# Problem Set 3

Jacob Hood

Due: Saturday, Apr. 17th, 11:59 pm

```
knitr::opts_chunk$set(echo = TRUE, results = "asis")
library(pacman)
p_load(tidyverse, kableExtra, scales, stargazer, effects)
```

## Part 1 Assumptions of OLS Regression

Recall that in our first lecture on regression, we talked about the **Gauss-Markov Assumptions**. If all these assumptions are met, the OLS estimator is the **Best Linear Unbiased Estimator** (**BLUE**). In a simple bivariate case, if the "true" data-generating process is $Y = \beta_0 + \beta_1 X + \epsilon$. The Gauss-Markov Assumptions can be state as the following:

(a) **Linearity**: A linear relationship between $X$ and $Y$ hold in the sample.

(b) **Exogeneity of Predictors**: The conditional mean of the error term, given the predictor, is zero ($\mathbf{x} = [x_1, x_2, ..., x_n]^\top$ is the value vector of $X$):

$$E[\epsilon_i|\mathbf{x}] = 0, \text{ for all } i = 1, 2, ..., n.$$

(c) **No Perfect Collinearity**: Explanatory variables cannot be perfectly correlated.

(d) **Homoskedasticity**:

- No Heteroskedasticity: The conditional variance of the error term, given the predictor, is constant: $Var[\epsilon_i|\mathbf{x}] = \sigma^2$, for all $i = 1, 2, ..., n$.
- No Autocorrelation: Conditional on the predictor, the error terms are uncorrelated across the observations: $Cov[\epsilon_i, \epsilon_j|\mathbf{x}] = 0$ , $i \neq j$.

**1.** [15pts] For each of the assumptions, discuss what will go wrong when the assumption is violated. Be brief in your answers. *Note*: In addition to class materials, you can learn more about these assumptions in the Wikipedia article on the Gauss–Markov theorem, particularly the "Gauss–Markov theorem as stated in econometrics" section. (You can skip all the mathematical proofs and remarks.)

(a) **Linearity**: If the linearity assumption is violated, then an OLS model would not be appropriate for these data. A non-linear model (like quadratic) might be used instead.
(b) **Exogeneity of Predictors**: If this assumption is violated (meaning the error term is correlated with explanatory variables), potentially because of omitted variables bias, then part of the error term is predictable and our parameter estimates become biased.

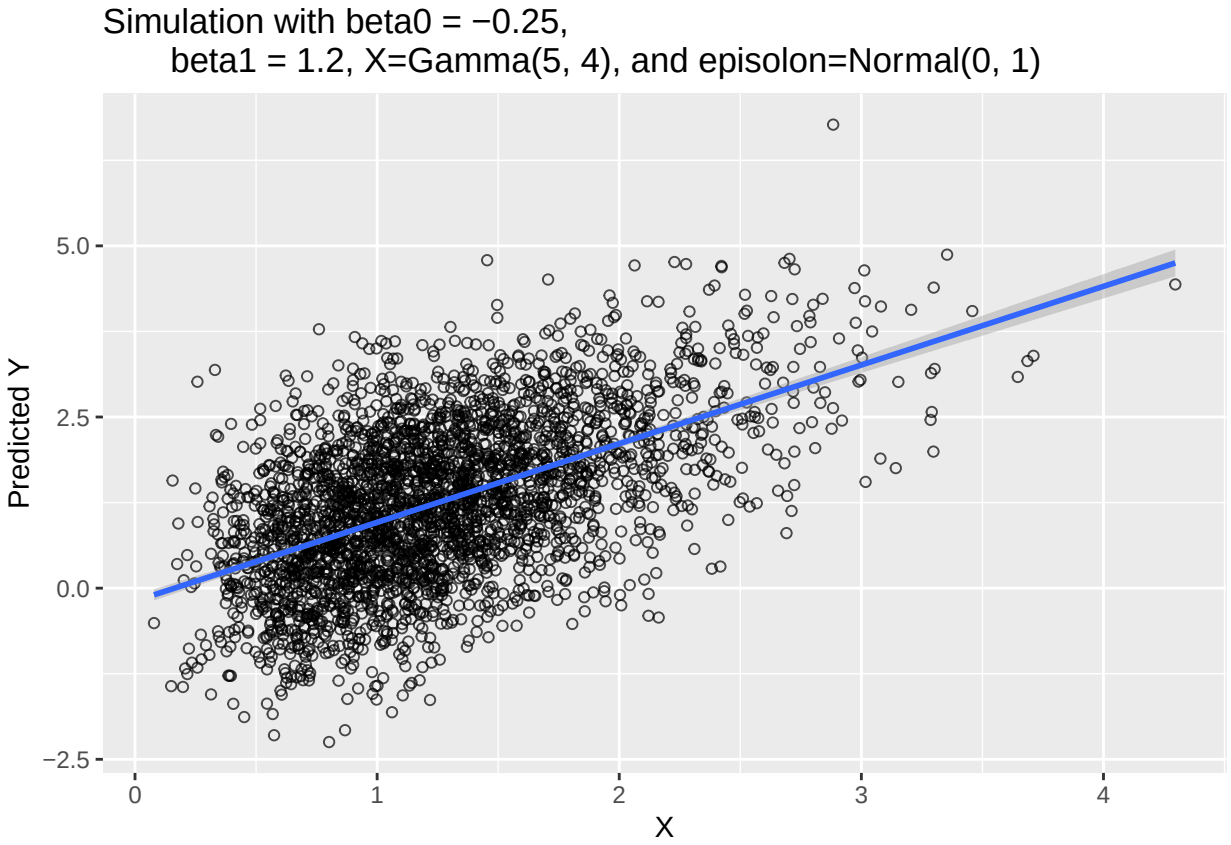
(c) **No Perfect Collinearity**: If there is perfect or multi-collinearity between variables, it will be difficult to interpret our regression coefficients because the standard errors have been inflated.
(d) **Homoskedasticity**: If there is heteroskedasticity, our beta coefficients do not change but our standard errors of the coefficients are no longer reliable. Thus, our confidence intervals based on these standard errors would either be too narrow or too wide.

**2.** [5pts] Let $\beta_0 = -0.25$, $\beta_1 = 1.2$, $X \sim \Gamma(5, 4)$, and $\epsilon \sim Normal(0, 1)$. Here, $\Gamma(\alpha, \psi)$ denotes the Gamma distribution with shape parameter $\alpha$ and rate parameter $\psi$. (You can search how to use R to simulate from this distribution.)
Simulate a dataset of size $n = 3,000$ from this process in which all of the assumptions you've discussed above hold. Estimate a OLS model and plot regression diagnostics of this model.

```
set.seed(1234)
X <- rgamma(3000,5,4)
Y_predicted <- -0.25 + 1.2*X + rnorm(3000,0,1)

df <- data.frame(
  X,
  Y_predicted)
```

```
# Creating the scatterplot
df %>% ggplot(aes(x = X, y = Y_predicted)) +
  geom_point(shape =1, alpha = 0.7) +
  geom_smooth(method = 'lm') +
  labs(title = "Simulation with beta0 = -0.25,
       beta1 = 1.2, X=Gamma(5, 4), and episolon=Normal(0, 1)", x = "X", y = "Predicted Y")
```

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




**Bonus Question** [10pts]: From assumption (a), (b) and (d), choose one assumption and simulate a data that violates that assumption (all other assumptions should be satisfied). Create a plot which illustrates how the violation of the assumption affects the regression results. This can be a scatterplot with both the "true" and "false" OLS lines, a sampling distribution of the OLS estimator (comparing your estimate model results with actual simulations), or anything that shows how the violation leads us to false decisions if we assume the assumption is true. (The point is to demonstrate a contrast between the "true" and the "false", not just diagnostics of the "false".)

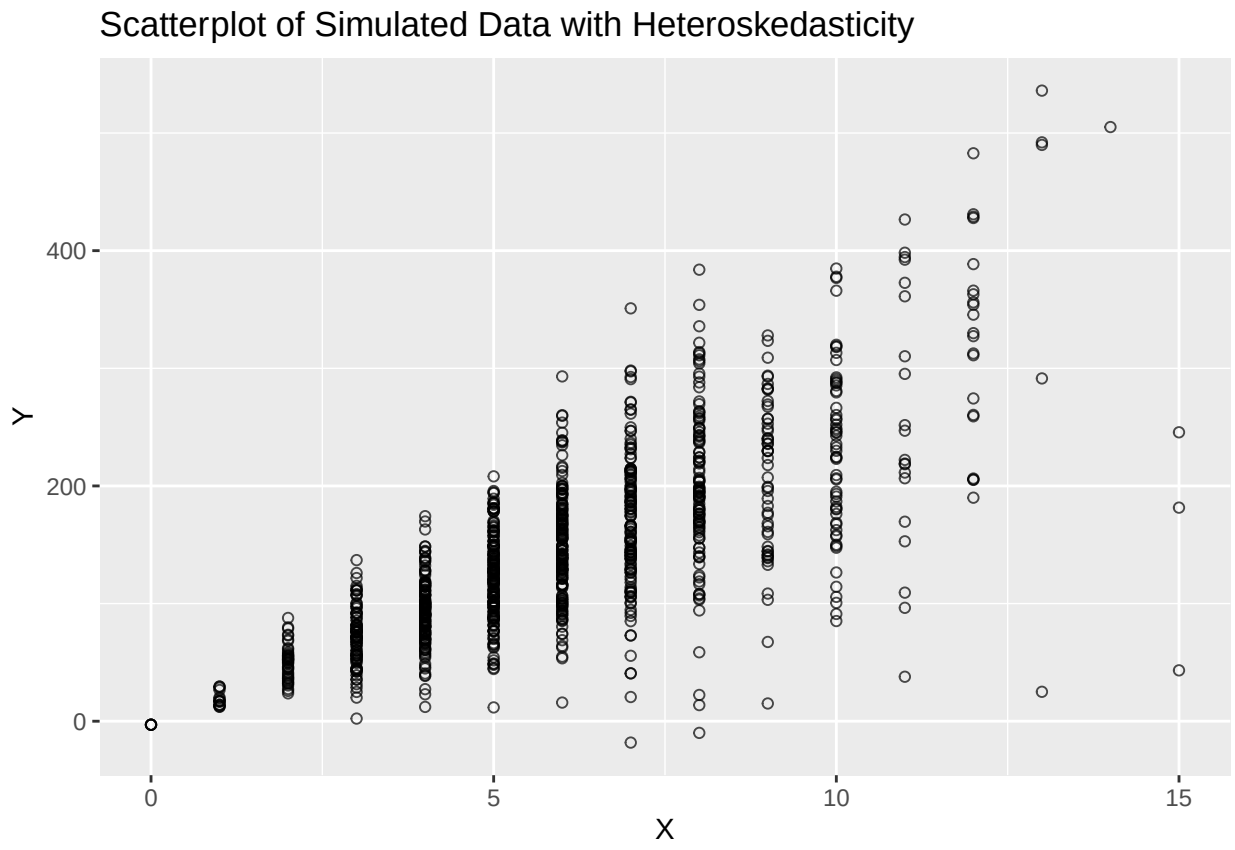
When simulating data, you don't have to use the parameters set in the previous problem.

*Hint*: You can search how to use `+ stat_function()` to plot a nonlinear line when plotting with `ggplot()`, or search how to use the base R functions such as `plot()` and `curve()`.

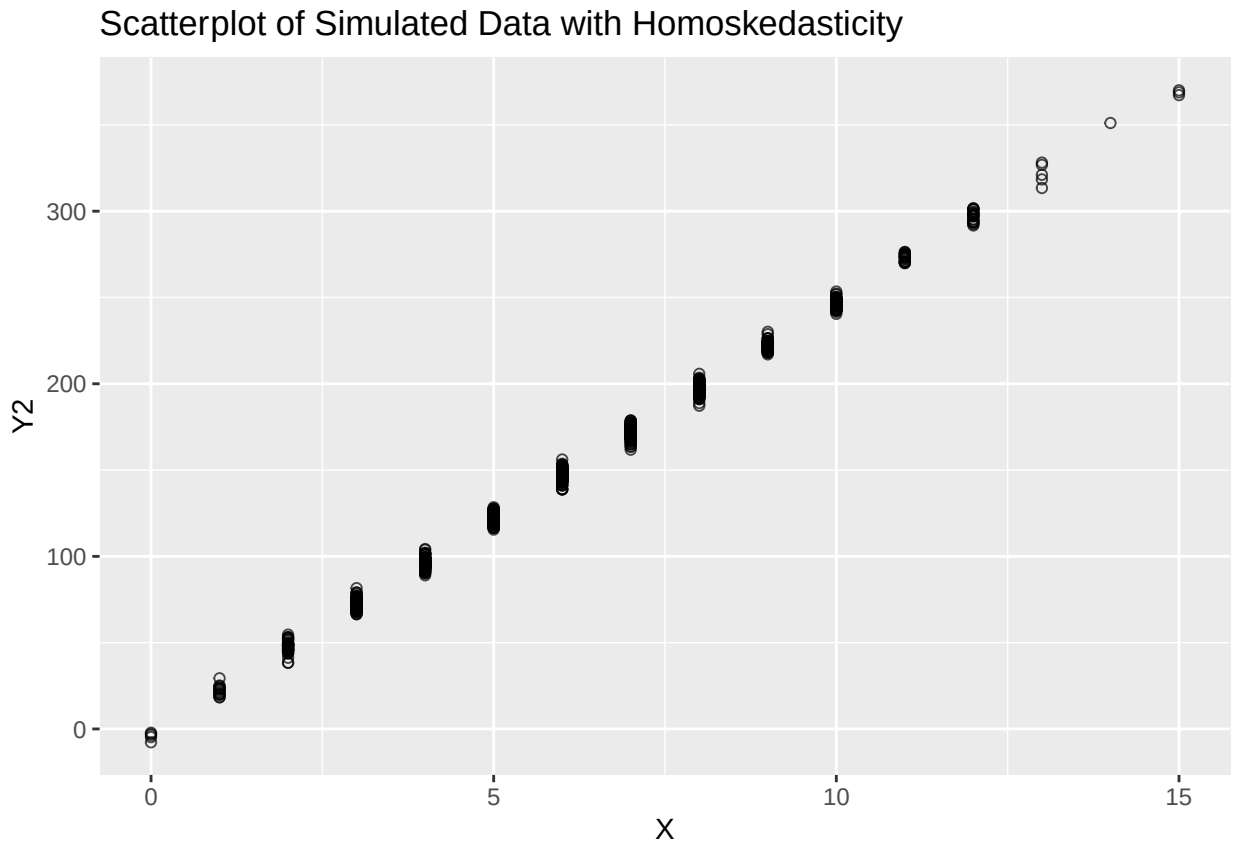
## Simulating a violation of homoskedasticity

```
# Simulate IV
set.seed(1234)
X <- rpois(1000, lambda = 6)

# Simulate error term  to violate homoskedasticity
set.seed(1234)
error_violate <- rnorm(1000, 0, 8*X)

#Simulate error term to satisfy homoskedasticity
set.seed(1234)
```

```r
error_satsify <-rnorm(1000, 0, 3) # Here, error term isn't dependent on X

# Calculate DV for violation
Y <- -3 + 25*X + error_violate
# Calculate DV for satisfy
Y2 <--3 + 25*X + error_satsify

# Put variables into a dataframes
violate_df<- tibble(X = X, Y = Y)
satisfy_df <- tibble(X = X, Y = Y2)

# Plot Y against X, violate_df
violate_df %>%
  ggplot(aes(x = X, y = Y)) +
  geom_point(shape = 1, alpha = 0.7) +
  labs(title = "Scatterplot of Simulated Data with Heteroskedasticity")
```




```r
# Plot Y against X, satisfy_df
satisfy_df %>%
  ggplot(aes(x = X, y = Y2)) +
  geom_point(shape = 1, alpha = 0.7) +
  labs(title = "Scatterplot of Simulated Data with Homoskedasticity")
```




```
m_violate <- lm(Y ~ X, violate_df)
m_satisfy <- lm(Y2 ~ X, satisfy_df)
stargazer(m_violate, m_satisfy, type = "latex")
```

% Table created by stargazer v.5.2.2 by Marek Hlavac, Harvard University. E-mail: hlavac at fas.harvard.edu
% Date and time: Fri, Apr 16, 2021 - 14:16:29

**As shown in the regression table, when there is heteroskedasticity our standard error is inflated compared to when there is not heteroskedasticity (0.689 vs. 0.037, respectively).**

## Part 2 Causality

A study on COVID-19 constructed a "COVID risk factor" score based on the COVID infection rate of a given area (defined by zip code).

A researcher wants to estimate the effect of having a vaccination center in the area on that area's COVID risk factor score. She compiled a dataset that contains each area's COVID risk factor score and whether the area has a vaccination center. She then estimated the effect of having a vaccination center using the "naive estimator" we discussed in class.

You noted that the quality of information residents have about COVID and the vaccine can be a confounding variable that affects both the area's infection rate and whether there is a vaccination center in the area. Assume that you are able to estimate the relationships this "informedness" confounder (`info`) and the original "vaccination center" predictor (`vaccine`) have with the COVID risk factor score (`covid_risk`), which can be simulated using the following code (`n` is sample size):

Table 1:

| | *Dependent variable:* | |
|---|---|---|
| | Y | Y2 |
| | (1) | (2) |
| X | 24.032*** | 24.991*** |
| | (0.689) | (0.037) |
| Constant | 1.107 | −3.027*** |
| | (4.545) | (0.246) |
| Observations | 1,000 | 1,000 |
| $R^2$ | 0.549 | 0.998 |
| Adjusted $R^2$ | 0.549 | 0.998 |
| Residual Std. Error (df = 998) | 55.343 | 2.993 |
| F Statistic (df = 1; 998) | 1,214.921*** | 449,078.600*** |
| *Note:* | | *p<0.1; **p<0.05; ***p<0.01 |

```
set.seed(1234) # set the same seed to ensure identical results
e = rnorm(n, 0, 0.5)
covid_risk = rescale( 0 - 7*vaccine - 2*info + e, to = c(0, 100))
```

**1.** [5pts] Import the data `covid.csv`, according to the counterfactual framework, constructing a counterfactual "risk factor" in the dataframe.

```
# Importing data
covid_data <- read.csv("covid.csv")
```

```
 # set the same seed to ensure identical results
n = nrow(covid_data)
set.seed(1234)
e = rnorm(n, 0, 0.5)
covid_risk = rescale(0 - 7*covid_data$vaccine - 2*covid_data$info + e, to = c(0, 100))

# Add counterfactual variable
covid_data <- covid_data %>%
  mutate(vaccine_cf = ifelse(vaccine == 1, 0, 1),
         risk_cf = rescale(0 - 7*vaccine_cf - 2*info + e, to = c(0,100)))
```

**2.** [10pts] Fill out the table below (round to 1 decimal points):

| Group | $Y^T$ | $Y^C$ |
|---|---|---|
| Treatment Group ($D = 1$) | $E[Y^T\|D = 1] = 41.8$ | $E[Y^C\|D = 1] = 49.7$ |
| Control Group ($D = 0$) | $E[Y^T\|D = 0] = 67.3$ | $E[Y^C\|D = 0] = 58.1$ |

```
# Creating variables for treatment and control, both observed and counterfactual
# Calculating means
treat <- covid_data %>% filter(covid_data$vaccine == 1)
control <- covid_data %>% filter(covid_data$vaccine == 0)
cf_treat <- covid_data %>% filter(covid_data$vaccine_cf == 0)
cf_control <- covid_data %>% filter(covid_data$vaccine_cf == 1)
meantreat <- mean(treat$covid_risk)
meancontrol <- mean(control$covid_risk)
meancf_treat <- mean(cf_treat$risk_cf)
meancf_control <- mean(cf_control$risk_cf)
```

```
mean(treat$covid_risk) - mean(cf_treat$risk_cf)
```

[1] -16.31397

(c) Treatment Effect on the Control

```
mean(control$covid_risk) - mean(cf_control$risk_cf)
```

[1] 17.61613

(d) Selection Bias

```
mean(cf_treat$risk_cf) - mean(control$covid_risk)
```

[1] -9.185033

**4**. [15pts] Write a non-technical, short summary reporting your results in response to the above mentioned researcher who used the naive estimation. Imagine that you are explaining this to an audience who may not be familiar with the specific terminologies of the counterfactual framework (such as ATE or Treatment Effect on the Treated), but is interested in your substantive findings.

There is one big problem in trying to determine the effect of having a vaccination center on that area's COVID risk: it's not possible to observe the effects of both having and not having a vaccination center in the same area at the same time. This is the biggest problem we face when trying to claim that "X" variable affects "Y" variable in a certain way. Constructing a counterfactual framework can help us make stronger claims. Basically, we construct extra variables that simulate the other outcome we can't directly observe in our data: what would happen to COVID risk if an area that received the treatment (having a vaccination center) didn't receive the treatment, for instance. If we think that the quality of information might be an unobserved factor, then we have a problem of selection bias. This means that we can expect those who receive higher quality information will be more likely to want and ultimately receive vaccinations. People aren't randomly assigned to live in these different areas, so we can also expect that the quality of information people get about COVID is associated with where they choose to live. Thus, there are important differences between those in the control group and those in the treatment group that we must consider to figure out the effect of living in an area with a vaccination center.

A biased or "naive" guess of the average treatment effect (ATE) is, that for areas having a vaccination center (treatment) compared to those that don't (control), these areas have a 25.5 unit lower risk of COVID. It's a biased and thus unreliable guess because 1) it can't take into account both potential outcomes happening in the same area and 2) the quality of information creates an issue of selection bias. Our counterfactual framework helps here. The treatment effect on the treated we calculate with the help of this framework (16.31 unit decrease in COVID risk) is not equal to the biased guess we calculated earlier (25.5). This shows that our hunch about selection bias in the data is true. If people were randomly assigned to these areas, these two numbers would be equal, so information quality is playing an important role here. Also with this framework, we can calculate the effect of not having a vaccination center for the areas we actually observed in this condition: a risk increase of 17.6 units. Since we know quality of information residents receive is significant here, we need to calculate the selection bias. We take the difference between the effect on COVID risk from our variable simulating what would happen if areas that have vaccination centers didn't have vaccination centers, and the effect on infection rates we actually observe in those areas without vaccination centers, giving us -9.19. By taking into account different qualities of information residents receive, we can see that this variable causes a problem of self-selection where people with higher quality information are more likely to be in the treatment group (having vaccination center) and thus have lower risk.

# Part 3 Linear Probability Model and Logistic Regression

`admin.csv` contains a dataset of graduate school admission results with the following variables:

| Variable Name | Variable Detail |
|---|---|
| `admit` | Admission Dummy (Admitted is 1) |
| `gre` | GRE score |
| `gpa` | GPA |
| `rank` | Institution Tier (Tier 1 to 4) |

**1**. [10pts] Import `admin.csv` to your R environment. Estimate (a) a linear probability model and (b) a logistic regression model to predict the probability of being admitted based on the applicant's GRE, GPA, and institution tier. Display the two modeling results in a table.

```
# Importing data
admin_data <- read.csv("admin.csv")
```

(a) Estimating linear probability model and (b) Estimate logistic regression model, display both in table

```
lpm1 <- lm(admit ~ gre + gpa + rank, admin_data)
logit1 <- glm(admit ~ gre + gpa + rank, admin_data, family = binomial(link="logit"))

stargazer(lpm1, logit1, type = "latex", omit.stat = c("ser","f"))
```

% Table created by stargazer v.5.2.2 by Marek Hlavac, Harvard University. E-mail: hlavac at fas.harvard.edu
% Date and time: Fri, Apr 16, 2021 - 14:16:29

**2**. [10pts] In one or two paragraphs, summarize your modeling result for each model.

In the linear probability model, for a 1 unit increase in GRE score, the likelihood of being admitted increases by 0.0004, or 0.04%, holding all else constant. For a 1 unit increase in GPA, the likelihood of being admitted increases by 0.151, or 15.1%, holding all else constant. For a 1 unit increase in school ranking (meaning schools with lower rankings), the likelihood of being admitted increases by -0.182, or -18.2%, holding all else constant.

In the logistic model, for a 1 unit increase in GRE score, the odds of being admitted increases by a factor of e^0.002, or 0.005, holding all else constant. For a 1 unit increase in GPA, the odds of being admitted increases by a factor of e^0.777, or 2.18, holding all else constant. For a 1 unit increase in school ranking (meaning schools with lower rankings), the odds of being admitted increases by a factor of e^-0.560, or -0.057, holding all else constant. These effects are statistically significant at the $p<0.01$ level

**3**. [15pts] Plot the predicted probability of admission based on one's GPA percentile and institution rank (holding GRE at the mean) for the logistic regression model. For the purpose of this exercise, please set the value of `gpa` to range from 1 to 4. Make sure to add appropriate title and labels to your figure.

```
# Using effect function to predict admission by gpa and rank, gre held at mean
pred_logit <- Effect(
  # Key IVs
  c("gpa", "rank"),
  # Model
  logit1,
  # Values of IVs
  xlevels = list(
```

Table 4:

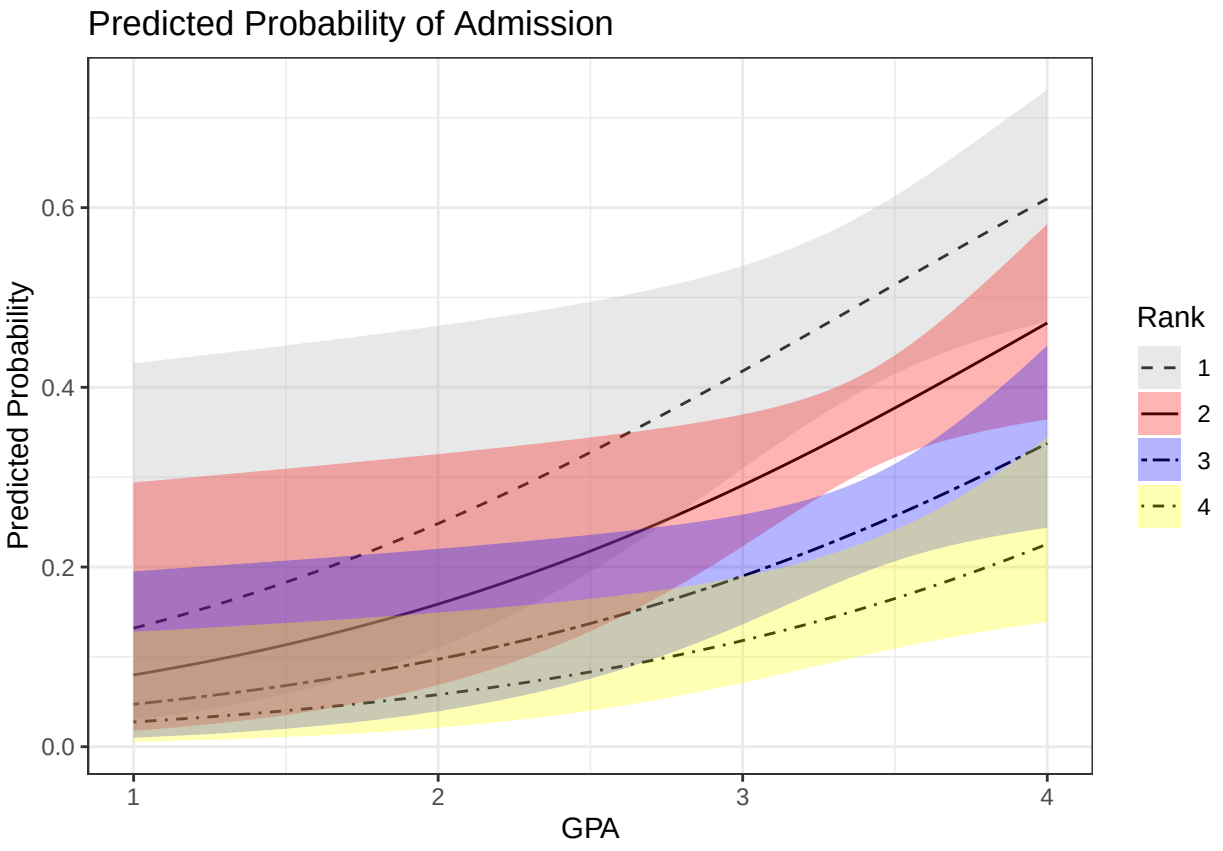
| | *Dependent variable:* | |
|---|---|---|
| | admit | |
| | *OLS* | *logistic* |
| | (1) | (2) |
| gre | 0.0004** | 0.002** |
| | (0.0002) | (0.001) |
| gpa | 0.151** | 0.777** |
| | (0.063) | (0.327) |
| rank | −0.110*** | −0.560*** |
| | (0.024) | (0.127) |
| Constant | −0.182 | −3.450*** |
| | (0.217) | (1.133) |
| Observations | 400 | 400 |
| $R^2$ | 0.096 | |
| Adjusted $R^2$ | 0.089 | |
| Log Likelihood | | −229.721 |
| Akaike Inf. Crit. | | 467.442 |
| *Note:* | *p<0.1; **p<0.05; ***p<0.01 | |

```
    gpa = seq(1,4, 0.01),
    rank = c(1,2,3,4),
    gre = mean(admin_data$gre)
    ),
  # Set confidence intervals
  confidence.level = .95) %>%
  # Convert to dataframe
  as.data.frame()

# Check the df:
head(pred_logit, 10) %>% kbl("latex") %>% kable_classic_2(full_width = F)
```

```
# Plotting the probability of admission
pred_logit %>%
  ggplot(aes(x = gpa, y = fit, ymax = upper, ymin = lower,
             fill = as.factor(rank), linetype = as.factor(rank))) +
  geom_line() +
  geom_ribbon(alpha = 0.3) +
  labs(title = "Predicted Probability of Admission",
       x = "GPA",
       y = "Predicted Probability") +
  scale_fill_manual(name = "Rank",
                    values = c("grey70", "red", "blue", "yellow"),
                    label = c("1", "2", "3", "4")) +
```

| gpa | rank | fit | se | lower | upper |
|---|---|---|---|---|---|
| 1.00 | 1 | 0.1318836 | 0.0928490 | 0.0300643 | 0.4267995 |
| 1.01 | 1 | 0.1327757 | 0.0930159 | 0.0304748 | 0.4271801 |
| 1.02 | 1 | 0.1336730 | 0.0931805 | 0.0308907 | 0.4275611 |
| 1.03 | 1 | 0.1345754 | 0.0933427 | 0.0313120 | 0.4279427 |
| 1.04 | 1 | 0.1354829 | 0.0935025 | 0.0317388 | 0.4283247 |
| 1.05 | 1 | 0.1363956 | 0.0936599 | 0.0321712 | 0.4287072 |
| 1.06 | 1 | 0.1373134 | 0.0938149 | 0.0326092 | 0.4290903 |
| 1.07 | 1 | 0.1382365 | 0.0939673 | 0.0330530 | 0.4294738 |
| 1.08 | 1 | 0.1391647 | 0.0941173 | 0.0335025 | 0.4298579 |
| 1.09 | 1 | 0.1400982 | 0.0942647 | 0.0339578 | 0.4302424 |

```
scale_linetype_manual(name = "Rank",
                      values = c("dashed", "solid" , "twodash", "dotdash"),
                      label = c("1", "2", "3", "4")) +
theme_bw()
```




## Part 4 (Not Graded) Final Replication Project

At this point, you should complete most of the data cleaning and start replicating the descriptive tables and figure. You can submit an additional PDF file if you have made progress in replication Table A1a, Table A1b, and Figure 1.

**I've just started putting together the tables so I don't have anything ready to knit to pdf just yet**