Interactions

Matt Lawrence

December 1, 2021

Setting Up

We'll use the `midd_survey.csv` file on Canvas. These are selected responses to a survey of Middlebury students conducted by Professor Peggy Nelson and her research methods class in Fall 2011. Load it as a data frame called `midd_survey` and load the usual packages.

```
library(tidyverse)
library(huxtable)
midd_survey <- read.csv("https://raw.githubusercontent.com/mjclawrence/soci385_f21/main/data/midd_survey
```

Warm Up

Everyone: Regress gpa on number of siblings:

REPLACE THIS LINE WITH YOUR CODE

```
gpa_sibs_model <- lm(gpa ~ siblings, data = midd_survey)
gpa_sibs_model2 <- lm(gpa ~ siblings + class, data = midd_survey)

huxreg(list("Model 1" = gpa_sibs_model, "Model 2" = gpa_sibs_model2),
       statistics = c("N. obs." = "nobs"),
       coefs = c("Siblings" = "siblings",
                 "Middle Class" = "classMiddle Class",
                 "Upper Middle Class" = "classUpper Middle Class",
                 "Upper Class" = "classUpper Class")) |>
  theme_article()
```

Warm Up 1: Add control for gender to original model

REPLACE THIS LINE WITH YOUR CODE

```
gpa_sibs_gender_model <- lm(gpa ~ siblings + gender,
      data = midd_survey)

summary(gpa_sibs_gender_model)
```

	Model 1	Model 2
Siblings	-0.032 ***	-0.027 **
	(0.009)	(0.009)
Middle Class		0.130 ***
		(0.032)
Upper Middle Class		0.192 ***
		(0.028)
Upper Class		0.159 ***
		(0.036)
N. obs.	985	985
***** $p < 0.001$; ** $p < 0.01$; * $p < 0.05$.		

```
##
## Call:
## lm(formula = gpa ~ siblings + gender, data = midd_survey)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.3859 -0.1459  0.0541  0.2141  0.6248
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  3.531540   0.021186 166.689  < 2e-16 ***
## siblings    -0.031269   0.009187  -3.404 0.000691 ***
## genderOther -0.058342   0.084524  -0.690 0.490210
## genderWoman  0.045629   0.019653   2.322 0.020455 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2996 on 981 degrees of freedom
## Multiple R-squared:  0.01848,    Adjusted R-squared:  0.01548
## F-statistic: 6.157 on 3 and 981 DF,  p-value: 0.0003801
```

Predict gpa for men with 3 siblings and women with 4 siblings

REPLACE THIS LINE WITH YOUR CODE

```
# For Men With 3 Siblings:
3.531540 - .031269*3
```

```
## [1] 3.437733
```

```
# For Women With 4 Siblings:
3.531540 - .031269*4 + .045629
```

```
## [1] 3.452093
```

Warm Up 2: Regress gpa on number of siblings, controlling for class.

REPLACE THIS LINE WITH YOUR CODE

```
# Reorder class
midd_survey <- midd_survey |>
     mutate(class = factor(class,
                           levels = c("Lower Class",
                                      "Middle Class",
                                      "Upper Middle Class",
                                      "Upper Class")))

gpa_sibs_class_model <- lm(gpa ~ siblings + class,
  data = midd_survey)

summary(gpa_sibs_class_model)
```

```
##
## Call:
## lm(formula = gpa ~ siblings + class, data = midd_survey)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.27129 -0.15165  0.02979  0.22156  0.62156
##
## Coefficients:
##                           Estimate Std. Error t value Pr(>|t|)
## (Intercept)               3.405229   0.029775 114.365  < 2e-16 ***
## siblings                 -0.026787   0.009059  -2.957  0.00318 **
## classMiddle Class         0.130127   0.031589   4.119 4.12e-05 ***
## classUpper Middle Class   0.191764   0.028396   6.753 2.48e-11 ***
## classUpper Class          0.158991   0.036167   4.396 1.22e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2938 on 980 degrees of freedom
## Multiple R-squared:  0.05695,    Adjusted R-squared:  0.0531
## F-statistic:  14.8 on 4 and 980 DF,  p-value: 9.6e-12
```

Predict gpa for middle class student with 2 siblings and upper middle class student with 1 sibling.

REPLACE THIS LINE WITH YOUR CODE

```
# For middle class student with 2 siblings:
3.405229 - .026787*2 + .130127
```

```
## [1] 3.481782
```

```
# For upper middle class student with 1 sibling:
3.405229 - .026787*1 + .191764
```

```
## [1] 3.570206
```

Warm Up 3: Add controls for gender and class to original model

```
gpa_sibs_class_gender_model <-
lm(gpa ~ siblings + class + gender, data = midd_survey)

summary(gpa_sibs_class_gender_model)
```

```
##
## Call:
## lm(formula = gpa ~ siblings + class + gender, data = midd_survey)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.28880 -0.14141  0.03547  0.20969  0.65715
##
## Coefficients:
##                           Estimate Std. Error t value Pr(>|t|)
## (Intercept)               3.368622   0.033079 101.837  < 2e-16 ***
## siblings                 -0.025774   0.009051  -2.848  0.00450 **
## classMiddle Class         0.132890   0.031883   4.168 3.34e-05 ***
## classUpper Middle Class   0.195771   0.028842   6.788 1.97e-11 ***
## classUpper Class          0.166871   0.036480   4.574 5.39e-06 ***
## genderOther               0.049050   0.084249   0.582  0.56056
## genderWoman               0.051690   0.019277   2.681  0.00745 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.293 on 978 degrees of freedom
## Multiple R-squared:  0.06387,    Adjusted R-squared:  0.05813
## F-statistic: 11.12 on 6 and 978 DF,  p-value: 5.013e-12
```

Predict gpa for lower class men with 0 siblings.

REPLACE THIS LINE WITH YOUR CODE

```
3.368622
```

```
## [1] 3.368622
```

Introducing Interactions

When we add a control to a model, we estimate different intercepts (or alphas) for each value of our control variable. When we add an interaction to a model, we estimate different slopes (or betas) for each value of our control variable. In words, the coefficient for an interaction term tells us if the average change in Y for a one unit change in X changes as the value of our control variable changes.

Since an interaction is a product of two variables, we use the multiplication symbol (*) to set it up. In this example, we want to regress gpa on number of siblings and class, and we want an interaction term between siblings and class.

```
gpa_sibsXclass_model <-
lm(gpa ~ siblings * class,
      data = midd_survey)

summary(gpa_sibsXclass_model)
```

```
##
## Call:
## lm(formula = gpa ~ siblings * class, data = midd_survey)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.26944 -0.15217  0.03884  0.20683  0.73316
##
## Coefficients:
##                                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)                       3.52009    0.04076  86.351  < 2e-16 ***
## siblings                         -0.09065    0.01801  -5.033 5.75e-07 ***
## classMiddle Class                -0.02612    0.05244  -0.498 0.618514
## classUpper Middle Class           0.05007    0.04787   1.046 0.295767
## classUpper Class                  0.01418    0.06503   0.218 0.827416
## siblings:classMiddle Class        0.09276    0.02606   3.559 0.000390 ***
## siblings:classUpper Middle Class  0.08165    0.02289   3.568 0.000377 ***
## siblings:classUpper Class         0.08201    0.03191   2.570 0.010328 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2917 on 977 degrees of freedom
## Multiple R-squared:  0.07307,    Adjusted R-squared:  0.06643
## F-statistic:    11 on 7 and 977 DF,  p-value: 1.97e-13
```

This model still has a reference group; in this example, it is lower class respondents with zero siblings. But the model now estimates coefficients for two kinds of effects:

- Main Effects: siblings, middle class, upper middle class, upper class
- Interaction Effects: siblingsXmiddle class, siblingsXupper middle class, and siblingsXupper class

The slope for our reference group is just the coefficient for siblings. The slope for our other groups is the coefficient for siblings plus the respective interaction term.

Interactions and Predictions

Writing out the full equation might make it easier to understand how interactions work. Here's the full equation for our model:

```
3.52009 - 0.09065*(siblings) - 0.02612*(middle class) +
0.05007*(upper middle class) + 0.01418*(upper class) +
0.09276*(siblings*middle class) +
0.08201*(siblings*upper class) +
0.08165*(siblings*upper middle class)
```

To estimate a predicted gpa for our reference group, every term inside the parentheses becomes a zero:

```
3.52009 - 0.09065*(0) - 0.02612*(0) +
0.05007*(0) + 0.01418*(0) +
0.09276*(0) +
0.08201*(0) +
0.08165*(0)
```

```
## [1] 3.52009
```

For any other group, replace every "siblings" with the number of siblings and replace the class variables with a 1 (if you are predicting for that group) or 0 (if you are not predicting for that group). Don't forget the interaction terms!

Try to estimate the predicted gpa for a middle class student with 2 siblings:

REPLACE THIS LINE WITH YOUR CODE

```
3.52009 - 0.09065*(2) - 0.02612*(1) +
0.05007*(0) + 0.01418*(0) +
0.09276*(2*1) +
0.08201*(0) +
0.08165*(0)
```

```
## [1] 3.49819
```

Plotting Interactions

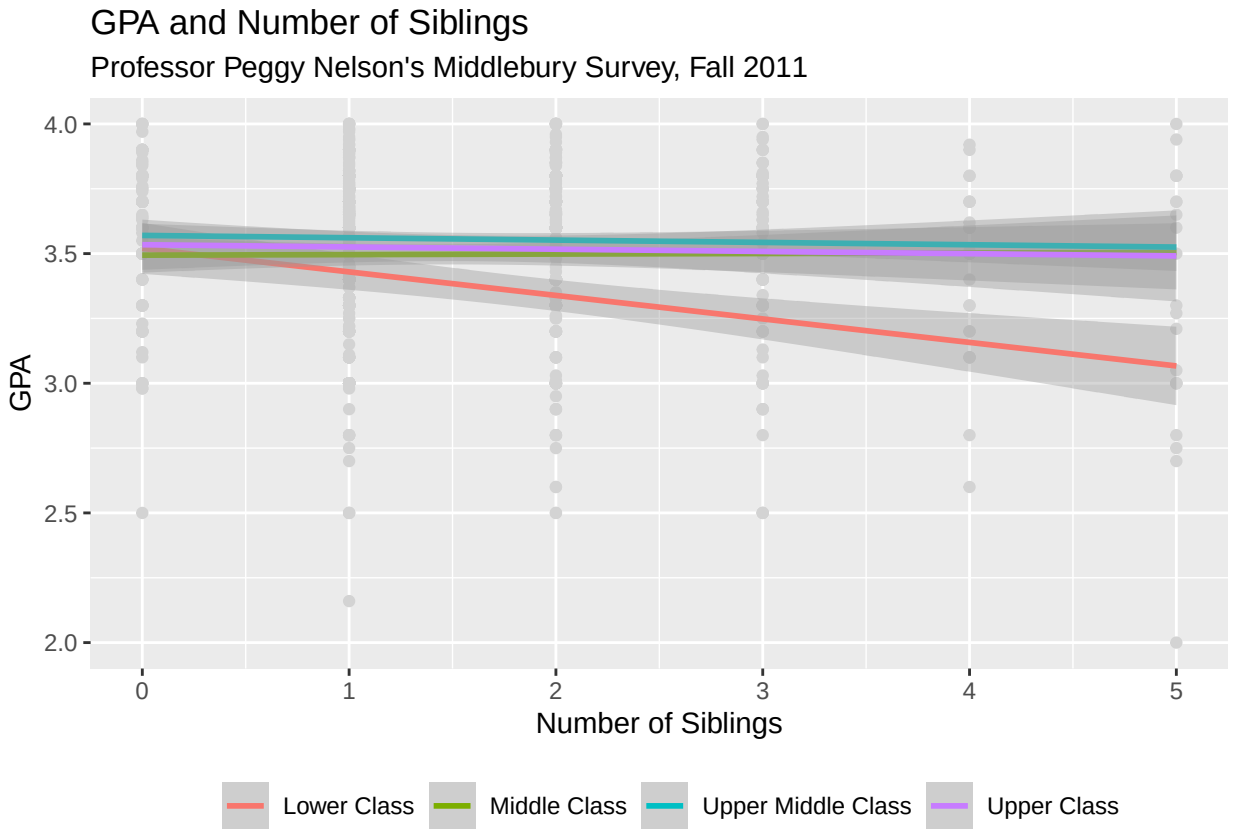
When we were simply controlling for variables, we used `geom_abline()` to manually add lines with different alphas. Plotting interaction terms is easier. Just add your control variable to the aesthetics map (as a `color` term if you want to change colors; `linetype` is another option). The regular `geom_smooth(method = lm)` function includes interactions by default.

```
gpa_sibs_class_plot <- ggplot(midd_survey,
      aes(x = siblings, y = gpa, color = class))

gpa_sibs_class_plot + geom_point(color = "Light Gray") +
     geom_smooth(method = lm) +
     labs(x = "Number of Siblings", y = "GPA",
```

```
      title = "GPA and Number of Siblings",
      subtitle = "Professor Peggy Nelson's Middlebury Survey, Fall 2011",
      color = " ") +
     theme(legend.position = "bottom")
```

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



Another Example

Let's look at the `midd_lookingfor_relationship` variable. Survey respondents were asked to rate their responses from 1 (strongly disagree) to 5 (strongly agree) to the following statement: "I am actively looking for someone to have a relationship with at Middlebury." We want to know if the average responses vary across genders, if school year could explain that variation, and whether any differences across genders might vary by school year.

Let's order the year levels before we continue:

```
midd_survey$year <- factor(midd_survey$year,
     levels = c("First Year", "Sophomore",
      "Junior", "Senior"))
```

Start by regressing `midd_lookingfor_relationship` on `gender`.

REPLACE THIS LINE WITH YOUR CODE

```
lookrel_gender_model <-
     lm(midd_lookingfor_relationship ~ gender,
      data = midd_survey)

summary(lookrel_gender_model)
```

```
##
## Call:
## lm(formula = midd_lookingfor_relationship ~ gender, data = midd_survey)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.9819 -0.9819  0.2201  1.0181  2.3846
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.98187    0.06678  44.650   <2e-16 ***
## genderOther -0.36648    0.36998  -0.991    0.322
## genderWoman -0.20200    0.08601  -2.349    0.019 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.312 on 982 degrees of freedom
## Multiple R-squared:  0.006035,   Adjusted R-squared:  0.00401
## F-statistic: 2.981 on 2 and 982 DF,  p-value: 0.0512
```

Now add `year` as a control variable.

REPLACE THIS LINE WITH YOUR CODE

```
lookrel_gender_year_model <-
lm(midd_lookingfor_relationship ~ gender + year,
      data = midd_survey)

summary(lookrel_gender_year_model)
```

```
##
## Call:
## lm(formula = midd_lookingfor_relationship ~ gender + year, data = midd_survey)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.0879 -1.0279  0.1027  1.1027  2.4178
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)    3.02792    0.09840  30.771   <2e-16 ***
## genderOther   -0.30582    0.36930  -0.828   0.4078
```

```
## genderWoman   -0.19060    0.08585  -2.220   0.0266 *
## yearSophomore  0.06002    0.11928   0.503   0.6149
## yearJunior     0.02784    0.12383   0.225   0.8221
## yearSenior    -0.25514    0.11535  -2.212   0.0272 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.307 on 979 degrees of freedom
## Multiple R-squared:  0.01601,    Adjusted R-squared:  0.01099
## F-statistic: 3.187 on 5 and 979 DF,  p-value: 0.007348
```

Finally add an interaction between `gender` and `year`.

REPLACE THIS LINE WITH YOUR CODE

```
lookrel_genderXyear_model <-
lm(midd_lookingfor_relationship ~ gender * year,
data = midd_survey)

summary(lookrel_genderXyear_model)
```

```
##
## Call:
## lm(formula = midd_lookingfor_relationship ~ gender * year, data = midd_survey)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.02041 -0.97059  0.07333  1.04950  2.66667
##
## Coefficients:
##                             Estimate Std. Error t value Pr(>|t|)
## (Intercept)                  2.95050    0.12975  22.740   <2e-16 ***
## genderOther                 -1.28383    0.76393  -1.681   0.0932 .
## genderWoman                 -0.02801    0.17325  -0.162   0.8716
## yearSophomore                0.06991    0.18489   0.378   0.7054
## yearJunior                   0.03774    0.19193   0.197   0.8442
## yearSenior                   0.02009    0.18304   0.110   0.9126
## genderOther:yearSophomore    3.26342    1.51696   2.151   0.0317 *
## genderWoman:yearSophomore   -0.06573    0.24228  -0.271   0.7862
## genderOther:yearJunior       0.62893    1.08182   0.581   0.5611
## genderWoman:yearJunior      -0.03959    0.25201  -0.157   0.8752
## genderOther:yearSenior       1.14657    0.94001   1.220   0.2229
## genderWoman:yearSenior      -0.48401    0.23681  -2.044   0.0412 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.304 on 973 degrees of freedom
## Multiple R-squared:  0.02732,    Adjusted R-squared:  0.01633
## F-statistic: 2.485 on 11 and 973 DF,  p-value: 0.004467
```

Summarizing Interactions

When interpreting or plotting interactions becomes messy, summarize predicted values. To calculate the predictions, use the `fitted.values` function and save them as a new variable.

```
midd_survey$pred_lookrel <- lookrel_genderXyear_model$fitted.values
```

Then use `group_by()` and `summarize()` to describe the predictions for each combination of the variables you are interacting.

```
lookrel_predictions <- midd_survey |>
  group_by(gender, year) |>
  summarize(agree_look_rel = round(mean(pred_lookrel),2))
```

```
## `summarise()` has grouped output by 'gender'. You can override using the `.groups` argument.
```

```
mytable <- as_hux(lookrel_predictions) |>
  set_contents(1, 1:3, c("Gender", "Year", "Agree Looking \\newline For Relationship")) |>
  set_align(everywhere, 2:3, "center") |>
  theme_article()

escape_contents(mytable) <- FALSE

mytable
```

Gender	Year	Agree LookingFor Relationship
Man	First Year	2.95
Man	Sophomore	3.02
Man	Junior	2.99
Man	Senior	2.97
Other	First Year	1.67
Other	Sophomore	5
Other	Junior	2.33
Other	Senior	2.83
Woman	First Year	2.92
Woman	Sophomore	2.93
Woman	Junior	2.92
Woman	Senior	2.46

Another Example

Think about answers to the question: "Are you satisfied with opportunities at Middlebury to meet new people?" (It is also a disagree (1) to agree (5) scale.) Would answers differ by race or by class? Would those racial differences vary by class?

```
newpeople_raceXclass_model <- lm(midd_opps_newpeople ~ race * class,
                                 data = midd_survey)

summary(newpeople_raceXclass_model)
```

```
##
## Call:
## lm(formula = midd_opps_newpeople ~ race * class, data = midd_survey)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.6667 -1.1237 -0.1237  0.7700  2.1667
##
## Coefficients:
##                                                  Estimate Std. Error t value
## (Intercept)                                      2.250000   0.315981   7.121
## raceAsian/Asian American                         0.592105   0.403612   1.467
## raceHispanic/Latino                              0.795455   0.392815   2.025
## raceOther                                        0.708333   0.386996   1.830
## raceWhite                                        0.943548   0.345208   2.733
## classMiddle Class                                0.305556   0.482668   0.633
## classUpper Middle Class                          0.750000   0.547294   1.370
## classUpper Class                                -1.250000   1.139284  -1.097
## raceAsian/Asian American:classMiddle Class       0.008589   0.577468   0.015
## raceHispanic/Latino:classMiddle Class           -0.851010   0.941537  -0.904
## raceOther:classMiddle Class                     -0.175654   0.564030  -0.311
## raceWhite:classMiddle Class                     -0.253821   0.509734  -0.498
## raceAsian/Asian American:classUpper Middle Class -0.430815   0.633435  -0.680
## raceHispanic/Latino:classUpper Middle Class     -0.962121   0.744096  -1.293
## raceOther:classUpper Middle Class               -0.636905   0.626293  -1.017
## raceWhite:classUpper Middle Class               -0.713524   0.567236  -1.258
## raceAsian/Asian American:classUpper Class        0.907895   1.249286   0.727
## raceHispanic/Latino:classUpper Class            -0.795455   1.597045  -0.498
## raceOther:classUpper Class                       1.958333   1.185303   1.652
## raceWhite:classUpper Class                       1.180163   1.153102   1.023
##                                                 Pr(>|t|)
## (Intercept)                                      2.1e-12 ***
## raceAsian/Asian American                         0.14270
## raceHispanic/Latino                              0.04314 *
## raceOther                                        0.06751 .
## raceWhite                                        0.00639 **
## classMiddle Class                                0.52685
## classUpper Middle Class                          0.17089
## classUpper Class                                 0.27284
## raceAsian/Asian American:classMiddle Class       0.98814
## raceHispanic/Latino:classMiddle Class            0.36630
## raceOther:classMiddle Class                      0.75554
## raceWhite:classMiddle Class                      0.61863
```

```
## raceAsian/Asian American:classUpper Middle Class  0.49659
## raceHispanic/Latino:classUpper Middle Class       0.19632
## raceOther:classUpper Middle Class                 0.30944
## raceWhite:classUpper Middle Class                 0.20873
## raceAsian/Asian American:classUpper Class         0.46757
## raceHispanic/Latino:classUpper Class              0.61854
## raceOther:classUpper Class                        0.09882 .
## raceWhite:classUpper Class                        0.30634
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.095 on 965 degrees of freedom
## Multiple R-squared:  0.03276,    Adjusted R-squared:  0.01372
## F-statistic:  1.72 on 19 and 965 DF,  p-value: 0.02794
```

In this model, none of the interactions are significant. But some of the differences are big. So there could be good theoretical reasons why you would want to keep the interaction terms in. And there could be good methodological reasons why the terms are not significant. For example, small sample sizes could be a cause: in this case there is only one upper class African/African American student.

Earlier we saw how to estimate the predicted values. Here's that code again:

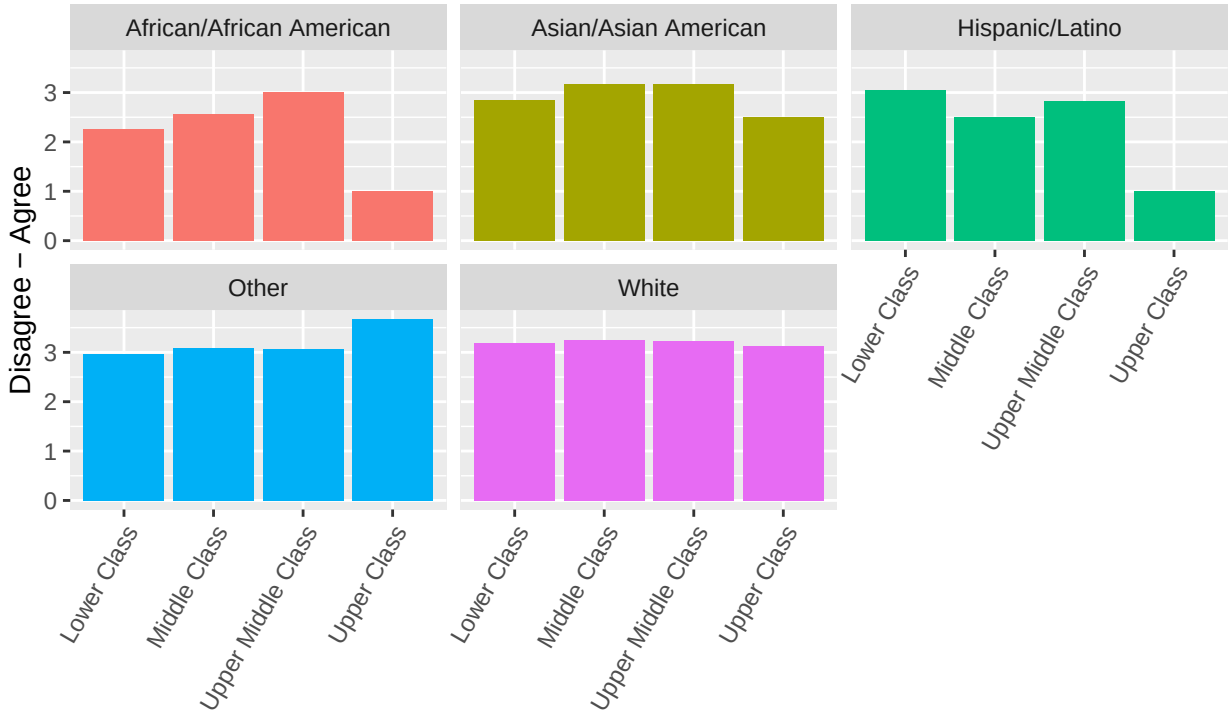
```
midd_survey$pred_newpeople <- fitted(newpeople_raceXclass_model)

new_people_predictions <- midd_survey |>
     group_by(race, class) |>
     summarize(agree_opps_newpeople = round(mean(pred_newpeople),2))
```

```
## `summarise()` has grouped output by 'race'. You can override using the `.groups` argument.
```

This time we'll plot the predictions rather than display them in a table.

```
plot3 <- ggplot(new_people_predictions, aes(x = class, y = agree_opps_newpeople, fill = race))
plot3 + geom_col() + facet_wrap(~race) + guides(fill = "none") +
     labs(x = "", y = "Disagree - Agree",
          title = "Satisfaction With Opportunities to Meet New People",
          subtitle = "Professor Peggy Nelson's Middlebury Survey, Fall 2011") +
     theme(axis.text.x = element_text(angle = 60, hjust = 1))
```


Satisfaction With Opportunities to Meet New People
Professor Peggy Nelson's Middlebury Survey, Fall 2011
African/African American
Asian/Asian American
Hispanic/Latino
Other
White
Disagree – Agree
0
1
2
3
Lower Class
Middle Class
Upper Middle Class
Upper Class