Assignment: Employee Retention

Dunja Novaković

2021-07-31

Instructions

This assignment reviews the *Employee Retention* analytical lecture. You will use the *retention.Rmd* file I reviewed in the video lectures to complete this assignment. You will *copy and paste* relevant code from that file and update it to answer the questions in this assignment. You will respond to questions in each section after executing relevant code to answer a question. You will submit this assignment to its *Submissions* folder on *D2L*. You will submit this *(1)* completed **R Markdown** script and *(2)* a *PDF*, *Word*, or *HTML* rendered version of it to *D2L* by the due date and time. As a first option, if you installed `TinyTeX` successfully, then I prefer a *PDF* version. As a second option, if you have *Microsoft Word*, then I prefer a *Word* version. As a third option, you can knit to *HTML*. The first two options work better with *D2L*.

To start:

For any analytical project, you want to create a clear project directory structure.
All materials from this course should exist in one folder on your computer. Inside of that main course folder, you should create folders to store course documentation, lecture analytical projects, assignments analytical projects, etc. Inside of your folder for assignments analytical projects, you should create folder for this assignment named *retention*.

Any analytical project folder should contain inside it at least three additional folders named *scripts*, *data*, and *plots*. Store this script in the *scripts* folder, the data for this assignment in the *data* folder, and any requested plots in the *plots* folder. Each analytical project should also contain a **.Rproj** file in its top-level directory. Go to the *File* menu in *RStudio*, select *New Project...*, choose *Existing Directory*, go to the folder you created to contain this analytical project. Select it as the top-level directory for this **RStudio Project**.

Global Settings

The first code chunk sets the global settings for the remaining code chunks in the document. Do *not* change anything in this code chunk.

Load Packages

In this code chunk, we load packages we need for this assignment:

1. **here**,
2. **tidyverse**,
3. **skimr**,
4. **broom**, and
5. **interactions**.

We will use functions from these packages to import the data, examine the data, calculate summaries on the data, build logistic regression models, and create visualizations from the data. Do *not* change anything in this code chunk.

```
### load libaries for use in current working session
## here for workflow
library(here)
```

```
## here() starts at C:/Users/novak/OneDrive/Desktop/MGT 591/Assignments/retention
```

```
## tidyverse for data manipulation and plotting
# loads eight different libraries simultaneously
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --

## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.2     v dplyr   1.0.7
## v tidyr   1.1.3     v stringr 1.4.0
## v readr   1.4.0     v forcats 0.5.1

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

```
## skimr for summary statistics
library(skimr)

## broom to work with model objects
library(broom)

## interactions to make 2D plots to capture interaction effects
library(interactions)
```

Task 1: Load and Clean Data

Load the **retention.rds** data file with the correct functions. Save the imported data as the data object: **reten_data**. Use **glimpse** on the data.

Question 1.1: How many variables and observations are there in the data?

Response 1.1: *1650 observations and 8 variables.*

Convert **BossGender**, **Gender**, **Country**, and **LeaverStatus** to factor variables and assign the correct factor levels to each of them. Rename **BossGender** and **Gender** to **Boss Gender** and **Emp. Gender**, respectively. Save all changes to **reten_data**. Print the updated data object such that you arrange it by descending **Age**.

Question 1.2: What is the age of the oldest employees in the data?

Response 1.2: *66.*

```
#### Q1.1
### load data via the readRDS and here functions
reten_data <- readRDS(here("data", "retention.rds"))

### examine sampled data
## using glimpse()
glimpse(reten_data)
```

```
## Rows: 1,650
## Columns: 8
## $ BossGender      <dbl> 0, 0, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ~
## $ Gender          <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ~
## $ Age             <dbl> 48, 26, 45, 52, 31, 49, 42, 30, 27, 38, 22, 40, 39, 32~
## $ LengthOfService <dbl> 0, 6, 26, 13, 1, 0, 25, 12, 11, 6, 4, 21, 22, 14, 15, ~
## $ AppraisalRating <dbl> 2.53, 6.13, 4.42, 6.14, 6.96, 2.75, 6.19, 7.78, 6.09, ~
## $ CareerSat       <dbl> 4.24, 4.97, 7.79, 7.24, 6.24, 4.86, 9.00, 5.72, 6.14, ~
## $ Country         <dbl> 1, 1, 1, 1, 1, 1, 1, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, ~
## $ LeaverStatus    <dbl> 1, 1, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, ~
```

```
#### Q1.2
### mutate variables
## overwrite data
reten_data <- reten_data %>%
  ## change vars to factors
  mutate_at(vars(BossGender, Gender, Country, LeaverStatus), ~ as_factor(.)) %>%
  ## assign gender levels
  mutate_at(vars(BossGender, Gender), ~ fct_recode(., `Female` = "0", `Male` = "1")) %>%
  ## change Country and LeaverStatus
         # LeaverStatus
  mutate(LeaverStatus = fct_recode(LeaverStatus, `Stayer` = "0", `Leaver` = "1"),
         # Country
         Country = fct_recode(Country, `Belgium` = "1", `Sweden` = "2", `Italy` = "3",
                              `France` = "4", `Poland` = "5", `Mexico` = "6", `Spain` = "7",
                              `UK` = "8", `USA` = "9", `Australia` = "10")) %>%
  ## rename variables for plotting aesthetics
  rename(`Boss Gender` = BossGender, `Emp. Gender` = Gender)

## print to view results
reten_data %>%
  arrange(desc(Age))
```

```
## # A tibble: 1,650 x 8
##    `Boss Gender` `Emp. Gender`   Age LengthOfService AppraisalRating CareerSat
##    <fct>         <fct>         <dbl>           <dbl>           <dbl>     <dbl>
##  1 Female        Female           66              15            4.86      5.59
##  2 Male          Male             66               6            3.91      7.08
##  3 Male          Female           64              11            4.28      5.09
##  4 Female        Male             64               5            5.86      4.02
##  5 Male          Male             63               0            3.47      3.75
##  6 Male          Male             63              28            7.61      5.01
##  7 Male          Male             62              31            5.13      7.33
##  8 Female        Male             62               6            9         6.85
##  9 Male          Male             61              31            6.35      5.6
```

```
## 10 Female        Female           61              24           3.7       5.45
## # ... with 1,640 more rows, and 2 more variables: Country <fct>,
## #   LeaverStatus <fct>
```

Task 2: Examine Data

Summarize the data with **skim_without_charts()** while grouping by **LeaverStatus**.

Question 2.1: What is the difference in the average *appraisal rating* between *stayers* and *leavers*?

Response 2.1: *5.54-5.38=0.16.*

Produce a density plot for **AppraisalRating** filled by **LeaverStatus**. Facet the plot such that rows represent **Emp. Gender** and columns represent **Boss Gender**. Note the faceting differs from the lecture script. Appropriately label the aesthetics. Beautify the facet grid labels as well.

Question 2.2: Is there a mean difference between *stayers* and *leavers* on *appraisal ratings* for any combination of *boss* and *employee gender*?

Response 2.2: *There is a mean difference between stayers and leavers on appraisal ratings for male employees with male bosses.*

In one chained command, produce a horizontal bar plot showing the percentage of *leavers* and *stayers* for each combination of *boss* and *employee gender*. Call **reten_data** and group by **Boss Gender**, **Emp. Gender**, and **LeaverStatus** in that order. Count the number of cases who compose those eight groups. Then, group by only **Boss Gender** and **Emp. Gender** and compute the percentage of *stayers* and *leavers* for each combination of *boss* and *employee gender*. Name the computed percentage variable: **pct**. Create a **ggplot** with the x-axis representing percentage of *stayers* and *leavers*, the y-axis representing the *female and male bosses*, facets representing *female and male employees*, and filling the bars by *stayers* and *leavers*. In the aesthetics, order **Boss Gender** by **LeaverStatus** and **pct**. You will need to call **facet_wrap** to facet by **Emp. Gender**. Use **labeller = label_both** inside of **facet_wrap** to clearly indicate the facets represent employee gender. Make sure to include a text geometry to label the percentage of *stayers* and *leavers* for each combination of *employee* and *boss gender*.

Question 2.3: Which combination of *employee* and *boss gender* has the highest percentage of *stayers*? Which combination of *employee* and *boss gender* has the highest percentage of *leavers*?

Response 2.3: *The group consisted of male employees with male bosses has the highest percentage of stayers. The group consisted of female employees with female bosses has the highest percentage of leavers.*

```
#### Q2.1
### summarize by LeaverStatus
## call data
reten_data %>%
  ## group by LeaverStatus
  group_by(LeaverStatus) %>%
  ## summarize
  skim_without_charts()
```

Table 1: Data summary

Name	Piped data
Number of rows	1650
Number of columns	8

Column type frequency:	
factor	3
numeric	4
Group variables	LeaverStatus

Variable type: factor

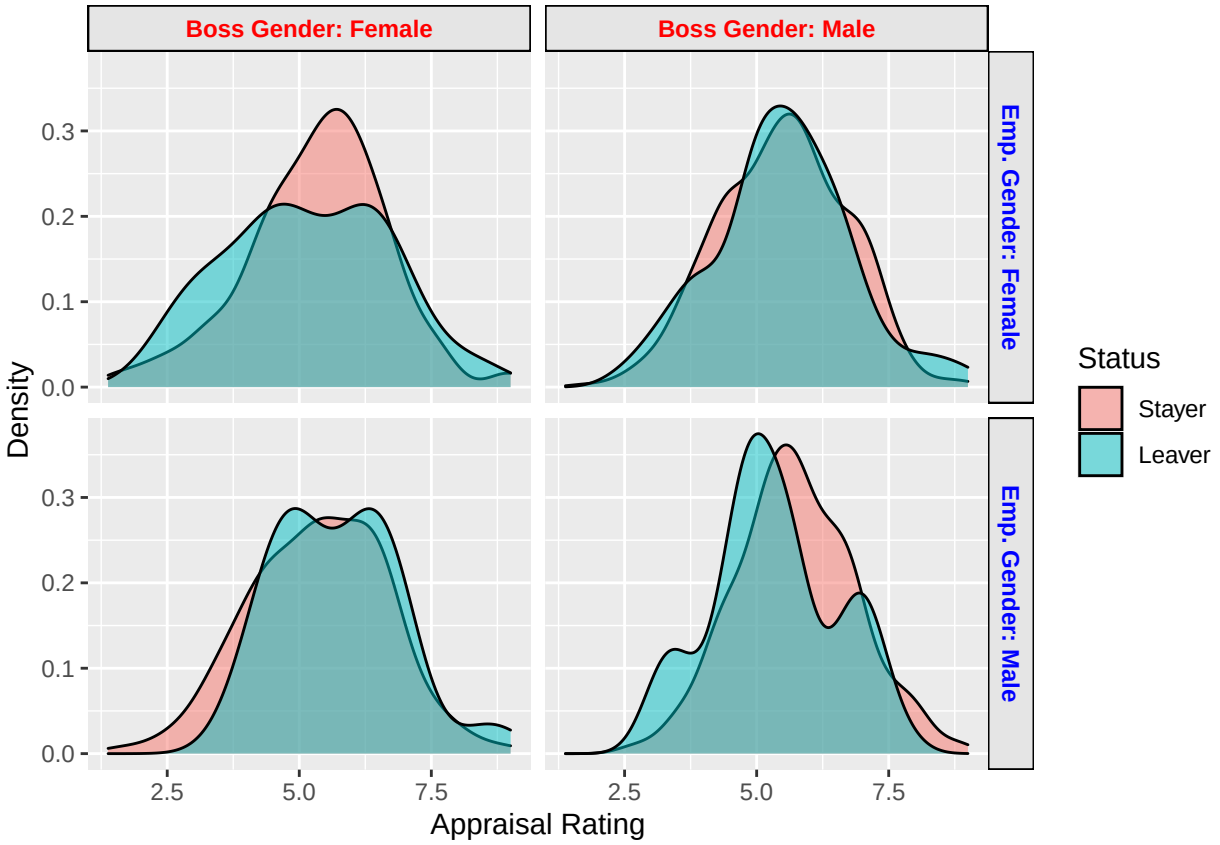
skim_variable	LeaverStatus	n_missing	complete_rate	ordered	n_unique	top_counts
Boss Gender	Stayer	0	1	FALSE	2	Mal: 1003, Fem: 438
Boss Gender	Leaver	0	1	FALSE	2	Mal: 128, Fem: 81
Emp. Gender	Stayer	0	1	FALSE	2	Mal: 745, Fem: 696
Emp. Gender	Leaver	0	1	FALSE	2	Fem: 136, Mal: 73
Country	Stayer	0	1	FALSE	10	USA: 949, UK: 168, Spa: 64, Ita: 55
Country	Leaver	0	1	FALSE	10	USA: 115, UK: 27, Ita: 15, Spa: 12

Variable type: numeric

skim_variable	LeaverStatus	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100
Age	Stayer	0	1	38.38	9.44	16.00	31.00	38.00	46.00	66.00
Age	Leaver	0	1	37.05	10.00	19.00	29.00	36.00	44.00	64.00
LengthOfService	Stayer	0	1	10.12	9.73	0.00	2.00	7.00	15.00	42.00
LengthOfService	Leaver	0	1	7.65	9.26	0.00	1.00	5.00	11.00	37.00
AppraisalRating	Stayer	0	1	5.54	1.23	1.38	4.72	5.58	6.39	9.00
AppraisalRating	Leaver	0	1	5.38	1.33	2.17	4.50	5.32	6.29	9.00
CareerSat	Stayer	0	1	5.84	1.22	1.91	5.01	5.86	6.69	9.00
CareerSat	Leaver	0	1	5.20	1.15	2.43	4.40	5.11	5.92	8.21

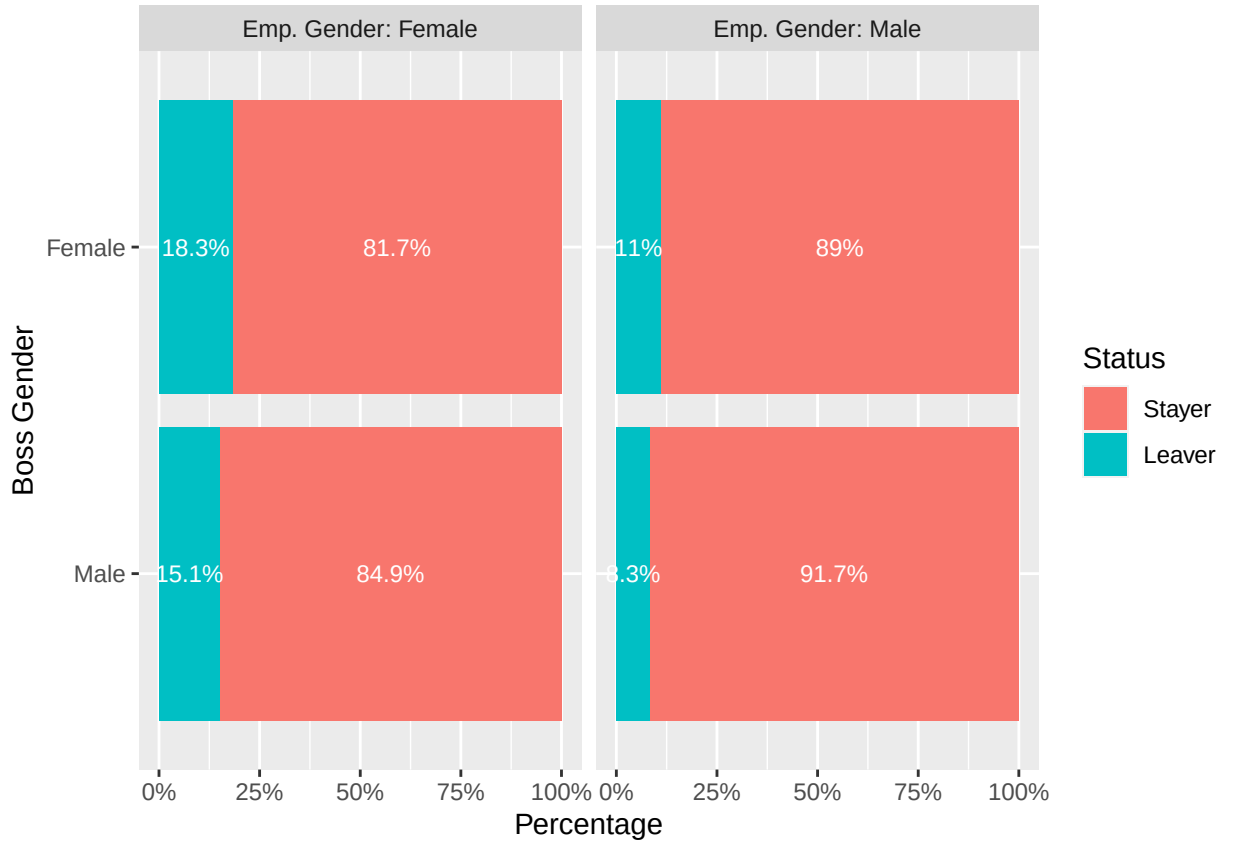
```
#### Q2.2
### distributions
## call data and set aesthetics
ggplot(reten_data, aes(x = AppraisalRating, fill = LeaverStatus)) +
  ## density geometry
  geom_density(alpha = 0.5) +
  ## facet by boss and employee gender
  facet_grid(`Emp. Gender` ~ `Boss Gender`, labeller = label_both) +
  ## aesthetic labels
  labs(x = "Appraisal Rating", y = "Density", fill = "Status") +
  ## change aesthetics of gender labels
  theme(
    # change employee gender
    strip.text.x = element_text(
      color = "red", face = "bold"
    ),
    # change boss gender
    strip.text.y = element_text(
      color = "blue", face = "bold"
    ),
    # change background
```

```
    strip.background = element_rect(
      color = "black", fill = "grey90"
    )
  )
```




```
#### Q2.3
### bar plot
## call data
reten_data %>%
  ## group by Boss Gender, Emp. Gender and LeaverStatus
  group_by(`Boss Gender`, `Emp. Gender`, LeaverStatus) %>%
  ## count
  count() %>%
  ## group by Boss Gender, Emp. Gender
  group_by(`Boss Gender`, `Emp. Gender`) %>%
  ## calculate percentage
  mutate(pct = round(n/sum(n), digits = 3)) %>%
  ## call plot
  ggplot(aes(x =  fct_rev(fct_reorder2(`Boss Gender`, LeaverStatus, pct)),
             y = pct, fill = LeaverStatus)) + facet_wrap(~`Emp. Gender`, labeller = label_both)+
    ## bar geometry
    geom_bar(position = "fill", stat = "identity") +
    ## text geometry
    geom_text(aes(label = paste0(pct*100, "%")), size = 3,
            position = position_stack(vjust = 0.5), color = "white") +
```

```
## y-axis
scale_y_continuous(labels = scales::percent_format()) +
## aesthetic labels
labs(x = "Boss Gender", y = "Percentage", fill = "Status") +
## flip coordinates
coord_flip()
```



Task 3: Simple Logistic Regression

Estimate a simple logistic regression where *appraisal rating* predicts *employee retention* using the correct variables. Name the model: **mod_1**. Examine the levels and contrats for *employee retention* if needed.

Question 3.1: How do you correctly interpret the logit regression coefficient for *appraisal rating*?

Response 3.1: *For a one unit increase in appraisal rating, the log odds of an employee leaving (versus not-leaving) decreases by 0.1063.*

Calculate the *logit*, *odds ratio*, and *probability* predictions from **mod_1** and save them to **reten_data**. Select the relevant variables from **reten_data** and arrange it by ascending *appraisal rating*.

Question 3.2: What is the *highest probability* of someone leaving in this data based on **mod_1**? What is the predicted *probability of leaving* for the individual with an *appraisal rating* equal to 1.64?

Response 3.2: *0.183. 0.179.*

Produce a **ggplot** that represents **mod_1**. Label the axes appropriately.

Question 3.3: Do higher values of *appraisal rating* associate with a greater or lower probability of an employee leaving?

Response 3.3: *Higher values of appraisal rating associate with a lower probability of an employee leaving.*

```
#### Q3.1
### examine categorical outcome
## contrasts
contrasts(reten_data$LeaverStatus)
```

```
##        Leaver
## Stayer      0
## Leaver      1
```

```
### estimate simple logistic regression model
mod_1 <- glm(LeaverStatus ~ AppraisalRating, family = "binomial", data = reten_data)

## examine summary
summary(mod_1)
```

```
##
## Call:
## glm(formula = LeaverStatus ~ AppraisalRating, family = "binomial",
##     data = reten_data)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.6356  -0.5352  -0.5125  -0.4861   2.1918
##
## Coefficients:
##                  Estimate Std. Error z value Pr(>|z|)
## (Intercept)      -1.3501     0.3297  -4.095 4.23e-05 ***
## AppraisalRating  -0.1063     0.0595  -1.787   0.0739 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1254.0  on 1649  degrees of freedom
## Residual deviance: 1250.8  on 1648  degrees of freedom
## AIC: 1254.8
##
## Number of Fisher Scoring iterations: 4
```

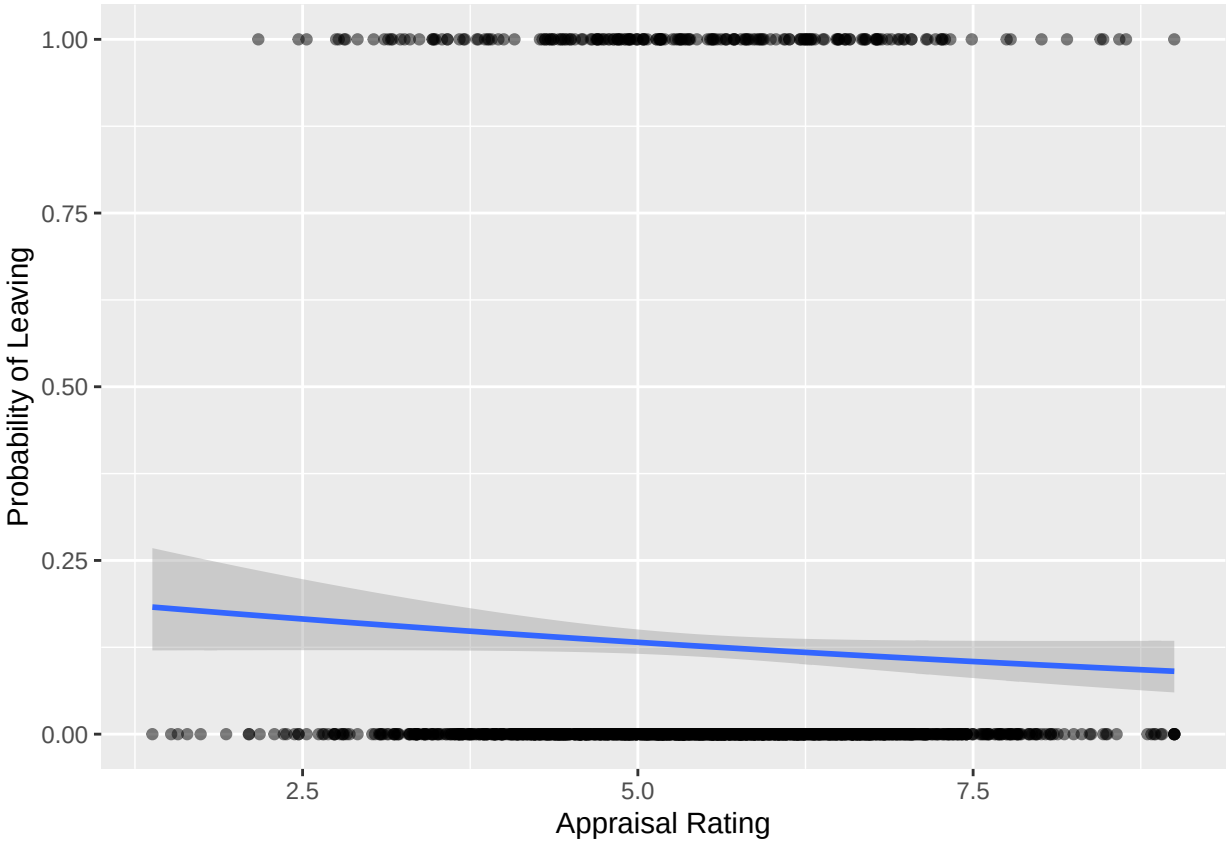
```
#### Q3.2
### predictions
## call data
reten_data <- reten_data %>%
         ## probability predictions
  mutate(mod_1_prob = fitted(mod_1),
         ## odds ratio predictions
         mod_1_or = exp(predict(mod_1)),
         ## logit predictions
```

```
         mod_1_log = predict(mod_1)
        )

### examine predictions
## call data
reten_data %>%
  ## select variables
  select(AppraisalRating, LeaverStatus, mod_1_prob:mod_1_log) %>%
  ## arrange
  arrange(AppraisalRating)
```

```
## # A tibble: 1,650 x 5
##    AppraisalRating LeaverStatus mod_1_prob mod_1_or mod_1_log
##              <dbl> <fct>             <dbl>    <dbl>     <dbl>
##  1            1.38 Stayer            0.183    0.224     -1.50
##  2            1.52 Stayer            0.181    0.221     -1.51
##  3            1.57 Stayer            0.180    0.219     -1.52
##  4            1.64 Stayer            0.179    0.218     -1.52
##  5            1.74 Stayer            0.177    0.215     -1.54
##  6            1.93 Stayer            0.174    0.211     -1.56
##  7            2.1  Stayer            0.172    0.207     -1.57
##  8            2.1  Stayer            0.172    0.207     -1.57
##  9            2.17 Leaver            0.171    0.206     -1.58
## 10            2.18 Stayer            0.171    0.206     -1.58
## # ... with 1,640 more rows
```

```
#### Q3.3
### Plot
## Choose data and mapping
ggplot(reten_data, aes(x = AppraisalRating, y = as.numeric(LeaverStatus) - 1)) +
  ## Point geometry
  geom_point(alpha = 0.5) +
  ## Smooth geometry
  geom_smooth(method = "glm", se = TRUE,
              method.args = list(family = "binomial")) +
  ## Axes labels
  labs(x = "Appraisal Rating", y = "Probability of Leaving")
```

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



Task 4: Multiple Logistic Regression

Estimate a multiple logistic regression where *appraisal rating*, *career satisfaction*, *age*, and *employee gender* predict *employee retention* using the correct variables. Name the model: **mod_2**. Examine the levels and contrasts for *employee gender* and cross-tabs between *employee gender* and *retention* if needed.

Question 4.1: How do you correctly interpret the *logit regression coefficient* for *career satisfaction*? What is the *odds ratio regression coefficient* for *career satisfaction*?

Response 4.1: *For a one unit increase in career satisfaction, the log odds of an employee leaving (versus not-leaving) decreases by 0.43492, holding the other predictors constant. The odds ratio regression coefficient for career satisfaction is 0.6473188.*

Calculate the *logit*, *odds ratio*, and *probability* predictions from **mod_2** and save them to **reten_data**. Select the relevant variables from **reten_data** and arrange it by *descending predicted probability*.

Question 4.2: What is the *highest probability* of someone leaving in this data based on **mod_2**? What is the predicted *odds ratio of leaving to not leaving* for the fourth individual in the list?

Response 4.2: *0.451. 0.757.*

Compute the number of *true* and *false positives* and *true* and *false negatives* based on **mod_2** and using a *0.25* probability threshold. Compute the relevant accuracy metrics for **mod_2** and this *0.25* probability threshold.

Question 4.3: How well does **mod_2** do at predicting who will leave the organization at a *0.25* probability threshold? What accuracy metric are you using to base your conclusion?

Response 4.3: *The model does not successfully predict who will leave the organization. The model can accurately distinguish true positives from false positives in only 29.2% of cases (as indicated by positive*

accuracy). In addition, the model successfully distinguishes true positives from false negatives in only 16.7% of cases (as indicated by sensitivity).

Create a new tibble named: **mod_2_pred_data** consisting of the four predictors in **mod_2**. The tibble should consist of *200* rows with all rows reflecting the average *age* and *appraisal rating* for the data, while *career satisfaction* from its minimum to its maximum value in the data should mix with *employee gender*. After initially creating **mod_2_pred_data**, apply **augment()** to it to calculate the *probability of leaving* for each row and the respective (95%) lower and upper boundary of each prediction. Produce a **ggplot** to with *career satisfaction* on the x-axis and the *probability of leaving* on the y-axis, separate lines for *employee gender*, and a ribbon to represent the upper and lower boundaries of prediction for each line.

Question 4.4: Is the difference in the *probability of leaving* for male and female employees greater for lower or higher *career satisfaction*?

Response 4.4: *The difference in the probability of leaving for male and female employees is greater for lower career satisfaction levels.*

```
#### Q4.1
### examine categorical predictors
## contrasts
# Emp. Gender
contrasts(reten_data$`Emp. Gender`)
```

```
##        Male
## Female    0
## Male      1
```

```
## cross-tabulations
# retention and country
xtabs(~ LeaverStatus + `Emp. Gender`, data = reten_data)
```

```
##             Emp. Gender
## LeaverStatus Female Male
##       Stayer    696  745
##       Leaver    136   73
```

```
### estimate multiple logistic regression model
mod_2 <- glm(LeaverStatus ~ AppraisalRating + CareerSat +
               Age + `Emp. Gender`,
             family = "binomial", data = reten_data)

## examine results
summary(mod_2)
```

```
##
## Call:
## glm(formula = LeaverStatus ~ AppraisalRating + CareerSat + Age +
##     `Emp. Gender`, family = "binomial", data = reten_data)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.0708  -0.5617  -0.4431  -0.3332   2.7241
##
## Coefficients:
```

```
##                   Estimate Std. Error z value Pr(>|z|)
## (Intercept)        1.63171    0.55227   2.955  0.00313 **
## AppraisalRating   -0.07378    0.06094  -1.211  0.22602
## CareerSat         -0.43492    0.06366  -6.831 8.41e-12 ***
## Age               -0.01272    0.00809  -1.572  0.11588
## `Emp. Gender`Male -0.65115    0.15749  -4.135 3.56e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1254.0  on 1649  degrees of freedom
## Residual deviance: 1179.7  on 1645  degrees of freedom
## AIC: 1189.7
##
## Number of Fisher Scoring iterations: 5
```

```
## convert logit regression coefficients to odds ratios regression coefficients
exp(coef(mod_2))
```

```
##       (Intercept)   AppraisalRating         CareerSat               Age
##         5.1126015         0.9288725         0.6473188         0.9873609
## `Emp. Gender`Male
##         0.5214456
```

```
#### Q4.2
### predictions
## call data
reten_data <- reten_data %>%
        ## probability predictions
  mutate(mod_2_prob = fitted(mod_2),
        ## odds ratio predictions
        mod_2_or = exp(predict(mod_2)),
        ## logit predictions
        mod_2_log = predict(mod_2)
       )

### examine predictions
## call data
reten_data %>%
  ## select variables
  select(AppraisalRating, CareerSat, Age,
        `Emp. Gender`, LeaverStatus,
        mod_2_prob:mod_2_log) %>%
  ## arrange
  arrange(desc(mod_2_prob)) %>%
  ## print
  print(width = Inf)
```

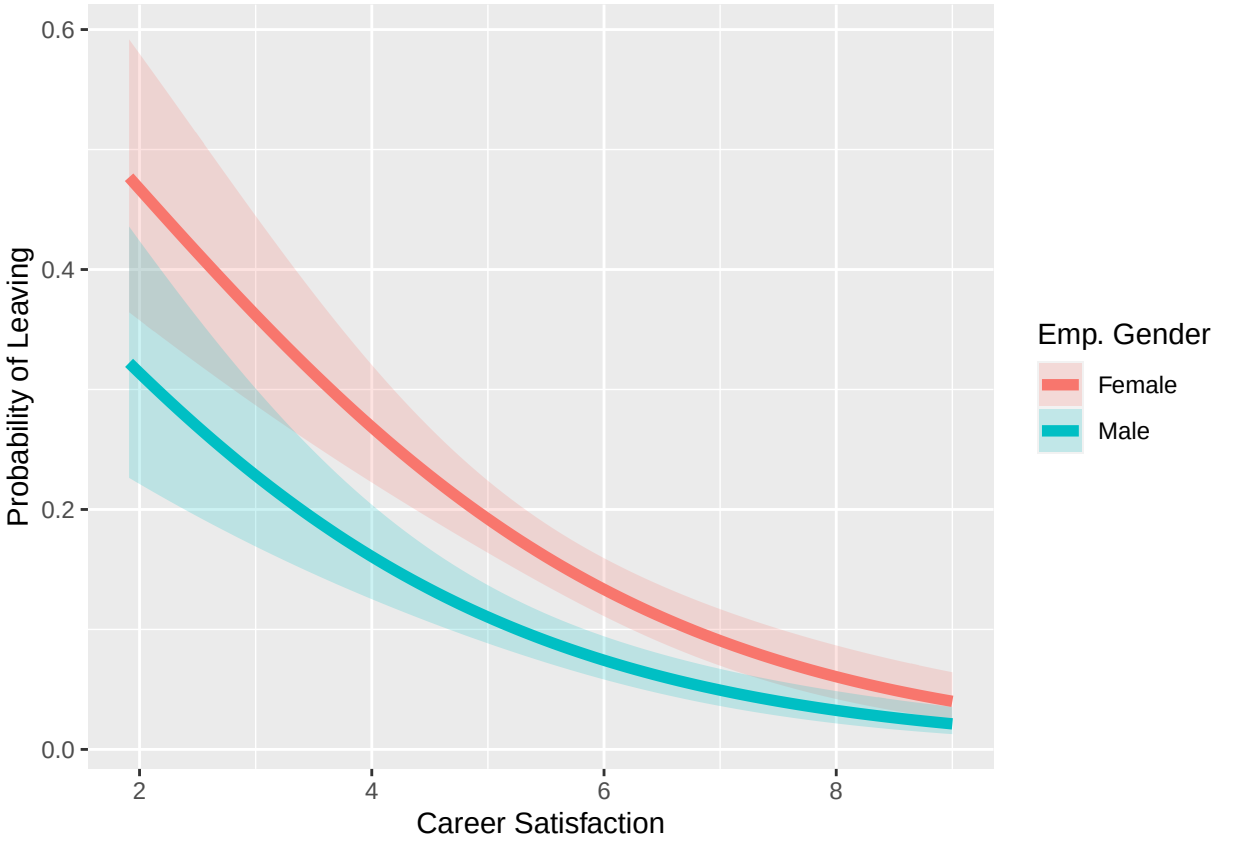
```
## # A tibble: 1,650 x 8
##    AppraisalRating CareerSat   Age `Emp. Gender` LeaverStatus mod_2_prob
##              <dbl>     <dbl> <dbl> <fct>         <fct>             <dbl>
##  1            4.43      2.55    31 Female        Leaver            0.451
```

```
##  2            6.24      2.43    25 Female        Leaver            0.449
##  3            5.6       2.63    26 Female        Stayer            0.436
##  4            3.23      3.17    23 Female        Leaver            0.431
##  5            4.72      2.85    29 Female        Leaver            0.419
##  6            4.7       2.86    32 Female        Leaver            0.410
##  7            6.31      2.75    27 Female        Stayer            0.408
##  8            2.81      3.57    26 Female        Leaver            0.387
##  9            4.37      2.91    40 Female        Leaver            0.386
## 10            6.69      2.93    26 Female        Leaver            0.385
##    mod_2_or mod_2_log
##       <dbl>     <dbl>
##  1    0.820    -0.198
##  2    0.816    -0.204
##  3    0.774    -0.256
##  4    0.757    -0.278
##  5    0.723    -0.325
##  6    0.694    -0.366
##  7    0.688    -0.373
##  8    0.632    -0.459
##  9    0.628    -0.465
## 10    0.627    -0.467
## # ... with 1,640 more rows
```

```
#### Q4.3
### accuracy of predictions
## name result and choose data
acc_mod_2 <- reten_data %>%
  ## summarize
            # true positives
  summarize(tp = sum(mod_2_prob >= 0.25 & LeaverStatus == "Leaver"),
            # true negatives
            tn = sum(mod_2_prob < 0.25 & LeaverStatus == "Stayer"),
            # false positives
            fp = sum(mod_2_prob >= 0.25 & LeaverStatus == "Stayer"),
            # false negatives
            fn = sum(mod_2_prob < 0.25 & LeaverStatus == "Leaver"))
## accuracy computations
acc_mod_2 %>%
  ## summarize
            # overall accuracy
  summarize(overall = (tp + tn)/(tp + tn + fp + fn),
            # positive accuracy
            positive = tp/(tp + fp),
            # negative accuracy
            negative = tn/(tn + fn),
            # sensitivity
            sensitivity = tp/(tp + fn),
            # specificity
            specificity = tn/(tn + fp))
```

```
## # A tibble: 1 x 5
##   overall positive negative sensitivity specificity
##     <dbl>    <dbl>    <dbl>       <dbl>       <dbl>
## 1   0.843    0.292    0.886       0.167       0.941
```

```r
#### Q4.4
### create new data for prediction
## name new data object
mod_2_pred_data <- with(reten_data,
  ## create data frame
  tibble(
    # mean of appraisal rating
    AppraisalRating = mean(AppraisalRating),
    # mean of age
    Age = mean(Age),
    # set career satisfaction
    CareerSat = rep(seq(min(CareerSat), max(CareerSat), length.out = 100), 2),
    # set employee gender
    `Emp. Gender` = factor(rep(c("Female", "Male"), each = 100))
  ))

### calculate probability bands
## overwrite data
mod_2_pred_data <- augment(mod_2, newdata = mod_2_pred_data, se_fit = TRUE) %>%
         ## Calculate probability
  mutate(prob = plogis(.fitted),
         # Calculate lower band probability
         prob_lower = plogis(.fitted - 1.96*.se.fit),
         # Calculate upper band probability
         prob_upper = plogis(.fitted + 1.96*.se.fit))

### plot
## choose data and mapping
ggplot(mod_2_pred_data, aes(x = CareerSat, y = prob)) +
  ## ribbon geometry for bands
  geom_ribbon(aes(ymin = prob_lower, ymax = prob_upper, fill = `Emp. Gender`),
              alpha = 0.2) +
  ## line geometry for predictions
  geom_line(aes(color = `Emp. Gender`), size = 2) +
  ## axes labels
  labs(x = "Career Satisfaction", y = "Probability of Leaving")
```



Task 5: Moderated Logistic Regression

Estimate a moderated logistic regression where *appraisal rating*, *career satisfaction*, and their interaction predict *employee retention* using the correct variables. Name the model: **mod_3**. Make sure to mean center *appraisal rating* and *career satisfaction* before estimating the model.

Question 5.1: What is the estimate of the interaction effect of *career satisfaction* and *appraisal rating* on *employee retention*?

Response 5.1: *0.14251.*

Use **interact_plot** to visualize the interaction effect from **mod_3**. Use **sim_slopes** to calculate the simple slopes.

Question 5.2: For individuals with low levels of *appraisal rating*, do individuals with low or high levels of *career satisfaction* have a higher *probability of leaving*? What is the simple slope estimate between *appraisal rating* and *employee retention* for an individual who is one standard deviation above the mean on *career satisfaction*?

Response 5.2: *Individuals with low levels of career satisfaction have a higher probability of leaving for low levels of appraisal ratings. Simple slope estimate: 0.13.*

Calculate the *logit*, *odds ratio*, and *probability* predictions from **mod_3** and save them to **reten_data**. Select the relevant variables from **reten_data** and arrange it by *descending predicted probability*.

Question 5.3: What is the *highest probability* of someone leaving in this data based on **mod_3**? What is the predicted *logit* for the fourth individual in the list?

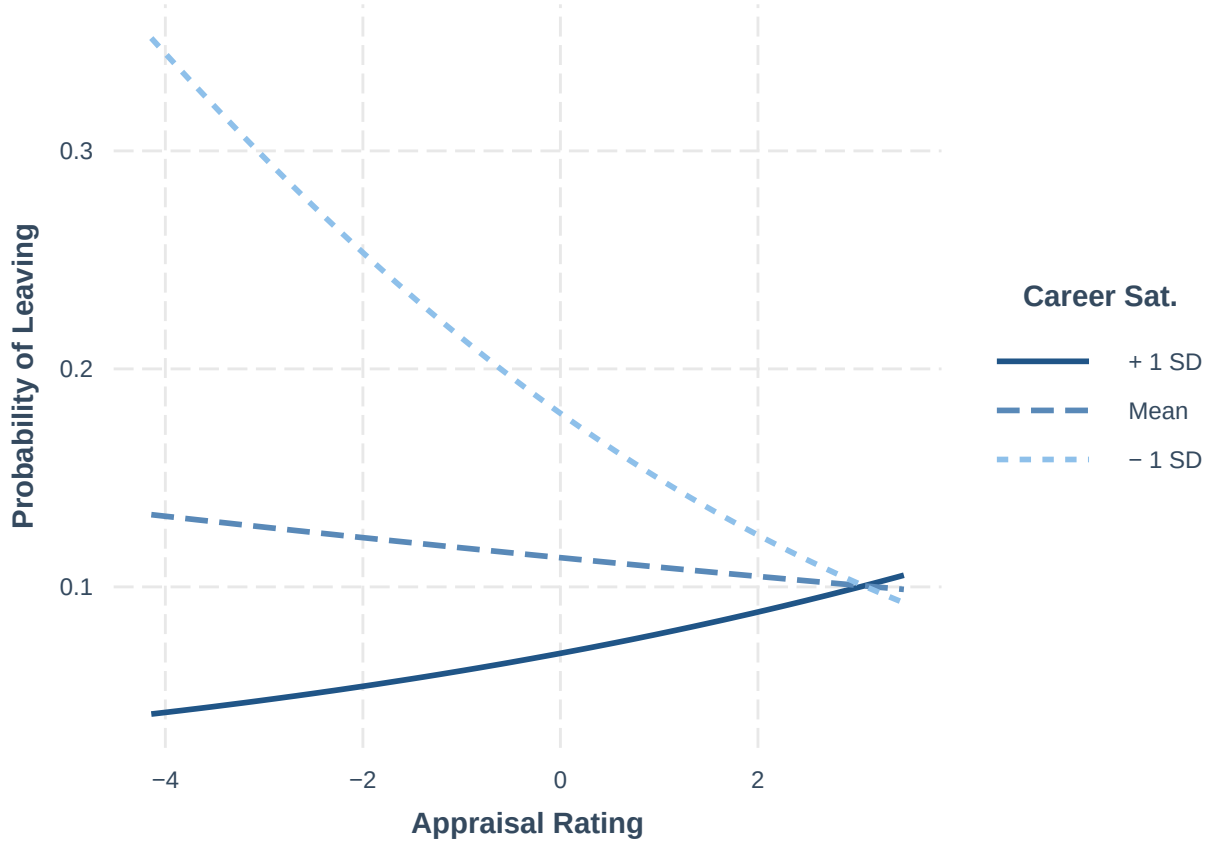
Response 5.3: *0.506. -0.292.*

```r
#### Q5.1
### create centered variables
## overwrite data
reten_data <- reten_data %>%
  ## center variables
  mutate_at(vars(AppraisalRating, CareerSat), list(cent = ~ . - mean(.)))

### estimate moderated logistic regression model
mod_3 <- glm(LeaverStatus ~ AppraisalRating_cent * CareerSat_cent,
             family = "binomial", data = reten_data)

## examine results
summary(mod_3)
```

```
##
## Call:
## glm(formula = LeaverStatus ~ AppraisalRating_cent * CareerSat_cent,
##     family = "binomial", data = reten_data)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.0220  -0.5433  -0.4598  -0.3702   2.6249
##
## Coefficients:
##                                   Estimate Std. Error z value Pr(>|z|)
## (Intercept)                       -2.05652    0.08185 -25.126  < 2e-16 ***
## AppraisalRating_cent              -0.04417    0.06509  -0.679  0.49742
## CareerSat_cent                    -0.43786    0.06391  -6.851 7.32e-12 ***
## AppraisalRating_cent:CareerSat_cent  0.14251    0.05310   2.684  0.00728 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1254.0  on 1649  degrees of freedom
## Residual deviance: 1193.3  on 1646  degrees of freedom
## AIC: 1201.3
##
## Number of Fisher Scoring iterations: 5
```

```r
#### Q5.2
### visualize interaction
## call plot and model
interact_plot(mod_3,
  # x-axis variable
  pred = AppraisalRating_cent,
  # moderator variable
  modx = CareerSat_cent,
  # x-axis label
  x.label = "Appraisal Rating",
  # y-axis label
  y.label = "Probability of Leaving",
  # legend label
```

```
  legend.main = "Career Sat.")
```




```
## simple slopes
sim_slopes(mod_3,
  # x-axis variable
  pred = AppraisalRating_cent,
  # moderator variable
  modx = CareerSat_cent)
```

```
## JOHNSON-NEYMAN INTERVAL
##
## When CareerSat_cent is OUTSIDE the interval [-0.54, 2.80], the slope of
## AppraisalRating_cent is p < .05.
##
## Note: The range of observed values of CareerSat_cent is [-3.85, 3.24]
##
## SIMPLE SLOPES ANALYSIS
##
## Slope of AppraisalRating_cent when CareerSat_cent = -1.227565e+00 (- 1 SD):
##
##    Est.   S.E.   z val.      p
## ------- ------ -------- ------
##   -0.22   0.08    -2.90   0.00
##
## Slope of AppraisalRating_cent when CareerSat_cent =  3.959123e-16 (Mean):
```

```
##
##    Est.   S.E.   z val.      p
## ------- ------ -------- ------
##   -0.04   0.07    -0.68   0.50
##
## Slope of AppraisalRating_cent when CareerSat_cent =  1.227565e+00 (+ 1 SD):
##
##   Est.   S.E.   z val.      p
## ------ ------ -------- ------
##   0.13   0.11     1.23   0.22
```

```
#### Q5.3
### predictions
## call data
reten_data <- reten_data %>%
         ## probability predictions
  mutate(mod_3_prob = fitted(mod_3),
         ## odds ratio predictions
         mod_3_or = exp(predict(mod_3)),
         ## logit predictions
         mod_3_log = predict(mod_3)
        )

### examine predictions
## call data
reten_data %>%
  ## select variables
  select(CareerSat_cent, AppraisalRating_cent, LeaverStatus,
         mod_3_prob:mod_3_log) %>%
  ## arrange
  arrange(desc(mod_3_prob)) %>%
  ## print
  print(width = Inf)
```

```
## # A tibble: 1,650 x 6
##    CareerSat_cent AppraisalRating_cent LeaverStatus mod_3_prob mod_3_or
##             <dbl>                <dbl> <fct>             <dbl>    <dbl>
##  1          -2.59               -2.29  Leaver            0.506    1.02
##  2          -3.21               -1.09  Leaver            0.474    0.900
##  3          -2.19               -2.71  Leaver            0.466    0.874
##  4          -2.85               -1.15  Leaver            0.428    0.747
##  5          -2.89               -0.912 Stayer            0.407    0.686
##  6          -2.90               -0.822 Leaver            0.398    0.662
##  7          -1.82               -2.79  Stayer            0.398    0.660
##  8          -2.91               -0.802 Leaver            0.397    0.660
##  9          -2.36               -1.54  Stayer            0.392    0.645
## 10          -2.07               -2.10  Stayer            0.392    0.644
##    mod_3_log
##        <dbl>
##  1    0.0220
##  2   -0.105
##  3   -0.134
##  4   -0.292
##  5   -0.377
```

```
##  6   -0.413
##  7   -0.415
##  8   -0.416
##  9   -0.439
## 10   -0.440
## # ... with 1,640 more rows
```