Mathematica curates extensive datasets. I downloaded one of their datasets on U.S. States: `states_m.csv`. `read.csv` this file[1], look at its `str`. 51 observations (includes DC) of 64 variables. My focus is on `GiniIndex`—a measure of income inequality, where larger Gini means more unequal distribution of income. Clearly with 64 features (columns) and 51 'states' (rows) overfitting is going to be easy. Lets start with the twenty (essentially random) features that follow the column (27) for `GiniIndex`.

```
out0=lm(GiniIndex~.,data=df[,27:47])
summary(out0)
```

In all the following models (but not this one, for I hope obvious reasons) record the 'Residual standard error', 'Adjusted R-squared', and AIC (Akaike's Information Criterion). Report any/all significant variables. This model should have resulted in zero residuals (a 'perfect' fit that means nothing). Let's cut the variables down a bit

```
out1=lm(GiniIndex~.,data=df[,27:37])
summary(out1)
AIC(out1)
```

Interesting: the higher the state's elevation the better (smaller) the Gini. I think I could make up somewhat plausible explanations for these relationships! Would you be convinced?

I've sorted many of the data.frame's features into broad groups. I want you to use the `pairs` function (which *xy* plots all the combinations of pairs of columns) to quickly examine the relationships between the various features in a group, and then pick out two features (a pair, not to be confused with the `pair` R function) to represent that group. Use whatever criteria you feel like for your pair. Continue for all my groups:

```
pairs(df[,c(3:7,9:12,25:26)])                    #education
pairs(df[,c(1,13,16,23,35,41,42,51,52,57)])      #crime
pairs(df[,c(15,17,27,29,32,38,40,46,47,54)])     #economics
pairs(df[,c(30,33,37)])                          #health
pairs(df[,c(20,21,28,54)])                       #government
pairs(df[,c(2,18,19,29,31,34,36,49,50,58)])      #msc stats
pairs(df[,c(8,32,39,44,45,48)])                  #houses
pairs(df[,c(55:56)])                             #voting
```

Do note the mix of colons (e.g., `3:7=3,4,5,6,7`) and commas in these lists. The first elements of my selected collection of pairs was

```
AverageACTCompositeScore
CrimeRate
PerCapitaPersonalIncome
HealthInsuranceCoverageRate
FederalGovernmentExpenditurePerCapita
Employment
MedianSalePrice
TotalVotingRate
```

[1] What worked for me on GitHub was: navigate to file, click on it (reports too big), click Raw, (appears in browser), use browser 'Save Page As' functionality. On Friday's larger files, Download was the offered option rather than Raw. Apparently GitHub is not reporting the MIME type as text/csv.

It will help to make reduced data.frames with just the relevant variables (including `GiniIndex`). Clearly you will use—and hence need to determine—the column numbers of your selected variables (and Gini), for me something like:

```
dfA=df[,c(3,16,47,30,21,18,39,56,27)]
```

Similarly make a reduced data.frame `dfB` with the second elements of your pairs. For both `dfA, dfB` make a linear model using all your selected variables, e.g.,

```
outA1=lm(GiniIndex~., data=dfA)
```

then try another linear model where all the significant variables of this first test model plus the lowest p-value variable of the insignificant variables are in the model. Repeat the process with `dfB`. I sure hope you don't have much confidence in the importance of your results! Correlation most certainly does not mean cause[2]! For example, I don't think you can sell a program to haul rocks to mountain tops based on the correlation between state maximum elevation and improved Gini. This process would be more convincing if we had a larger dataset (say counties not states) and used the proper test/train methodology.

Three years ago, I collected together a bunch of data on liberal arts colleges/universities into the file `schools.csv`. Find it on GitHub; read it into R. The column names I hope are clear; some harder examples: `UnitID`= a government supplied ID of no consequence to this project, `Istaff`= number of instructional staff, `fresh`= number of first year students, `undergrad`= number of undergraduate students, `SFR`= student/faculty ratio, `Psalary`= average professor salary, `USNews`= ranking by USNews. Aim: understand how to achieve a better (lower) USNews ranking. As usual since `USNews` ranking spans more than a decade, you will be making linear models of `log(USNews)`. Play around with fitting the data, come up with the most credible model you can. Print out the summary of that model. Write in words, as if to your boss, starting: "Correlation does not imply causation, but", you might try to change this set of dependent variables (report whether to increase or decrease) in order to reduce (i.e., improve) the rank. At the same time hand your boss a copy of the cartoon in footnote on this page. My collection of saved xkcd cartoons: `www.physics.csbsju.edu/xkcd`.

---

p. 123, problem 10. . . start:

```
library("ISLR")
str(Carseats)
out=lm(Sales~Price+Urban+US,data=Carseats)
summary(out)
```

[2] `https://xkcd.com/552/`