Math 150 - Methods in Biostatistics - Homework 4

Ethan Ashby

Due: Friday, February 26, 2021

Assignment Summary (Goals)

- fluent use of the logistic model for prediction and for coefficient interpretation
- practice using `ggplot()` so that visualizations can inform the larger analysis

Note that if you don't know the R code either check my notes or ask me!!! Happy to scaffold, debug, send resources, etc. Don't go down a rabbit hole trying to figure out an R function or syntax.

Also, note that you'll need to get the data from Sakai and use it for this analysis. Look back to your own HW1 file to see the line of code **you** used to import the `games1.csv` dataset. Ask me if it isn't obvious to you after you look at your own HW1.

Q1. PodQ Describe one thing you learned from someone in your pod this week (it could be: content, logistical help, background material, R information, etc.) 1-3 sentences.

Annie explained that the inflection point is dictated by $-\beta_0/\beta_1$ #### Q2. Chp 7, A1

Based on the description of the Challenger disaster O-ring concerns, identify which variable in the `Shuttle` data set in Table 7.1 should be the explanatory variable and which should be the response variable.

The explanatory variable is temperature (continuous) and the response variable is successful launch (binary).

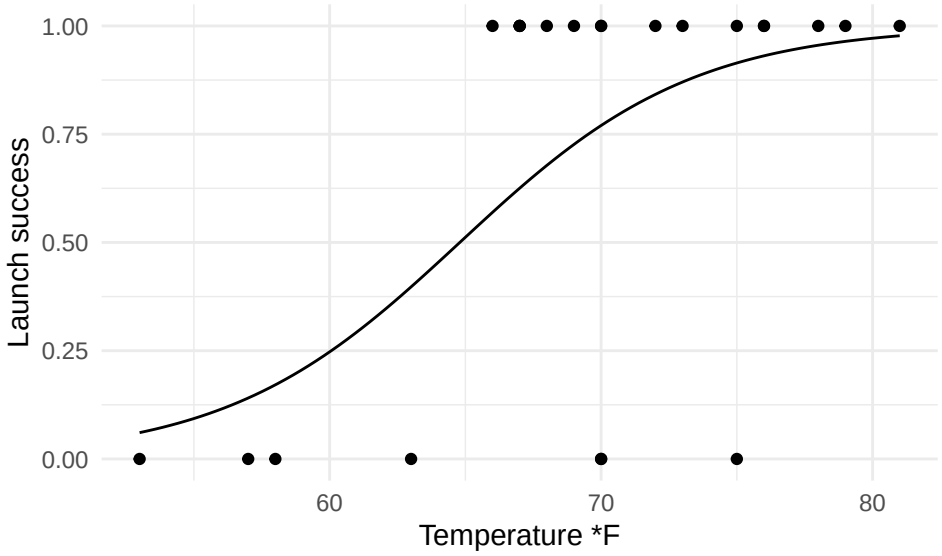
Q3. Chp 7, A2 Imagine you were an engineer working for Thiokol Corporation prior to January 1986. Create a few graphs of the data in Table 7.1. Is it obvious that temperature is related to the success of the O-rings? Submit any charts or graphs you have created that show a potential relationship between temperature and O-ring damage.

note: the data is coded with missing values represented by `*`. You may need to account for that. See how I did it below. Again, **ask** me if you are having trouble!

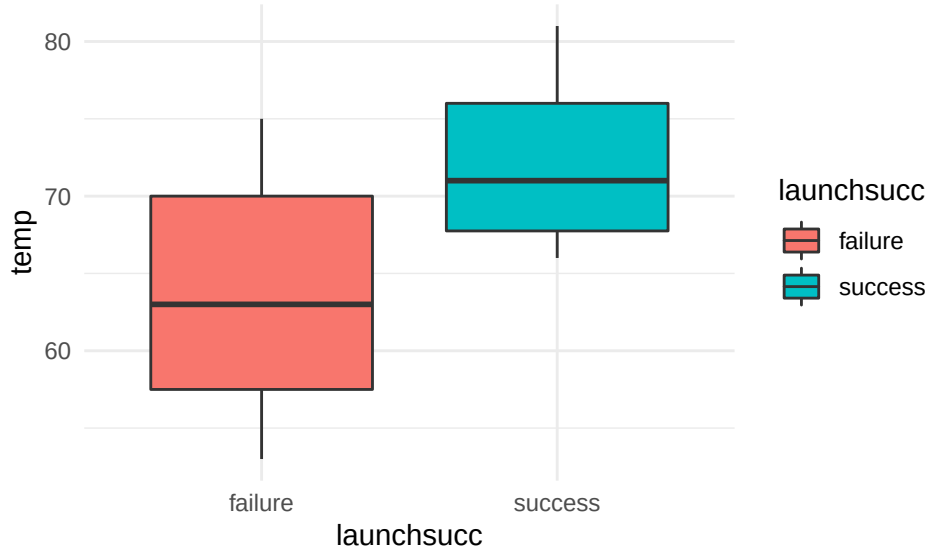
note on graphs: if you tell me the type of graph you want, and you don't know how to make it, ask me and I'll send you code! Remember, your response is binary and your explanatory variable is continuous.

```
shuttle <- read.delim("C7 Shuttle.txt", na="*")

# new names that make the data easier to work with:
# mine loads with an empty 5th column
names(shuttle) <- c("flight", "date", "temp", "launch", "X5")

# remove the row that has a missing value for launch
# also create a character variable for success
shuttle <- shuttle %>%
  filter(!is.na(launch)) %>%
```

```r
  mutate(launchsucc = as.factor(ifelse(launch == 1, "success", "failure")))

##########
#Logistic regression plot

fit = glm(launch ~ temp, data=shuttle, family=binomial)
newdat <- data.frame(temp=seq(min(shuttle$temp), max(shuttle$temp),len=100))
newdat$y = predict(fit, newdata=newdat, type="response")

ggplot(shuttle)+geom_point(aes(x=temp, y=launch))+geom_line(data=newdat, aes(x=temp, y=y))+xlab("Tempera
```




```r
##########
#boxplot

ggplot(shuttle)+geom_boxplot(aes(x=launchsucc, y=temp, fill=launchsucc))+theme_minimal()
```



The logistic regression plot shows that the successful launches all take place at higher temperatures... while the failures are generally at lower temps.

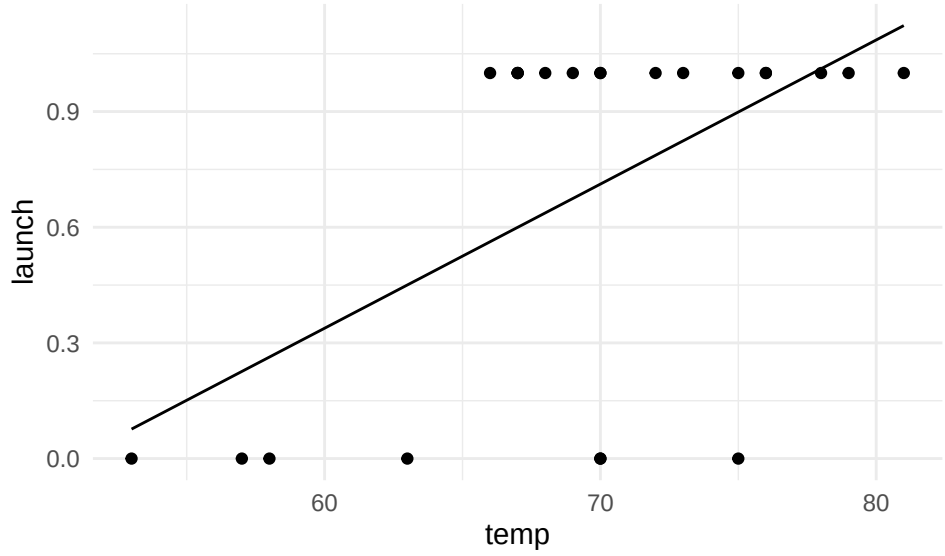
The boxplot shows that the distribution of temperatures for the successful launches are much higher than the distribution of temperatures for the failed launches.

Q4. Chp 7, A3 Use the data in Table 7.1 to create a scatterplot with a least squares regression line for the space shuttle data. Calculate the predicted response values ($\hat{y} = b_0 + b_1 x$) when the temperature is 60F and when the temperature is 85F.

```
lmfit<-lm(launch~temp, data=shuttle)
lmfit %>% tidy()
```

```
## # A tibble: 2 x 5
##   term        estimate std.error statistic p.value
##   <chr>          <dbl>     <dbl>     <dbl>   <dbl>
## 1 (Intercept)  -1.90      0.842      -2.26 0.0344
## 2 temp          0.0374    0.0120      3.10 0.00538
```

```
newdata=data.frame(temp=seq(min(shuttle$temp), max(shuttle$temp), len=100))
newdata$y<-predict(lmfit, newdata, type='response')

ggplot(shuttle)+geom_point(aes(x=temp, y=launch))+geom_line(data=newdata, aes(x=temp, y=y))+theme_minima
```




```
predict(lmfit, data.frame(temp=c(60, 85)), type='response')
```

```
##         1         2
## 0.3380952 1.2726190
```

For a SLR: $b_0 = -1.9$ and $b_1 = 0.0374$.

These predicted values $0.338, 1.272$ are difficult to interpret, since we are dealing with a binary outcome!

Q5. Chp 7, A4 Solve Equation (7.5) for π_i to show that Equation (7.6) is true. Note that your text uses π_i to represent the true model (akin to p_i that has been used in class). The difference is only in notation, not in meaning.

$$\ln\left(\frac{\pi_i}{1-\pi_i}\right) = \beta_0 + \beta_1 x_i$$
$$\frac{\pi_i}{1-\pi_i} = \exp(\beta_0 + \beta_1 x_i)$$
$$\pi_i = (1-\pi_i)\exp(\beta_0 + \beta_1 x_i)$$
$$\pi_i(1 + \exp(\beta_0 + \beta_1 x_i)) = \exp(\beta_0 + \beta_1 x_i)$$
$$\pi_i = \boxed{\frac{\exp(\beta_0 + \beta_1 x_i)}{1 + \exp(\beta_0 + \beta_1 x_i)}}$$

Q6. Chp 7, A5 Use Equation (7.6) to create twelve graphs: In each graph plot the explanatory variable (x) versus the expected probability of success (pi) using the following values:

	1	2	3	4	5	6	7	8	9	10	11	12
β_0	-10	-10	-10	-5	-5	-5	10	10	10	5	5	5
β_1	0.5	1	1.5	0.5	1	1.5	-0.5	-1	-1.5	-0.5	-1	-1.5

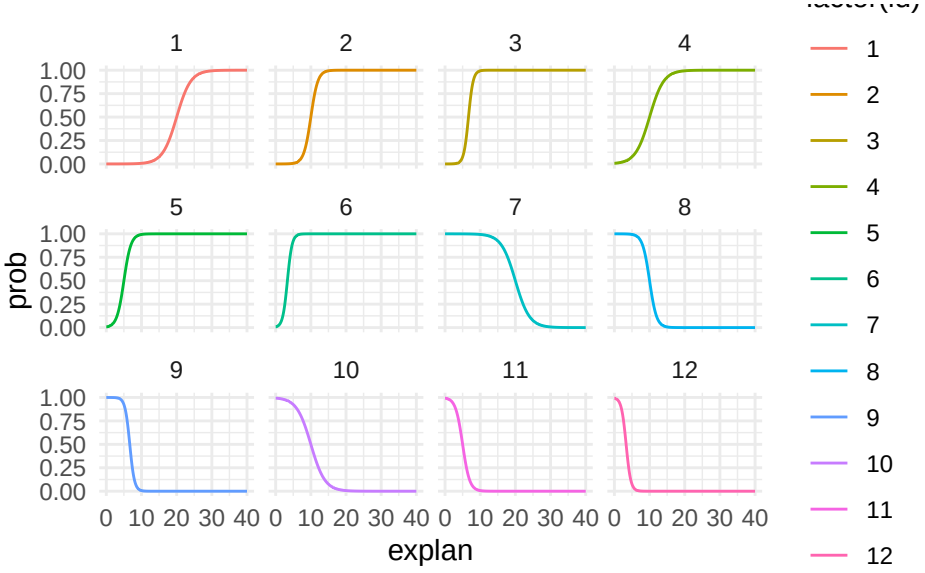
(a) Do not submit the graphs, but explain the impact of changing β_0 and β_1.

Increasing the magnitude of the slope β_1 increases the steepness of the sigmoid curve. A positive β_1 indicates upward slope while a negative β_1 indicates downward slope. Increasing the magnitude of β_0 can shift the curve to the left.

(b) For all of the graphs, at what value of π does there appear to be the steepest slope?

For $\boxed{\pi = 0.5}$.

```
#set the parameters
probfunc <- function(b0, b1, ex){
  exp(b0 + b1*ex) / (1 + exp(b0 + b1*ex))
}

betas0<-c(-10, -10, -10, -5, -5, -5, 10, 10, 10, 5, 5, 5)
betas1<-c(0.5, 1, 1.5, 0.5, 1, 1.5, -0.5, -1, -1.5, -0.5, -1, -1.5)

datatoplot <- data.frame(explan = NA, prob=NA, id=NA)
for(i in 1:length(betas0)){
  beta0 <- betas0[i]
  beta1 <- betas1[i]
  valuesofX <- seq(0,40,by=0.01)  # create a vector of X values

  valuesofY <- probfunc(beta0, beta1, valuesofX)

  datatoplot <- rbind(datatoplot, data.frame(explan = valuesofX, prob = valuesofY, id=i))}

datatoplot<-datatoplot %>% na.omit()

ggplot(datatoplot) +
  geom_line(aes(x = explan, y = prob, color=factor(id)))+theme_minimal()+facet_wrap(~id)
```



Q7. Chp 7, A6 [For the shuttle data:] Use statistical software to calculate the maximum likelihood estimates of β_0 and β_1. Compare the maximum likelihood estimates to the least squares estimates in A3. Use `glm(response ~ explanatory, family = "binomial", data = yourdataset) %>% tidy()`.

```
glm(launch ~ temp, data=shuttle, family='binomial') %>% tidy()
```

```
## # A tibble: 2 x 5
##   term        estimate std.error statistic p.value
##   <chr>          <dbl>     <dbl>     <dbl>   <dbl>
## 1 (Intercept)  -15.0       7.38      -2.04  0.0415
## 2 temp           0.232     0.108      2.14  0.0320
```

$\hat{\beta}_0 = -15$ and $\hat{\beta}_1 = 0.232$. These estimates are much better than those obtained by least squares, since now we are modeling the response as binary!

Q8. Chp 7, A7 Use Equation (7.9) to predict the probability that a launch has no O-ring damage when the temperature is 31F, 50F, and 75F.

Use the formula below: $\pi_i = \frac{\exp(b_0+b_1 x_i)}{1+\exp(b_0+b_1 x_i)}$

$\pi_{31^*} = \frac{\exp(-15+0.232*31)}{1+\exp(-15+0.232*31)} = 4.06 \times 10^{-4}$

$\pi_{50^*} = \frac{\exp(-15+0.232*50)}{1+\exp(-15+0.232*50)} = 3.22 \times 10^{-2}$

$\pi_{75^*} = \frac{\exp(-15+0.232*75)}{1+\exp(-15+0.232*75)} = 9.17 \times 10^{-1}$

Q9. Chp 7, A8 Calculate the odds of a launch with no O-ring damage when the temperature is 60F and when the temperature is 70F.

$Odds = \frac{\pi_i}{1-\pi_i} = \exp(b_0 + b_1 x_i)$

$Odds_{60^*} = \exp(-15 + 0.232 \cdot 60) = 0.34$

$Odds_{70^*} = \exp(-15 + 0.232 \cdot 70) = 3.46$

Q10. Chp 7, A9 For the shuttle model above, when x_i increases by 10, state in terms of e^{b_1} how much you would expect the odds to change. (Here you are calculating the odds ratio for an increase in 10 degrees.)

$$\hat{OR} = \frac{Odds_{x+10}}{Odds_x} = \frac{\exp(-15 + 0.232 \cdot (x+10))}{\exp(-15 + 0.232 \cdot x)} = \exp(0.232 \cdot 10) = \exp(b_1)^{10}$$

A 10 degree temperature change causes the odds ratio to be $\exp(b_1)^{10}$.

Q11. Chp 7, A10 The difference between the odds of success at 60F and 59F is about 0.3285 - 0.2605 = 0.068. Would you expect the difference between the odds at 52F and 51F to also be about 0.068? Explain why or why not.

No the difference in odds is not constant. The odds of success at 60F is $\exp(-15 + 0.232 \cdot 60) = 0.339$ and the odds of success at 59F is $\exp(-15 + 0.232 \cdot 59) = 0.269$ generating a difference in odds of around 0.07. The odds of success at 52F is $\exp(-15 + 0.232 \cdot 52) = 0.053$ and the odds of success at 59F is $\exp(-15 + 0.232 \cdot 51) = 0.042$ generating a difference in odds of around 0.011. Thus, the difference in odds is not constant. This is because precisely because the odds are equal to $\exp$(Linear component), which doesn't have constant slope over various 1* temperature intervals.

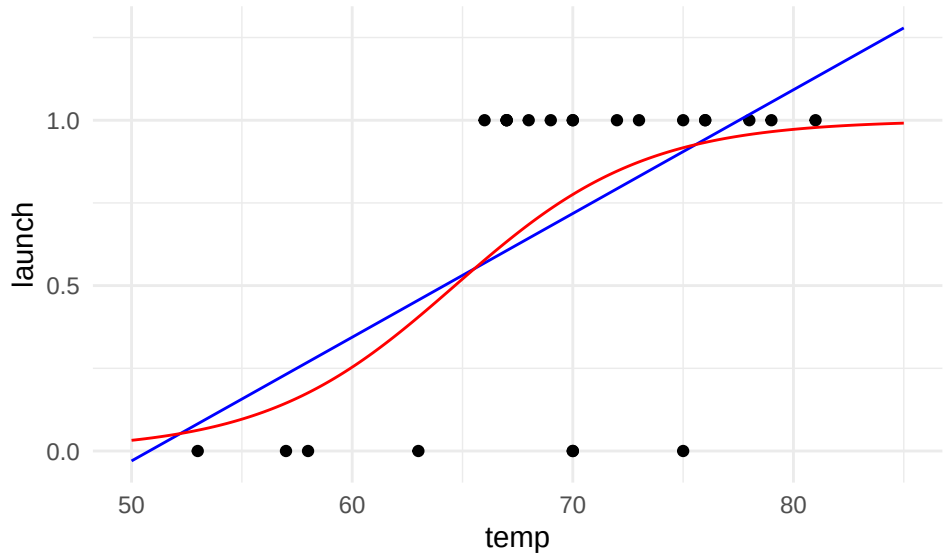
Q12. Chp 7, A11 Create a plot of two logistic regression models. Plot temperature versus the estimated probability using maximum likelihood estimates from A6, and plot temperature versus the estimated probability using the least squares estimates from A3.

R code: Step1. Look up at `probfunc()` above. Write a very similar function that is linear instead. Give it a different name.

Step2. Using the two sets of coefficients (one from the linear and one from the logistic), predict the "y" value for both models for a vector of possible explanatory variables (e.g., `valuesofX <- seq(50,85,by=0.01)`). You should have two different vectors of predictions (and the vector of X, the explanatory variable).

Step3. Create a `data.frame()` with three columns. Let's say you call it `mypredictons`. The `ggplot` code will look like this. Have fun with coloring the plot or changing the line types or something!

```
probfunc <- function(b0, b1, ex){
  exp(b0 + b1*ex) / (1 + exp(b0 + b1*ex))
}

linfunc <- function(b0, b1, ex){
  b0 + b1*ex
}

temps <- seq(50,85,by=0.01)

linfun_y=linfunc(b0=-1.9, b1=0.0374, ex=temps)
logreg_y=probfunc(b0=-15, b1=0.232, ex=temps)

mypredictions=data.frame(temps=temps, lin=linfun_y, log=logreg_y)

ggplot(shuttle) +
   geom_point(aes(x = temp, y = launch)) +
   geom_line(data = mypredictions, aes(x = temps, y = lin), color="blue") +
   geom_line(data = mypredictions, aes(x = temps, y = log), color="red") +
    theme_minimal()
```




```
praise()
```

```
## [1] "You are beautiful!"
```