```
> install.packages('datasets')
> install.packages('tidyverse')
> install.packages('dplyr')
> rm(list = ls())
> set.seed(1000660251)
>
> ### Question 2 ###
> library(datasets)
> data(warpbreaks)
> head(warpbreaks)
  breaks wool tension
1     26    A       L
2     30    A       L
3     54    A       L
4     25    A       L
5     70    A       L
6     52    A       L
>
> ## 2a) Poisson Regression ##
> summary(glm(breaks ~ wool + tension, family = poisson, data = warpbreaks))

Call:
glm(formula = breaks ~ wool + tension, family = poisson, data = warpbreaks)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-3.6871  -1.6503  -0.4269   1.1902   4.2616

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  3.69196    0.04541  81.302  < 2e-16 ***
woolB       -0.20599    0.05157  -3.994 6.49e-05 ***
tensionM    -0.32132    0.06027  -5.332 9.73e-08 ***
tensionH    -0.51849    0.06396  -8.107 5.21e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for poisson family taken to be 1)
```

```
    Null deviance: 297.37  on 53  degrees of freedom
Residual deviance: 210.39  on 50  degrees of freedom
AIC: 493.06

Number of Fisher Scoring iterations: 4

> # Our mean value would be exp(intercept) = exp(3.69196)
>
> # For woolB this is the estimate that for one increase point while other variables remain constant.
> # For an increase in woolB by one point, the difference in the logs of expected counts would be
> # expected to decrease by 0.20599 units, while holding the other variables in the model constant.
> # woolB's p value is much smaller than an alpha level of 0.05, therefore we reject the null hypothesis
> # and woolB is statistically significant.
>
> # For tensionM its estimate for one point increase would be the difference in the logs of expected
> # counts and it would decrease by 0.32132, while holding other variables constant.
> # its p value < 0.05 therefore we reject the null hypothesis that tensionM has no effect on wool breaks
> # and tensionM is statistically significant.
>
> # For tensionH its estimate for one point increase would be the difference in the logs of expected
> # counts and it would decrease by 0.051849, when other variables are held constant.
> # tensionH's pvalue < 0.05, therefore we reject the null hypothesis and tensionH is statistically significant.
>
>
> ## 2b) Negative Binomial Regression ##
> library(MASS)
> summary(glm.nb(breaks ~ wool + tension, data = warpbreaks))

Call:
glm.nb(formula = breaks ~ wool + tension, data = warpbreaks,
    init.theta = 9.944385436, link = log)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.0144  -0.9319  -0.2240   0.5828   1.8220

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)   3.6734     0.0979  37.520  < 2e-16 ***
```

```
woolB        -0.1862     0.1010  -1.844   0.0651 .
tensionM     -0.2992     0.1217  -2.458   0.0140 *
tensionH     -0.5114     0.1237  -4.133 3.58e-05 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for Negative Binomial(9.9444) family taken to be 1)

    Null deviance: 75.464  on 53  degrees of freedom
Residual deviance: 53.723  on 50  degrees of freedom
AIC: 408.76

Number of Fisher Scoring iterations: 1


              Theta:  9.94
          Std. Err.:  2.56

 2 x log-likelihood:  -398.764
>
> # the mean for this model is the dispersion parameter of 9.94
> # for one increase in woolB there is a decrease in the difference in the logs of the
> # expected counts by 0.1862.
> # woolB's pvalue > alpha = 0.05 therefore we fail to reject the null hypothesis
>
> # for one increase in tensionM there is a decrease in the diff in the logs of the expected
> # counts by 0.2992.
> # tensionM's pvalue < alpha = 0.05 therefore we reject the null hypothesis and tensionM is
> # statistically significant.
>
> # for one increase in tensionH there is a decrease in the diff in the logs of the expected
> # counts by 0.5114.
> # tensionH's pvalue <<<< alpha=0.05 therefore we reject the null hypothesis and tensionH
> # is statistically significant
>
> ## 2c) Model Comparison ##
> # The AIC for the poisson regression is 493.06, whereas for the negative
> # binomial regeression it is 408.76.
> # Therefore the negative binomial regression is a better fit given the
```

```
> # smaller AIC value.
> # For neg. bin. the dispersion factor is: 1/k = 1/theta = 1/9.94
> # This value is not close to zero.
> # Therefore we cannot use the poisson regression model since it would lead to
> # overdispersion. Negative binomaial regression model is a better model to use.
>
>
> ### Question 3 ###
> ## 3a) Random Simulations ##
> set.seed(1000660251)
> n <- 500
> X1 <- runif(n, 0, 1)
> X2 <- runif(n, 0, 1)
> X3 <- runif(n, 0, 1)
> X4 <- runif(n, 0, 1)
> X5 <- runif(n, 0, 1)
> fX <- 4*(sin(pi*X1*X2) + 8*(X3 - 0.5)^3 + 1.5*X4 - X5 - 0.77)
> pX <- (exp(fX))/(1 + exp(fX))
> Y <- rbinom(n, 1, pX)
>
> ## 3b) Logistic Regression ##
> library(nnet)
> library(pROC)
> df <- c(list(X1, X2, X3, X4, X5))
> #df
> multi.mod <- multinom(Y ~ X1 + X2 + X3 + X4 + X5, data = df)
# weights:  7 (6 variable)
initial  value 346.573590
iter  10 value 208.899879
final  value 208.839465
converged
> summary(multi.mod)
Call:
multinom(formula = Y ~ X1 + X2 + X3 + X4 + X5, data = df)

Coefficients:
             Values Std. Err.
(Intercept) -5.257012 0.6177702
X1           2.155847 0.4400079
```

```
X2           2.625741 0.4496319
X3           3.534753 0.4688883
X4           5.467585 0.5802606
X5          -3.390456 0.4882242

Residual Deviance: 417.6789
AIC: 429.6789
> logit.mod <- glm(Y ~ X1 + X2 + X3 + X4 + X5, family = binomial(link = logit), data = df)
> summary(logit.mod)

Call:
glm(formula = Y ~ X1 + X2 + X3 + X4 + X5, family = binomial(link = logit),
    data = df)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-3.0276  -0.6113  -0.1468   0.6685   2.2369

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  -5.2570     0.6178  -8.510  < 2e-16 ***
X1            2.1558     0.4400   4.900 9.60e-07 ***
X2            2.6258     0.4496   5.840 5.22e-09 ***
X3            3.5348     0.4689   7.539 4.74e-14 ***
X4            5.4676     0.5802   9.423  < 2e-16 ***
X5           -3.3904     0.4882  -6.945 3.79e-12 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 692.50  on 499  degrees of freedom
Residual deviance: 417.68  on 494  degrees of freedom
AIC: 429.68

Number of Fisher Scoring iterations: 5

>
> # ROC Curve #
```

```
> library(pROC)
> pred <- predict(logit.mod, type = "response")
> roc_logit <- roc(Y ~ pred)
Setting levels: control = 0, case = 1
Setting direction: controls < cases
> ## The True Positive Rate ##
> TPR <- roc_logit$sensitivities
> ## The False Positive Rate ##
> FPR <- 1 - roc_logit$specificities
> pdf("ROC_LOG_REG.pdf")
> plot(FPR, TPR, xlim = c(0,1), ylim = c(0,1), type = 'l', lty = 1, lwd = 2,col = 'red',
+      main = "Logistic Model Predictors vs Outcome (3b)")
> abline(a = 0, b = 1, lty = 2, col = 'blue')
> text(0.7,0.4,label = paste("AUC = ", round(auc(roc_logit),2)))
> dev.off()
null device
          1
> auc(roc_logit)
Area under the curve: 0.8888
> # We see that the AUC is 0.8888 indicating the model can discriminate between true
> # positive rate and a false positive rate 88% of the time.
> # From the coefficients we see that for a one point increase in X1, X2, X3, X4
> # there is an increase in the difference of the log odds.
> # While there is a decrease in the difference in the log odds of 3.3904 for an increase in X5
>
> ## 3c) Linear Transformation Predictor ##
> new_X1 <- 4*(sin(pi*X1*X2))
> new_X2 <- 32*(X3 - 0.5)^3
> new_X3 <- 4*(1.5*X4)
> new_X4 <- 4*(-X5 - 0.77)
>
> transformed.logit.mod <- glm(Y ~ new_X1 + new_X2 + new_X3 + new_X4,
+                              family = binomial(link = logit), data = df)
> summary(transformed.logit.mod)

Call:
glm(formula = Y ~ new_X1 + new_X2 + new_X3 + new_X4, family = binomial(link = logit),
    data = df)
```

```
Deviance Residuals:
     Min        1Q    Median        3Q       Max
-2.98659  -0.54712  -0.09419   0.55662   2.81098

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  -0.1854     0.6390  -0.290    0.772
new_X1        0.9183     0.1124   8.168 3.14e-16 ***
new_X2        0.8001     0.1032   7.753 8.96e-15 ***
new_X3        1.0129     0.1069   9.476  < 2e-16 ***
new_X4        0.9512     0.1325   7.178 7.09e-13 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 692.50  on 499  degrees of freedom
Residual deviance: 377.23  on 495  degrees of freedom
AIC: 387.23

Number of Fisher Scoring iterations: 6

>
> # ROC Curve #
> library(pROC)
> transformed.pred <- predict(transformed.logit.mod, type = "response")
> roc_logit.transformed <- roc(Y ~ transformed.pred)
Setting levels: control = 0, case = 1
Setting direction: controls < cases
> ## The True Positive Rate ##
> TPR.transformed <- roc_logit.transformed$sensitivities
> ## The False Positive Rate ##
> FPR.transformed <- 1 - roc_logit.transformed$specificities
> pdf("ROC_LOG_REG_TRANSFORMED.pdf")
> plot(FPR.transformed, TPR.transformed, xlim = c(0,1), ylim = c(0,1), type = 'l', lty = 1, lwd = 2,col = 'red',
+      main = "Linearly Transformed Predictor vs Outcome (3C)")
> abline(a = 0, b = 1, lty = 2, col = 'blue')
> text(0.7,0.4,label = paste("AUC = ", round(auc(roc_logit.transformed),2)))
> dev.off()
```

```
null device
          1
> auc(roc_logit.transformed)
Area under the curve: 0.9105
> # Here we see that the AUC is 0.9105,indicating that this model can discriminate
> # between a true positive rate from a false positive rate 91% of the time.
> # All four newly transformed variables provide an increase by some value (different for each
> # transformed predictor) in the difference in the log odds with a one point increase in their
> # respective transformed predictor value.
>
> ## 3d) Interpretation ##
> # First we see that the logit.mod (from 3b) has a higher AIC value (429.68) compared
> # to the transformed.logit.mod (in 3c) with an AIC value of 387.23.
> # This indicates that the transformed.logit.mod has a better goodness of fit.
>
> # Secondly, the coefficents in the transformed model(3c) are smaller absolute values
> # than the logistic regression(3b).
> # and there is 1 negative coefficent in the transformed model, whereas there
> # are 2 in the logistic regression.
> # This could suggest that the transformed model has more accurate odds given the
> # very large and extreme odds found in the logistic regression coefficients.
>
> # Thirdly, the AUC for the transformed model is 0.9105 and the AUC for the logistic
> # regression model is 0.8888.
> # This suggests the transformed model is much better at discriminating TPR from
> # FPR compared to the logistic model.
>
> # All in all, these findings indicate that the transformed model is a better
> # model for predicting compared to the logistic model in 3b.
> # This could be due to the fact that the new_X variables(3c) that were linearly
> # transformed to match the fX function,
> # which was used in the simulating Y, the outcome variable, better
> # than the X variables in 3b.
> # In other words the linearly transformed variables were better at simulating
> # values that Y was also producing.
> # This lead to better overall prediction due to lower AIC score,
> # goodness of fit and stronger coefficients and AUC.

>
```