Lab 9

```
require("openintro")
require("dplyr")
```

For today, we'll start with the satGPA dataset in the openintro package, which includes data for 1000 students at an unnamed college. The variables are sex, SATV (Verbal SAT percentile), SATM (Math SAT percentile), SATSum (Total of verbal and math SAT percentiles), HSGPA (high school GPA), and FYGPA (first year of college GPA).

```
data(satGPA)
str(satGPA)
```

```
## 'data.frame':    1000 obs. of  6 variables:
##  $ sex   : int  1 2 2 1 1 2 1 1 2 1 ...
##  $ SATV  : int  65 58 56 42 55 55 57 53 67 41 ...
##  $ SATM  : int  62 64 60 53 52 56 65 62 77 44 ...
##  $ SATSum: int  127 122 116 95 107 111 122 115 144 85 ...
##  $ HSGPA : num  3.4 4 3.75 3.75 4 4 2.8 3.8 4 2.6 ...
##  $ FYGPA : num  3.18 3.33 3.25 2.42 2.63 2.91 2.83 2.51 3.82 2.54 ...
```

Suppose that we want to predict first year college GPAs using students' SAT scores and high school GPA. We also want to explore the relationship between SAT scores/HS GPAs and college GPAs. In order to build our model, we'll look at each covariate separately, then create an additive model (question: why am I not including SATSum in the model with all covariates? What would happen if I included it?):

```
lm_HSGPA = lm(FYGPA ~ HSGPA, data = satGPA)
lm_SATM = lm(FYGPA ~ SATM, data = satGPA)
lm_SATV = lm(FYGPA ~ SATV, data = satGPA)
lm_all_vars = lm(FYGPA ~ HSGPA + SATM + SATV, data = satGPA)
summary(lm_HSGPA)$coef
```

```
##               Estimate Std. Error    t value     Pr(>|t|)
## (Intercept) 0.09131887 0.11788863  0.7746199 4.387478e-01
## HSGPA       0.74313847 0.03634501 20.4467809 6.932446e-78
```

```
summary(lm_SATM)$coef
```

```
##               Estimate  Std. Error   t value     Pr(>|t|)
## (Intercept) 0.62189917 0.140848907  4.415364 1.118346e-05
## SATM        0.03393788 0.002558712 13.263654 4.243134e-37
```

```
summary(lm_SATV)$coef
```

```
##               Estimate  Std. Error  t value     Pr(>|t|)
## (Intercept) 0.70079352 0.129434776  5.41426 7.710336e-08
## SATV        0.03611306 0.002608456 13.84461 5.298818e-40
```

```
summary(lm_all_vars)$coef
```

```
##                Estimate  Std. Error   t value     Pr(>|t|)
## (Intercept) -0.86692662 0.149201736 -5.810433 8.379772e-09
## HSGPA        0.58006828 0.038463344 15.081067 1.920447e-46
## SATM         0.01239756 0.002601136  4.766208 2.156524e-06
## SATV         0.01645881 0.002648354  6.214730 7.546945e-10
```

What do you notice about the coefficients? For example, is the coefficient on HSGPA in the original model the same as the coefficient on HSGPA in the full additive model? . . . No! Why might this be? Let's look at the correlations between these variables:

```
cor(satGPA$HSGPA, satGPA$SATV)
```

```
## [1] 0.3595001
```

```
cor(satGPA$HSGPA, satGPA$SATM)
```

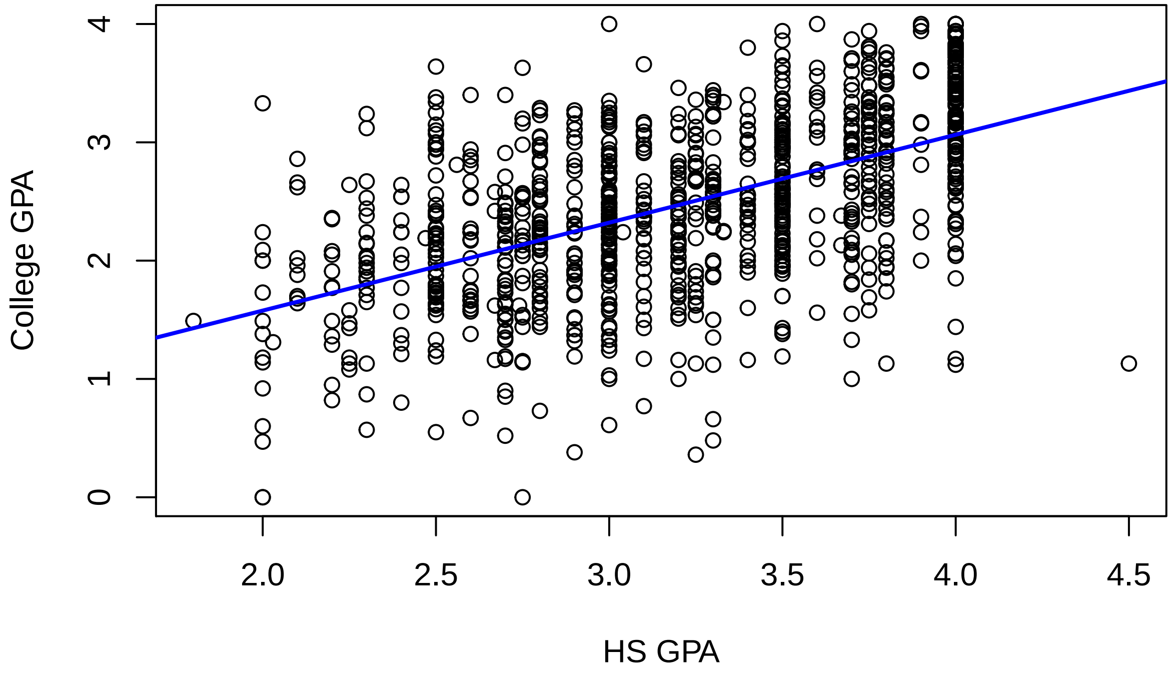
```
## [1] 0.3780704
```

```
cor(satGPA$SATM, satGPA$SATV)
```

```
## [1] 0.4665806
```

Pretend for a moment that you are reporting to your boss (maybe an admissions officer?), who wants to know how HS GPA is related to college GPA. If you didn't have SAT information, you would probably start with the first model:

```
plot(satGPA$HSGPA, satGPA$FYGPA, xlab="HS GPA", ylab="College GPA")
abline(lm_HSGPA, lwd=2, col=4)
```



Based on this plot and regression line, you would estimate that students with a 1 point higher GPA in High School are expected to have first year college GPAs that are 0.74 points higher on average.

However, if you got access to the SAT data, your conclusions would now change. Now you would report: Controlling for SAT scores, students with a 1 point higher GPA in High School are expected to have first year college GPAs that are 0.58 points higher on average.

This happens because all of these variables are positively correlated with each other. Once we include all three, they each account for less of the variation in first year college GPAs than any of them did on their own. This is important to keep in mind when interpreting regression coefficients. The coefficient on any particular variable may change, depending on what other variables are included in the model. And our coefficient estimates will be biased if our model is mispecified.

We can investigate this further via a Shiny App: https://a3sr.shinyapps.io/QM_Bias_Lab/

Just for fun, let's create a bunch of potential models for first year college GPA and compare them:

```r
lm1 = lm(FYGPA ~ HSGPA, data = satGPA)
lm2 = lm(FYGPA ~ HSGPA + SATM, data = satGPA)
lm3 = lm(FYGPA ~ HSGPA + SATV, data = satGPA)
lm4 = lm(FYGPA ~ HSGPA + SATM + SATV, data = satGPA)
lm5 = lm(FYGPA ~ HSGPA + SATM + SATV + SATM * SATV, data = satGPA)
lm6 = lm(FYGPA ~ HSGPA + SATM + SATV + SATM * SATV + I(HSGPA^2), data = satGPA)
lm7 = lm(FYGPA ~ HSGPA * SATM * SATV, data = satGPA)
lm8 = lm(FYGPA ~ HSGPA * SATM * SATV + I(HSGPA^2), data = satGPA)
anova(lm1, lm2, lm3, lm4, lm5, lm6, lm7, lm8)
```

```
## Analysis of Variance Table
##
## Model 1: FYGPA ~ HSGPA
## Model 2: FYGPA ~ HSGPA + SATM
## Model 3: FYGPA ~ HSGPA + SATV
## Model 4: FYGPA ~ HSGPA + SATM + SATV
## Model 5: FYGPA ~ HSGPA + SATM + SATV + SATM * SATV
## Model 6: FYGPA ~ HSGPA + SATM + SATV + SATM * SATV + I(HSGPA^2)
## Model 7: FYGPA ~ HSGPA * SATM * SATV
## Model 8: FYGPA ~ HSGPA * SATM * SATV + I(HSGPA^2)
##   Res.Df    RSS Df Sum of Sq       F    Pr(>F)
## 1    998 386.38
## 2    997 365.27  1   21.1170 59.9221 2.427e-14 ***
## 3    997 359.65  0    5.6156
## 4    996 351.63  1    8.0200 22.7577 2.113e-06 ***
## 5    995 351.41  1    0.2200  0.6244    0.4296
## 6    994 350.53  1    0.8787  2.4933    0.1146
## 7    992 349.48  2    1.0551  1.4971    0.2243
## 8    991 349.24  1    0.2419  0.6863    0.4076
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Data analysis

For the second part of this lab, we'll return to the 2012 US census dataset. To make things a little more manageable, I've provided some code to subset the data to a smaller number of variables.

```r
data(acs12)
acs12 = acs12 %>% select(income, gender, edu, age, employment, hrs_work, race) %>% na.omit()
```

The goal for today is to come up with some potential models (make sure they are nested) for predicting income and compare them. Can you find the best model?? You should come up with at least 5 potential models. At least one of your models should include an interaction term and a higher order polynomial term.