Data Science - Regression Models - Quiz 4 - Coursera

Quiz 4

This is Quiz 4 from Coursera's Regression Models class within the Data Science Specialization. This publication is intended as a learning resource, all answers are documented and explained. Datasets are available in R packages.

---

1. Consider the space shuttle data `?shuttle` in the `MASS` library. Consider modeling the use of the autolander as the outcome (variable name `use`). Fit a logistic regression model with autolander (variable auto) use (labeled as "auto" 1) versus not (0) as predicted by wind sign (variable wind). Give the estimated odds ratio for autolander use comparing head winds, labeled as "head" in the variable headwind (numerator) to tail winds (denominator).

- **Answer:** 0.969

Explanation:

Fitting the model using a binomial distribution gives a beta coefficient of .031.

```
library(MASS)
data(shuttle)
head(shuttle)
##   stability error sign wind   magn vis  use
## 1     xstab    LX   pp head  Light  no auto
## 2     xstab    LX   pp head Medium  no auto
## 3     xstab    LX   pp head Strong  no auto
## 4     xstab    LX   pp tail  Light  no auto
## 5     xstab    LX   pp tail Medium  no auto
## 6     xstab    LX   pp tail Strong  no auto
#Checking out the data
unique(shuttle$use)
## [1] auto   noauto
## Levels: auto noauto
unique(shuttle$wind)
## [1] head tail
## Levels: head tail
```

```
#Creating 0,1 variable for auto/noauto factor
shuttle$use <- as.numeric(shuttle$use == "auto")


#generating model
mdl <- glm(factor(use)~factor(wind)-1,binomial,data = shuttle)


exp(mdl$coef[1])/exp(mdl$coef[2])
## factor(wind)head
##        0.9686888
```

---

2. Consider the previous problem. Give the estimated odds ratio for autolander use comparing head winds (numerator) to tail winds (denominator) adjusting for wind strength from the variable magn.

- **0.969**

Explanation:

The unadjusted beta values are higher. Weight is confounding significantly.

```
#Checking out the factor levels
unique(shuttle$magn)
## [1] Light  Medium Strong Out
## Levels: Light Medium Out Strong
mdl2 <- glm(factor(use)~factor(wind)+factor(magn)-1,binomial,data = shuttle)
summary(mdl2)
##
## Call:
## glm(formula = factor(use) ~ factor(wind) + factor(magn) - 1,
##     family = binomial, data = shuttle)
##
## Deviance Residuals:
##    Min      1Q  Median      3Q     Max
## -1.349  -1.321   1.015   1.040   1.184
##
## Coefficients:
##                        Estimate Std. Error z value Pr(>|z|)
```

```
## factor(wind)head    3.635e-01  2.841e-01   1.280    0.201
## factor(wind)tail    3.955e-01  2.844e-01   1.391    0.164
## factor(magn)Medium -1.010e-15  3.599e-01   0.000    1.000
## factor(magn)Out    -3.795e-01  3.568e-01  -1.064    0.287
## factor(magn)Strong -6.441e-02  3.590e-01  -0.179    0.858
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 354.89  on 256  degrees of freedom
## Residual deviance: 348.78  on 251  degrees of freedom
## AIC: 358.78
##
## Number of Fisher Scoring iterations: 4
exp(mdl2$coef[1])/exp(mdl2$coef[2])
## factor(wind)head
##        0.9684981
```

---

3. If you fit a logistic regression model to a binary variable, for example use of the autolander, then fit a logistic regression model for one minus the outcome (not using the autolander) what happens to the coefficients?

- **The coefficients reverse their signs.**

Explanation:

The sign of the ceofficient flips. One minus a binary variable flips zeros with 1 and vice versa.

```
mdl3 <- glm(1-use~factor(wind)-1,binomial,data = shuttle)
mdl3$coef
## factor(wind)head factor(wind)tail
##       -0.2513144       -0.2831263
mdl$coef
## factor(wind)head factor(wind)tail
##        0.2513144        0.2831263
```

---

4. Consider the insect spray data `InsectSprays`. Fit a Poisson model using spray as a factor level. Report the estimated relative rate comapring spray A (numerator) to spray B (denominator).

- **0.9457**

Explanation:

Mtcars reports the weight in units of 1000 lbs. Using I(wt*.5) doubles the weight coefficient from the previous model. This reflects a 2000 lbs (1 ton) increase holding the factor variable fixed.

```
data("InsectSprays")
mdl4 <- glm(count~spray-1,poisson,data = InsectSprays)
exp(mdl4$coef[1])/exp(mdl4$coef[2])
##    sprayA
## 0.9456522
```

---

5. Consider a Poisson glm with an offset, t. So, for example, a model of the form `glm(count ~ x + offset(t), family = poisson)` where `x` is a factor variable comparing a treatment (1) to a control (0) and `t` is the natural log of a monitoring time. What is impact of the coefficient for `x` if we fit the model `glm(count ~ x + offset(t2), family = poisson)` where 2 <- `log`(10) + `t`? In other words, what happens to the coefficients if we change the units of the offset variable. (Note, adding log(10) on the log scale is multiplying by 10 on the original scale.)

- **The coefficient estimate is unchanged**

Explanation:

Coefficient stays because poisson regression is modeling odds so the multiplicative offset will cancel out.

```
mdl5 <- glm(count~spray,poisson,offset = log(count+1),data = InsectSprays)
mdl6 <- glm(count~spray,poisson,offset = log(10)+log(count+1),data = InsectSprays)
mdl6$coef
##  (Intercept)       sprayB       sprayC       sprayD       sprayE
## -2.369276467  0.003512473 -0.325350713 -0.118451059 -0.184623054
##       sprayF
##  0.008422466
mdl5$coef
##  (Intercept)       sprayB       sprayC       sprayD       sprayE
```

```
## -0.066691374  0.003512473 -0.325350713 -0.118451059 -0.184623054
##       sprayF
##  0.008422466
```

---

6. Consider the data

```
x <- -5:5
y <- c(5.12, 3.93, 2.67, 1.87, 0.52, 0.08, 0.93, 2.05, 2.54, 3.87, 4.97)
```

Using a knot point at 0, fit a linear model that looks like a hockey stick with two lines meeting at x=0. Include an intercept term, x and the knot point term. What is the estimated slope of the line after 0?

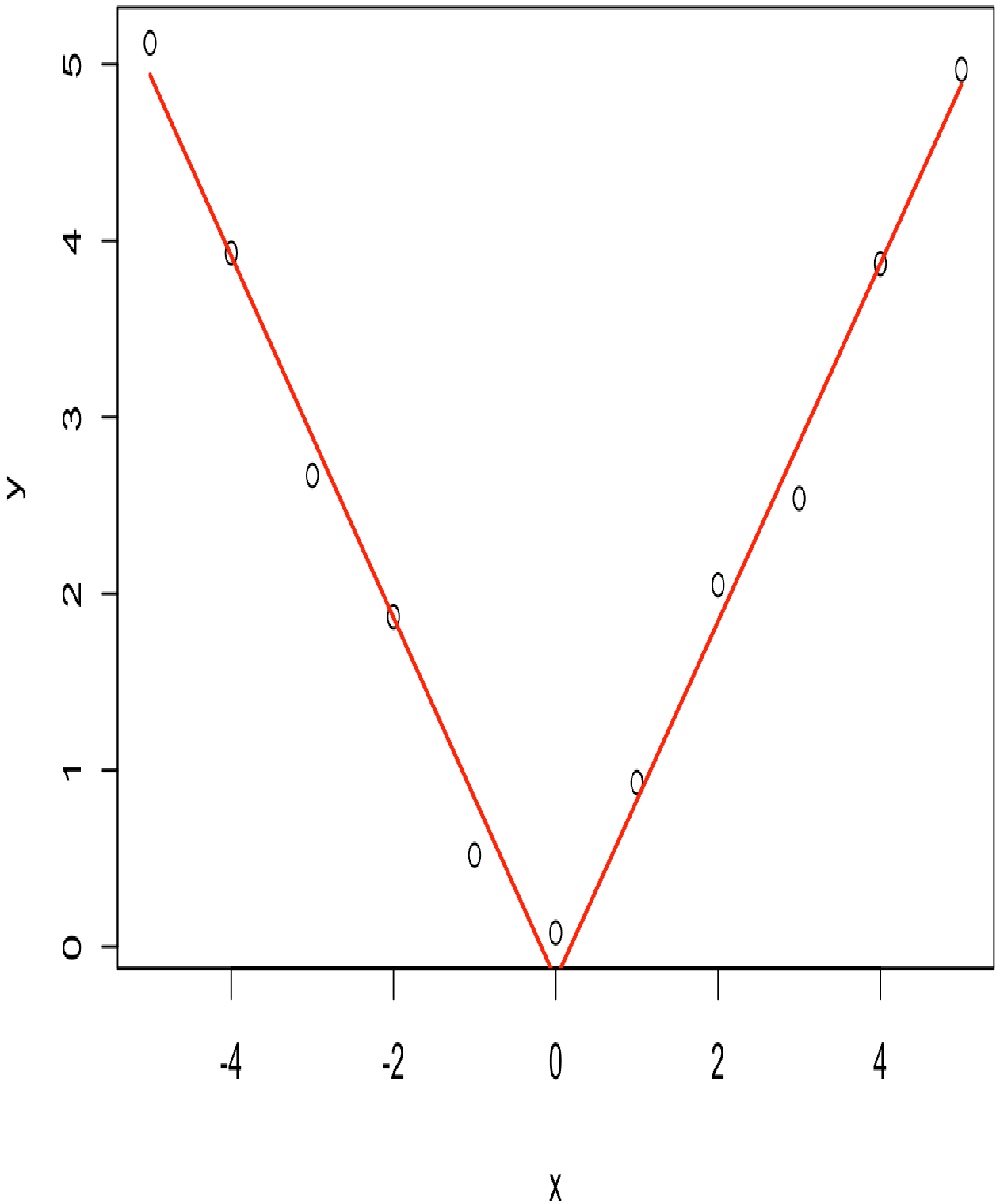
- **1.013**

Explanation:

To give the coefficients R automatically subtracted the mean slope of the first line from that of the second, so we can simply add it back to get the true value.

```
x <- -5:5
y <- c(5.12, 3.93, 2.67, 1.87, 0.52, 0.08, 0.93, 2.05, 2.54, 3.87, 4.97)


k<-c(0)
split<-sapply(k,function(k) (x>k)*(x-k))
xmat<-cbind(1,x,split)
mdl7 <- lm(y~xmat-1)
yhat<-predict(mdl7)
mdl7$coef
##       xmat      xmatx       xmat
## -0.1825806 -1.0241584  2.0372258
mdl7$coef[3]+mdl7$coef[2]
##     xmat
## 1.013067
plot(x,y)
lines(x,yhat, col= "red", lwd =2)
```



5
4
3
2
1
0
y
-4
-2
0
2
4
x