HW6

Nicholas Wendt

2025-04-08

Questions: 3.3, 3.5, 3.8

Question 3.3

Part a The prediction equation is: $E(Y) = 0.00255 + 0.00109x$

Where x = mother's alcohol consumption and Y = whether a baby has sex organ malformation

The intercept is the probability of a baby having a sex organ malformation when the mother doesn't consume alcohol during pregnancy

The slope is the increase in probability of a baby having a sex organ malformation due to a 1 unit increase in alcohol consumption

Part b

(i)

```
model.0 <- 0.00255+0.00109*0
model.7 <- 0.00255+0.00109*7
print(model.0)
```

```
## [1] 0.00255
```

```
print(model.7)
```

```
## [1] 0.01018
```

(ii)

```
rr <- model.7/model.0
rr
```

```
## [1] 3.992157
```

The a baby is about 4 times more likely to have a sex organ malformation when the mother consumes 7 units of alcohol compared to a baby who's mother consumed no alcohol.

```
alcohol.consumption <- matrix(data = c(17066,48,14464,38,788,5,126,1,37,1), byrow = TRUE, nrow = 5, ncol
colnames(alcohol.consumption) <- c("Absent", "Present")
rownames(alcohol.consumption) <- c("0","<1","1-2","3-5",">5")
alcohol.consumption
```

Part c

```
##     Absent Present
## 0    17066      48
## <1   14464      38
## 1-2    788       5
## 3-5    126       1
## >5      37       1
```

```
new.scores <- c(0,1,2,3,4)
old.scores <- c(0,0.5,1.5,4,7)
alcohol1=data.frame(alcohol.consumption,old.scores, new.scores)
alcohol1
```

```
##     Absent Present old.scores new.scores
## 0    17066      48        0.0          0
## <1   14464      38        0.5          1
## 1-2    788       5        1.5          2
## 3-5    126       1        4.0          3
## >5      37       1        7.0          4
```

```
alc.model.new = glm( cbind(Present, Absent) ~ new.scores,family=binomial(link = "identity"), data=alcoho
alc.model.old <- glm( cbind(Present, Absent) ~ old.scores,family=binomial(link = "identity"), data=alcol
summary(alc.model.new)
```

```
##
## Call:
## glm(formula = cbind(Present, Absent) ~ new.scores, family = binomial(link = "identity"),
##     data = alcohol1)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) 0.0025977  0.0003797   6.842 7.84e-12 ***
## new.scores  0.0005044  0.0005276   0.956    0.339
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 6.2020  on 4  degrees of freedom
## Residual deviance: 4.9336  on 3  degrees of freedom
## AIC: 27.56
##
## Number of Fisher Scoring iterations: 9
```

```r
summary(alc.model.old)
```

```
##
## Call:
## glm(formula = cbind(Present, Absent) ~ old.scores, family = binomial(link = "identity"),
##     data = alcohol1)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) 0.0025476  0.0003523   7.232 4.77e-13 ***
## old.scores  0.0010872  0.0008324   1.306    0.192
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 6.2020  on 4  degrees of freedom
## Residual deviance: 2.9795  on 3  degrees of freedom
## AIC: 25.606
##
## Number of Fisher Scoring iterations: 10
```

```r
model.0 <- 0.0025977+0.0005044*0
model.7 <- 0.0025977+0.0005044*7
rr <- model.7/model.0
rr
```

```
## [1] 2.359202
```

```r
(fit_old=fitted(alc.model.old))
```

```
##           0          <1         1-2         3-5          >5
## 0.002547627 0.003091221 0.004178410 0.006896382 0.010157948
```

```r
(fit_new=fitted(alc.model.new))
```

```
##           0          <1         1-2         3-5          >5
## 0.002597699 0.003102096 0.003606493 0.004110889 0.004615286
```

```r
rr_old=fit_old[5]/fit_old[1]
rr_new=fit_new[5]/fit_old[1]
rr_old
```

```
##       >5
## 3.987219
```

```r
rr_new
```

```
##       >5
## 1.811602
```

```
altered.alc <- matrix(data = c(17066,48,14464,38,788,5,126,1), byrow = TRUE, nrow = 4, ncol = 2)
colnames(altered.alc) <- c("Absent", "Present")
rownames(altered.alc) <- c("0","<1","1-2","3-5")

scores <- c(0,0.5,1.5,4)
alter.w.scores <- data.frame(altered.alc, scores)
altered.model = glm( cbind(Present, Absent) ~ scores,family=binomial(link = "identity"), data=alter.w.s
summary(altered.model)
```

Part d

```
##
## Call:
## glm(formula = cbind(Present, Absent) ~ scores, family = binomial(link = "identity"),
##     data = alter.w.scores)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) 0.0026026  0.0003595   7.240 4.48e-13 ***
## scores      0.0008184  0.0008627   0.949    0.343
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 3.5131  on 3  degrees of freedom
## Residual deviance: 2.1795  on 2  degrees of freedom
## AIC: 22.833
##
## Number of Fisher Scoring iterations: 8
```

```
altered.0 <- 0.0026026 + 0.0008184*0
altered.7 <- 0.0026026 + 0.0008184*7
rr <- altered.7/altered.0
rr
```

```
## [1] 3.201183
```

The model is sensitive to this change.

Question 3.5

```
crab <- read.table("https://users.stat.ufl.edu/~aa/cat/data/Crabs.dat", header = TRUE)
crab
```

```
##     crab sat y weight width color spine
## 1      1   8 1  3.050  28.3     2     3
## 2      2   0 0  1.550  22.5     3     3
```

```
## 3      3   9 1  2.300  26.0     1     1
## 4      4   0 0  2.100  24.8     3     3
## 5      5   4 1  2.600  26.0     3     3
## 6      6   0 0  2.100  23.8     2     3
## 7      7   0 0  2.350  26.5     1     1
## 8      8   0 0  1.900  24.7     3     2
## 9      9   0 0  1.950  23.7     2     1
## 10    10   0 0  2.150  25.6     3     3
## 11    11   0 0  2.150  24.3     3     3
## 12    12   0 0  2.650  25.8     2     3
## 13    13  11 1  3.050  28.2     2     3
## 14    14   0 0  1.850  21.0     4     2
## 15    15  14 1  2.300  26.0     2     1
## 16    16   8 1  2.950  27.1     1     1
## 17    17   1 1  2.000  25.2     2     3
## 18    18   1 1  3.000  29.0     2     3
## 19    19   0 0  2.200  24.7     4     3
## 20    20   5 1  2.700  27.4     2     3
## 21    21   4 1  1.950  23.2     2     2
## 22    22   3 1  2.300  25.0     1     2
## 23    23   1 1  1.600  22.5     2     1
## 24    24   2 1  2.600  26.7     3     3
## 25    25   3 1  2.000  25.8     4     3
## 26    26   0 0  1.300  26.2     4     3
## 27    27   3 1  3.150  28.7     2     3
## 28    28   5 1  2.700  26.8     2     1
## 29    29   0 0  2.600  27.5     4     3
## 30    30   0 0  2.100  24.9     2     3
## 31    31   4 1  3.200  29.3     1     1
## 32    32   0 0  2.600  25.8     1     3
## 33    33   0 0  2.000  25.7     2     2
## 34    34   8 1  2.000  25.7     2     1
## 35    35   5 1  2.700  26.7     2     1
## 36    36   0 0  1.850  23.7     4     3
## 37    37   0 0  2.650  26.8     2     3
## 38    38   6 1  3.150  27.5     2     3
## 39    39   0 0  1.900  23.4     4     3
## 40    40   6 1  2.800  27.9     2     3
## 41    41   3 1  3.100  27.5     3     3
## 42    42   5 1  2.800  26.1     1     1
## 43    43   6 1  2.500  27.7     1     1
## 44    44   5 1  3.300  30.0     2     1
## 45    45   9 1  3.250  28.5     3     1
## 46    46   4 1  2.800  28.9     3     3
## 47    47   6 1  2.600  28.2     2     3
## 48    48   4 1  2.100  25.0     2     3
## 49    49   3 1  3.000  28.5     2     3
## 50    50   3 1  3.600  30.3     2     1
## 51    51   5 1  2.100  24.7     4     3
## 52    52   5 1  2.900  27.7     2     3
## 53    53   6 1  2.700  27.4     1     1
## 54    54   4 1  1.600  22.9     2     3
## 55    55   5 1  2.000  25.7     2     1
## 56    56  15 1  3.000  28.3     2     3
```

```
## 57    57   3 1  2.700  27.2     2     3
## 58    58   3 1  2.300  26.2     3     3
## 59    59   0 0  2.750  27.8     2     1
## 60    60   0 0  2.250  25.5     4     3
## 61    61   0 0  2.550  27.1     3     3
## 62    62   5 1  2.050  24.5     3     3
## 63    63   3 1  2.450  27.0     3     1
## 64    64   5 1  2.150  26.0     2     3
## 65    65   1 1  2.800  28.0     2     3
## 66    66   8 1  3.050  30.0     2     3
## 67    67  10 1  3.200  29.0     2     3
## 68    68   0 0  2.400  26.2     2     3
## 69    69   0 0  1.300  26.5     2     1
## 70    70   3 1  2.400  26.2     2     3
## 71    71   7 1  2.800  25.6     3     3
## 72    72   1 1  1.650  23.0     3     3
## 73    73   0 0  1.800  23.0     3     3
## 74    74   6 1  2.250  25.4     2     3
## 75    75   0 0  1.900  24.2     3     3
## 76    76   0 0  1.600  22.9     2     2
## 77    77   3 1  2.200  26.0     3     2
## 78    78   4 1  2.250  25.4     2     3
## 79    79   0 0  1.200  25.7     3     3
## 80    80   5 1  2.100  25.1     2     3
## 81    81   0 0  2.250  24.5     3     2
## 82    82   0 0  2.900  27.5     4     3
## 83    83   0 0  1.650  23.1     3     3
## 84    84   4 1  2.550  25.9     3     1
## 85    85   0 0  2.300  25.8     2     3
## 86    86   3 1  2.250  27.0     4     3
## 87    87   0 0  3.050  28.5     2     3
## 88    88   0 0  2.750  25.5     4     1
## 89    89   0 0  1.900  23.5     4     3
## 90    90   0 0  1.700  24.0     2     2
## 91    91   5 1  3.850  29.7     2     1
## 92    92   0 0  2.550  26.8     2     1
## 93    93   0 0  2.450  26.7     4     3
## 94    94   0 0  3.200  28.7     2     1
## 95    95   0 0  1.550  23.1     3     3
## 96    96   1 1  2.800  29.0     2     1
## 97    97   0 0  2.250  25.5     3     3
## 98    98   1 1  1.967  26.5     3     3
## 99    99   1 1  2.200  24.5     3     3
## 100  100   1 1  3.000  28.5     3     3
## 101  101   1 1  2.867  28.2     2     3
## 102  102   1 1  1.600  24.5     2     3
## 103  103   1 1  2.550  27.5     2     3
## 104  104   4 1  2.550  24.7     2     2
## 105  105   1 1  2.000  25.2     2     1
## 106  106   1 1  2.900  27.3     3     3
## 107  107   1 1  2.400  26.3     2     3
## 108  108   1 1  3.100  29.0     2     3
## 109  109   2 1  1.900  25.3     2     3
## 110  110   4 1  2.300  26.5     2     3
```

```
## 111  111   3 1  3.250  27.8     2     3
## 112  112   6 1  2.500  27.0     2     3
## 113  113   0 0  2.100  25.7     3     3
## 114  114   2 1  2.100  25.0     2     3
## 115  115   2 1  3.325  31.9     2     3
## 116  116   0 0  1.800  23.7     4     3
## 117  117  12 1  3.225  29.3     4     3
## 118  118   0 0  1.400  22.0     3     3
## 119  119   5 1  2.400  25.0     2     3
## 120  120   6 1  2.500  27.0     3     3
## 121  121   6 1  1.800  23.8     3     3
## 122  122   2 1  3.275  30.2     1     1
## 123  123   0 0  2.225  26.2     3     3
## 124  124   2 1  1.650  24.2     2     3
## 125  125   3 1  2.900  27.4     2     3
## 126  126   0 0  2.300  25.4     2     2
## 127  127   3 1  3.200  28.4     3     3
## 128  128   4 1  1.475  22.5     4     3
## 129  129   2 1  2.025  26.2     2     3
## 130  130   6 1  2.300  24.9     2     1
## 131  131   6 1  1.950  24.5     1     2
## 132  132   0 0  1.800  25.1     2     3
## 133  133   4 1  2.900  28.0     2     1
## 134  134  10 1  2.250  25.8     4     3
## 135  135   7 1  3.050  27.9     2     3
## 136  136   0 0  2.200  24.9     2     3
## 137  137   5 1  3.100  28.4     2     1
## 138  138   5 1  2.400  27.2     3     3
## 139  139   6 1  2.250  25.0     2     2
## 140  140   6 1  2.625  27.5     2     3
## 141  141   7 1  5.200  33.5     2     1
## 142  142   3 1  3.325  30.5     2     3
## 143  143   3 1  2.925  29.0     3     3
## 144  144   0 0  2.000  24.3     2     1
## 145  145   0 0  2.400  25.8     2     3
## 146  146   8 1  2.100  25.0     4     3
## 147  147   4 1  3.725  31.7     2     1
## 148  148   4 1  3.025  29.5     2     3
## 149  149  10 1  1.900  24.0     3     3
## 150  150   9 1  3.000  30.0     2     3
## 151  151   4 1  2.850  27.6     2     3
## 152  152   0 0  2.300  26.2     2     3
## 153  153   0 0  2.000  23.1     2     1
## 154  154   0 0  1.600  22.9     2     1
## 155  155   0 0  1.900  24.5     4     3
## 156  156   4 1  1.950  24.7     2     3
## 157  157   0 0  3.200  28.3     2     3
## 158  158   2 1  1.850  23.9     2     3
## 159  159   0 0  1.800  23.8     3     3
## 160  160   4 1  3.500  29.8     3     2
## 161  161   4 1  2.350  26.5     2     3
## 162  162   3 1  2.275  26.0     2     3
## 163  163   8 1  3.050  28.2     2     3
## 164  164   0 0  2.150  25.7     4     3
```

```
## 165  165   7 1  2.750  26.5     2     3
## 166  166   0 0  2.200  25.8     2     3
## 167  167   0 0  1.800  24.1     3     3
## 168  168   2 1  2.175  26.2     3     3
## 169  169   3 1  2.750  26.1     3     3
## 170  170   4 1  3.275  29.0     3     3
## 171  171   0 0  2.625  28.0     1     1
## 172  172   0 0  2.625  27.0     4     3
## 173  173   0 0  2.000  24.5     2     2
```

```
crab.model <- glm(y ~ weight, data = crab, family = binomial(link = "identity"), start = c(0.01, 0.001))
```

Part a

```
## Warning: step size truncated: out of bounds
## Warning: step size truncated: out of bounds
## Warning: step size truncated: out of bounds
## Warning: step size truncated: out of bounds
## Warning: step size truncated: out of bounds
## Warning: step size truncated: out of bounds
## Warning: step size truncated: out of bounds
## Warning: step size truncated: out of bounds
## Warning: step size truncated: out of bounds
## Warning: step size truncated: out of bounds
## Warning: step size truncated: out of bounds
## Warning: step size truncated: out of bounds
## Warning: step size truncated: out of bounds
## Warning: step size truncated: out of bounds
## Warning: step size truncated: out of bounds
## Warning: step size truncated: out of bounds
## Warning: step size truncated: out of bounds
## Warning: step size truncated: out of bounds
## Warning: step size truncated: out of bounds
## Warning: step size truncated: out of bounds
## Warning: step size truncated: out of bounds
## Warning: step size truncated: out of bounds
## Warning: step size truncated: out of bounds
## Warning: step size truncated: out of bounds
## Warning: step size truncated: out of bounds
```

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

```
## Warning: glm.fit: algorithm stopped at boundary value
```

```
summary(crab.model)
```

```
##
## Call:
## glm(formula = y ~ weight, family = binomial(link = "identity"),
```

```
##     data = crab, start = c(0.01, 0.001))
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  0.09783    0.06887    1.42    0.155
## weight       0.17349    0.01324   13.10   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 225.76  on 172  degrees of freedom
## Residual deviance: 216.30  on 171  degrees of freedom
## AIC: 220.3
##
## Number of Fisher Scoring iterations: 25
```

```
predict(crab.model, newdata = data.frame(weight = 5.2))
```

```
## 1
## 1
```

```
predict
```

```
## function (object, ...)
## UseMethod("predict")
## <bytecode: 0x000001e25a3a12c0>
## <environment: namespace:stats>
```

The linear model predicts that a crab with a width of 5.2 will have a satellite

```
crab.log <- glm(y ~ weight, data = crab, family = binomial)
summary(crab.log)
```

Part b

```
##
## Call:
## glm(formula = y ~ weight, family = binomial, data = crab)
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -3.6947     0.8802  -4.198 2.70e-05 ***
## weight        1.8151     0.3767   4.819 1.45e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
```

```
##     Null deviance: 225.76  on 172  degrees of freedom
## Residual deviance: 195.74  on 171  degrees of freedom
## AIC: 199.74
##
## Number of Fisher Scoring iterations: 4
```

```
predict(crab.log, newdata = data.frame(weight = 5.2), type = "response")
```

```
##         1
## 0.9968084
```

the logistic regression predicts that there is a %99.68% chance for a crab with width 5.2 to have a satellite.

Question 3.8

```
snoring <- matrix(c(24,1355,35,603,21,192,30,224), nrow = 4, ncol = 2, byrow = TRUE)
colnames(snoring) <- c("Yes","No")
rownames(snoring) <- c("Never", "Occasional", "Nearly every night", "Every night")
snoring
```

```
##                    Yes   No
## Never               24 1355
## Occasional          35  603
## Nearly every night  21  192
## Every night         30  224
```

```
scores <- c(0,2,4,5)
snoring.df <- data.frame(snoring, scores)
snoring.model <- glm(cbind(Yes,No) ~ scores, data = snoring.df, family = binomial(link = "logit"))
summary(snoring.model)
```

```
##
## Call:
## glm(formula = cbind(Yes, No) ~ scores, family = binomial(link = "logit"),
##     data = snoring.df)
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -3.86625    0.16621 -23.261  < 2e-16 ***
## scores       0.39734    0.05001   7.945 1.94e-15 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 65.9045  on 3  degrees of freedom
## Residual deviance:  2.8089  on 2  degrees of freedom
## AIC: 27.061
##
## Number of Fisher Scoring iterations: 4
```

$H_0 : \beta_1 = 0$

$H_a : \beta_1 \neq 0$

Reject H_0 since p-val $= 1.94e-15 < \alpha = 0.05$

Snoring is statistically significant when predicting heart disease

```
confint(snoring.model)
```

```
## Waiting for profiling to be done...
```

```
##                   2.5 %     97.5 %
## (Intercept) -4.2072190 -3.5544117
## scores       0.2999362  0.4963887
```

Since 0 isn't in the CI we can reject.