Lec19_1

R Markdown

```
aba <- read.csv("Abalone.csv")
head(aba)
```

```
##   Sex Length Diameter Height  Whole Shucked Viscera Shell Rings
## 1   1  0.455    0.365  0.095 0.5140  0.2245  0.1010 0.150    15
## 2   1  0.350    0.265  0.090 0.2255  0.0995  0.0485 0.070     7
## 3   2  0.530    0.420  0.135 0.6770  0.2565  0.1415 0.210     9
## 4   1  0.440    0.365  0.125 0.5160  0.2155  0.1140 0.155    10
## 5   0  0.330    0.255  0.080 0.2050  0.0895  0.0395 0.055     7
## 6   0  0.425    0.300  0.095 0.3515  0.1410  0.0775 0.120     8
```

```
library(dplyr)
library(mcprofile)
```

a.

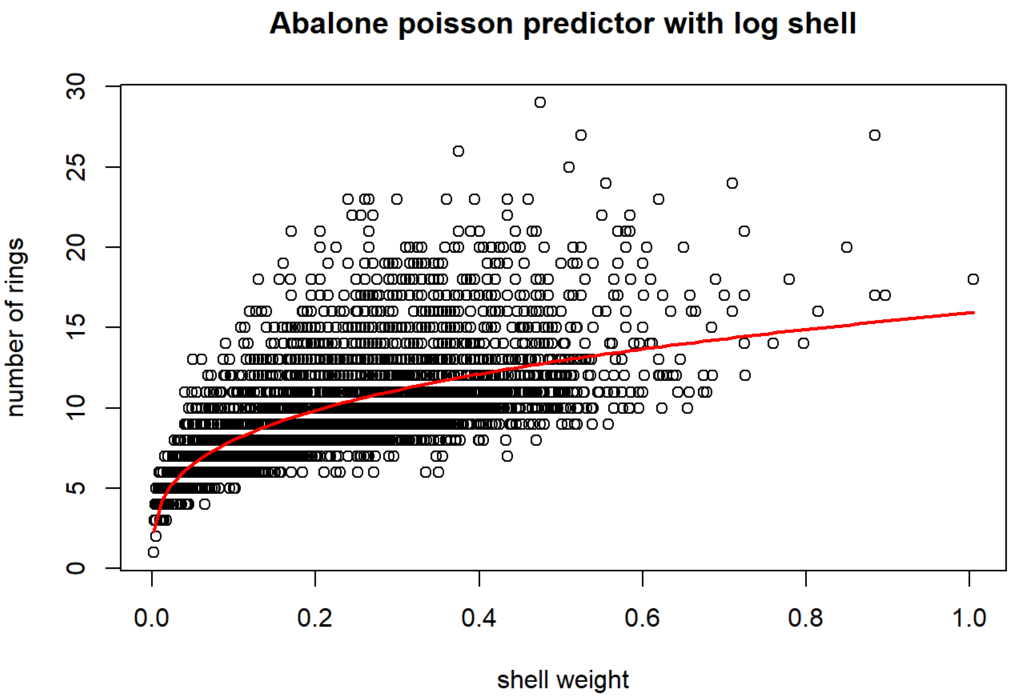
```
x <- c(0.1, 0.2, 0.3, 0.4)
shell.fit <- glm(Rings ~ log(Shell), family = poisson(link = "log"), data = aba)
K.means <- cbind(1,log(x))
meanpro <- mcprofile(shell.fit, CM=K.means)
CI.est <- exp(confint(meanpro))
CI.est$estimate %>% unlist() -> s
CI.est$confint %>% unlist() -> ss
Means <- data.frame(Rings = s, lower = ss[1:4], ipper = ss[5:8])
Means <- `rownames<-`(Means, c("Weight 0.1", "Weight 0.2", "Weight 0.3", "Weight 0.4"))
K.means
```

```
##      [,1]       [,2]
## [1,]    1 -2.3025851
## [2,]    1 -1.6094379
## [3,]    1 -1.2039728
## [4,]    1 -0.9162907
```

```
Means
```

```
##                Rings     lower     ipper
## Weight 0.1  8.020632  7.881182  8.161429
## Weight 0.2  9.858468  9.744843  9.972957
## Weight 0.3 11.123016 10.984797 11.262369
## Weight 0.4 12.117423 11.936338 12.300537
```

```
plot(x = aba$Shell, y = aba$Rings, xlab = "shell weight", ylab = "number of rings", main = "Abalone poisson
predictor with log shell")
curve(expr = exp(shell.fit$coefficients[1] + log(x)*shell.fit$coefficients[2]), lwd = 2, add = TRUE, col = "
red")
```



Comparing to figure 5, 2 graphs are indentical.

b.

```
##log mean difference of a c-unit change is
##beta_1*(log(x+c)-log(x))
x1 <- c(0.1, 0.2, 0.3)
c <- 0.1
change <- (log(x1+c)-log(x1))
K.ratio <- cbind(0, change)
mean.ratio <- mcprofile(shell.fit, CM = K.ratio)
CI.change <- exp(confint(mean.ratio))

##convert to percentage change
d <- unlist(CI.change$estimate) ##work on vectors so math manipulations are valid
f <- unlist(CI.change$confint)
100*(d-1) -> i ##Estiamte in percentage
100*(f-1) -> ii  ##CIs in percentage
percent.change <- data.frame(estimate = i, lower = c(ii[1:3]), upper = c(ii[4:6]))
percent.change <- `row.names<-`(percent.change, c("change from 0.1 to 0.2", "change from 0.2 to 0.3", "chang
e from 0.3 to 0.4"))
K.ratio
```

```
##            change
## [1,] 0 0.6931472
## [2,] 0 0.4054651
## [3,] 0 0.2876821
```

```
percent.change
```

```
##                         estimate     lower     upper
## change from 0.1 to 0.2 22.913855 21.735715 24.109021
## change from 0.2 to 0.3 12.827024 12.193148 13.467488
## change from 0.3 to 0.4  8.940085  8.505481  9.378486
```

```
##check against part a
ests <- unlist(CI.est$estimate)
##convert to percentage change
m <- data.frame(manual = c((ests[2]-ests[1])/ests[1], (ests[3]-ests[2])/ests[2], (ests[4]-ests[3])/ests[3]))
cbind(m, percent.change)
```

```
##                manual  estimate     lower     upper
## Estimate2 0.22913855 22.913855 21.735715 24.109021
## Estimate3 0.12827024 12.827024 12.193148 13.467488
## Estimate4 0.08940085  8.940085  8.505481  9.378486
```

From the comparison we can see percentage changes from mcprofile and rate of changes from manual calculation are identical.