STAT-HW-8

AUTHOR
Damini Vadrevu

1

```
c <- read.csv("C:/Users/vadre/Downloads/caesarian.csv")


# Check the relative frequency of the response variable 'Caesarian'
table(c$Caesarian)
```

```
 0  1
34 46
```

```
# Convert the response variable 'Caesarian' from numeric to a factor variable
c$Caesarian <- as.factor(c$Caesarian)

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

```
# Set seed for reproducibility
set.seed(123457)

# Define the proportion for the training set
train.prop <- 0.80

# Create stratification variable
strats <- c$Caesarian

# Create indices for stratified sampling
rr <- split(1:length(strats), strats)
idx <- sort(as.numeric(unlist(sapply(rr, function(x) sample(x, length(x)*train.prop)))))

# Split the data into training and test sets
c_train <- c[idx, ]
c_test <- c[-idx, ]

summary(c_train$Caesarian)/nrow(c_train)
```

```
       0        1
0.4285714 0.5714286
```

```r
summary(c_test$Caesarian)/nrow(c_test)
```

```
        0         1 
0.4117647 0.5882353 
```

```r
summary(c$Caesarian)/nrow(c)
```

```
    0     1 
0.425 0.575 
```

```r
logit_model_age <- glm(Caesarian ~ age, data = c_train, family = binomial(link="logit"))
summary(logit_model_age)
```

```

Call:
glm(formula = Caesarian ~ age, family = binomial(link = "logit"), 
    data = c_train)

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept) -0.67091    1.40960  -0.476    0.634
age          0.03445    0.04994   0.690    0.490

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 86.046  on 62  degrees of freedom
Residual deviance: 85.566  on 61  degrees of freedom
AIC: 89.566

Number of Fisher Scoring iterations: 4
```

```r
# Load the necessary library
library(stats)

# Create the null model with only 'Age'
null.logit <- glm(Caesarian ~ age, data = c_train, family = binomial(link="logit"))

# Create the full model with all predictors
full.logit <- glm(Caesarian ~ ., data = c_train, family = binomial(link="logit"))

# Perform stepwise selection using both backward and forward steps
both.logit <- step(null.logit,
                   list(lower = formula(null.logit),
                        upper = formula(full.logit)),
                   direction = "both", trace = 0, data = c_train)

# Output the formula of the selected model
formula(both.logit)
```

```
Caesarian ~ age + heart_problem
```

```
# Calculate the reduction in deviance
deviance_reduction <- null.logit$deviance - both.logit$deviance
deviance_reduction
```

```
[1] 10.15274
```

Residual deviance for the model with only 'Age': 85.566 Residual deviance for the model with both 'Age' and 'Heart Problem': 75.413 The reduction in deviance is the difference between these two values: Reduction in deviance = 85.566 − 75.413 =10.153.

This reduction in deviance by approximately 10.153 indicates that including 'heart_problem' as a predictor along with 'age' provides a better fit to the data than 'age' alone. Given the significance of the 'heart_problem' coefficient and the reduction in deviance, we can conclude that 'heart_problem' is an important predictor for the incidence of a Caesarian section when considered along with 'age'

Forward Variable Selection to choose the best additional predictor alongside 'age'

```
model_additional <- step(
  logit_model_age,
  scope = list(
    lower = ~ age,
    upper =  ~ age + delivery_number + delivery_time + blood_pressure + heart_problem
  ),
  direction = "forward"
)
```

```
Start:  AIC=89.57
Caesarian ~ age

                 Df Deviance    AIC
+ heart_problem    1   75.413 81.413
+ blood_pressure   2   80.860 88.860
<none>                 85.566 89.566
+ delivery_time    2   82.631 90.631
+ delivery_number  3   82.718 92.718

Step:  AIC=81.41
Caesarian ~ age + heart_problem

                 Df Deviance    AIC
<none>                 75.413 81.413
+ blood_pressure   2   71.990 81.990
+ delivery_time    2   72.563 82.563
+ delivery_number  3   72.730 84.730
```

```
logit_model_ageandheart <- glm(Caesarian ~ age + heart_problem, data = c_train, family = binomial
```

```
summary(logit_model_ageandheart)
```

```
Call:
glm(formula = Caesarian ~ age + heart_problem, family = binomial(link = "logit"),
    data = c_train)

Coefficients:
               Estimate Std. Error z value Pr(>|z|)
(Intercept)    -0.482326   1.550942  -0.311  0.75581
age             0.003674   0.056000   0.066  0.94769
heart_problem1  1.810558   0.610215   2.967  0.00301 **
---
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: 75.413  on 60  degrees of freedom
AIC: 81.413

Number of Fisher Scoring iterations: 4
```

The AIC of the model with age and heart_problem is 81.413, much less than the AIC of the model with age indicating this be a better fit.

For heart_problem, the p-value is 0.00301, which is below 0.01, hence it is marked with '**' indicating it is statistically significant. This means that having a heart problem is associated with a higher likelihood of a Caesarian section, holding age constant.

Fitting Model with all predictor variables

```
logit_model_all <- glm(Caesarian ~ ., data = c_train, family = binomial)
summary(logit_model_all)
```

```
Call:
glm(formula = Caesarian ~ ., family = binomial, data = c_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
```

COEFFICIENTS:

(Intercept): The estimated coefficient for the intercept is 2.10729 with a standard error of 1.96780. The z-value is 1.071, with a p-value of 0.2842, which is not statistically significant at the conventional 0.05 level. This means that when all other variables are held at 0, the log-odds of the outcome variable is 2.10729, but this estimate is not statistically significant.

- age: The coefficient for age is -0.02548, which suggests a slight decrease in the log-odds of the outcome with a one-year increase in age. However, with a p-value of 0.7003, this effect is not statistically significant.
- delivery_number2 has a coefficient of 1.15576, but it's not statistically significant (p = 0.1680).
- delivery_number3 has a coefficient of 1.12106, which is also not significant (p = 0.2958).
- delivery_number4 has an extremely large coefficient of 17.82109 with a very high standard error, which results in a z-value of 0.008 and a nonsignificant p-value of 0.9939. This could indicate a data issue or an outlier due to a very small number of cases in this category.
- delivery_time1 has a negative coefficient (-1.33456) which is not significant (p = 0.1217).
- delivery_time2 has a significant negative coefficient (-1.78609) with a p-value of 0.0444, which is statistically significant at the 0.05 level. This indicates that compared to the reference category, delivery_time2 is associated with a lower log-odds of the outcome.
- blood_pressure1 has a statistically significant negative coefficient (-2.54558) with a p-value of 0.0112. This suggests a lower log-odds of the outcome when blood pressure is in category 1.
- blood_pressure2 has a negative coefficient (-1.21159) but it's not statistically significant (p = 0.1970).
- heart_problem1: With a coefficient of 1.73554 and a p-value of 0.0118, having a heart problem (presumably, this is a binary variable with 0 as the reference category) is significantly associated with an increased log-odds of the outcome.

The Null deviance has decreased from 86.046 with 62 degrees of freedom (df) to a Residual deviance of 62.231 with 53 df, indicating that the model with predictors provides a better fit to the data than the null model.

```
library(caret)
```

```
Loading required package: ggplot2

Loading required package: lattice
```

```
pred.full <- predict(logit_model_all, newdata = c_train, type="response")

f <- ifelse(pred.full > 0.65,1,0)
cm.full <- confusionMatrix(reference=as.factor(c_train$Caesarian),
            data=as.factor(f), mode="everything")
cm.full
```

```
Confusion Matrix and Statistics

          Reference
Prediction  0  1
         0 24 12
         1  3 24

               Accuracy : 0.7619
                 95% CI : (0.6379, 0.8602)
    No Information Rate : 0.5714
    P-Value [Acc > NIR] : 0.001302

                  Kappa : 0.5333

 Mcnemar's Test P-Value : 0.038867

            Sensitivity : 0.8889
            Specificity : 0.6667
         Pos Pred Value : 0.6667
         Neg Pred Value : 0.8889
              Precision : 0.6667
                 Recall : 0.8889
                     F1 : 0.7619
             Prevalence : 0.4286
         Detection Rate : 0.3810
   Detection Prevalence : 0.5714
      Balanced Accuracy : 0.7778

       'Positive' Class : 0
```

Confusion Matrix:

- True negatives (TN): 24 (actual 0, predicted 0)
- False positives (FP): 12 (actual 0, predicted 1)

- False negatives (FN): 3 (actual 1, predicted 0)
- True positives (TP): 24 (actual 1, predicted 1)

Statistics:

- Accuracy: 0.7619 or 76.19% of the predictions made by the model are correct.
- P-Value [Acc > NIR]: 0.001302. This p-value tests the null hypothesis that the model is no better than random guessing. A value less than 0.05 rejects this hypothesis, indicating that the model has predictive power.
- Kappa: 0.5333: A kappa of 0.5333 indicates a moderate agreement. No Information Rate (NIR): This is the accuracy that could be achieved by always predicting the most frequent class. Here, the NIR is 57.14%, which means that if the model always predicted the majority class, it would be correct about 57.14% of the time.
- P-Value [Acc > NIR]: This tests the null hypothesis that the accuracy is no better than the NIR. A p-value of 0.001302 is low and suggests that the model's accuracy is significantly better than the NIR
- Sensitivity This means that the model correctly identifies 88.89% of the actual positives.
- Specificity (true negative rate): 0.6667 or 66.67%. The model correctly identifies 66.67% of the actual negatives.

```
library(pROC)
```

```
Type 'citation("pROC")' for a citation.


Attaching package: 'pROC'

The following objects are masked from 'package:stats':

    cov, smooth, var
```

```
roc.full <- roc(c_train$Caesarian, pred.full, levels=c(1,0))
```

```
Setting direction: controls > cases
```

```
auc(c_train$Caesarian, pred.full)
```

```
Setting levels: control = 0, case = 1

Setting direction: controls < cases

Area under the curve: 0.8272
```

Test Data

```
library(caret)
pred.test <- predict(logit_model_all, newdata = c_test, type="response")

l <- ifelse(pred.test > 0.65,1,0)
cm.test <- confusionMatrix(reference=as.factor(c_test$Caesarian),
            data=as.factor(l), mode="everything")
cm.test
```

```
Confusion Matrix and Statistics

          Reference
Prediction 0 1
         0 5 5
         1 2 5

               Accuracy : 0.5882
                 95% CI : (0.3292, 0.8156)
    No Information Rate : 0.5882
    P-Value [Acc > NIR] : 0.6022

                  Kappa : 0.2013

 Mcnemar's Test P-Value : 0.4497

            Sensitivity : 0.7143
            Specificity : 0.5000
         Pos Pred Value : 0.5000
         Neg Pred Value : 0.7143
              Precision : 0.5000
                 Recall : 0.7143
                     F1 : 0.5882
             Prevalence : 0.4118
         Detection Rate : 0.2941
   Detection Prevalence : 0.5882
      Balanced Accuracy : 0.6071

       'Positive' Class : 0
```

Balanced Accuracy: The average of sensitivity and specificity is 60.71%, slightly better than flipping a coin.

Confusion Matrix:

- True Negatives (TN): 5 (the model correctly predicted the negative class)
- True Positives (TP): 5 (the model correctly predicted the positive class)
- False Positives (FP): 5 (the model incorrectly predicted the positive class when it was actually negative)
- False Negatives (FN): 2 (the model incorrectly predicted the negative class when it was actually positive)

Performance Metrics:

- Accuracy: The model has an accuracy of 58.82%, meaning it correctly predicted the outcome about 58.82% of the time. The confidence interval of 95% CI (32.92% to 81.56%) indicates that we can be 95% confident that the accuracy of the model in the general population would fall within this range.
- No Information Rate (NIR): This rate is 58.82%, which is the proportion of the largest class. Since it equals the model's accuracy, it implies the model does no better than random guessing.
- P-Value [Acc > NIR]: The p-value of 0.6022 indicates there is no statistical evidence that the model's accuracy is different from the NIR; the model is not significantly better than random chance at the conventional 0.05 significance level.
- Kappa: A Kappa of 0.2013 suggests that there is slight agreement between the predictions and the actual values, corrected for chance.
- Sensitivity (Recall for the 'Positive' class): The model correctly predicts the positive class 71.43% of the time when it is actually positive.
- Specificity: The model correctly predicts the negative class 50.00% of the time when it is actually negative.
- Precision: Identical to Positive Predictive Value, at 50.00%. Recall: Identical to Sensitivity, at 71.43%.

2

```
lf=read.csv("C:/Users/vadre/Downloads/leveefailure.csv")
head(lf)
```

```
  Failure Year Rivermile Sediments Borrowpit Meander ChannelWidth Floodwaywidth
1       1 1890       847         0         0       4      1347.00       2025.54
2       1 1890       787         1         0       3      2580.99       4122.89
3       1 1890       776         1         0       3      3378.96       7998.81
4       1 1890       776         1         0       3      3507.43       8537.52
5       1 1890       773         1         0       2      1704.10       4173.75
6       1 1910       830         1         0       1      2822.97       5206.98
  ConstrictionFactor Landcover VegWidth Sinuosity Dredging Revetment  X
1             1.0000         4   391.31    1.9484        0         0 NA
2             1.0000         2   150.63    2.3318        0         0 NA
3             1.0000         4   145.15    1.9238        0         0 NA
4             1.0000         2   301.95    2.0305        0         0 NA
5             1.0000         2  1027.68    1.5503        0         0 NA
6             0.7004         3   399.08    1.3511    81115         0 NA
```

```
#Removing coloumn X as it contains only NA values
lf <- subset(lf, select = -X)
head(lf)
```

```
  Failure Year Rivermile Sediments Borrowpit Meander ChannelWidth Floodwaywidth
1       1 1890       847         0         0       4      1347.00       2025.54
2       1 1890       787         1         0       3      2580.99       4122.89
3       1 1890       776         1         0       3      3378.96       7998.81
4       1 1890       776         1         0       3      3507.43       8537.52
5       1 1890       773         1         0       2      1704.10       4173.75
6       1 1910       830         1         0       1      2822.97       5206.98
  ConstrictionFactor Landcover VegWidth Sinuosity Dredging Revetment
1             1.0000         4   391.31    1.9484        0         0
2             1.0000         2   150.63    2.3318        0         0
3             1.0000         4   145.15    1.9238        0         0
4             1.0000         2   301.95    2.0305        0         0
5             1.0000         2  1027.68    1.5503        0         0
6             0.7004         3   399.08    1.3511    81115         0
```

```
lf$Landcover <- as.factor(lf$Landcover)
lf$Meander <- as.factor(lf$Meander)
lf$Borrowpit <- as.factor(lf$Borrowpit)
lf$Sediments <- as.factor(lf$Sediments)
lf$Failure <- as.factor(lf$Failure)
```

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

```
table(lf$Failure)
```

```
 0  1
41 41
```

```
summary(lf_train$Failure)/nrow(lf_train)
```

```
  0   1
0.5 0.5
```

```
summary(lf_test$Failure)/nrow(lf_test)
```

```
  0   1
0.5 0.5
```

```
summary(lf$Failure)/nrow(lf)
```

```
  0   1
0.5 0.5
```

```
lf.full.logit <- glm(Failure ~ .  ,data = lf_train,
                     family = binomial(link = "logit"))
summary(lf.full.logit)
```

```
Call:
glm(formula = Failure ~ ., family = binomial(link = "logit"),
    data = lf_train)

Coefficients:
                    Estimate Std. Error z value Pr(>|z|)
(Intercept)        -2.247e+02  3.570e+03  -0.063   0.9498
Year                1.334e-01  6.475e-02   2.060   0.0394 *
Rivermile          -1.315e-02  1.155e-02  -1.139   0.2547
Sediments1         -9.245e-01  1.047e+00  -0.883   0.3774
Borrowpit1         -1.756e+00  1.702e+00  -1.032   0.3022
Meander2           -2.482e+00  1.578e+00  -1.573   0.1158
Meander3            1.880e+00  1.602e+00   1.173   0.2407
Meander4           -1.086e+00  1.354e+00  -0.802   0.4227
ChannelWidth        1.367e-03  8.100e-04   1.688   0.0915 .
Floodwaywidth      -4.926e-04  3.400e-04  -1.449   0.1473
ConstrictionFactor -1.564e+00  1.182e+00  -1.323   0.1858
Landcover2         -1.875e+01  3.567e+03  -0.005   0.9958
Landcover3         -2.036e+01  3.567e+03  -0.006   0.9954
Landcover4         -1.823e+01  3.567e+03  -0.005   0.9959
VegWidth            2.195e-05  4.509e-04   0.049   0.9612
Sinuosity           1.676e+00  1.366e+00   1.226   0.2201
Dredging           -5.963e-06  3.012e-06  -1.980   0.0477 *
Revetment          -1.796e+01  3.300e+03  -0.005   0.9957
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 88.723  on 63  degrees of freedom
Residual deviance: 45.187  on 46  degrees of freedom
AIC: 81.187

Number of Fisher Scoring iterations: 17
```

COEFFICIENTS:

- (Intercept): The estimate is extremely large in magnitude (-2.247e+02) with a very high standard error (3.570e+03), which results in an insignificant z-value (-0.063) and a high p-value (0.9498). This suggests that the intercept is not significantly different from zero.

- Year: The estimate is 0.1334, with a standard error of 0.06475, yielding a z-value of 2.060, which is significant at the 0.05 level (p-value = 0.0394). This suggests that as the year increases, the log-odds of Failure increases slightly.
- Rivermile, Sediments1, Borrowpit1, Meander2-4, ChannelWidth, Floodwaywidth, ConstrictionFactor, Landcover2-4, VegWidth, Sinuosity: These predictors have high p-values (greater than 0.05), suggesting that they are not significantly associated with the outcome Failure.
- Dredging: This predictor has an estimate of -5.963e-06 with a p-value of 0.0477, suggesting it is significantly associated with the outcome at the 0.05 level. The negative sign indicates that as dredging increases, the log-odds of Failure decrease slightly.

```
lf.null.logit <- glm(Failure~1, data = lf_train,
                     family = binomial(link = "logit"))
summary(lf.null.logit)
```

```
Call:
glm(formula = Failure ~ 1, family = binomial(link = "logit"),
    data = lf_train)

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) 1.665e-16  2.500e-01       0        1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 88.723  on 63  degrees of freedom
Residual deviance: 88.723  on 63  degrees of freedom
AIC: 90.723

Number of Fisher Scoring iterations: 2
```

The model appears to favour the complete model over the null model, as indicated by the high p-value of 0.5.

Anomalies

```
# Load necessary libraries
library(car) # for vif() function
```

```
Loading required package: carData
```

```
library(lmtest) # for coeftest() function
```

```
Loading required package: zoo
```

```
Attaching package: 'zoo'

The following objects are masked from 'package:base':

    as.Date, as.Date.numeric
```

```
library(broom) # for augment() function

# Check for multicollinearity
vif_results <- vif(lf.full.logit)
print(vif_results)
```

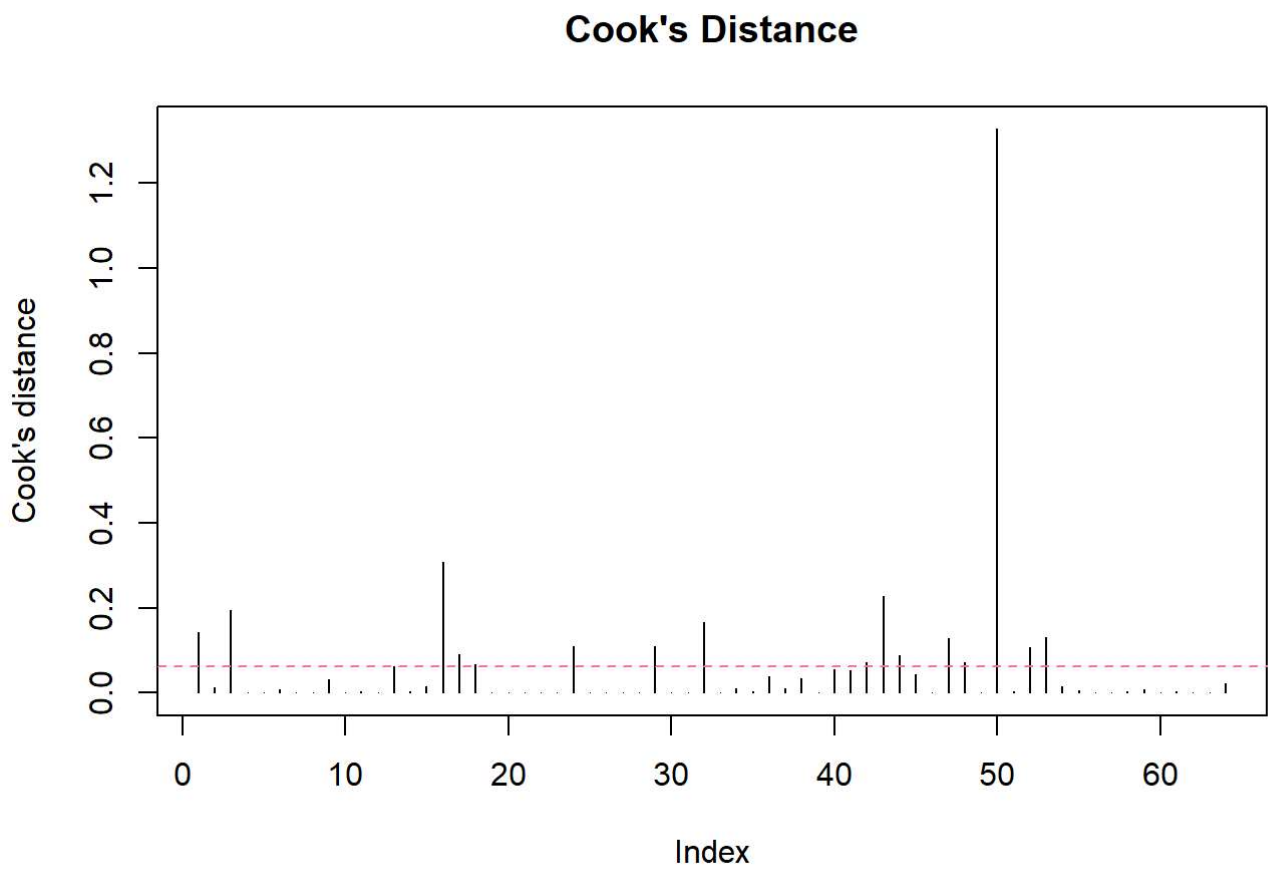
```
                     GVIF Df GVIF^(1/(2*Df))
Year                8.396035  1        2.897591
Rivermile           5.383449  1        2.320226
Sediments           1.943436  1        1.394072
Borrowpit           5.338071  1        2.310427
Meander             8.234749  3        1.421047
ChannelWidth        2.622252  1        1.619337
Floodwaywidth       4.135006  1        2.033471
ConstrictionFactor 2.423412  1        1.556731
Landcover           3.697442  3        1.243513
VegWidth            3.977019  1        1.994247
Sinuosity           1.806429  1        1.344035
Dredging            6.452806  1        2.540237
Revetment           1.000000  1        1.000000
```

```
# Identify influential observations using Cook's distance
cooks_distances <- cooks.distance(lf.full.logit)
influential <- which(cooks_distances > (4 / length(cooks_distances)))

# Print the cases that are influential
print(influential)
```

```
1  5 21 22 24 31 37 41 55 57 58 61 62 64 66 68
1  3 16 17 18 24 29 32 42 43 44 47 48 50 52 53
```

```
# Plot Cook's distance
plot(cooks_distances, type = "h", main = "Cook's Distance", ylab = "Cook's distance")
abline(h = 4 / length(cooks_distances), col = "hotpink", lty = 2) # threshold line
```




```
# Use the 'augment' function from broom to add the Cook's distance to the dataset
augmented_data <- augment(lf.full.logit)

# Investigate the influential cases
influential_cases <- augmented_data[influential, ]
print(influential_cases)
```

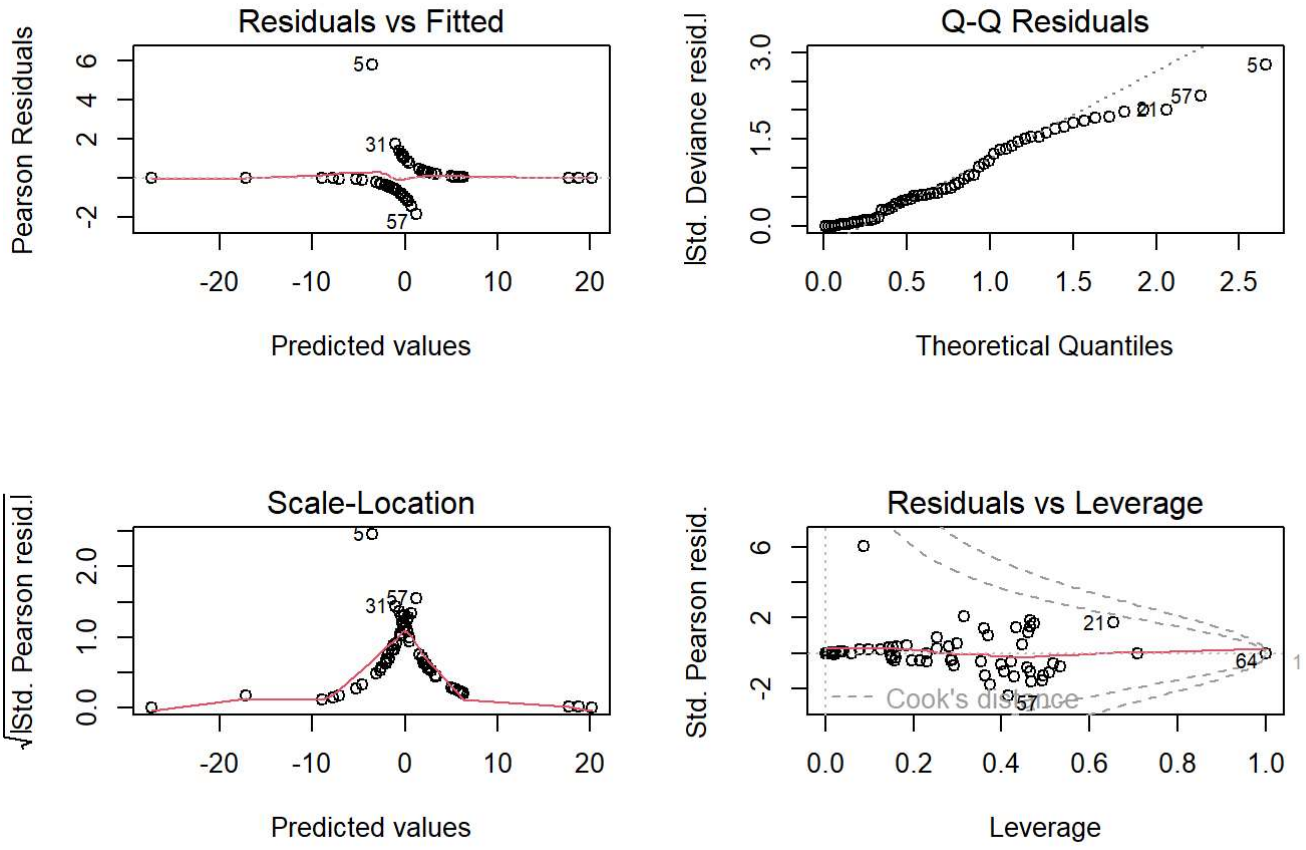
```
# A tibble: 16 × 21
   .rownames Failure  Year Rivermile Sediments Borrowpit Meander ChannelWidth
   <chr>     <fct>   <int>     <dbl> <fct>     <fct>     <fct>          <dbl>
 1 1         1        1890      847  0         0         4              1347
 2 5         1        1890      773  1         0         2              1704.
 3 21        1        1937      952  1         1         2               998.
 4 22        1        1937      939  0         1         1              1439.
 5 24        1        1937      924  1         1         4              3268.
 6 31        1        1937      893  0         1         2              1025.
 7 37        1        1937      830  1         1         4              3879
 8 41        1        1937      716  1         1         3              2153.
 9 55        0        1910      820. 0         0         2              1721.
10 57        0        1910      789. 1         0         4              2123.
11 58        0        1910      759. 1         0         2              2923.
12 61        0        1910      729. 1         0         1              2128.
13 62        0        1937      939. 0         1         1              1656.
```

```
14 64        0        1937      912. 0         1         1              1926.
15 66        0        1937      905. 0         1         4              2018.
16 68        0        1937      884. 1         1         3              2004.
# ℹ 13 more variables: Floodwaywidth <dbl>, ConstrictionFactor <dbl>,
#   Landcover <fct>, VegWidth <dbl>, Sinuosity <dbl>, Dredging <int>,
#   Revetment <int>, .fitted <dbl>, .resid <dbl>, .hat <dbl>, .sigma <dbl>,
#   .cooksd <dbl>, .std.resid <dbl>
```

```r
# plot diagnostic plots for logistic regression
par(mfrow=c(2, 2))
plot(lf.full.logit)
```

```
Warning in sqrt(crit * p * (1 - hh)/hh): NaNs produced

Warning in sqrt(crit * p * (1 - hh)/hh): NaNs produced
```



Most of the points have a Cook's distance close to zero, indicating that they have little influence on the model. However, there is one particular observation, right around the index number 50, with a Cook's distance that dramatically exceeds 1. This indicates that this point is highly influential and has a large impact on the calculation of the regression coefficients. Depending on the context and further analysis, this point could be an outlier or an indication of some substantive issue with the model such as missing an important predictor or having a wrong functional form.

The dashed lines represent Cook's distance, a measure of the influence of each observation. Most data points are within the Cook's distance contours, suggesting they are not influential to the model's fit. However, points labeled with numbers (like 21 and 64) fall outside the Cook's distance lines, suggesting they are influential points and outliers that could be unduly affecting the model's predictions.

The points largely follow the 45-degree reference line.The points seem to form a slight "U" shape (look at points around -10 and 10 on the x-axis), which suggests that there might be some non-linearity in the relationship between the variables.

Using Forward Variable Selections to select best predictor variables

```
f = step(lf.null.logit, trace=0,
      scope=list(lower=formula(lf.null.logit),
                 upper=formula(lf.full.logit)), direction="forward")
formula(f)
```

```
Failure ~ Sinuosity + ChannelWidth + Floodwaywidth + Meander
```

```
summary(f)
```

```
Call:
glm(formula = Failure ~ Sinuosity + ChannelWidth + Floodwaywidth +
    Meander, family = binomial(link = "logit"), data = lf_train)

Coefficients:
               Estimate Std. Error z value Pr(>|z|)
(Intercept)   -2.4071967  2.1235994  -1.134   0.2570
Sinuosity      1.7676452  0.9466071   1.867   0.0619 .
ChannelWidth   0.0007846  0.0004488   1.748   0.0804 .
Floodwaywidth -0.0003343  0.0001708  -1.957   0.0504 .
Meander2      -1.7623320  0.9896817  -1.781   0.0750 .
Meander3       0.5085299  0.9301440   0.547   0.5846
Meander4      -0.8200030  0.8766930  -0.935   0.3496
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 88.723  on 63  degrees of freedom
Residual deviance: 64.706  on 57  degrees of freedom
AIC: 78.706

Number of Fisher Scoring iterations: 4
```

Fitting the model with best predictors (additional step)

```
lf.both.logit = glm(Failure~Sinuosity + ChannelWidth + Floodwaywidth + Meander , data=lf_train,fa
```

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

```
Call:
glm(formula = Failure ~ Sinuosity + ChannelWidth + Floodwaywidth +
    Meander, family = binomial(link = "logit"), data = lf_train)

Coefficients:
               Estimate Std. Error z value Pr(>|z|)
(Intercept)   -2.4071967  2.1235994  -1.134   0.2570
Sinuosity      1.7676452  0.9466071   1.867   0.0619 .
ChannelWidth   0.0007846  0.0004488   1.748   0.0804 .
Floodwaywidth -0.0003343  0.0001708  -1.957   0.0504 .
Meander2      -1.7623320  0.9896817  -1.781   0.0750 .
Meander3       0.5085299  0.9301440   0.547   0.5846
Meander4      -0.8200030  0.8766930  -0.935   0.3496
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 88.723  on 63  degrees of freedom
Residual deviance: 64.706  on 57  degrees of freedom
AIC: 78.706

Number of Fisher Scoring iterations: 4
```

COEFFICIENTS:

- (Intercept): The estimate is -2.407 with a standard error of 2.123, giving a z-value of -1.134 which is not statistically significant (p-value = 0.2570).
- Sinuosity: Has a positive estimate of 1.7676, with a standard error of 0.9466. It has a z-value of 1.867, which is marginally significant (p-value = 0.0619), suggesting that it is close to being a significant predictor of Failure.
- ChannelWidth: The coefficient is small (0.0007846) and has a p-value of 0.0804, which is close to the conventional significance level of 0.05, indicating a trend towards significance.
- Floodwaywidth: Has a negative coefficient (-0.0003343) and is marginally significant with a p-value of 0.0504, suggesting it is also close to being a significant predictor.
- Meander2: This categorical variable has a negative coefficient (-1.7623) with a p-value of 0.0750, indicating it is not significantly different from the baseline category of Meander at the 0.05 level, but it is close.
- Meander3 and Meander4: These coefficients are not significant with p-values of 0.5846 and 0.3496, respectively.

Assessing Test Data

```
pred.both <- predict(lf.both.logit, newdata = lf_test, type="response")
pred.full <- predict(lf.full.logit, newdata = lf_test, type="response")
```

```
(table.both <- table(pred.both > 0.5, lf_test$Failure))
```

```
       0 1
 FALSE 4 3
 TRUE  5 6
```

```
(table.full <- table(pred.full > 0.5, lf_test$Failure))
```

```
       0 1
 FALSE 3 3
 TRUE  6 6
```

```
(accuracy.both <- round((sum(diag(table.both))/sum(table.both))*100,2))
```

```
[1] 55.56
```

```
(accuracy.full <- round((sum(diag(table.full))/sum(table.full))*100,2))
```

```
[1] 50
```

The accuracy of the lf.both.logit seems to be a better fit due to the higher accuracy, relative to the logit.full.model.

```
library(pROC)
roc.both <- roc(lf_test$Failure, pred.both, levels=c(1,0))
```

```
Setting direction: controls > cases
```

```
roc.full <- roc(lf_test$Failure, pred.full, levels=c(1,0))
```

```
Setting direction: controls > cases
```

Assessing Train Data

```
pred.both <- predict(lf.both.logit, newdata = lf_train, type="response")
pred.full <- predict(lf.full.logit, newdata = lf_train, type="response")
(table.both <- table(pred.both > 0.5, lf_train$Failure))
```

```
       0  1
FALSE 24  8
TRUE   8 24
```

```
(table.full <- table(pred.full > 0.5, lf_train$Failure))
```

```
       0  1
FALSE 27  8
TRUE   5 24
```

```
(accuracy.both <- round((sum(diag(table.both))/sum(table.both))*100,2))
```

```
[1] 75
```

```
(accuracy.full <- round((sum(diag(table.full))/sum(table.full))*100,2))
```

```
[1] 79.69
```

```
library(pROC)
roc.both <- roc(lf_train$Failure, pred.both, levels=c(1,0))
```

```
Setting direction: controls > cases
```

```
roc.full <- roc(lf_train$Failure, pred.full, levels=c(1,0))
```

```
Setting direction: controls > cases
```

For the train data, the accuracy seems to be better for the lf.full.logit with an accuracy of 79.69

Best Predictors with Backwards Elimination

```
b <- step(lf.full.logit, trace = 0)
formula(b)
```

```
Failure ~ Year + Rivermile + Meander + Floodwaywidth + ConstrictionFactor +
    Landcover + Sinuosity + Dredging
```

```
summary(b)
```

```

Call:
glm(formula = Failure ~ Year + Rivermile + Meander + Floodwaywidth +
    ConstrictionFactor + Landcover + Sinuosity + Dredging, family = binomial(link = "logit"),
    data = lf_train)

Coefficients:
```

```
                    Estimate Std. Error z value Pr(>|z|)
(Intercept)        -1.792e+02  2.045e+03  -0.088   0.9302
Year                1.112e-01  5.536e-02   2.009   0.0445 *
Rivermile          -1.892e-02  8.723e-03  -2.169   0.0301 *
Meander2           -2.710e+00  1.231e+00  -2.201   0.0277 *
Meander3            1.477e+00  1.314e+00   1.124   0.2611
Meander4           -3.924e-01  1.054e+00  -0.372   0.7096
Floodwaywidth      -3.097e-04  2.092e-04  -1.480   0.1388
ConstrictionFactor -1.620e+00  9.789e-01  -1.655   0.0979 .
Landcover2         -1.701e+01  2.043e+03  -0.008   0.9934
Landcover3         -1.943e+01  2.043e+03  -0.010   0.9924
Landcover4         -1.766e+01  2.043e+03  -0.009   0.9931
Sinuosity           2.577e+00  1.262e+00   2.043   0.0411 *
Dredging           -6.496e-06  2.642e-06  -2.458   0.0140 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 88.723  on 63  degrees of freedom
Residual deviance: 50.644  on 51  degrees of freedom
AIC: 76.644

Number of Fisher Scoring iterations: 16
```

```
pred <- predict(b, newdata = lf_test, type="response")
(table <- table(pred > 0.5, lf_test$Failure))
```

```
       0 1
  FALSE 4 3
  TRUE  5 6
```

```
(accuracy <- round((sum(diag(table.both))/sum(table))*100,2))
```

```
[1] 266.67
```

```
roc <- roc(lf_test$Failure, pred, levels=c(1,0))
```

```
Setting direction: controls > cases
```

The AIC of the model using backward variable selection is relatively less than the AIC of the model created using forward variable selection. In comparison to the other models, it.also has a high AUC-ROC of 0.67 and its accuracy is comparable to the logit.both.model but superior to the entire model's 55% accuracy.