

# Day8 exercise solutions

Ali Movasati

Oct. 4th, 2024

```
# Set global code chunk options
knitr::opts_chunk$set(warning = FALSE)

# load required libraries
library("skimr")
library("dplyr")
library("magrittr")
library("ggplot2")

# define functions
`%notin%` <- Negate(`%in%`)
```

## Problem 1

```
data(bliss, package = "faraway")
```

### 1.A)

```
skim(bliss)
```

Table 1: Data summary

Name	bliss
Number of rows	5
Number of columns	3
Column type frequency:	
numeric	3
Group variables	None

### Variable type: numeric

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
dead	0	1	15	10.32	2	8	15	23	27	
alive	0	1	15	10.32	3	7	15	22	28	
conc	0	1	2	1.58	0	1	2	3	4	

```
head(bliss)
```

```
##   dead alive conc
## 1    2    28    0
## 2    8    22    1
## 3   15    15    2
## 4   23     7    3
## 5   27     3    4
```

## 1.B)

```
bliss %<>% mutate(ratio = dead/(alive + dead))
```

```
logit_model <- glm(cbind(dead,alive)~conc, family = binomial(link=logit), data = bliss)
```

```
summary(logit_model)
```

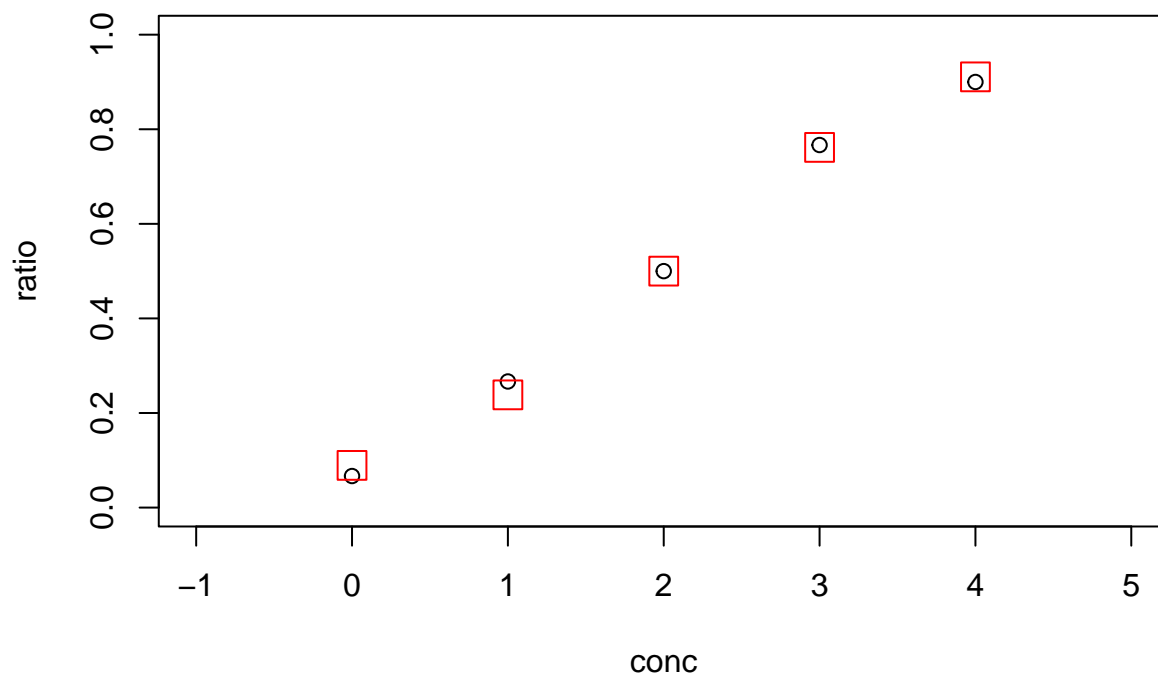
```
##
## Call:
## glm(formula = cbind(dead, alive) ~ conc, family = binomial(link = logit),
##      data = bliss)
##
## Deviance Residuals:
##      1       2       3       4       5
## -0.4510  0.3597  0.0000  0.0643 -0.2045
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -2.3238      0.4179  -5.561 2.69e-08 ***
## conc           1.1619      0.1814   6.405 1.51e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 64.76327  on 4  degrees of freedom
## Residual deviance:  0.37875  on 3  degrees of freedom
## AIC: 20.854
##
## Number of Fisher Scoring iterations: 4
```

## 1.C)

```
logit_model_out <- predict(logit_model, newdata=data.frame(conc=seq(-1,to=5,length.out=100)), type="response")
```

```
plot(ratio~conc, xlim = c(-1,5), ylim = c(0,1), data=bliss)
```

```
points(bliss$conc, logit_model$fitted.values, pch=0, col="red", cex=2)
```

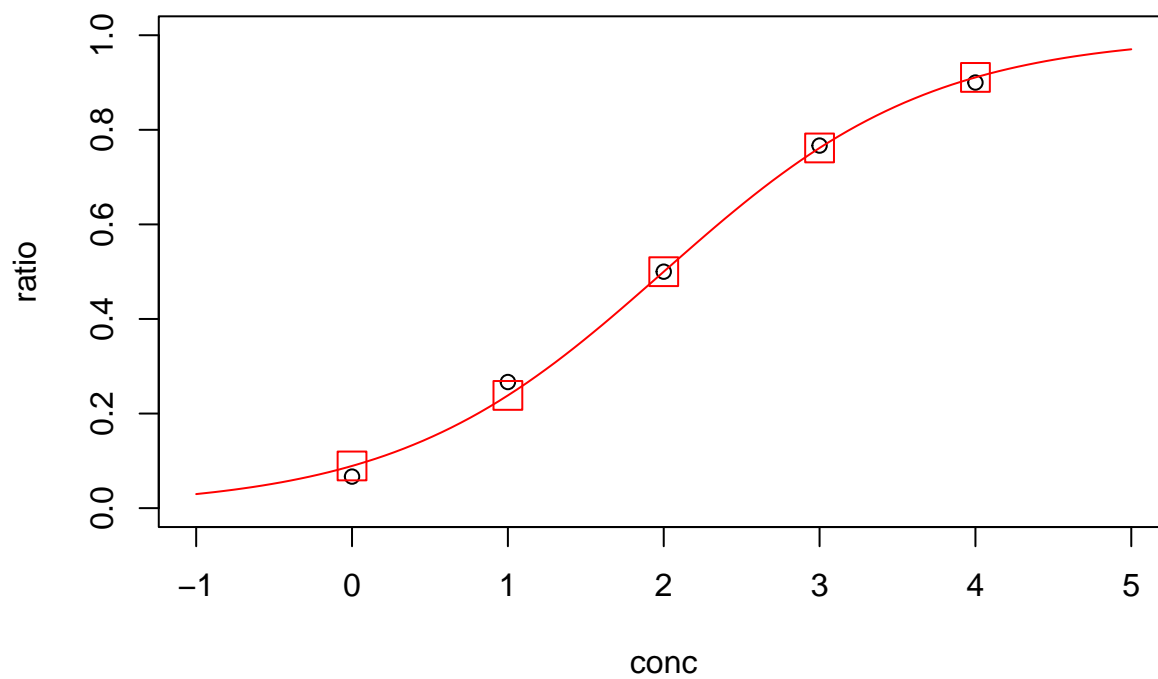


1.D)

```
plot(ratio~conc, xlim = c(-1,5), ylim = c(0,1), data=bliss)
```

```
points(bliss$conc, logit_model$fitted.values, pch=0, col="red", cex=2)
```

```
lines(seq(-1,to=5,length.out=100),logit_model_out, col = "red")
```



1.E)

```
logit_model_out2 <- predict(logit_model, newdata=data.frame(conc=seq(-1,to=5,length.out=100)), type="link")

new_data <- data.frame(conc = seq(-1,to=5,length.out=100))

new_data$fit <- logit_model_out2$fit
new_data$lower <- logit_model_out2$fit - qnorm(0.975) * logit_model_out2$se.fit
new_data$upper <- logit_model_out2$fit + qnorm(0.975) * logit_model_out2$se.fit

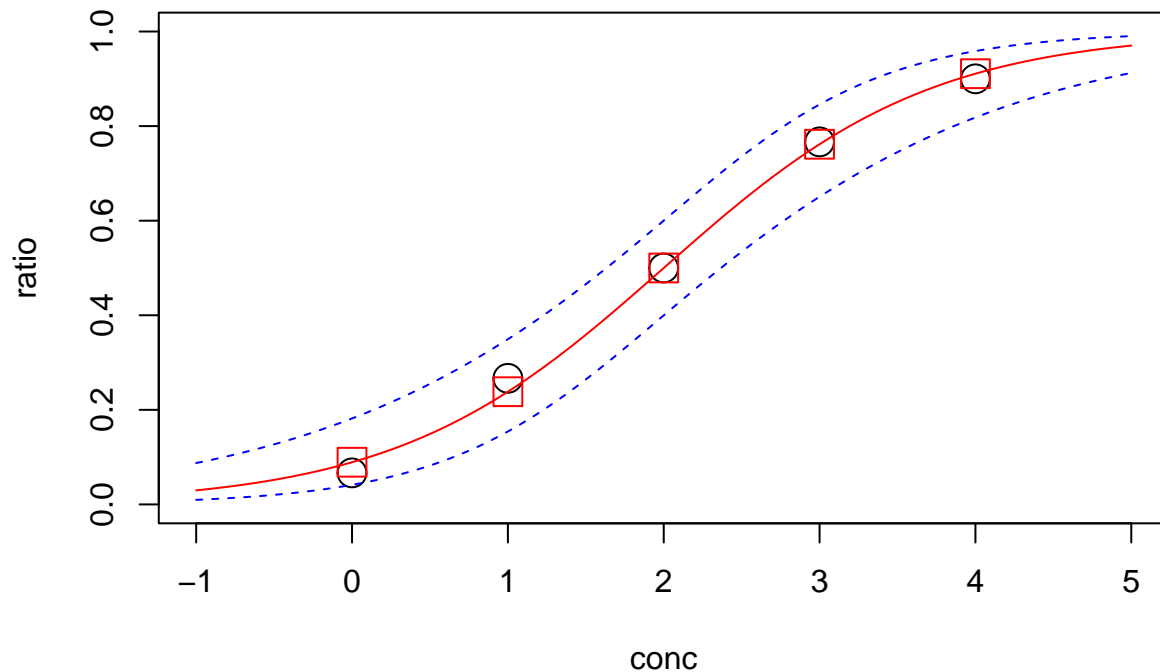
new_data$fit_prob <- 1 / (1 + exp(-new_data$fit))
new_data$upper_prob <- 1 / (1 + exp(-new_data$upper))
new_data$lower_prob <- 1 / (1 + exp(-new_data$lower))

# plot

plot(ratio~conc, xlim = c(-1,5), ylim = c(0,1), cex=2, data=bliss)

points(bliss$conc, logit_model$fitted.values, pch=0, col="red", cex=2)

lines(new_data$conc,new_data$fit_prob, col = "red")
lines(new_data$conc,new_data$lower_prob, col = "blue", lty = "dashed")
lines(new_data$conc,new_data$upper_prob, col = "blue", lty = "dashed")
```



1.F)

```
probit_model <- glm(cbind(dead, alive)~conc, family = binomial(link=probit), data = bliss)
```

```
summary(probit_model)
```

```
##
## Call:
## glm(formula = cbind(dead, alive) ~ conc, family = binomial(link = probit),
##      data = bliss)
##
## Deviance Residuals:
##      1      2      3      4      5
## -0.35863  0.27493  0.01893  0.18230 -0.27545
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -1.37709    0.22781  -6.045 1.49e-09 ***
## conc         0.68638    0.09677   7.093 1.31e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 64.76327  on 4  degrees of freedom
## Residual deviance:  0.31367  on 3  degrees of freedom
## AIC: 20.789
##
## Number of Fisher Scoring iterations: 4
```

```
probit_model_out <- predict(probit_model, newdata=data.frame(conc=seq(-1,to=5,length.out=100)), type="r
```

```
plot(ratio~conc, xlim = c(-1,5), ylim = c(0,1), data=bliss)
```

```
points(bliss$conc, probit_model$fitted.values, pch=0, col="blue", cex=2)
```

```
lines(seq(-1,to=5,length.out=100),probit_model_out, col = "blue")
```

