STAT 505: Final Exam
Name: Connor Demorest

Please turn in the exam to GitHub and include the R Markdown code and a PDF or Word file with output. Please verify that all of the code has compiled and the graphics look like you think they should on your files, you are welcome to upload image files directly if they look distorted in the output file.

While the exam is open book and you can use any resources from class or freely available on the internet, this is strictly an individual endeavor and **you should not discuss the problems with anyone outside the course instructor including group mates or class members.** All resources, including websites, should be acknowledged.

For full credit, include your code and graphics for each question and create neat output by using options like `kable()` for tables and writing results in line with R commands.

Short Answer Questions (16 points)

For questions in this section, keep your answers concise. You are welcome to use a combination of prose, math, and pseudocode, but your responses should be well thought out and defended.

1. (4 points)

Detail the process for conducting a posterior predictive check and then describe how they can be used for assessing model fit.

A posterior predictive check can be done by comparing plots of the density of the observed data with the posterior distribution and looking for differences between the two groups. If the observed data does not look similar to the data found in the posterior distribution, then there could be a problem with the modeling scheme or the priors used.

2. (4 points)

Make an argument (to a collaborator) for centering and/or standardizing continuous predictors.

When a predictor is standardized, the interpretation of a 1 unit increase changes from the scale of the predictor to a different scale, usually to a normal distribution where a 1 unit increase in the predictor is transformed into a 1 SD increase in the predictor. When a predictor is centered, the default value of the predictor changes from 0 (which may not make much sense, e.g. a house with 0 square feet) to a different default value, typically the mean (a house with the mean amount of square feet, or a specified value as default).

3. (4 points)

Why should inferences about sampling units be characterized as differences in predictors *between* units rather than differences in predictors *within* a sampling unit?

When you want to use inference, you want to learn something that is unknown using data that you've collected as a way to give evidence. The predictors themselves are not of interest usually, but their effect on the outcome is. So the differences between predictors within a single sampling unit is not what the goal of inference is, but instead you want to characterize what the differences in predictors mean for the new observation and that outcome.

I don't really understand what this question is asking and I wish I would have asked about it earlier in the week before I left for Siberia (aka Eastern MT) last Thursday. (My grandparents don't have internet at their farm.) I guess I'm hoping you're keeping that in mind when you grade this question and have no idea what my answer is saying :). I feel like I just rephrased the question without answering it and I'm just wasting your time by putting an answer at all.

4. (4 points)

Consider the distribution of an expected outcome given a set of predictors and the distribution of a new observation given a set of predictors. Describe how the point estimate and uncertainty would differ (or not) for the two situations.

A distribution of a predicted outcome given predictors is different than a distribution of a predicted observation. For the same predictor values, the point estimates will be the same for both the distribution and the new observation, but the amount of uncertainty around the point estimates will be different. The uncertainty in a point estimate for a new observation will be greater.

Code Interpretation (16 points)

For this question, we will use a subset of a dataset that contains Indian recipes.

```
indian_food <- read_csv('https://raw.githubusercontent.com/stat408/final_exam/master/indian.csv') %>%
  filter(course != 'starter') %>%
  select(-ingredients, -diet, -region) %>%
  mutate(flavor_profile = factor(flavor_profile), course = factor(course))
```

```
## Parsed with column specification:
## cols(
##   name = col_character(),
##   ingredients = col_character(),
##   diet = col_character(),
##   prep_time = col_double(),
##   cook_time = col_double(),
##   flavor_profile = col_character(),
##   course = col_character(),
##   region = col_character()
## )
```

```
summary(indian_food)
```

```
##      name              prep_time        cook_time      flavor_profile
##  Length:179         Min.   :  5.00   Min.   :  5.00   spicy:106
##  Class :character   1st Qu.: 10.00   1st Qu.: 25.00   sweet: 73
##  Mode  :character   Median : 10.00   Median : 30.00
##                     Mean   : 34.76   Mean   : 41.32
##                     3rd Qu.: 20.00   3rd Qu.: 45.00
##                     Max.   :500.00   Max.   :720.00
##          course
##  dessert    :70
##  main course:81
##  snack      :28
##
##
##
```

1. (4 points)

Using the following model specification write out the complete linear model and define all of the coefficients in the model.

```
model_specification <- formula(cook_time ~ prep_time + flavor_profile + course)
model_specification
```

```
## cook_time ~ prep_time + flavor_profile + course
```

$cook_time = \beta_0 + \beta_1 prep_time + \beta_2 flavor_profile = Sweet + \beta_3 course = MainCourse + \beta_4 course = Snack + \epsilon$
where:

β_0 *is the estimated cook time for a meal with meal with 0 minutes of prep time, spicy flavor profile, and dessert course.*

β_1 *is the increase in estimated cook time due to a 1 unit increase in prep time*

β_2 *is the increase in estimated cook time for foods with a sweet flavor profile*

β_3 *is the increase in estimated cook time for foods that are a main course*

β_4 is the increase in estimated cook time for foods that are a snack

ϵ is the error between the actual cook time and the estimated cook time and is $\sim N(0, \sigma^2)$

2. (4 points)

Interpret the results.

```
lm(model_specification, data = indian_food) %>% summary()
```

```
##
## Call:
## lm(formula = model_specification, data = indian_food)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -60.43 -16.52  -6.99   3.76 673.48
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)           37.12660   32.91436   1.128   0.2609
## prep_time              0.09342    0.05583   1.673   0.0961 .
## flavor_profilesweet    8.46149   32.33476   0.262   0.7939
## coursemain course     -1.81583   32.37908  -0.056   0.9553
## coursesnack          -10.76795   34.77966  -0.310   0.7572
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 54.81 on 174 degrees of freedom
## Multiple R-squared:  0.02858,    Adjusted R-squared:  0.006252
## F-statistic:  1.28 on 4 and 174 DF,  p-value: 0.2797
```

There is not enough evidence to conclude that the model fits the data significantly better than the naive model ($F = 1.28$ on 4 df1 and 174 df2, $p = 0.280$). The R^2 is 0.0286, so only 2.86% of the variability in the data is explained by the model. None of the predictors are statistically significant at the $\alpha = 0.05$ level, which indicates that there is no association between each predictor and the outcome after accounting for the other predictors.

The estimated increase in cook time for a 1 minute increase in prep time is 0.09 minutes (95% CI = -0.018, 0.205) when accounting for other predictors. The estimated increase in cook time for a food that sweet instead of spicy is 8.46 minutes (95% CI: -56.2, 73.1) when accounting for other predictors. The estimated increase in cook time for a main course instead of a dessert is -1.82 minutes (95% CI: -66.6, 62.9) when accounting for other predictors. The estimated increase in cook time for snack compared to a dessert is -10.8 minutes (95% CI: -80.4, 58.8) when accounting for other predictors.

This model isn't significantly better than a model that just uses the overall mean of all the cook times and none of the predictors are associated with a change in cook time when accounting for the other predictors.

3. (4 points)

Interpret the results.

```
stan_glm(model_specification, data = indian_food, refresh = 0) %>% print(digits = 2)
```

```
## stan_glm
##  family:       gaussian [identity]
##  formula:      cook_time ~ prep_time + flavor_profile + course
##  observations: 179
```

```
##  predictors:   5
## ------
##                     Median MAD_SD
## (Intercept)          36.90  32.60
## prep_time             0.09   0.05
## flavor_profilesweet   8.74  31.85
## coursemain course    -1.54  31.81
## coursesnack         -10.67  33.93
##
## Auxiliary parameter(s):
##       Median MAD_SD
## sigma 55.01   2.93
##
## ------
## * For help interpreting the printed output see ?print.stanreg
## * For info on the priors used see ?prior_summary.stanreg
```

The effect of a 1 minute increase in prep time on cook time is an increase of 0.09 minutes (95% interval: -0.01, 0.19) when controlling for flavor and course. The effect of the food being sweet instead of spicy is 7.36 minute increase (95% interval: -54.4, 69.12) when controlling for prep time and course. The change in cook time of a food being a main course instead of a dessert is -3.00 minutes (95% interval: -65.08, 59.08) when controlling for prep time and flavor profile. The change in cook time of a food being a snack instead of a dessert is -11.85 minutes (95% interval: -80.23, 56.53) when controlling for prep time and flavor profile.

All of the 95% intervals contain 0, so there is not much evidence that the predictors are useful for predicting the cook time of the foods.

4. (4 points)

What is the problem with trying to fit this model?

```
lm(cook_time ~ prep_time + flavor_profile + course + flavor_profile:course,
   data = indian_food) %>% summary()
```

```
##
## Call:
## lm(formula = cook_time ~ prep_time + flavor_profile + course +
##     flavor_profile:course, data = indian_food)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -60.43 -16.52  -6.99   3.76 673.48
##
## Coefficients: (2 not defined because of singularities)
##                                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)                           37.12660   32.91436   1.128   0.2609
## prep_time                              0.09342    0.05583   1.673   0.0961 .
## flavor_profilesweet                    8.46149   32.33476   0.262   0.7939
## coursemain course                     -1.81583   32.37908  -0.056   0.9553
## coursesnack                          -10.76795   34.77966  -0.310   0.7572
## flavor_profilesweet:coursemain course        NA         NA      NA       NA
## flavor_profilesweet:coursesnack              NA         NA      NA       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 54.81 on 174 degrees of freedom
```

```
## Multiple R-squared:  0.02858,    Adjusted R-squared:  0.006252
## F-statistic:  1.28 on 4 and 174 DF,  p-value: 0.2797
```

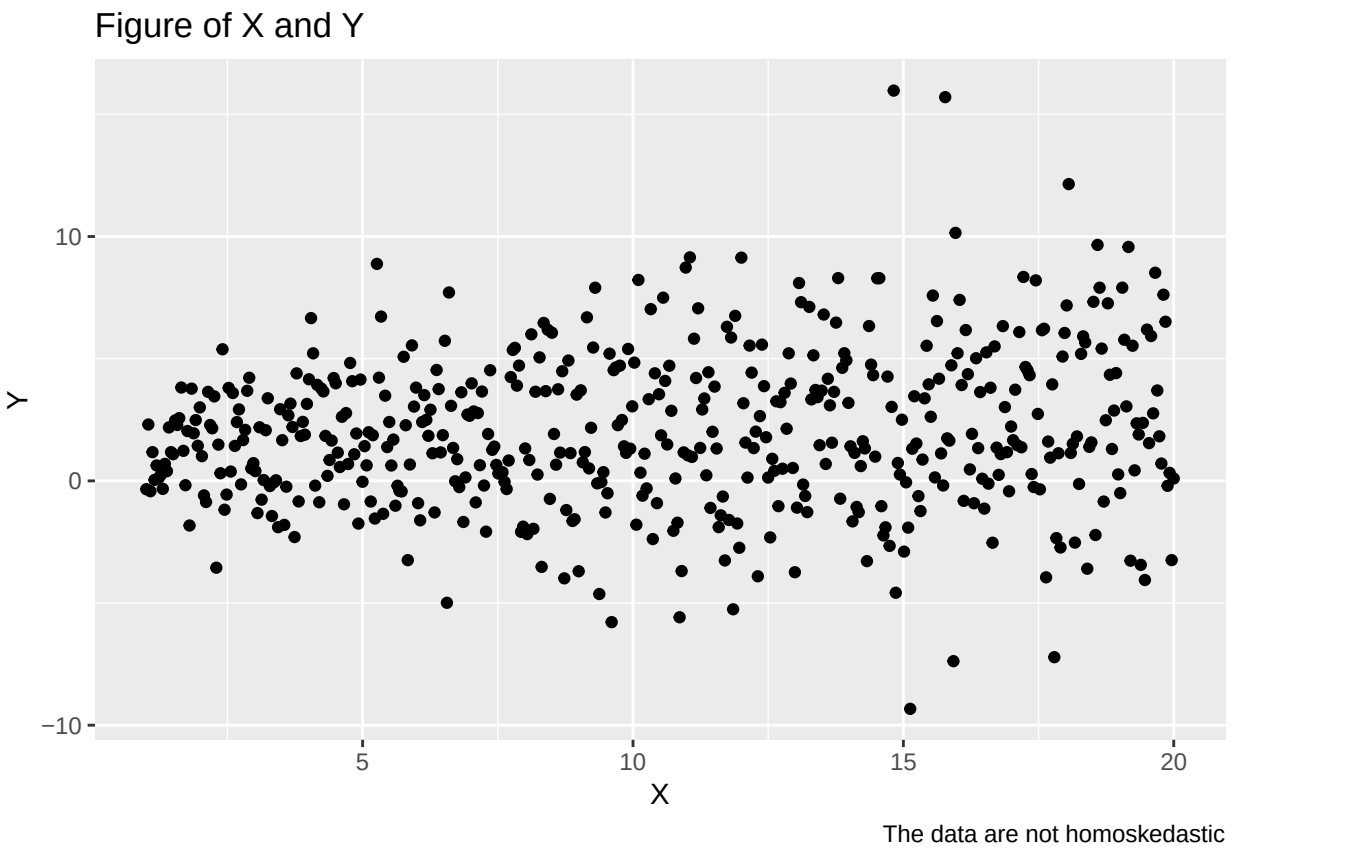
That model doesn't work because there are no spicy desserts or sweet snacks in the dataset, so it's impossible to estimate the interaction effect between flavor profiles and course without there being at least one observation at each combination of each level of the factors.

Simulation Question (12 points)

1. (4 points)

Consider the code below. Create a figure of x and y. What linear regression assumption does this data violate?

```
set.seed(11112020)
n <- 500
x <- seq(1,20, length.out = n)
beta <- c(1, .1)
sigma <- sqrt(x)
y <- rnorm(n, beta[1] + beta[2] * x, sd = sigma)

ggplot() +
  geom_point(aes(x = x, y = y)) +
  labs(x = "X", y = "Y", title = "Figure of X and Y", caption = "The data are not homoskedastic")
```



There is not homoskedasticity for the data. The variability in Y increases as X increases.

2. (4 points)

How well does a linear regression model recover the point estimates of beta? Justify your answer (simulation may be useful).

```
# Find if a 95% CI contains the true betas one time
contains_val = function() {

  # Simulate data
  n <- 500
```

```
  x <- seq(1,20, length.out = n)
  beta <- c(1, .1)
  sigma <- sqrt(x)
  y <- rnorm(n, beta[1] + beta[2] * x, sd = sigma)

  # Find coefficients to make 95% ci
  est_beta0 = summary(lm(y ~ x))$coefficients[1,1]
  se_beta0 = summary(lm(y ~ x))$coefficients[1,2]
  est_beta1 = summary(lm(y ~ x))$coefficients[2,1]
  se_beta1 = summary(lm(y ~ x))$coefficients[2,2]

  # Find bounds for 95% CI
  beta0_95ci_lb = (est_beta0 + qnorm(0.025) * se_beta0)
  beta0_95ci_ub = (est_beta0 + qnorm(0.975) * se_beta0)
  beta1_95ci_lb = (est_beta1 + qnorm(0.025) * se_beta1)
  beta1_95ci_ub = (est_beta1 + qnorm(0.975) * se_beta1)

  # If beta is greater than lb and less than ub then it is in 95% CI and increase counter by 1
  counter = c(ifelse(beta[1] > beta0_95ci_lb & beta[1] < beta0_95ci_ub, 1, 0),
              ifelse(beta[2] > beta1_95ci_lb & beta[2] < beta1_95ci_ub, 1, 0))

  return(counter)
}
set.seed(11282020)
apply(replicate(10000, contains_val()), MARGIN = 1, FUN = mean)
```

```
## [1] 0.9887 0.9499
```

The 95% CI should cover the true value of β_0 and β_1 approximately 95% of the time in the long run. For estimating β_0, the 95% CI contains the true value 98.9% of the time with 10,000 repetitions, which is too much. That would suggest that the SE estimate for β_0 is too large. When estimating β_1, the true value was contained in the confidence interval 95.0% of the time with 10,000 repetitions, so it seems that the estimates for β_1 are pretty accurate.

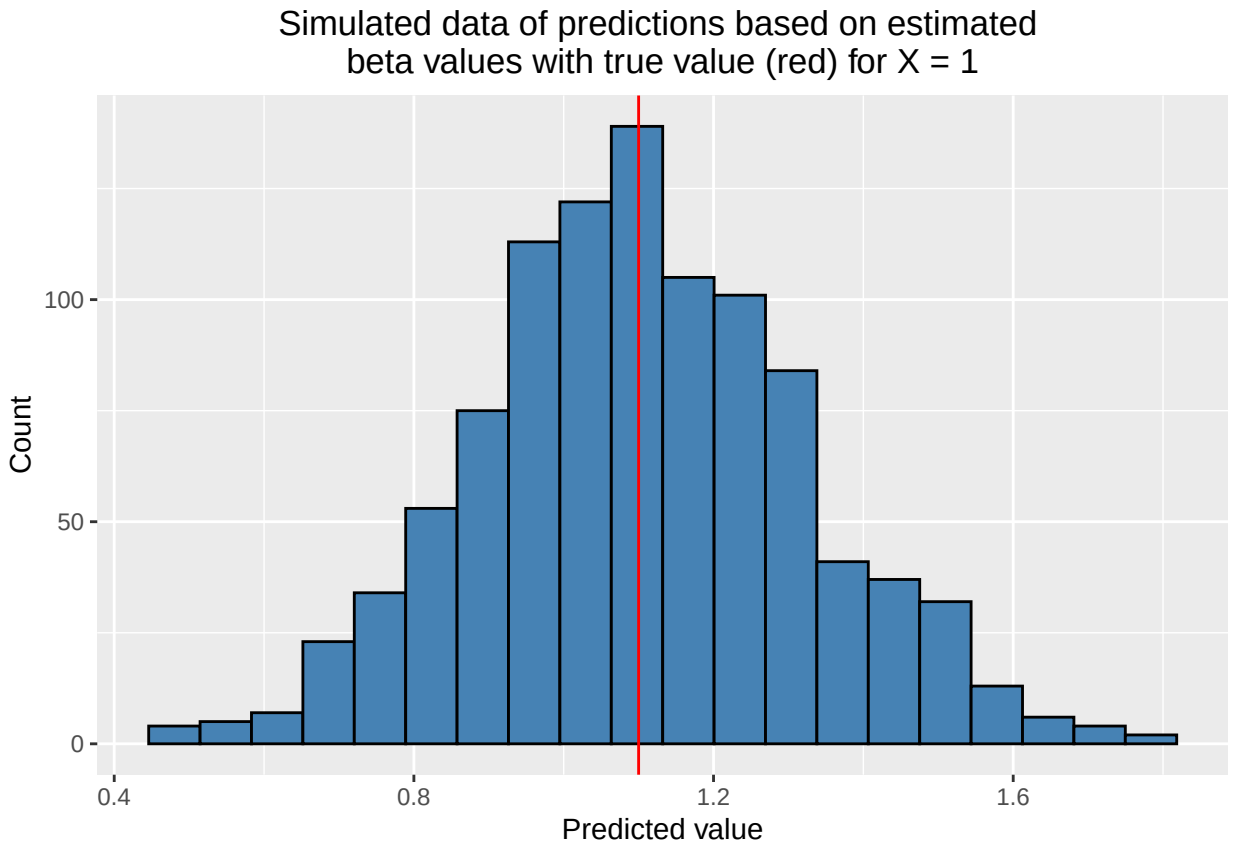
2. (4 points)

How well does a linear regression model capture the uncertainty in a predictions for y conditional on

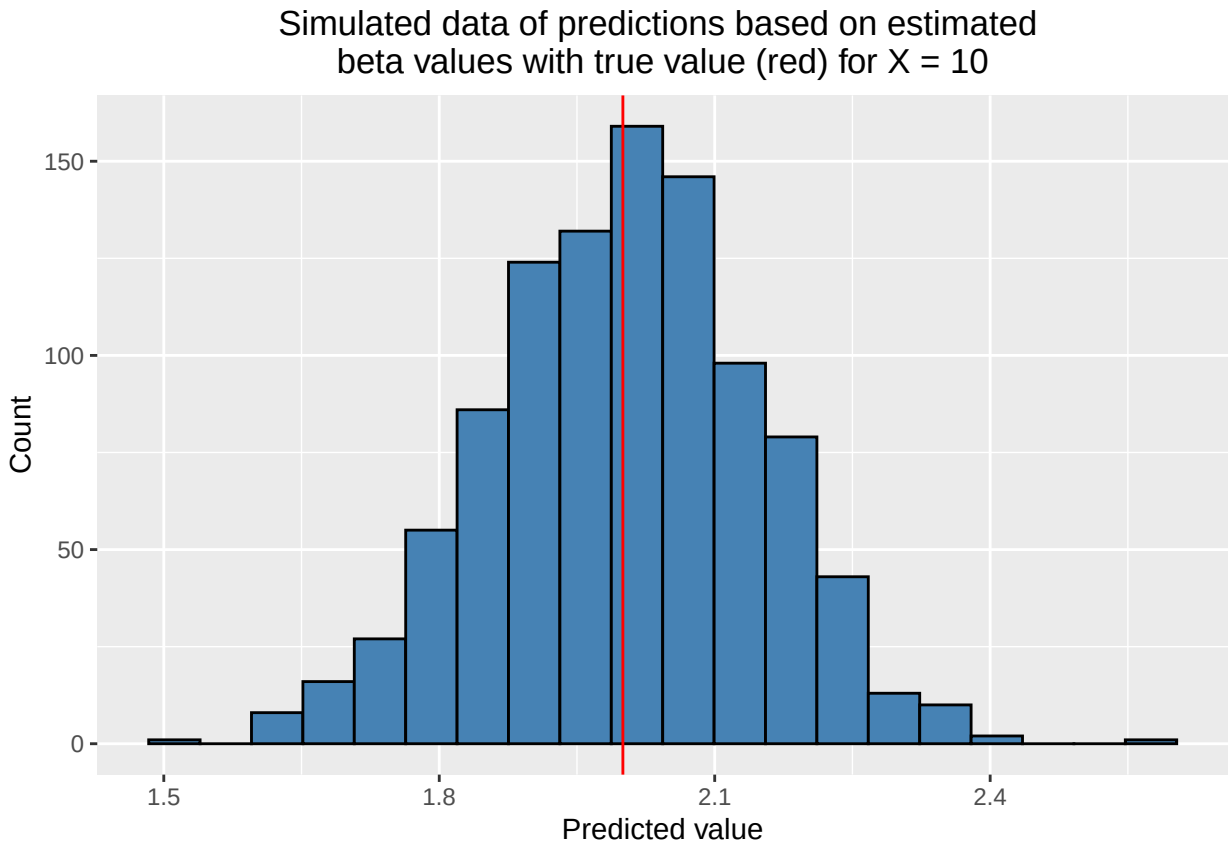
- x = 1
- x = 10
- x = 20

```
# Get prediction based on estimated beta values
fun = function(xval) {
  # Simulate data
  n <- 500
  x <- seq(1,20, length.out = n)
  beta <- c(1, .1)
  sigma <- sqrt(x)
  y <- rnorm(n, beta[1] + beta[2] * x, sd = sigma)

  # Estimate beta values
  lm = lm(y ~ x)
  # Predict the simulated estimate for E(Y|X = xval)
  pred = predict(object = lm, newdata = data.frame(x = xval))
```

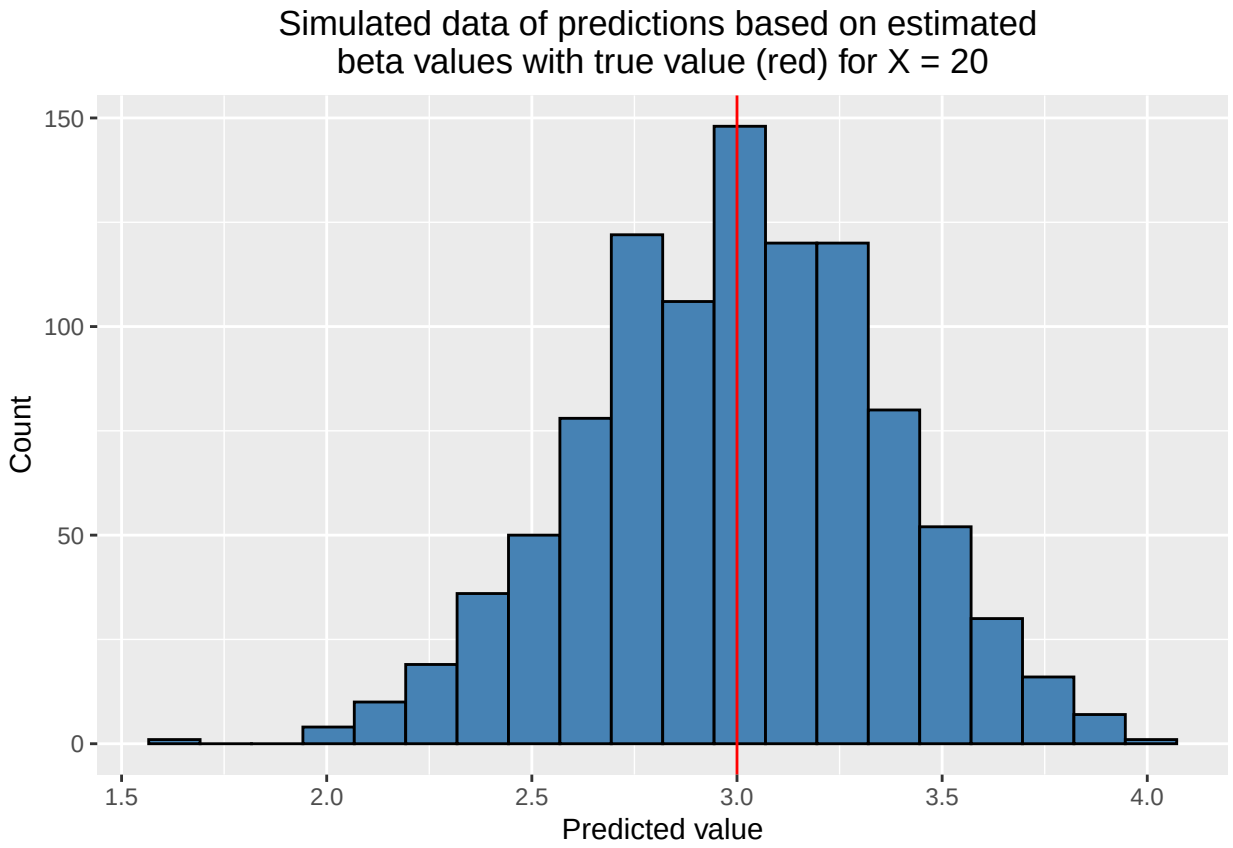
```
}

# Compare histogram of many simulations of predictions to actual data
xvals = c(1, 10, 20)
dat1 = t(replicate(1000, fun(xvals)))

ggplot() +
  geom_histogram(aes(x = dat1[,1]), fill = "steelblue", col = "black", bins = 20) +
  geom_vline(xintercept = 1 + 0.1 * xvals[1], col = "red") +
  labs(x = "Predicted value",
       y = "Count",
       title = "Simulated data of predictions based on estimated \nbeta values with true value (red) fo
  theme(plot.title = element_text(hjust = 0.5))
```




```
ggplot() +
  geom_histogram(aes(x = dat1[,2]), fill = "steelblue", col = "black", bins = 20) +
  geom_vline(xintercept = 1 + 0.1 * xvals[2], col = "red") +
  labs(x = "Predicted value",
       y = "Count",
       title = "Simulated data of predictions based on estimated \nbeta values with true value (red) fo
  theme(plot.title = element_text(hjust = 0.5))
```




```
ggplot() +
  geom_histogram(aes(x = dat1[,3]), fill = "steelblue", col = "black", bins = 20) +
  geom_vline(xintercept = 1 + 0.1 * xvals[3], col = "red") +
  labs(x = "Predicted value",
       y = "Count",
       title = "Simulated data of predictions based on estimated \nbeta values with true value (red) fo
  theme(plot.title = element_text(hjust = 0.5))
```



For $X = 1$, 10, and 20, the predictions look very normal and centered at what the true value is, which would indicate that even though the SE estimates may be off from what I found in the last question, the predictions are still accurate.

Data Analysis (Scaled to be worth 26 points)

Using the Indian recipe dataset fit a logistic regression model to model the probability of a dish being classified as a main course. Write your results in a short report (shorter than the projects). Turn this document in separately. Including figures and tables, I am setting a four page maximum using standard PDF output settings in RMD. This will require careful selection and sizing of tables and figures. The page limit does not apply to references or code in the appendix.

Report generalities	**Points**
Spelling, grammar, writing clarity, paragraphs, section labels	/8
Citations/Acknowledgments for papers and packages used	/4
Code in appendix	/4

Introduction + Data Overview	**Points**
Research question	/4
Variables with units and descriptive statistics	/4
Data Viz: Figure Clarity (Titles, Labels, and Captions)	/4

Statistical Procedures	**Points**
Define model to fit with complete notation (including priors)	/8
Defense of model choice	/4

Results + Discussion	**Points**
Discuss Results in the context of the research question	/4
Summarize estimates from final model including uncertainty	/8

Source used:

I didn't use any sources for help on the internet except ones for general debugging. I used the course notes and the textbook for some sections, especially the simulation question.