

C240424_5

YUSEF IBRAHIM GOMAA MAHMOUD MABROUK

2024-12-22

Exercise 5-1

- The link function's purpose in a GLM to connect the expected value of the response variable (μ) to the linear predictors $\alpha + \beta_1x_1 + \beta_2x_2 + \dots + \beta_px_p$. It ensures the model handles different types of data properly by transforming μ to ensure that predictions are meaningful for the type of data being modeled.
- For binomial models, the identity link $g(\mu) = \mu$ is simple but difficult to work with as it allows predictions for probabilities (π) outside the valid range $[0,1]$.
 - As a measure to avoid this, we use the logit link: $g(\pi) = \log\left(\frac{\pi}{1-\pi}\right)$, where $0 \leq \pi \leq 1$.

Exercise 5-2

(a) (i)

```
HDi.df <- data.frame(Yes = c(24, 35, 21, 30),
                      No = c(1355, 603, 192, 224),
                      x = c(0, 2, 4, 6)
                      )
rownames(HDi.df) <- c("Never", "Occasional",
                      "Nearly every night", "Every night"
                      )
HDi.df
```

##	Yes	No	x
## Never	24	1355	0
## Occasional	35	603	2
## Nearly every night	21	192	4
## Every night	30	224	6

```
glm(cbind(Yes, No) ~ x,
    family = binomial(link = "logit"),
    data = HDi.df
)
```

```
##
## Call: glm(formula = cbind(Yes, No) ~ x, family = binomial(link = "logit"),
## data = HDi.df)
##
## Coefficients:
## (Intercept)          x
##      -3.7774      0.3273
##
## Degrees of Freedom: 3 Total (i.e. Null); 2 Residual
## Null Deviance:      65.9
## Residual Deviance: 6.24 AIC: 30.49
```

(ii)

```
HDii.df <- data.frame(Yes = c(24, 35, 21, 30),
                      No = c(1355, 603, 192, 224),
                      x = c(0, 1, 2, 3)
)
rownames(HDii.df) <- c("Never", "Occasional",
                      "Nearly every night", "Every night"
)
HDii.df
```

```
##           Yes   No x
## Never      24 1355 0
## Occasional  35  603 1
## Nearly every night 21 192 2
## Every night  30  224 3
```

```
glm(cbind(Yes, No) ~ x,
    family = binomial(link = "logit"),
    data = HDii.df
)
```

```
##
## Call: glm(formula = cbind(Yes, No) ~ x, family = binomial(link = "logit"),
## data = HDii.df)
##
## Coefficients:
## (Intercept)          x
##      -3.7774      0.6545
##
## Degrees of Freedom: 3 Total (i.e. Null); 2 Residual
## Null Deviance:      65.9
## Residual Deviance: 6.24 AIC: 30.49
```

(iii)

```
HDiii.df <- data.frame(Yes = c(24, 35, 21, 30),
                      No = c(1355, 603, 192, 224),
                      x = c(1, 2, 3, 4)
                      )
rownames(HDiii.df) <- c("Never", "Occasional",
                       "Nearly every night", "Every night"
                       )
HDiii.df
```

```
##              Yes    No x
## Never          24 1355 1
## Occasional     35  603 2
## Nearly every night 21  192 3
## Every night    30  224 4
```

```
glm(cbind(Yes, No) ~ x,
    family = binomial(link = "logit"),
    data = HDiii.df
)
```

```
##
## Call: glm(formula = cbind(Yes, No) ~ x, family = binomial(link = "logit"),
##      data = HDiii.df)
##
## Coefficients:
## (Intercept)              x
##      -4.4319         0.6545
##
## Degrees of Freedom: 3 Total (i.e. Null);  2 Residual
## Null Deviance:          65.9
## Residual Deviance: 6.24  AIC: 30.49
```

- Based on the results, the coefficient of x (β) is inversely proportional to the scale of the scores. However, shifting the scores had little to no effect on it, but instead adjusted the intercept.

(b)

```
HD_new <- data.frame(Yes = c(24, 35, 21, 30),
                    No = c(1355, 603, 192, 224),
                    x_new = c(0, 2, 6, 7))

HD_text <- data.frame(Yes = c(24, 35, 21, 30),
                    No = c(1355, 603, 192, 224),
                    x_text = c(0, 2, 4, 5))
```

```
fit_new <- glm(cbind(Yes, No) ~ x_new,
              family = binomial(link = "logit"),
              data = HD_new)
```

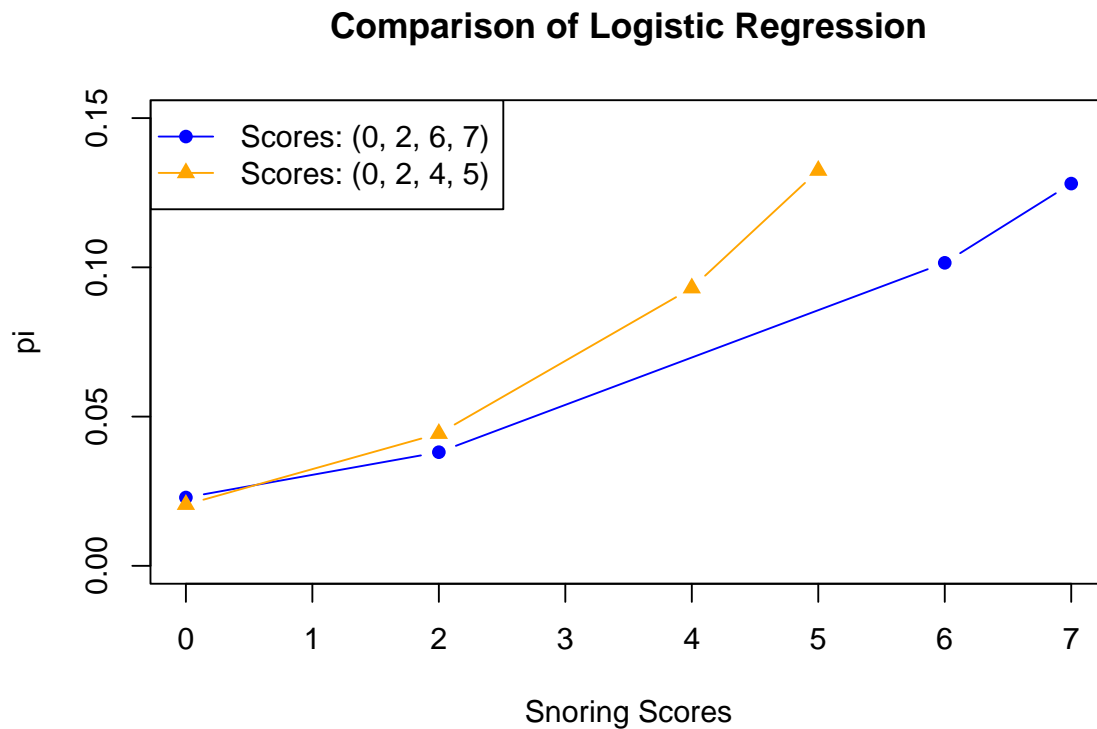
```
fit_text <- glm(cbind(Yes, No) ~ x_text,
               family = binomial(link = "logit"),
               data = HD_text)
```

```

HD_new$fitted_new <- predict(fit_new, type = "response")
HD_text$fitted_text <- predict(fit_text, type = "response")

plot(HD_new$x_new, HD_new$fitted_new,
     type = "b", col = "blue", pch = 16, ylim = c(0, 0.15),
     xlab = "Snoring Scores", ylab = "pi",
     main = "Comparison of Logistic Regression")
lines(HD_text$x_text, HD_text$fitted_text,
     type = "b", col = "orange", pch = 17)
legend("topleft",
     legend = c("Scores: (0, 2, 6, 7)", "Scores: (0, 2, 4, 5)"),
     col = c("blue", "orange"), pch = c(16, 17), lty = 1)

```



- Yes, it is slightly sensitive. As the change in the score slightly affected α , β and π .

Exercise 5-3

- (a) $\because x_A = 0, x_B = 1, \log \mu = \alpha + \beta x$
 $\therefore \log \mu_A = \alpha + \beta * (0) = \alpha$
 $\& \log \mu_B = \alpha + \beta * (1) = \alpha + \beta$
 $\therefore \log \mu_B - \log \mu_A = \alpha + \beta - \alpha = \beta = \log\left(\frac{\mu_B}{\mu_A}\right)$, raising to exponent for R.H.S and L.H.S
 $\therefore e^\beta = e^{\log\left(\frac{\mu_B}{\mu_A}\right)} = \frac{\mu_B}{\mu_A}$

```
(b) treat.df <- data.frame(
      treatment <- c(rep(0, 10), rep(1, 10)),
      imperfections <- c(8, 7, 6, 6, 3, 4, 7, 2, 3, 4, 9, 9, 8, 14, 8, 13, 11, 5, 7, 6)
    )
treat.fit <- glm(imperfections ~ treatment,
  family = poisson(link = "log"),
  data = treat.df
)
summary(treat.fit)
```

```
##
## Call:
## glm(formula = imperfections ~ treatment, family = poisson(link = "log"),
##      data = treat.df)
##
## Coefficients:
##              Estimate Std. Error z value      Pr(>|z|)
## (Intercept)   1.6094     0.1414  11.380 < 0.0000000000000002 ***
## treatment     0.5878     0.1764   3.332     0.000861 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##      Null deviance: 27.857  on 19  degrees of freedom
## Residual deviance: 16.268  on 18  degrees of freedom
## AIC: 94.349
##
## Number of Fisher Scoring iterations: 4
```

- (c) $z_w = \frac{\hat{\beta}_{treatment}}{SE(\hat{\beta}_{treatment})} = \frac{0.06355}{0.04971} = 3.3321995$
 $\therefore p < .05$, at significance level $\alpha = .05$, we reject the null hypothesis that $\beta_{treatment} = 0$
 This means there is a statistically significant difference in the imperfections between treatment A and treatment B.

```
confint.default(treat.fit)
```

```
##              2.5 %    97.5 %
## (Intercept) 1.332258 1.8866181
## treatment   0.242082 0.9334913
```

#Exercise 5-4

```
(a) intercept.fit <- glm(imperfections ~ 1,
  family = poisson(link = "log"),
  data = treat.df
)
logLik(intercept.fit)
```

```
## 'log Lik.' -50.96924 (df=1)
```

(b) `logLik(treat.fit)`

```
## 'log Lik.' -45.17455 (df=2)
```

```
deviance <- 2 * (logLik(treat.fit)[1] - logLik(intercept.fit)[1])
deviance
```

```
## [1] 11.58937
```

$D_0 - D_1 = 2(L_1 - L_0) = 2[-45.17455 - (-50.96924)] = 11.58938 \chi^2(2-1) = \chi^2(1)$

```
p_value <- pchisq(deviance, df = 1, lower.tail = FALSE)
p_value
```

```
## [1] 0.0006632976
```

```
drop1(treat.fit, test = "LRT")
```

```
## Single term deletions
```

```
##
```

```
## Model:
```

```
## imperfections ~ treatment
```

```
##           Df Deviance      AIC      LRT  Pr(>Chi)
```

```
## <none>          16.268   94.349
```

```
## treatment    1    27.857 103.938 11.589 0.0006633 ***
```

```
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

$\therefore p\text{-value} < (\alpha = .05)$, we reject the null hypothesis that $\beta_{treatment} = 0$.