453 HW2

Max Tjen

2023-06-15

Packages

```r
library(car)
library(logistf)
library(dplyr)
library(tidyverse)
```

Question 13

```r
# table from textbook
bird <- data.frame(first = c ("made", "missed"),
                   success = c(251, 48),
                   trials = c(285, 53)) |>
  mutate(first = factor(first))

bird
```

```
##    first success trials
## 1   made     251    285
## 2 missed      48     53
```

```r
# create new dataframe of data
data13 <- data.frame(matrix(nrow = 338, ncol = 2))
colnames(data13) <- c("made_first", "made_second")
firstVals <- c()
secondVals <- c()

for (count in 1:338) {
  if (count <= 285) {
    if (count <= 251) {
      first <- 1
      second <- 1
    } else {
      first <- 1
      second <- 0
    }
  } else {
    if (count <= 333) {
      first <- 0
      second <- 1
    } else {
      first <- 0
      second <- 0
    }
  }
  firstVals <- append(firstVals, first)
  secondVals <- append(secondVals, second)
}

data13$made_first <- firstVals
data13$made_second <- secondVals

table(data13)
```

```
##           made_second
## made_first   0   1
##          0   5  48
##          1  34 251
```

Part B

```r
mod13 <- glm(formula = made_second ~ made_first,
             family = binomial(link = logit),
             data = data13)

mod13
```

```
##
## Call:  glm(formula = made_second ~ made_first, family = binomial(link = logit),
##     data = data13)
##
## Coefficients:
## (Intercept)   made_first
##      2.2618      -0.2627
##
## Degrees of Freedom: 337 Total (i.e. Null);  336 Residual
## Null Deviance:       241.8
## Residual Deviance: 241.5     AIC: 245.5
```

Here, we create a logistic regression model for the probability of success on the second free throw shot using the result of the first shot as the predictor variable.

Part C

```
# odds ratio
exp(coef(mod13)[2])
```

```
## made_first
##  0.7689951
```

```
# Wald interval
beta.ci <- confint.default(object = mod13, parm = "made_first", level = 0.95)
exp(beta.ci)
```

```
##                 2.5 %   97.5 %
## made_first 0.2862814 2.065637
```

```
# Profile LR interval
beta.ci <- confint(object = mod13, parm = "made_first", level = 0.95)
```

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

```
exp(beta.ci)
```

```
##     2.5 %    97.5 %
## 0.2537936 1.9072332
```

The estimated odds ratio of `made_first` is 0.769, which indicates that the odds of Larry Bird making his second free throw shot decreases if he made the first shot rather than miss. The two calculated intervals were relatively similar, with the Wald interval being (0.286, 2.066) and the profile LR interval being (0.254, 1.907).

The profile LR wasn't calculated in the Larry Bird example in section 1.2.5, but the example odds ratio is the same (0.77) as well as the Wald confidence interval of (0.29, 2.07).

Part D

```
Anova(mod13, test.statistic = "Wald")
```

```
## Analysis of Deviance Table (Type II tests)
##
## Response: made_second
##            Df  Chisq Pr(>Chisq)
## made_first  1 0.2715     0.6024
```

```
Anova(mod13, test.statistic = "LR")
```

```
## Analysis of Deviance Table (Type II tests)
##
## Response: made_second
##            LR Chisq Df Pr(>Chisq)
## made_first  0.28575  1      0.593
```

To perform a hypothesis test of $H_0 : \beta_1 = 0$ vs. $H_0 : \beta_1 \neq 0$, we used the Anova() function from the car package. We can see that with the Wald statistic, the p-value is 0.602 and the LR statistic p-value is 0.593. This means that we can't reject the null hypothesis that $\beta_1 = 0$.

In section 1.2.3, the Wald statistic p-value is 0.602 and the LR statistic p-value is 0.593. Both of these p-values are the same as those that we got.

Part E

The similarity in calculated values in this section and the previous example is likely because there is only one binary predictor variable. With this, there is very little room for variation because there are few possible outcomes with the outcome variable also being binary. As such, the little variation probably leads to similar calculations.

Question 14

```
# table from textbook
table <- array (data = c(57, 142, 200688, 201087),
                dim = c(2, 2),
                dimnames = list(Treatment = c("vaccine", "placebo"),
                                Result = c("polio", "polio free")))

table
```

```
##          Result
## Treatment polio polio free
##   vaccine    57     200688
##   placebo   142     201087
```

```
# create new dataframe of data
polio <- data.frame(matrix(nrow = 401974, ncol = 2))
colnames(polio) <- c("got_vaccine", "polio_free")

polio <- polio |>
  mutate(got_vaccine = 1,
         polio_free = 1)

polio$got_vaccine[200746:401974] <- 0
polio$polio_free[1:57] <- 0
polio$polio_free[200746:200887] <- 0

table(polio)
```

```
##            polio_free
## got_vaccine      0      1
##           0    142 201087
##           1     57 200688
```

Part C

```
mod14 <- glm(formula = polio_free ~ got_vaccine,
             family = binomial(link = logit),
             data = polio)

mod14
```

```
##
## Call:  glm(formula = polio_free ~ got_vaccine, family = binomial(link = logit),
##     data = polio)
##
## Coefficients:
## (Intercept)  got_vaccine
##      7.2557       0.9108
##
## Degrees of Freedom: 401973 Total (i.e. Null);  401972 Residual
## Null Deviance:       3427
## Residual Deviance: 3390  AIC: 3394
```

Here, we create a logistic regression model for the probability of being polio free where the predictor variable is got_vaccine. For got_vaccine, a value of 1 means that the patient received a vaccine and a value of 0 means that they received the placebo.

Part D

```
# odds ratio
exp(coef(mod14)[2])
```

```
## got_vaccine
##    2.486285
```

```
# Wald interval
beta.ci <- confint.default(object = mod14, parm = "got_vaccine", level = 0.95)
exp(beta.ci)
```

```
##                2.5 %   97.5 %
## got_vaccine 1.828335 3.381006
```

```
# Profile LR interval
beta.ci <- confint(object = mod14, parm = "got_vaccine", level = 0.95)
```

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

```
exp(beta.ci)
```

```
##    2.5 %   97.5 %
## 1.839526 3.406080
```

The estimated odds ratio of `got_vaccine` is 2.486, which indicates that the odds of a patient being polio free increases if they are given the vaccine rather than the placebo. The two calculated intervals were relatively similar, with the Wald interval being (1.828, 3.381) and the profile LR interval being (1.840, 3.406).

The profile LR wasn't calculated in the example in section 1.2.5, but the example odds ratio is the same (2.49) as well as the Wald confidence interval of (1.83, 3.38).

Part E

```r
Anova(mod14, test.statistic = "Wald")
```

```
## Analysis of Deviance Table (Type II tests)
##
## Response: polio_free
##             Df  Chisq Pr(>Chisq)
## got_vaccine  1 33.726  6.343e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
Anova(mod14, test.statistic = "LR")
```

```
## Analysis of Deviance Table (Type II tests)
##
## Response: polio_free
##             LR Chisq Df Pr(>Chisq)
## got_vaccine   37.313  1  1.006e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

To see if there's a difference we ran an ANOVA test to perform a hypothesis test of $H_0 : \beta_1 = 0$ vs. $H_0 : \beta_1 \neq 0$. The p-values when using both the Wald and LR statistic are very small, meaning that we can reject the null hypothesis and determine that there is a difference in the probability of being polio free based on treatment.

Part F

A logistic regression model would be an easier way to assess the effectiveness than a contingency table if more explanatory variables are added because of the coefficients calculated. With these, we can get the odds ratio for treatment while factoring in the effects of the other explanatory variables. This is difficult to do with a contingency table because you only get the raw count values and with more than one explanatory variable, it's hard to interpret.

Question 19

```r
healthcare <- read_csv("/Users/mtjen/Desktop/453/hw2/healthcare_worker.csv")

# create new dataframe of data
health <- data.frame(matrix(nrow = 10654, ncol = 2))
```

```
colnames(health) <- c("group", "has_hepatitis")

health <- health |>
  mutate(group = "Exposure prone",
         has_hepatitis = 0)

# change values
health$group[2206:8412] <- "Fluid contact"
health$group[8413:8945] <- "Lab Staff"
health$group[8946:10183] <- "Patient contact"
health$group[10184:nrow(health)] <- "No patient contact"

health$has_hepatitis[1:5] <- 1
health$has_hepatitis[2206:2222] <- 1
health$has_hepatitis[8414:8416] <- 1
health$has_hepatitis[8948:8949] <- 1
health$has_hepatitis[10187:10189] <- 1

# check distribution
table(health)
```

```
##                      has_hepatitis
## group                     0    1
##   Exposure prone       2200    5
##   Fluid contact        6190   17
##   Lab Staff             530    3
##   No patient contact    468    3
##   Patient contact      1236    2
```

```
# build model
mod19 <- glm(formula = has_hepatitis ~ group,
             family = binomial(link = logit),
             data = health)

# anova
Anova(mod19, test.statistic = "Wald")
```

```
## Analysis of Deviance Table (Type II tests)
##
## Response: has_hepatitis
##       Df  Chisq Pr(>Chisq)
## group  4 4.1987     0.3798
```

```
Anova(mod19, test.statistic = "LR")
```

```
## Analysis of Deviance Table (Type II tests)
##
## Response: has_hepatitis
##       LR Chisq Df Pr(>Chisq)
## group    3.735  4     0.4431
```

By creating a logistic regression model and then running an Anova test on it to get p-values using both the Wald and LR statistics, we can determine that there isn’t sufficient evidence that occupational group effects hepatitis status. This may be a preferable result as it shows that hepatitis equally impacts different occupational groups rather than being concentrated in one or a few.

Question 24

Part A

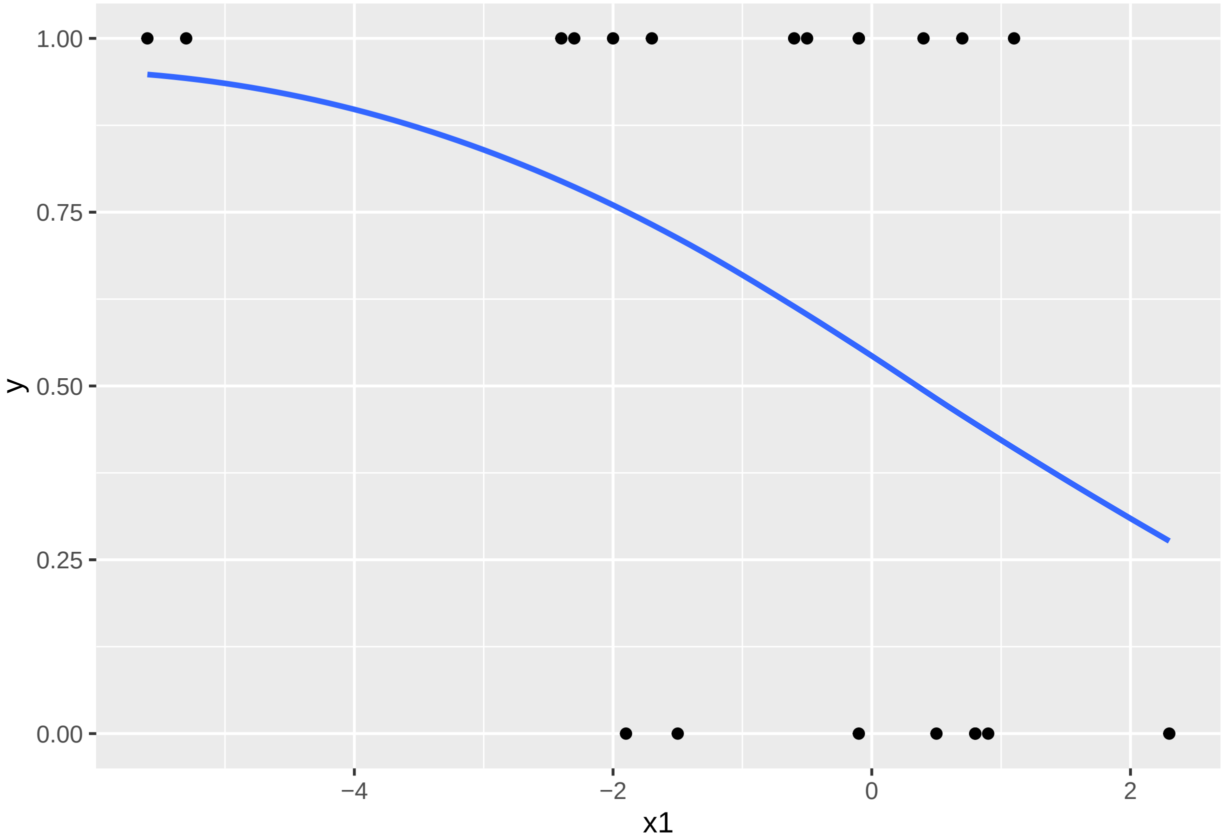
```r
incontinence <- read_csv("/Users/mtjen/Desktop/453/hw2/incontinence.csv")

# x1 variable
mod24a1 <- logistf(formula = y ~ x1,
                   data = incontinence)

## get coefficient values
b0 <- mod24a1$coefficients[1]
b1 <- mod24a1$coefficients[2]

## get predicted values
preds = plogis(b0 + b1 * incontinence$x1)
predDf <- data.frame(x1 = incontinence$x1,
                     y = preds)

## plot observed values with modeled overlay
ggplot(incontinence, aes(x = x1, y = y)) +
  geom_point() +
  stat_smooth(data = predDf, se = FALSE)
```



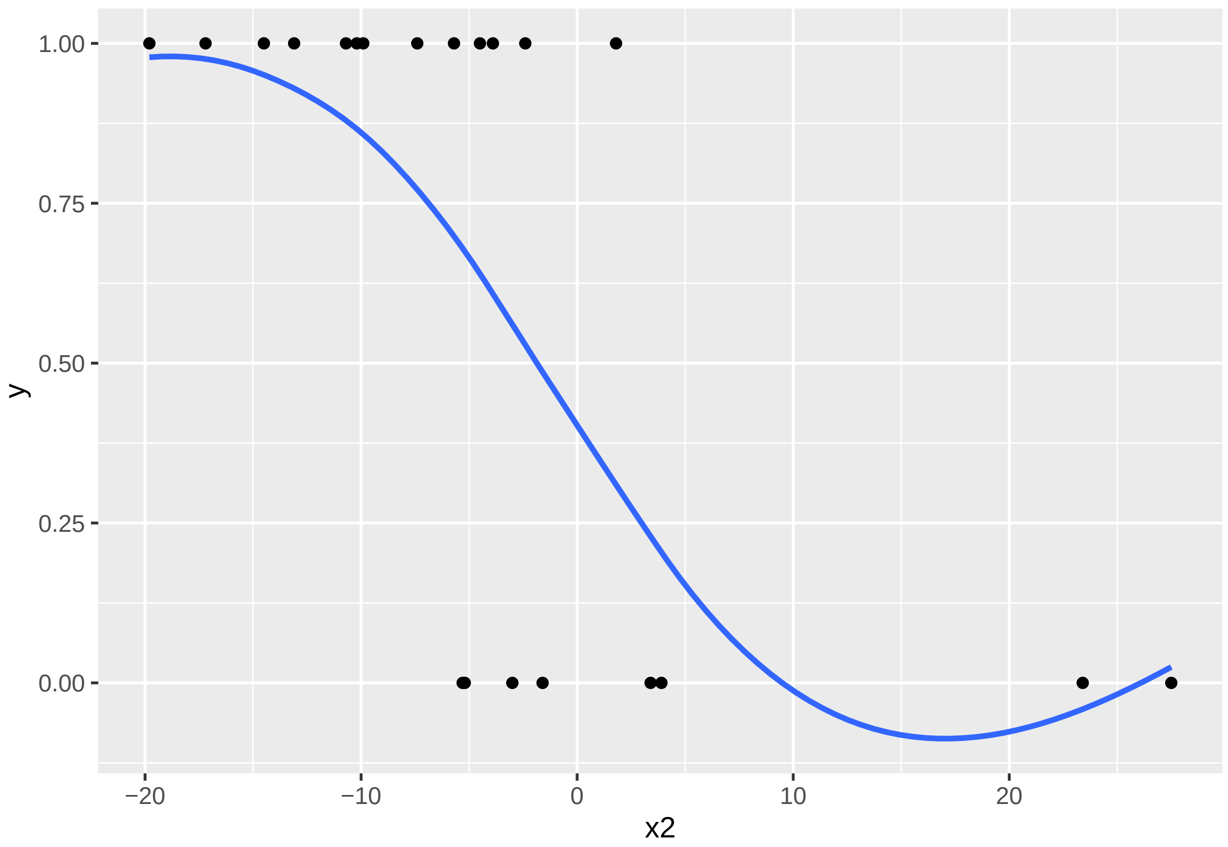
```
# x2 variable
mod24a2 <- logistf(formula = y ~ x2,
                   data = incontinence)

b0 <- mod24a2$coefficients[1]
b1 <- mod24a2$coefficients[2]

preds = plogis(b0 + b1 * incontinence$x2)

predDf <- data.frame(x2 = incontinence$x2,
                     y = preds)

ggplot(incontinence, aes(x = x2, y = y)) +
  geom_point() +
  stat_smooth(data = predDf, se = FALSE)
```



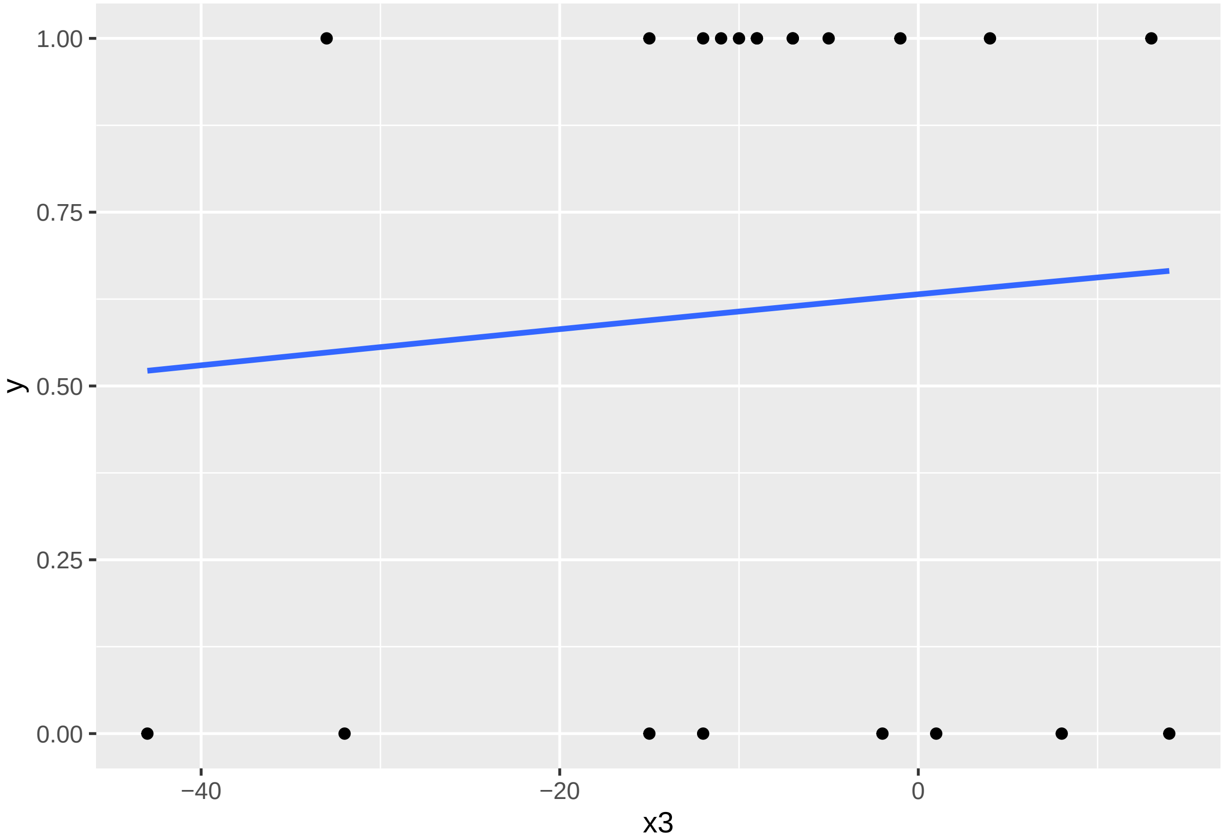
```
# x3 variable
mod24a3 <- logistf(formula = y ~ x3,
                   data = incontinence)

b0 <- mod24a3$coefficients[1]
b1 <- mod24a3$coefficients[2]

preds = plogis(b0 + b1 * incontinence$x3)

predDf <- data.frame(x3 = incontinence$x3,
                     y = preds)

ggplot(incontinence, aes(x = x3, y = y)) +
  geom_point() +
  stat_smooth(data = predDf, se = FALSE)
```



After looking at the plots, there doesn't appear to be any issues with complete separation.

Part B

```
mod24b <- logistf(formula = y ~ x1 + x2 + x3,
                   data = incontinence)
```

```
## Warning in logistpl(x, y, beta, i, LL.0, firth, -1, offset, weight, plcontrol,
## : fitted probabilities numerically 0 or 1 occurred for variable x2
```

```
## Warning in logistpl(x, y, beta, i, LL.0, firth, 1, offset, weight, plcontrol, :
## fitted probabilities numerically 0 or 1 occurred for variable x3
```

```
## Warning in logistf(formula = y ~ x1 + x2 + x3, data = incontinence):
## Nonconverged PL confidence limits: maximum number of iterations for variables:
## x2, x3 exceeded. Try to increase the number of iterations by passing
## 'logistpl.control(maxit=...)' to parameter plcontrol
```

```
b0 <- mod24b$coefficients[1]
b1 <- mod24b$coefficients[2]
b2 <- mod24b$coefficients[3]
b3 <- mod24b$coefficients[4]
```

```
preds = plogis(b0 + b1 * incontinence$x1 +
                 b2 * incontinence$x2 +
                 b3 * incontinence$x3)

preds
```

```
##  [1] 0.2715881520 0.1991366440 0.2014987573 0.0002055636 0.3240669633
##  [6] 0.4674562437 0.1738438229 0.0003295809 0.9999050409 0.9995944059
## [11] 0.6719904187 0.9938558707 0.9563363357 0.9875460461 0.6887687683
## [16] 0.9852424391 0.9600110238 0.9195336333 0.9905624499 0.8935177051
## [21] 0.3072666738
```

The predicted values seem to be clumped into two groups - with one being around 0.2 and the other being in the 0.9s. This leads us to believe that there may be an issue with coomplete separation, particularly with the high concentration oof values close to 1.