Candidate number : 41380

2024-11-19

```r
#Clearing the envrionment
rm(list =ls())

#Loading libraries
library(ggplot2)
```

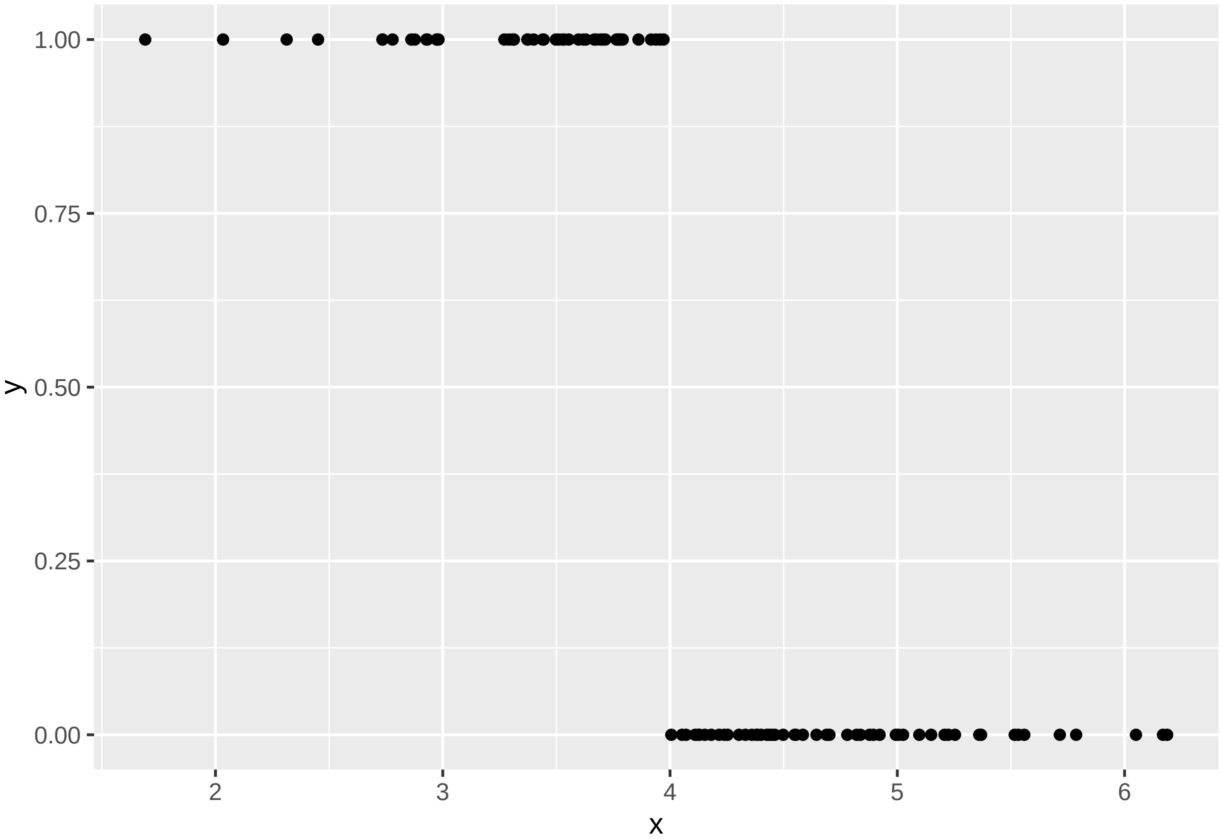
Part 1

```r
#For reproducibiltiy
set.seed(123)

#Setting parameters
n <- 100
c <- 4 # X* = c is the boundary

#Generating data
x <- rnorm(n, mean = c, sd = 1)
y <- ifelse(x < c, 1, 0)
data <- data.frame(x = x, y = y)

#Plotting the data
ggplot(data = data, aes(x = x, y =y )) +
geom_point()
```



This plot shows complete separation.

Part 2

```
#Logistic regression
logistic_model <- glm(y ~ x, family = binomial(link = "logit"))
```

```
## Warning: glm.fit: algorithm did not converge
```

```
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
```

```
summary(logistic_model)
```

```
##
## Call:
## glm(formula = y ~ x, family = binomial(link = "logit"))
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)    3889.8   428450.1   0.009    0.993
## x              -975.2   107438.0  -0.009    0.993
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1.3847e+02  on 99  degrees of freedom
## Residual deviance: 2.1701e-07  on 98  degrees of freedom
## AIC: 4
##
## Number of Fisher Scoring iterations: 25
```

Interpretation :

Coefficients: The intercept estimate is 3889.8, and the coefficient for x is -975.2. These values are very large, this is due to separation in the data.

Standard Errors: The standard errors are extremely large (428450.1 for the intercept and 107438.0 for x), making the estimates unreliable. Large standard errors indicate the difficulty in estimating coefficients precisely.

z-values and p-values: The z-values (0.009 and -0.009) are close to zero, and the p-values (0.993) are very high. This shows that neither the intercept nor x is statistically significant in predicting y, but these results are likely distorted by instability in the model.

Deviance: The null deviance (1.3847e+02) represents the deviance of the model with only the intercept. The residual deviance is nearly zero (2.1701e-07), suggesting the model perfectly fits the data.

AIC: The AIC is 4, which is low and reflects the apparent perfect fit. However, such a low AIC is misleading in this context, as it arises from the model's inability to generalise the fit.

Convergence and Iterations: The model required 25 iterations to converge, which is relatively high and the model struggled to estimate parameters reliably.

Explanation of Warnings :

Warning (1: glm.fit: algorithm did not converge):

Logistic regression uses maximum likelihood to estimate the coefficients because it has has a fully parametric model of y conditional on x. The likelihood function increases without bound as the coefficients tend to infinity, due to this complete separation which is shown on the plot above.

Warning (2: glm.fit: fitted probabilities numerically 0 or 1 occurred):

Due to the perfect separation, the probabilities are 0 or 1. In addition, the standard errors of the coefficients are very large and the p -values are very close to 1. There is a large uncertainty given by the model, so it is quite unreliable.

Part 3

To address the issue of complete seperation in logistic regression and prevent the coefficients becoming very large and uncertain, a penalty term could be added.The lasso regression model introduces a parameter lambda, such that lambda > 0. To avoid the large coefficients, the residual sum of squares criterion is penalised with the squared lengths of the coefficients. By penalising large coefficients, lambda ensures stability and prevents divergence even in cases of perfect seperation.

So the method would consist of fitting a logistics regression with lasso regularisation, then using cross validation to find the best lambda. Using this model the coefficients can then be extracted.

Part 4

I expect to see a significant p - value for the coefficients of x1 and x2, because the outcome y is completely determined by x1 and x2. In addition, because the other predictors are just noise, the p-value for them should be close to 1. The binary outcome of y also means that the model will have high variability, so the standard errors for the coefficients should be very high.

```r
n <- 100
p <- 10
x <- matrix(rnorm(n * p), ncol = p)
y <- (sign(x[, 1] + x[, 2]) + 1)/2
d <- data.frame(y = y, x = x)

model <- glm(y ~ ., data = d, family = binomial(link = "logit"))
```

```
## Warning: glm.fit: algorithm did not converge
```

```
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
```

```r
summary(model)
```

```
##
## Call:
## glm(formula = y ~ ., family = binomial(link = "logit"), data = d)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)     7.183   7941.956   0.001    0.999
## x.1           179.786  46273.422   0.004    0.997
## x.2           198.001  50629.037   0.004    0.997
## x.3           -15.205  10566.353  -0.001    0.999
## x.4            -7.446   9143.372  -0.001    0.999
## x.5            -8.581   8391.972  -0.001    0.999
## x.6             0.947   6409.091   0.000    1.000
## x.7            -6.695  12669.743  -0.001    1.000
## x.8            -8.916  13592.999  -0.001    0.999
## x.9            -4.152  12209.882   0.000    1.000
## x.10           -1.476  16578.408   0.000    1.000
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1.3863e+02  on 99  degrees of freedom
## Residual deviance: 2.7798e-08  on 89  degrees of freedom
## AIC: 22
##
## Number of Fisher Scoring iterations: 25
```

We see the same warnings as above because that same problem is apparent here. y is a binary variable. It is also only dependent of the predictor x1 and x2, so the variables from x3 to x10 are noise.

From the summary we can see the coefficients of x1 and x2 (135.1334 and 129.1501) are much larger than the rest. They also have large standard errors (58095.2044 and 54457.2632). The p - value is 1 for all coefficients expect x1 and x2 where it is 0.998. So the model cannot confidently detect which predictors are significant. This should not be the case because only x1 and x2 drive the outcome of y.

This result could be driven by the inclusion on the irrelevant predictors of x3, .. , x10 which adds noise and unnecessarily increases the complexity of the model.