

HW 5

Enter your name and EID here: Austine Do (ahd589)

You will submit this homework assignment as a pdf file on Gradescope.

For all questions, include the R commands/functions that you used to find your answer (show R chunk). Answers without supporting code will not receive credit. Write full sentences to describe your findings.

Part 1

We will work with data from the following article:

Hickey, W. (2007). The Ultimate Halloween Candy Power Ranking. FiveThirtyEight.

<https://fivethirtyeight.com/videos/the-ultimate-halloween-candy-power-ranking/>

[\(https://fivethirtyeight.com/videos/the-ultimate-halloween-candy-power-ranking/\)](https://fivethirtyeight.com/videos/the-ultimate-halloween-candy-power-ranking/)




















```
# Upload data from github
candy <- read_csv("https://raw.githubusercontent.com/laylaguyot/datasets/main//Halloween-
candy.csv")

# Take a quick look
head(candy)
```

```
## # A tibble: 6 × 13
##   competitorname chocolate fruity caramel peanutyalmondy nougat crispedricewafer
##   <chr>           <dbl> <dbl> <dbl>           <dbl> <dbl>           <dbl>
## 1 100 Grand         1     0     1             0     0             1
## 2 3 Musketeers      1     0     0             0     1             0
## 3 One dime         0     0     0             0     0             0
## 4 One quarter      0     0     0             0     0             0
## 5 Air Heads        0     1     0             0     0             0
## 6 Almond Joy       1     0     0             1     0             0
## # i 6 more variables: hard <dbl>, bar <dbl>, pluribus <dbl>,
## #   sugarpercent <dbl>, pricepercent <dbl>, winpercent <dbl>
```

This dataset is the result of an experiment: “Pit dozens of fun-sized candy varieties against one another, and let the wisdom of the crowd decide which one was best. While we don’t know who exactly voted, we do know this: 8,371 different IP addresses voted on about 269,000 randomly generated matchups.”

Here are the top 19 winners:

RK	CANDY	WIN PERCENTAGE
1	Reese's Peanut Butter Cup	84.2% 
2	Reese's Miniatures	81.9 
3	Twix	81.6 
4	Kit Kat	76.8 
5	Snickers	76.7 
6	Reese's Pieces	73.4 
7	Milky Way	73.1 
8	Reese's Stuffed With Pieces	72.9 
9	Peanut Butter M&M's	71.5 
10	Butterfinger	70.7 
11	Peanut M&M's	69.5 
12	3 Musketeers	67.6 
13	Starburst	67.0 
14	100 Grand	67.0 
15	M&M's	66.6 
16	Crunch	66.5 
17	Rolo	65.7 
18	Milky Way Simply Caramel	64.4 
19	Skittles original	63.1 

Question 1: (2 pts)

How many rows are there in the `candy` dataset? How many columns? What does one row in `candy` represent?

Visit the data dictionary on the following GitHub page and describe each variable as numeric or categorical:

<https://github.com/fivethirtyeight/data/tree/master/candy-power-ranking>

(<https://github.com/fivethirtyeight/data/tree/master/candy-power-ranking>)

```
# structure of `candy` dataset
str(candy)
```

```
## spc_tbl_ [85 × 13] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
## $ competitorname : chr [1:85] "100 Grand" "3 Musketeers" "One dime" "One quarter"
## ...
## $ chocolate      : num [1:85] 1 1 0 0 0 1 1 0 0 0 ...
## $ fruity         : num [1:85] 0 0 0 0 1 0 0 0 0 1 ...
## $ caramel        : num [1:85] 1 0 0 0 0 0 1 0 0 1 ...
## $ peanutyalmondy : num [1:85] 0 0 0 0 0 1 1 1 0 0 ...
## $ nougat         : num [1:85] 0 1 0 0 0 0 1 0 0 0 ...
## $ crispedricewafer: num [1:85] 1 0 0 0 0 0 0 0 0 0 ...
## $ hard           : num [1:85] 0 0 0 0 0 0 0 0 0 0 ...
## $ bar            : num [1:85] 1 1 0 0 0 1 1 0 0 0 ...
## $ pluribus       : num [1:85] 0 0 0 0 0 0 0 1 1 0 ...
## $ sugarpercent    : num [1:85] 0.732 0.604 0.011 0.011 0.906 ...
## $ pricepercent    : num [1:85] 0.86 0.511 0.116 0.511 0.511 ...
## $ winpercent      : num [1:85] 67 67.6 32.3 46.1 52.3 ...
## - attr(*, "spec")=
## .. cols(
## ..   competitorname = col_character(),
## ..   chocolate = col_double(),
## ..   fruity = col_double(),
## ..   caramel = col_double(),
## ..   peanutyalmondy = col_double(),
## ..   nougat = col_double(),
## ..   crispedricewafer = col_double(),
## ..   hard = col_double(),
## ..   bar = col_double(),
## ..   pluribus = col_double(),
## ..   sugarpercent = col_double(),
## ..   pricepercent = col_double(),
## ..   winpercent = col_double()
## .. )
## - attr(*, "problems")=<externalptr>
```

There are 85 columns and 13 columns. Each one of the rows in the candy dataset represents an individual candy and its associated characteristics. Chocolate, fruity, caramel, peanutyalmondy, nougat, crispedricewafer, hard, bar, pluribus are all categorical variables. Sugarpercent, pricepercent, and winpercent are all numerical variables.

Question 2: (3 pts)

Fit a linear regression model that uses the sugar percentile to predict the win percentage of a candy.

```
# linear model using `sugarpercentile` to predict `winpercentage` of the candy
lin_model_sp <- lm(winpercent ~ sugarpercent, data = candy)
summary(lin_model_sp)
```

```
##
## Call:
## lm(formula = winpercent ~ sugarpercent, data = candy)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -24.924 -11.066  -1.168   9.252  36.851
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   44.609      3.086   14.455  <2e-16 ***
## sugarpercent   11.924      5.560    2.145   0.0349 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 14.41 on 83 degrees of freedom
## Multiple R-squared:  0.05251,    Adjusted R-squared:  0.04109
## F-statistic:  4.6 on 1 and 83 DF,  p-value: 0.0349
```

Write the expression of the model:

winpercent = 11.924 * sugarpercent + 44.609

Predict the win percentage for your favorite candy in this dataset. Calculate and interpret its residual:

```
# Twix is my favorite candy so I will use the model to predict its win percentage
twix <- data.frame(sugarpercent = 0.546)
predict_twin_win_percent <- predict(lin_model_sp, newdata = twix)
actual_twix_win_percent <- as.numeric(candy[candy$competitorname == 'Twix', "winpercent"])

actual_twix_win_percent - predict_twin_win_percent
```

```
##      1
## 30.52304
```

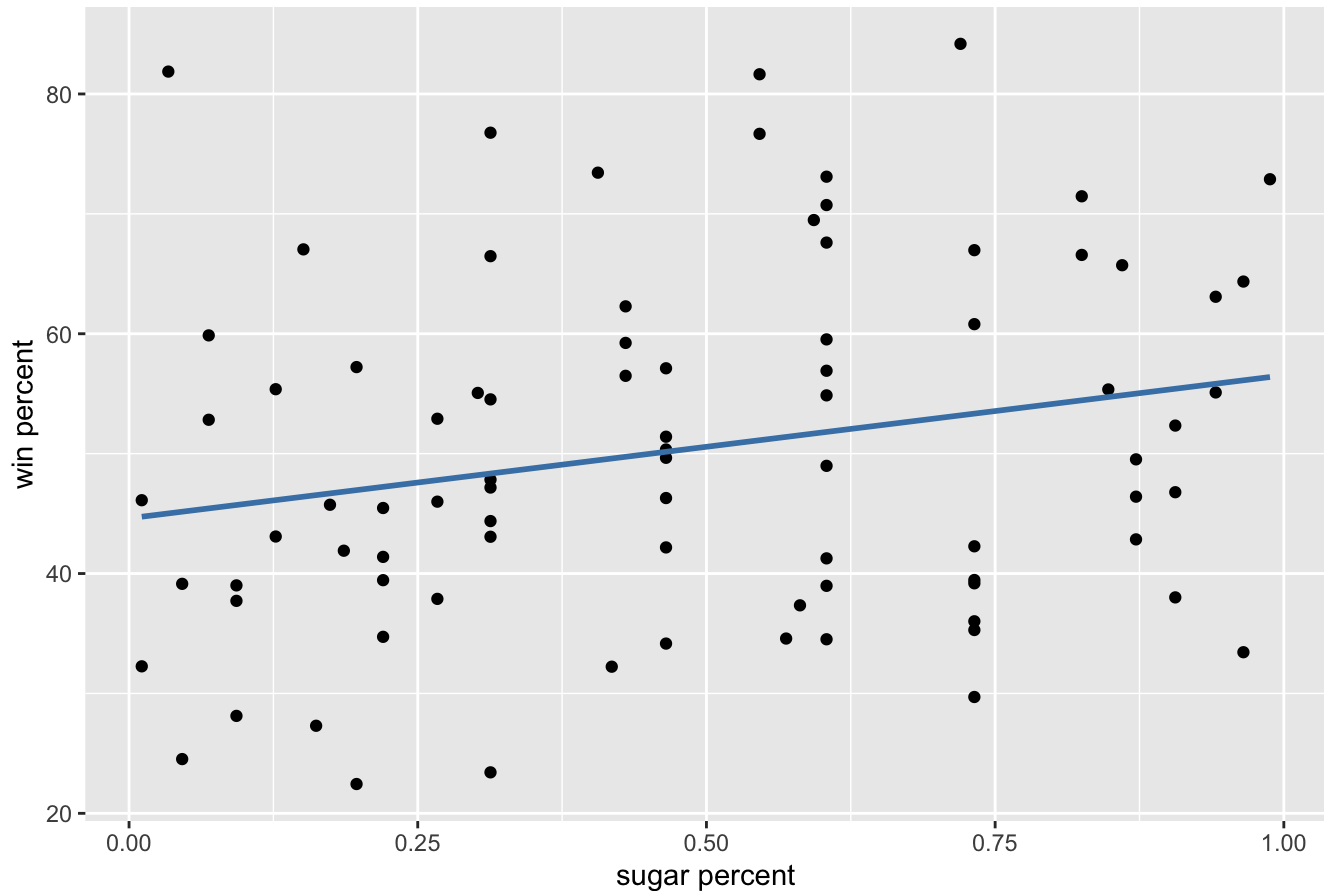
The calculated residual indicates that the observed win percent for Twix is higher than what the model based on sugar percent predicts it to be

Make a visualization to represent the linear regression model for the sugar percentile to predict the win percentage. Is there a strong relationship between the two variables?

```
# Plot of sugar percent and win percent with linear model
candy |>
  ggplot(aes(x = sugarpercent, y = winpercent)) +
    geom_point() +
    geom_smooth(method = "lm", se = FALSE, color = "steelblue", size = 1) +
    labs(title = 'Plot of sugar percent and win percent with linear model',
         x = 'sugar percent',
         y = 'win percent')
```

```
## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use `linewidth` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

Plot of sugar percent and win percent with linear model



No there does not seem to be a strong relationship between the 2 variables

Report two metrics to evaluate the performance of the model (no interpretation needed for now):

```
# RSME and Adjusted R-squared of the linear model with sugarpercent as the predictor
sqrt(mean(resid(lin_model_sp)^2))
```

```
## [1] 14.23832
```

```
summary(lin_model_sp)$adj.r.squared
```

```
## [1] 0.04109448
```

Question 3: (3 pts)

Choose a categorical predictor and fit a linear regression model to predict the win percentage of a candy only based on this categorical predictor.

```
# Linear model with chocolate as the predictor variable
lin_model_chocolate <- lm(winpercent ~ chocolate, data = candy)
summary(lin_model_chocolate)
```

```
##
## Call:
## lm(formula = winpercent ~ chocolate, data = candy)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -26.1995  -7.5633  -0.2379   8.5623  24.8954
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   42.142      1.648  25.574 < 2e-16 ***
## chocolate     18.779      2.498   7.519 5.86e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 11.42 on 83 degrees of freedom
## Multiple R-squared:  0.4052, Adjusted R-squared:  0.398
## F-statistic: 56.53 on 1 and 83 DF,  p-value: 5.86e-11
```

Write the expression of the model:

winpercent = 18.779 * chocolate + 42.142

Predict the win percentage for each category. What do these predicted values represent?

```
# Prediction of candy with and without chocolate
candy_chocolate <- data.frame(chocolate = 1)
candy_no_chocolate <- data.frame(chocolate = 0)

predict(lin_model_chocolate, newdata = candy_chocolate)
```

```
##      1
## 60.92153
```

```
predict(lin_model_chocolate, newdata = candy_no_chocolate)
```

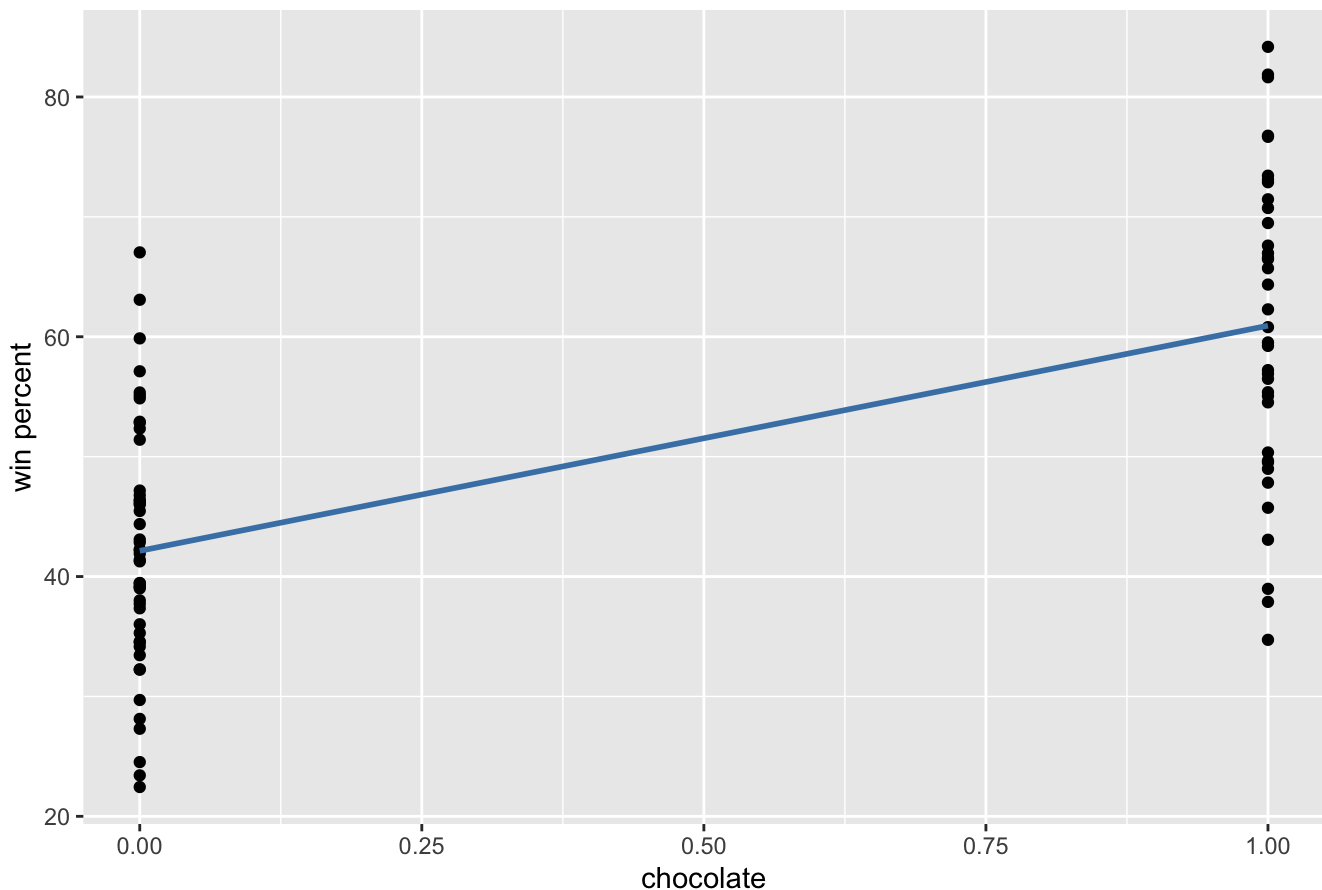
```
##      1
## 42.14226
```

The predicted values represent the win percentage of a candy with or without chocolate. A candy with chocolate has a 60.9 win percent while a candy without chocolate has a 42.1 win percent.

Make a visualization to represent the linear regression model for the categorical variable of your choice to predict the win percentage. Is there a strong difference in win percentage between the different categories?

```
# Plot of chocolate and win percent with linear model
candy |>
  ggplot(aes(x = chocolate, y = winpercent)) +
    geom_point() +
    geom_smooth(method = "lm", se = FALSE, color = "steelblue", size = 1) +
    labs(title = 'Plot of sugar percent and win percent with linear model',
         x = 'chocolate',
         y = 'win percent')
```

Plot of sugar percent and win percent with linear model



It seems like there is a difference between candies that have chocolate versus candies that don't have chocolate but I don't know if I would call it a strong difference since the win percent distribution between candies that don't have chocolate and candies that do overlaps a lot.

Report two metrics to evaluate the performance of the model. Has the performance improved compared to the model with the sugar percentile? Justify.

```
# RSME and Adjusted R-squared of linear model with sugar percent predictor
sqrt(mean(resid(lin_model_sp)^2))
```

```
## [1] 14.23832
```

```
summary(lin_model_sp)$adj.r.squared
```

```
## [1] 0.04109448
```

```
sqrt(mean(resid(lin_model_chocolate)^2))
```

```
## [1] 11.28168
```

```
summary(lin_model_chocolate)$adj.r.squared
```

```
## [1] 0.3979867
```

Yes, the performance has improved across both RSME and Adjusted R-squared, jumping from 14.23 to 11.28 and 0.04 to 0.39 across RSME and Adjusted R-Squared respectively.

Question 4: (3 pts)

Fit a linear regression model that uses all the predictors that make sense to predict the win percentage of a candy.

```
# Linear model with all variables excluding competitorname
lin_model_all_vars <- lm(winpercent ~ chocolate + fruity + caramel +
                        peanutyalmondy + nougat + crispedricewafer +
                        hard + bar + pluribus + sugarpercent + pricepercent,
                        data = candy)
summary(lin_model_all_vars)
```



```
##
## Call:
## lm(formula = winpercent ~ chocolate + fruity + caramel + peanutyalmondy +
##      nougat + crispedricewafer + hard + bar + pluribus + sugarpercent +
##      pricepercent, data = candy)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -20.2244  -6.6247   0.1986   6.8420  23.8680
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    34.5340     4.3199   7.994 1.44e-11 ***
## chocolate      19.7481     3.8987   5.065 2.96e-06 ***
## fruity          9.4223     3.7630   2.504 0.01452 *
## caramel         2.2245     3.6574   0.608 0.54493
## peanutyalmondy 10.0707     3.6158   2.785 0.00681 **
## nougat          0.8043     5.7164   0.141 0.88849
## crispedricewafer 8.9190     5.2679   1.693 0.09470 .
## hard          -6.1653     3.4551  -1.784 0.07852 .
## bar             0.4415     5.0611   0.087 0.93072
## pluribus       -0.8545     3.0401  -0.281 0.77945
## sugarpercent    9.0868     4.6595   1.950 0.05500 .
## pricepercent   -5.9284     5.5132  -1.075 0.28578
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 10.7 on 73 degrees of freedom
## Multiple R-squared:  0.5402, Adjusted R-squared:  0.4709
## F-statistic: 7.797 on 11 and 73 DF, p-value: 9.504e-09
```

Report two metrics to evaluate the performance of the model. Has the performance improved compared to the model with the previous models? Justify.

```
# RSME and Adjusted R-squared of linear model of all relevant variables
sqrt(mean(resid(lin_model_all_vars)^2))
```

```
## [1] 9.918633
```

```
summary(lin_model_all_vars)$adj.r.squared
```

```
## [1] 0.4709252
```

Yes the model did improve against both the previous models since the RSME is lower and the Adjusted R-squared is higher against both models.

Using all potential predictors to predict an outcome can make our model too specific to our data so it does not perform very well when we add new data. Consider the new candy below:

```
# Add new data
newcandy <- data.frame(chocolate = 1, fruity = 1, caramel = 1,
                      peanutyalmondy = 1, nougat = 1, crispedricewafer = 1,
                      hard = 1, bar = 1, pluribus = 1,
                      sugarpercent = 0.5, pricepercent = 0.5, winpercent = 20)
```

Predict the value of its win percent based on the model with all predictors that make sense. How could we be so much in error?

```
# Predicts newcandy win percent
predict(lin_model_all_vars, newdata = newcandy)
```

```
##           1
## 80.72375
```

A potential reason why our predicted value has a very high residual/error is that the predictors in our linear regression model may not have a linear relationship with our outcome variable, leading to high error in the predicted values. There could many other reasons causing our model to produce such an error.

Part 2

Let's analyze the overall sentiment of a Wikipedia page!

Question 5: (2 pts)

Choose a Wikipedia page you would like to explore. We will retrieve the text content from this Wikipedia page using the `rvest` package. Modify the following code to retrieve the text content from the page that you chose:

```
# Wikipedia page
wikipedia_page <- read_html("https://en.wikipedia.org/wiki/Data_science")

# Retrieve text content
wikipedia_text <- data.frame(text =
  wikipedia_page |>
  html_nodes("p") |>
  html_text())
```

Why did you choose that page?

I chose this site because its about data science and I am interested in a future in the data science field!

Question 6: (4 pts)

Using the text of the Wikipedia page, `wikipedia_text`, tokenize the text content into words. Then match these words with a sentiment from the `nrc` (National Research Council) lexicon. Finally, find the top 5 sentiments that occurred the most in this Wikipedia page.

```
# Table of top 5 sentiments on the Wikipedia page
wikipedia_text |>
  unnest_tokens(input = text, output = word) -> wiki_words

wiki_words |>
  inner_join(get_sentiments('nrc'), by = 'word') |>
  group_by(sentiment) |>
  summarize(frequency = n()) |>
  arrange(desc(frequency))
```

```
## Warning in inner_join(wiki_words, get_sentiments("nrc"), by = "word"): Detected an un
expected many-to-many relationship between `x` and `y`.
## i Row 6 of `x` matches multiple rows in `y`.
## i Row 10937 of `y` matches multiple rows in `x`.
## i If a many-to-many relationship is expected, set `relationship =
##   "many-to-many"` to silence this warning.
```

```
## # A tibble: 10 × 2
##   sentiment      frequency
##   <chr>          <int>
## 1 positive         98
## 2 trust            53
## 3 anticipation     21
## 4 negative        14
## 5 fear             7
## 6 surprise         7
## 7 joy              5
## 8 anger            4
## 9 disgust          3
## 10 sadness         3
```

Describe the overall sentiment in the Wikipedia page you chose:

The overall sentiment of the Wikipedia page I chose seems to be of both positivity, trust, and anticipation and negativity and fear.

Question 7: (2 pts)

What are some limitations of conducting sentiment analysis? Discuss at least two reasons why we should be careful when interpreting the results of sentiment analysis.

Some limitation of performing sentiment analysis is that language is highly context dependent which sentiment analysis may struggle to understand and sentiment is also highly subjective to each person and model. For these reasons we should be careful when using sentiment analysis to interpret text/language as it will not grasp all the nuance contained in human language.

Formatting: (1 pt)

Knit your file! You can knit into html and once it knits in html, click on `Open in Browser` at the top left of the window that pops out. **Print** your html file into pdf from your browser.

Is it working? If not, try to decipher the error message: look up the error message, consult websites such as stackoverflow (<https://stackoverflow.com/>) or crossvalidated (<https://stats.stackexchange.com/>).

Finally, remember to select pages for each question when submitting your pdf to Gradescope.