HW3

Ben Tankus

4/17/2021

1.

```r
library(vcdExtra)
```

```
## Warning: package 'vcdExtra' was built under R version 4.0.4

## Loading required package: vcd

## Warning: package 'vcd' was built under R version 4.0.4

## Loading required package: grid

## Loading required package: gnm

## Warning: package 'gnm' was built under R version 4.0.4
```

```r
library(tidyverse)
```

```
## Warning: package 'tidyverse' was built under R version 4.0.3

## -- Attaching packages ----------------------------------------------------------------------

## v ggplot2 3.3.3     v purrr   0.3.4
## v tibble  3.0.3     v dplyr   1.0.2
## v tidyr   1.1.2     v stringr 1.4.0
## v readr   1.4.0     v forcats 0.5.0

## Warning: package 'ggplot2' was built under R version 4.0.3

## Warning: package 'tidyr' was built under R version 4.0.3

## Warning: package 'readr' was built under R version 4.0.3

## Warning: package 'purrr' was built under R version 4.0.3

## Warning: package 'dplyr' was built under R version 4.0.3
```

```
## Warning: package 'forcats' was built under R version 4.0.3

## -- Conflicts --------------------------------------------------------------------------------
## x dplyr::filter()    masks stats::filter()
## x dplyr::lag()       masks stats::lag()
## x dplyr::summarise() masks vcdExtra::summarise()
```

```
library(magrittr)
```

```
##
## Attaching package: 'magrittr'

## The following object is masked from 'package:purrr':
##
##     set_names

## The following object is masked from 'package:tidyr':
##
##     extract
```

```
library(ggplot2)
logistic <- function(x){exp(x)/(1 + exp(x))}

df <- read.csv('admissions.csv')
```

(a)

```
fit.glm.gre <- glm(admit ~ gpa + gre + factor(rank), family = binomial(link = "logit"), data = df)
fit.glm <- glm(admit ~ gpa  + factor(rank), family = binomial(link = "logit"), data = df)


summary(fit.glm.gre)
```

```
##
## Call:
## glm(formula = admit ~ gpa + gre + factor(rank), family = binomial(link = "logit"),
##     data = df)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.6268  -0.8662  -0.6388   1.1490   2.0790
##
## Coefficients:
##                Estimate Std. Error z value Pr(>|z|)
## (Intercept)   -3.989979   1.139951  -3.500 0.000465 ***
## gpa            0.804038   0.331819   2.423 0.015388 *
## gre            0.002264   0.001094   2.070 0.038465 *
## factor(rank)2 -0.675443   0.316490  -2.134 0.032829 *
## factor(rank)3 -1.340204   0.345306  -3.881 0.000104 ***
## factor(rank)4 -1.551464   0.417832  -3.713 0.000205 ***
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 499.98  on 399  degrees of freedom
## Residual deviance: 458.52  on 394  degrees of freedom
## AIC: 470.52
##
## Number of Fisher Scoring iterations: 4
```

```
summary(fit.glm)
```

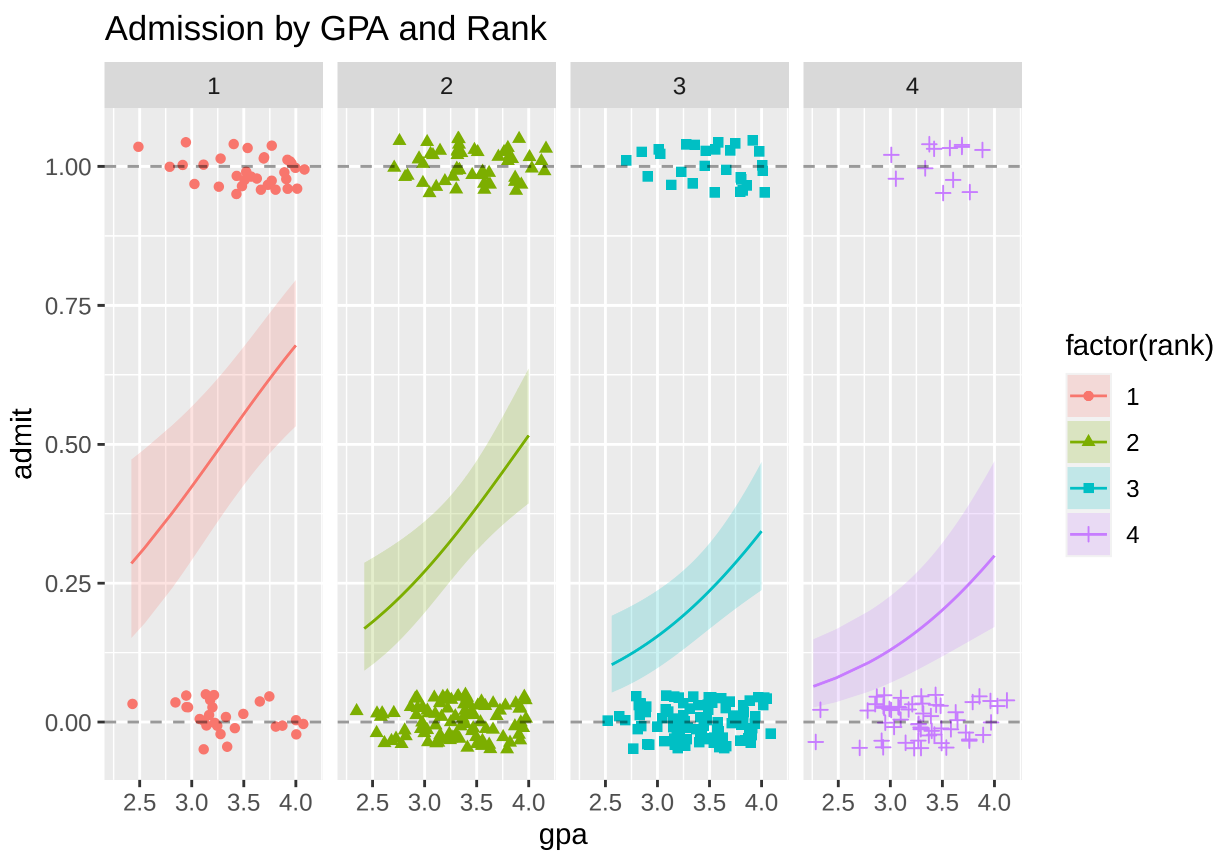
```
##
## Call:
## glm(formula = admit ~ gpa + factor(rank), family = binomial(link = "logit"),
##     data = df)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.5055  -0.8663  -0.6590   1.1505   2.0913
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)    -3.4636     1.1003  -3.148 0.001645 **
## gpa             1.0521     0.3102   3.392 0.000694 ***
## factor(rank)2  -0.6810     0.3141  -2.168 0.030181 *
## factor(rank)3  -1.3919     0.3419  -4.071 4.68e-05 ***
## factor(rank)4  -1.5943     0.4152  -3.840 0.000123 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 499.98  on 399  degrees of freedom
## Residual deviance: 462.88  on 395  degrees of freedom
## AIC: 472.88
##
## Number of Fisher Scoring iterations: 4
```

The model with the GRE only fits the data slightly better with a AIC of 470.52 VS 472.88. All fields are significant in both models.

(b)

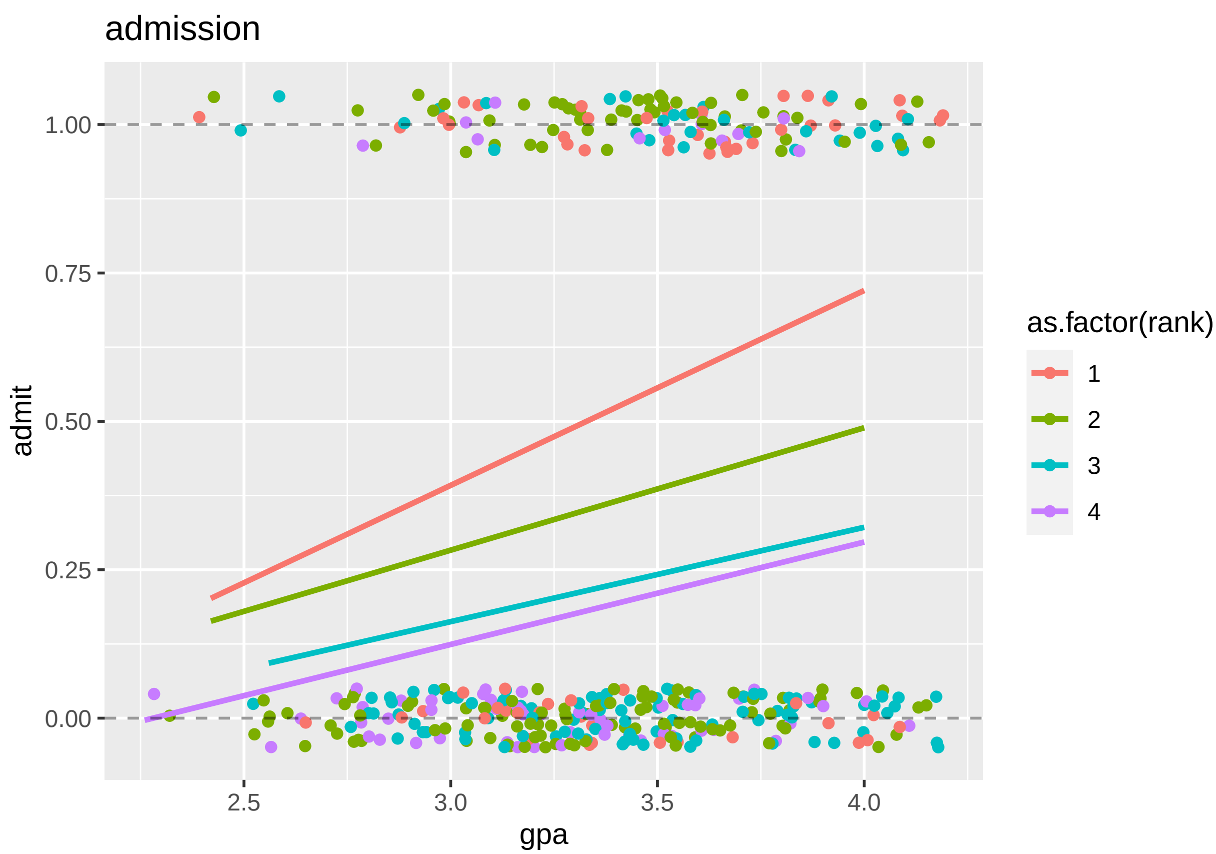
```
#plot(df$gpa, df$admit)

# Obtain 95% pointwise confidence bands from predict.glm()
glm_pred <- predict.glm(fit.glm, type="link", se.fit=TRUE)
low <- glm_pred$fit - 1.96 * glm_pred$se.fit
upp <- glm_pred$fit + 1.96 * glm_pred$se.fit

# back-transform everything to the data scale
```

```r
glm_fit <- logistic(glm_pred$fit)
glm_lower <- logistic(low)
glm_upper <- logistic(upp)

# augment the Donner data frame
augment_df <- as.data.frame(cbind(df, glm_fit, glm_lower, glm_upper))

# Big plot
ggplot(data = augment_df) +
  # plot jittered data
  geom_jitter(aes(x = gpa,
                  y = admit,
                  color = factor(rank),
                  shape = factor(rank)),
                  height = 0.05, width = 0.2) +

  # plot fitted lines
  geom_line(aes(x = gpa,
               y = glm_fit,
                color = factor(rank))) +

# plot 95% pointwise confidence bands
  geom_ribbon(aes(x = gpa,
                   fill = factor(rank),
                   ymin = glm_lower,
                   ymax = glm_upper),
              alpha = 0.2) +

  # plot reference lines at 0 and 1 (minimum and maximum possible probabilities)

  geom_hline(yintercept = 0, lty = 2, alpha = 0.4) +
  geom_hline(yintercept = 1, lty = 2, alpha = 0.4) +
  facet_grid(.~factor(rank)) +
  ggtitle("Admission by GPA and Rank")
```




```r
ggplot(data = df) +
  # original data
  geom_jitter(aes(x = gpa,
                  y = admit,
                  color = as.factor(rank)),
              height = 0.05, width = 0.2) +
  # This is a shortcut, within ggplot, to fitting a linear model and plotting the fits
  geom_smooth(mapping = aes(x = gpa, y = admit, color = as.factor(rank)),
              method = 'lm', se = FALSE) +
  # reference lines
  geom_hline(yintercept = 0, lty = 2, alpha = 0.4) +
  geom_hline(yintercept = 1, lty = 2, alpha = 0.4) +
  ggtitle("admission")
```

```
## `geom_smooth()` using formula 'y ~ x'
```



(c)

In all ranks, as GPA increases so does likelyhood of admittance. Ranks 1 and 2 seem to have a linear trend, where 4 has a clear non-linear, exponential trend. rank 3 seems on the border of both. There don't appear to be any unrepresented trends in the data, so I belive the model fits well.

It seems that if one comes from a good undergrad (rank 1 or 2) you have a much higher chance of getting into grad school. It also looks like gpa is a large indicator of whether or not one would be admitted.

Conceptual Questions

2.

(a)

In this case, one data point likely loves another data point, and would give their life for them. This makes the data dependent, as if one lives, the other may have to die.

(b)

There are 3 women who died *just* under 50, and none who died right over 50. There were also only 4 of the party aged over 50, so I would be hesitant to draw conclusions from this. I would reccomend cutting at 45 years to ensure a more representative analysis.

(c)

solve for age: log(p/(1-p)) = $\beta_0 + \beta_1 age + \beta_2 gender$

age = (log(p/(1-p)) - β_0 - $\beta_2 gender$))/β_1

```
p = 0.5

age.f = (log(p/(1-p)) - 1.63 - 1.60)/-0.078
age.f
```

```
## [1] 41.41026
```

```
age.m = (log(p/(1-p)) - 1.63)/-0.078
age.m
```

```
## [1] 20.89744
```

The survival probability is 50% for 41.4 year old women and 20.9 year old men.

3.

(a)

The remaining estimates changed with the removal of the head_length variable because they are likely dependent on one another. The largest change occured with skull_width which makes sense. If head length increases, it's likely the width will as well.

(b)

p/1-p = odds

solve for p p = o / (1+o)

```
b0 = 33.5095
b1 = -1.4207
b2 = -0.2787
b3 = 0.5687
b4 = -1.8057

odds = exp(b0 + b1*1 + b2*65 + b3*80 + b4*32)

p = odds/(1+odds)
p
```

```
## [1] 0.843816
```

There is a 84.4% chance that this possum is from Victoria using the reduced model.