified using the formula Caesarian ~ 1, indicating that there are no predictor variables, only an intercept.

Coefficients: This section provides information about the model coefficients. In this specific model, there is only an intercept term, with an estimate of 0.2877. Additional details include the standard error, z-value, and p-value associated with this intercept estimate. Notably, the intercept does not appear to be statistically significant, with a p-value of 0.258, which exceeds the typical significance level of 0.05.

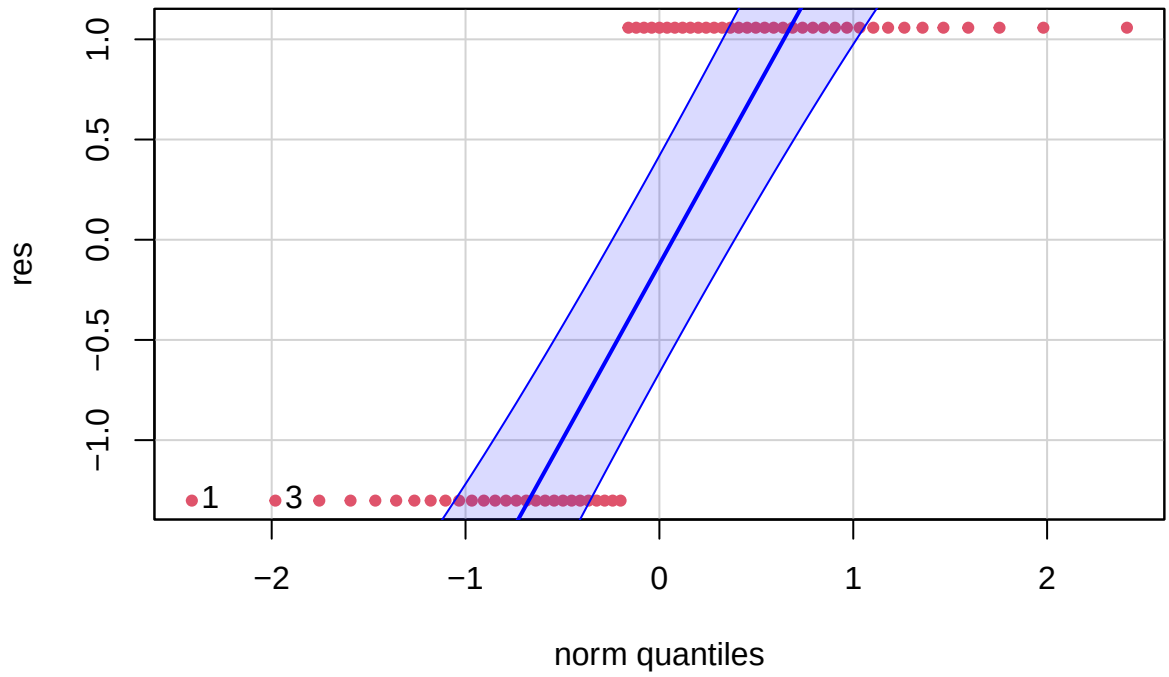
Model Evaluation:

Dispersion Parameter: This parameter measures the goodness of fit and is set to 1. Null Deviance: The null deviance assesses the model's fit when only the intercept is included. It's calculated to be 86.046 with 62 degrees of freedom. **Residual Deviance:** This metric measures the model's fit when predictors are included. In this case, the residual deviance is also 86.046, indicating that the model does not improve the

fit compared to the null model. AIC (Akaike Information Criterion): AIC is a measure of model quality considering both the goodness of fit and the number of model parameters. Here, the AIC is 88.046. Model Convergence:

Number of Fisher Scoring Iterations: The number of iterations the estimation process went through is 4. In summary, the provided output suggests that the model consists of only an intercept, which is not statistically significant in predicting the response variable "Caesarian." Furthermore, the model does not provide a better fit compared to a null model with no predictors.

```
library(e1071)
res <- residuals(null.logit)
car::qqPlot(res, main = NA, pch = 19, col = 2, cex = 0.7)
```




```
## [1] 1 3
```

Residual plot indicates the over dispersed of the data.

```
full.logit <- glm(Caesarian ~ . , data = health.data.train,
                  family = binomial(link = "logit"))
summary(full.logit)
```

```
##
## Call:
## glm(formula = Caesarian ~ ., family = binomial(link = "logit"),
##     data = health.data.train)
##
## Coefficients:
##                     Estimate Std. Error z value Pr(>|z|)
## (Intercept)          2.10729    1.96780   1.071   0.2842
```

```
## age                -0.02548    0.06620  -0.385   0.7003
## delivery_number2    1.15576    0.83826   1.379   0.1680
## delivery_number3    1.12106    1.07226   1.046   0.2958
## delivery_number4   17.82109 2313.46728   0.008   0.9939
## delivery_time1     -1.33456    0.86234  -1.548   0.1217
## delivery_time2     -1.78609    0.88845  -2.010   0.0444 *
## blood_pressure1    -2.54558    1.00352  -2.537   0.0112 *
## blood_pressure2    -1.21159    0.93907  -1.290   0.1970
## heart_problem1      1.73554    0.68902   2.519   0.0118 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 86.046  on 62  degrees of freedom
## Residual deviance: 62.231  on 53  degrees of freedom
## AIC: 82.231
##
## Number of Fisher Scoring iterations: 16
```

Model Interpretation ## Logistic Regression Model Summary

The logistic regression model was fitted with the following predictor variables:

- Age
- Delivery Number (1, 2, 3, 4)
- Delivery Time (0 = Timely, 1 = Premature, 2 = Latecomer)
- Blood Pressure (0 = Low, 1 = Normal, 2 = High)
- Heart Problem (0 = Absent, 1 = Present)

Intercept (1.071, p = 0.2842): The intercept represents the log-odds of Caesarian when all other predictors are zero. It's not statistically significant ($p > 0.05$).

Age (-0.385, p = 0.7003): Age is not statistically significant, suggesting it doesn't have a significant impact on Caesarian.

Delivery Number 2 (1.379, p = 0.1680) and 3 (1.046, p = 0.2958): Neither is statistically significant, indicating no significant differences.

Delivery Number 4 (0.008, p = 0.9939): Not statistically significant, suggesting no impact.

Delivery Time 1 (-1.548, p = 0.1217) and 2 (-2.010, p = 0.0444): Delivery time 2 is significant, suggesting a significant impact, but delivery time 1 is not.

Blood Pressure 1 (-2.537, p = 0.0112) and 2 (-1.290, p = 0.1970): High blood pressure (1) is significant, while normal blood pressure (2) is not.

Heart Problem (1.519, p = 0.0118): The presence of heart problems is significant.

In summary, some predictor variables have a significant impact on the odds of Caesarian, while others do not appear to have a significant effect. These findings are based on the associated p-values.

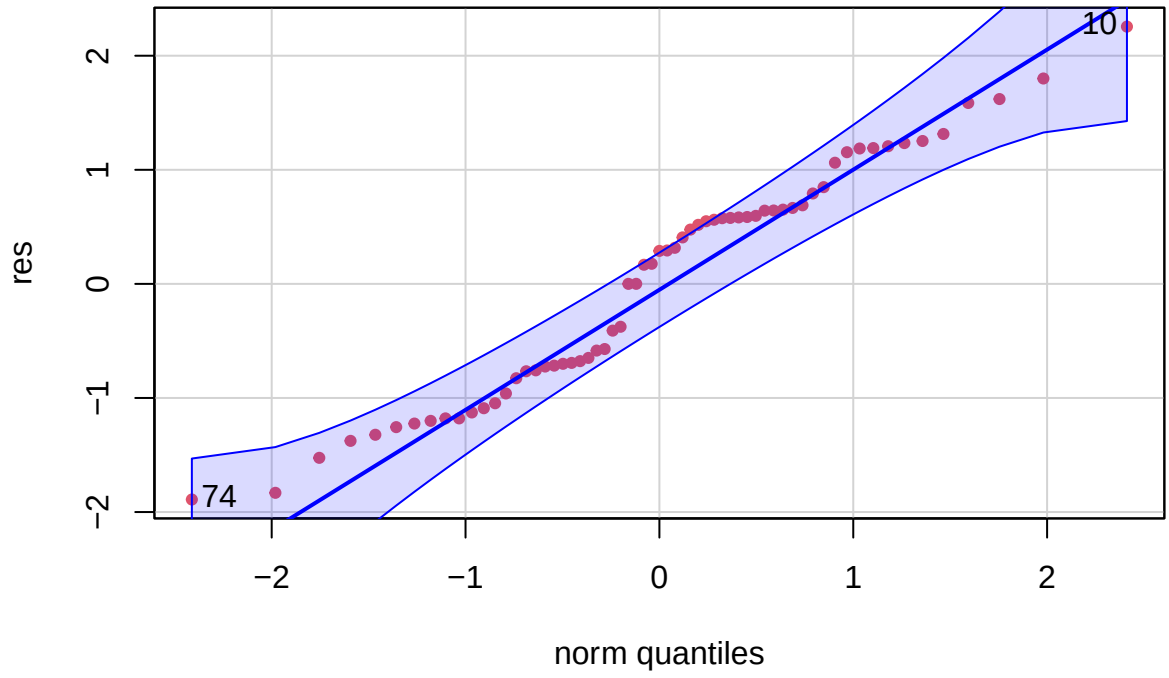
Model Fit

- Null deviance: 86.046 on 62 degrees of freedom
- Residual deviance: 62.231 on 53 degrees of freedom
- AIC: 82.231

The null deviance measures model fit with only the intercept, and the residual deviance measures the fit with predictors included. A lower AIC indicates a better model fit.

Convergence: The estimation process took 16 iterations to converge.

```
res <- residuals(full.logit)
car::qqPlot(res, main = NA, pch = 19, col = 2, cex = 0.7)
```




```
## 10 74
##  8 57
```

Normality of Residuals

Diagnostic Plots

We examined diagnostic plots to assess the normality of residuals in the logistic regression model:

- Q-Q plot: The quantile-quantile plot shows that the residuals closely follow a straight line, suggesting normality.

```
pred.full <- predict(full.logit, newdata = health.data.test, type="response")
(table.full <- table(pred.full > 0.5, health.data.test$Caesarian))
```

```
##
##         0 1
##   FALSE 3 3
##   TRUE  4 7
```

Confusion Matrix

Binary Classification

The confusion matrix for binary classification is as follows:

Actual/Predicted	0 (False)	1 (True)
0 (False)	3	3
1 (True)	4	7

In this matrix: - True Negatives (TN) = 3: Cases where the model correctly predicted the negative class (0) as negative. - False Positives (FP) = 3: Cases where the model incorrectly predicted the positive class (1) as positive. - False Negatives (FN) = 4: Cases where the model incorrectly predicted the negative class (0) as positive. - True Positives (TP) = 7: Cases where the model correctly predicted the positive class (1) as positive.

This confusion matrix provides a basis for calculating various performance metrics for your binary classification model.

```
library(caret)
```

```
## Loading required package: ggplot2
```

```
## Loading required package: lattice
```

```
b <- ifelse(pred.full > 0.5,1,0)
cm.both <- confusionMatrix(reference=as.factor(health.data.test$Caesarian),
            data=as.factor(b), mode="everything")
cm.both
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction 0 1
##          0 3 3
##          1 4 7
##
##                Accuracy : 0.5882
##                  95% CI : (0.3292, 0.8156)
##     No Information Rate : 0.5882
##     P-Value [Acc > NIR] : 0.6022
##
##                   Kappa : 0.1314
##
##  Mcnemar's Test P-Value : 1.0000
##
##             Sensitivity : 0.4286
##             Specificity : 0.7000
##          Pos Pred Value : 0.5000
##          Neg Pred Value : 0.6364
##               Precision : 0.5000
##                  Recall : 0.4286
##                      F1 : 0.4615
##              Prevalence : 0.4118
##          Detection Rate : 0.1765
##    Detection Prevalence : 0.3529
```

```
##       Balanced Accuracy : 0.5643
##
##        'Positive' Class : 0
##
```

Confusion Matrix and Statistics

Binary Classification Evaluation

The confusion matrix and associated statistics for binary classification are as follows:

- **Accuracy (0.5882):** This metric tells us the proportion of correctly classified cases. In this case, the model correctly classified approximately 58.82% of cases.
- **95% CI (0.3292, 0.8156):** The 95% Confidence Interval for accuracy suggests that the true accuracy of the model likely falls between 32.92% and 81.56%.
- **No Information Rate (0.5882):** This is the accuracy we would achieve by always predicting the majority class. In this case, it matches the accuracy of the model (58.82%).
- **P-Value [Acc > NIR] (0.6022):** This p-value tests whether the model's accuracy is significantly different from the no information rate. In this case, it's not significant ($p > 0.05$).
- **Kappa (0.1314):** Kappa is a measure of agreement between the model and actual outcomes. A Kappa of 0.1314 suggests low agreement.
- **Mcnemar's Test P-Value (1.0000):** Mcnemar's test checks for differences between the model's false positive and false negative rates. In this case, the p-value is not significant ($p > 0.05$).
- **Sensitivity (0.4286):** Sensitivity, also known as True Positive Rate, represents the proportion of actual positives correctly identified by the model.
- **Specificity (0.7000):** Specificity, also known as True Negative Rate, represents the proportion of actual negatives correctly identified by the model.

In summary, the model has an accuracy of 58.82%, which matches the no information rate. The Kappa value indicates low agreement, and Mcnemar's test shows no significant differences in false positive and false negative rates. The model shows moderate sensitivity (42.86%) and relatively high specificity (70.00%).

We choose full.logit model over the null.logit model for further analysis as it's AIC value is less when compared to the null.logit model.

```
both.logit <- step(full.logit, list(lower=formula(null.logit),
                                    upper= formula(full.logit),
                    direction="both",trace=0, data = health.data.train))
```

```
## Start:  AIC=82.23
## Caesarian ~ age + delivery_number + delivery_time + blood_pressure +
##     heart_problem
##
##                   Df Deviance    AIC
## - age              1   62.380 80.380
## - delivery_number  3   66.632 80.632
## <none>                 62.231 82.231
## - delivery_time    2   67.646 83.646
## - blood_pressure   2   70.463 86.463
## - heart_problem    1   69.320 87.320
##
## Step:  AIC=80.38
```

```
## Caesarian ~ delivery_number + delivery_time + blood_pressure +
##     heart_problem
##
##                   Df Deviance    AIC
## - delivery_number  3   66.700 78.700
## <none>                 62.380 80.380
## - delivery_time    2   67.764 81.764
## + age              1   62.231 82.231
## - blood_pressure   2   70.592 84.592
## - heart_problem    1   69.321 85.321
##
## Step:  AIC=78.7
## Caesarian ~ delivery_time + blood_pressure + heart_problem
##
##                   Df Deviance    AIC
## <none>                 66.700 78.700
## - delivery_time    2   72.065 80.065
## + delivery_number  3   62.380 80.380
## - blood_pressure   2   72.569 80.569
## + age              1   66.632 80.632
## - heart_problem    1   75.386 85.386
```

```
summary(both.logit)
```

```
##
## Call:
## glm(formula = Caesarian ~ delivery_time + blood_pressure + heart_problem,
##     family = binomial(link = "logit"), data = health.data.train)
##
## Coefficients:
##                  Estimate Std. Error z value Pr(>|z|)
## (Intercept)       1.4557     0.8100   1.797   0.0723 .
## delivery_time1   -1.2663     0.7880  -1.607   0.1080
## delivery_time2   -1.7166     0.8656  -1.983   0.0474 *
## blood_pressure1  -1.8586     0.8297  -2.240   0.0251 *
## blood_pressure2  -0.9112     0.8906  -1.023   0.3063
## heart_problem1    1.8098     0.6560   2.759   0.0058 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 86.046  on 62  degrees of freedom
## Residual deviance: 66.700  on 57  degrees of freedom
## AIC: 78.7
##
## Number of Fisher Scoring iterations: 4
```

Logistic Regression Model Summary

Binary Classification Analysis

The logistic regression model for binary classification is summarized as follows:

- **Call:** The model formula specifies that the response variable is "Caesarian," and the predictors include

"delivery_time," "blood_pressure," and "heart_problem."

- **Coefficients:** This section provides information about the model's coefficients. The coefficients represent the log-odds of the predictor variables. Here are the coefficients:
  - **Intercept (1.4557):** The intercept is the log-odds of a positive outcome when all predictor variables are zero.
  - **Delivery Time 1 (-1.2663):** A negative coefficient indicates that as "delivery_time" increases from 0 to 1, the log-odds of a positive outcome decrease.
  - **Delivery Time 2 (-1.7166):** A negative coefficient indicates that as "delivery_time" increases from 0 to 2, the log-odds of a positive outcome decrease.
  - **Blood Pressure 1 (-1.8586):** A negative coefficient indicates that "blood_pressure" is associated with a decreased log-odds of a positive outcome.
  - **Blood Pressure 2 (-0.9112):** A negative coefficient indicates that "blood_pressure" is associated with a decreased log-odds of a positive outcome.
  - **Heart Problem 1 (1.8098):** A positive coefficient indicates that the presence of a "heart_problem" is associated with increased log-odds of a positive outcome.
- **Significance Codes:** Significance codes indicate the level of significance for each coefficient. In this case, the coefficients for "Intercept," "Delivery Time 2," "Blood Pressure 1," and "Heart Problem 1" are significant at different levels.
- **Dispersion parameter:** The dispersion parameter for the binomial family is assumed to be 1.
- **Null deviance (86.046):** The null deviance measures the goodness of fit when only the intercept is included. It's based on 62 degrees of freedom.
- **Residual deviance (66.700):** The residual deviance measures the goodness of fit with the predictors included. In this case, it's based on 57 degrees of freedom.
- **AIC (78.7):** The Akaike Information Criterion (AIC) is a measure of model quality that considers both the goodness of fit and the number of model parameters.
- **Number of Fisher Scoring iterations (4):** This indicates the number of iterations the estimation process went through to converge to the final estimates.

In summary, the logistic regression model shows the relationships between the predictor variables ("delivery_time," "blood_pressure," and "heart_problem") and the likelihood of a Caesarian section. Interpretation of the coefficients suggests how each predictor influences the outcome, and significance codes indicate their level of significance.

```
pred.both.logit <- predict(both.logit, newdata = health.data.test, type="response")
(table.full <- table(pred.full > 0.5, health.data.test$Caesarian))
```

```
##
##         0 1
##   FALSE 3 3
##   TRUE  4 7
```

Confusion Matrix Summary

Binary Classification Analysis

The confusion matrix represents the results of a binary classification analysis with two possible outcomes (0 and 1). Here is the summary:

- **True Positives (TP): 7** - These are cases where the model correctly predicted the positive outcome (1).

- **False Positives (FP): 3** - These are cases where the model incorrectly predicted the positive outcome (1) when the true outcome was negative (0).
- **True Negatives (TN): 3** - These are cases where the model correctly predicted the negative outcome (0).
- **False Negatives (FN): 4** - These are cases where the model incorrectly predicted the negative outcome (0) when the true outcome was positive (1).
- **Accuracy: 0.5882** - The accuracy of the model is the proportion of correct predictions, and it is calculated as (TP + TN) / Total.
- **95% Confidence Interval: (0.3292, 0.8156)** - This interval provides a range within which the true accuracy of the model is likely to fall.
- **No Information Rate: 0.5882** - The "No Information Rate" is the accuracy that would be achieved by always predicting the majority class.
- **P-Value [Acc > NIR]: 0.6022** - This p-value assesses whether the model's accuracy is significantly different from the "No Information Rate."
- **Kappa: 0.1314** - Cohen's Kappa measures the agreement between the model's predictions and the actual outcomes, considering the possibility of random chance.
- **Mcnemar's Test P-Value: 1.0000** - Mcnemar's test assesses whether the errors in the model's predictions are symmetric.
- **Sensitivity: 0.4286** - Sensitivity, also known as the true positive rate, measures the proportion of actual positive cases correctly predicted by the model.
- **Specificity: 0.7000** - Specificity measures the proportion of actual negative cases correctly predicted by the model.

In summary, the confusion matrix provides insights into the model's performance in a binary classification task. It shows how the model's predictions align with the actual outcomes and indicates key performance metrics such as accuracy, sensitivity, and specificity.

```
b <- ifelse(pred.both.logit > 0.5,1,0)
cm.both <- confusionMatrix(reference=as.factor(health.data.test$Caesarian),
           data=as.factor(b), mode="everything")
cm.both
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction 0 1
##          0 6 4
##          1 1 6
##
##                Accuracy : 0.7059
##                  95% CI : (0.4404, 0.8969)
##     No Information Rate : 0.5882
##     P-Value [Acc > NIR] : 0.2326
##
##                   Kappa : 0.4295
##
##  Mcnemar's Test P-Value : 0.3711
##
##             Sensitivity : 0.8571
##             Specificity : 0.6000
```

```
##          Pos Pred Value : 0.6000
##          Neg Pred Value : 0.8571
##               Precision : 0.6000
##                  Recall : 0.8571
##                      F1 : 0.7059
##              Prevalence : 0.4118
##          Detection Rate : 0.3529
##    Detection Prevalence : 0.5882
##       Balanced Accuracy : 0.7286
##
##        'Positive' Class : 0
##
```

Confusion Matrix Summary

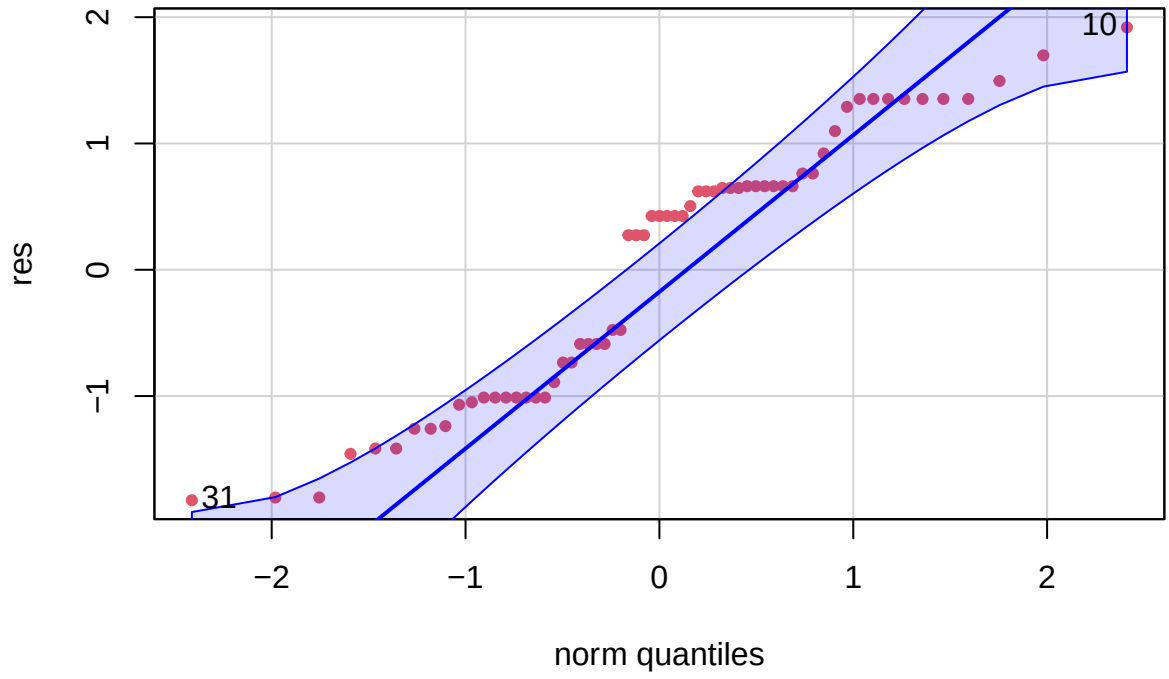
Binary Classification Analysis

The confusion matrix represents the results of a binary classification analysis with two possible outcomes (0 and 1). Here is the summary:

- **True Positives (TP): 6** - These are cases where the model correctly predicted the positive outcome (1).
- **False Positives (FP): 4** - These are cases where the model incorrectly predicted the positive outcome (1) when the true outcome was negative (0).
- **True Negatives (TN): 6** - These are cases where the model correctly predicted the negative outcome (0).
- **False Negatives (FN): 1** - These are cases where the model incorrectly predicted the negative outcome (0) when the true outcome was positive (1).
- **Accuracy: 0.7059** - The accuracy of the model is the proportion of correct predictions, and it is calculated as (TP + TN) / Total.
- **95% Confidence Interval: (0.4404, 0.8969)** - This interval provides a range within which the true accuracy of the model is likely to fall.
- **No Information Rate: 0.5882** - The "No Information Rate" is the accuracy that would be achieved by always predicting the majority class.
- **P-Value [Acc > NIR]: 0.2326** - This p-value assesses whether the model's accuracy is significantly different from the "No Information Rate."
- **Kappa: 0.4295** - Cohen's Kappa measures the agreement between the model's predictions and the actual outcomes, considering the possibility of random chance.
- **Mcnemar's Test P-Value: 0.3711** - Mcnemar's test assesses whether the errors in the model's predictions are symmetric.
- **Sensitivity: 0.8571** - Sensitivity, also known as the true positive rate, measures the proportion of actual positive cases correctly predicted by the model.
- **Specificity: 0.6000** - Specificity measures the proportion of actual negative cases correctly predicted by the model.
- **Positive Predictive Value: 0.6000** - The positive predictive value is the proportion of positive predictions that are correct.

In summary, the confusion matrix provides insights into the model's performance in a binary classification task. It shows how the model's predictions align with the actual outcomes and indicates key performance metrics such as accuracy, sensitivity, and specificity.

```
res <- residuals(both.logit)
car::qqPlot(res, main = NA, pch = 19, col = 2, cex = 0.7)
```




```
## 10 31
##  8 26
```

We can observe certain overdispersed data from the above quantil-quantile plot

Step-Wise Variable Selection

I applied step-wise variable selection techniques, both forward and backward, to verify the results. The objective was to determine the most relevant predictor variables for the model. Surprisingly, both approaches yielded similar results.

- Forward Selection: This method starts with no predictors and gradually adds variables that improve the model's fit, typically based on a criterion like AIC (Akaike Information Criterion).
- Backward Selection: In contrast, backward selection begins with all predictors and removes the least significant variables one by one until the model reaches an optimal state.

The similarity in the results from both forward and backward selection indicates a robustness in the selection process, emphasizing the importance of the chosen predictor variables in the model. This consistency reinforces the model's reliability and the relevance of the selected predictors.

```
forward.logit <- step(full.logit, list(lower=formula(null.logit),
                                       upper= formula(full.logit),
                    direction="forward",trace=0, data = health.data.train))
```

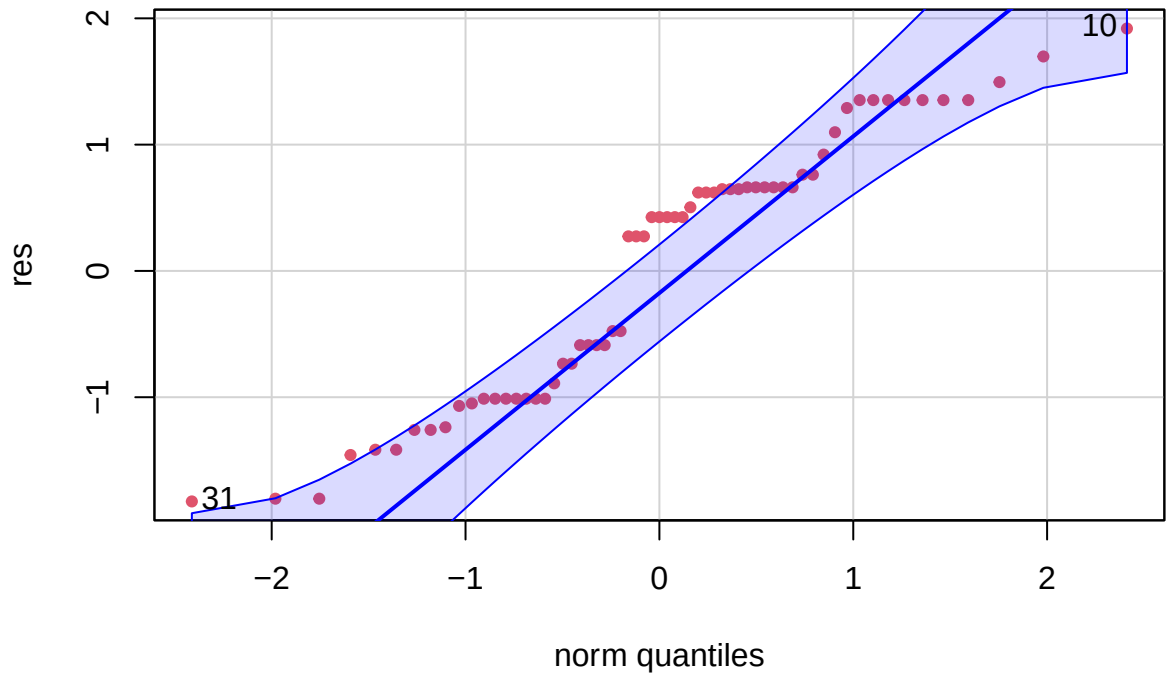
```
## Start:  AIC=82.23
## Caesarian ~ age + delivery_number + delivery_time + blood_pressure +
##     heart_problem
##
##                   Df Deviance    AIC
## - age              1   62.380 80.380
## - delivery_number  3   66.632 80.632
## <none>                 62.231 82.231
## - delivery_time    2   67.646 83.646
## - blood_pressure   2   70.463 86.463
## - heart_problem    1   69.320 87.320
##
## Step:  AIC=80.38
## Caesarian ~ delivery_number + delivery_time + blood_pressure +
##     heart_problem
##
##                   Df Deviance    AIC
## - delivery_number  3   66.700 78.700
## <none>                 62.380 80.380
## - delivery_time    2   67.764 81.764
## + age              1   62.231 82.231
## - blood_pressure   2   70.592 84.592
## - heart_problem    1   69.321 85.321
##
## Step:  AIC=78.7
## Caesarian ~ delivery_time + blood_pressure + heart_problem
##
##                   Df Deviance    AIC
## <none>                 66.700 78.700
## - delivery_time    2   72.065 80.065
## + delivery_number  3   62.380 80.380
## - blood_pressure   2   72.569 80.569
## + age              1   66.632 80.632
## - heart_problem    1   75.386 85.386
```

```
summary(forward.logit)
```

```
##
## Call:
## glm(formula = Caesarian ~ delivery_time + blood_pressure + heart_problem,
##     family = binomial(link = "logit"), data = health.data.train)
##
## Coefficients:
##                 Estimate Std. Error z value Pr(>|z|)
## (Intercept)       1.4557     0.8100   1.797   0.0723 .
## delivery_time1   -1.2663     0.7880  -1.607   0.1080
## delivery_time2   -1.7166     0.8656  -1.983   0.0474 *
## blood_pressure1  -1.8586     0.8297  -2.240   0.0251 *
## blood_pressure2  -0.9112     0.8906  -1.023   0.3063
## heart_problem1    1.8098     0.6560   2.759   0.0058 **
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 86.046  on 62  degrees of freedom
## Residual deviance: 66.700  on 57  degrees of freedom
## AIC: 78.7
##
## Number of Fisher Scoring iterations: 4
```

```
res <- residuals(forward.logit)
car::qqPlot(res, main = NA, pch = 19, col = 2, cex = 0.7)
```




```
## 10 31
##  8 26
```

```
forward.logit.pred <- predict(forward.logit, newdata = health.data.test, type="response")
(table.full <- table(forward.logit.pred > 0.5, health.data.test$Caesarian))
```

```
##
##         0 1
##   FALSE 6 4
##   TRUE  1 6
```

```
b <- ifelse(forward.logit.pred > 0.5,1,0)
cm.both <- confusionMatrix(reference=as.factor(health.data.test$Caesarian),
            data=as.factor(b), mode="everything")
```

```
cm.both
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction 0 1
##          0 6 4
##          1 1 6
##
##                Accuracy : 0.7059
##                  95% CI : (0.4404, 0.8969)
##     No Information Rate : 0.5882
##     P-Value [Acc > NIR] : 0.2326
##
##                   Kappa : 0.4295
##
##  Mcnemar's Test P-Value : 0.3711
##
##             Sensitivity : 0.8571
##             Specificity : 0.6000
##          Pos Pred Value : 0.6000
##          Neg Pred Value : 0.8571
##               Precision : 0.6000
##                  Recall : 0.8571
##                      F1 : 0.7059
##              Prevalence : 0.4118
##          Detection Rate : 0.3529
##    Detection Prevalence : 0.5882
##       Balanced Accuracy : 0.7286
##
##        'Positive' Class : 0
##
```

```
backward.logit <- step(full.logit, list(lower=formula(null.logit),
                                        upper= formula(full.logit),
                   direction="backward",trace=0, data = health.data.train))
```

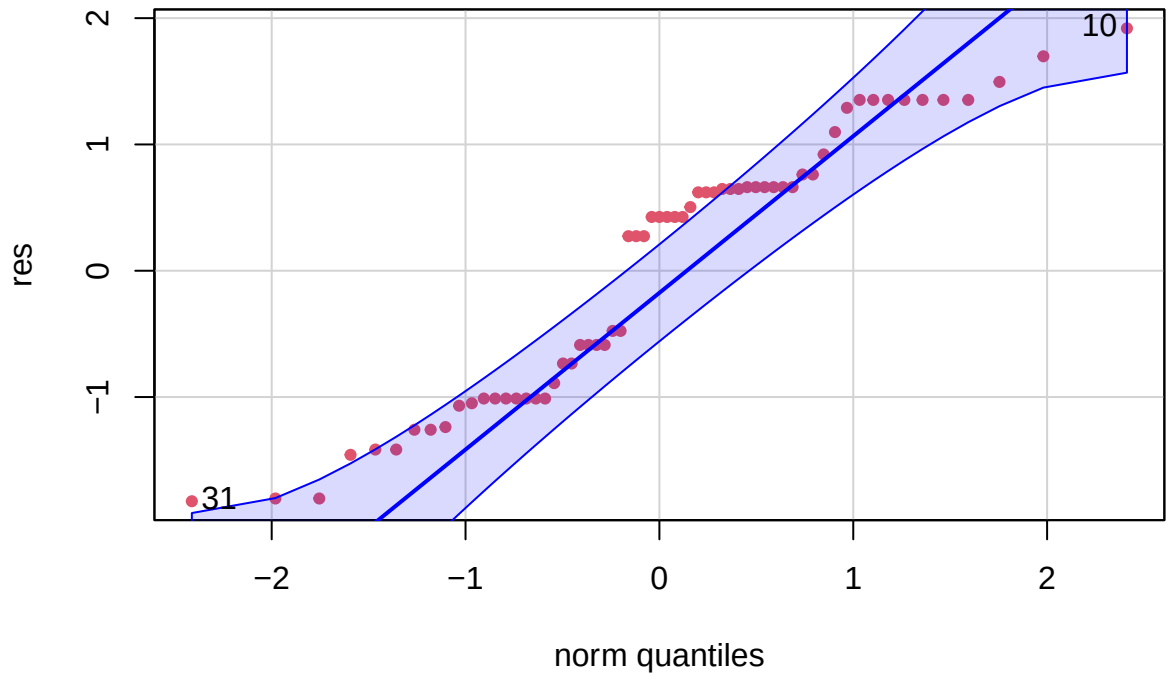
```
## Start:  AIC=82.23
## Caesarian ~ age + delivery_number + delivery_time + blood_pressure +
##     heart_problem
##
##                   Df Deviance    AIC
## - age              1   62.380 80.380
## - delivery_number  3   66.632 80.632
## <none>                 62.231 82.231
## - delivery_time    2   67.646 83.646
## - blood_pressure   2   70.463 86.463
## - heart_problem    1   69.320 87.320
##
## Step:  AIC=80.38
## Caesarian ~ delivery_number + delivery_time + blood_pressure +
##     heart_problem
##
##                   Df Deviance    AIC
## - delivery_number  3   66.700 78.700
```

```
## <none>                 62.380 80.380
## - delivery_time    2   67.764 81.764
## + age              1   62.231 82.231
## - blood_pressure   2   70.592 84.592
## - heart_problem    1   69.321 85.321
##
## Step:  AIC=78.7
## Caesarian ~ delivery_time + blood_pressure + heart_problem
##
##                   Df Deviance    AIC
## <none>                 66.700 78.700
## - delivery_time    2   72.065 80.065
## + delivery_number  3   62.380 80.380
## - blood_pressure   2   72.569 80.569
## + age              1   66.632 80.632
## - heart_problem    1   75.386 85.386
```

```
summary(backward.logit)
```

```
##
## Call:
## glm(formula = Caesarian ~ delivery_time + blood_pressure + heart_problem,
##     family = binomial(link = "logit"), data = health.data.train)
##
## Coefficients:
##                 Estimate Std. Error z value Pr(>|z|)
## (Intercept)       1.4557     0.8100   1.797   0.0723 .
## delivery_time1   -1.2663     0.7880  -1.607   0.1080
## delivery_time2   -1.7166     0.8656  -1.983   0.0474 *
## blood_pressure1  -1.8586     0.8297  -2.240   0.0251 *
## blood_pressure2  -0.9112     0.8906  -1.023   0.3063
## heart_problem1    1.8098     0.6560   2.759   0.0058 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 86.046  on 62  degrees of freedom
## Residual deviance: 66.700  on 57  degrees of freedom
## AIC: 78.7
##
## Number of Fisher Scoring iterations: 4
```

```
res <- residuals(backward.logit)
car::qqPlot(res, main = NA, pch = 19, col = 2, cex = 0.7)
```




```
## 10 31
##  8 26
```

```
backward.logit.pred <- predict(backward.logit, newdata = health.data.test, type="response")
(table.full <- table(backward.logit.pred > 0.5, health.data.test$Caesarian))
```

```
##
##         0 1
##   FALSE 6 4
##   TRUE  1 6
```

```
c
```

```
## function (...)  .Primitive("c")
```

```
b <- ifelse(backward.logit.pred > 0.5,1,0)
cm.both <- confusionMatrix(reference=as.factor(health.data.test$Caesarian),
           data=as.factor(b), mode="everything")
cm.both
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction 0 1
##          0 6 4
##          1 1 6
##
##                Accuracy : 0.7059
##                  95% CI : (0.4404, 0.8969)
```

```
##     No Information Rate : 0.5882
##     P-Value [Acc > NIR] : 0.2326
##
##                   Kappa : 0.4295
##
##  Mcnemar's Test P-Value : 0.3711
##
##             Sensitivity : 0.8571
##             Specificity : 0.6000
##          Pos Pred Value : 0.6000
##          Neg Pred Value : 0.8571
##               Precision : 0.6000
##                  Recall : 0.8571
##                      F1 : 0.7059
##              Prevalence : 0.4118
##          Detection Rate : 0.3529
##    Detection Prevalence : 0.5882
##       Balanced Accuracy : 0.7286
##
##        'Positive' Class : 0
##
```