1 Can observational studies replicate experiments?

In week 5 we studied data from the National Supported Work (NSW) experiment. Men were randomized to either receive job training (the treatment) or not (control). We found the average treatment effect was $1,794.34: treated men could expect to earn this much more than controls after the training. What if we didn’t have a randomized experiment? Could we still estimate this causal effect from observational data? Answering these questions is the goal of this problem.

What is the fundamental problem? We want to estimate the benefit that the 185 men got, on average, from going through the treatment. In lecture 5 we used the difference between the mean in the treated group and the mean in the control group to estimate this average treatment effect:

$$ b_T = \hat{\beta}_T = \bar{Y}_{T=1} - \bar{Y}_{T=0}. $$

We can still find the mean in the treated group, $\bar{Y}_{T=1}$, but we lack a control group to fill the second piece. Individuals may have nonrandomly selected to participate in the NSW experiment, so there is a sample selection problem here.

(This is one of the most important, most studied problems in economics and statistics, and the particular data set you’re about to use is one of the most-studied data sets in labor economics; many, many researchers have done (versions of) what you are about to do.)

Suppose that the same 185 men received job training, and we want to know what effect this training had on income, but we do not have access to the NSW control sample. That is, no experiment was done. We still worry about nonrandom selection into treatment, because even without the experimental control group, it’s still true that these 185 treated men were not randomly selected from the population at large: they signed up to participate in the NSW program and receive training. Therefore, we need a control sample to compare against. For this we have 2490 men drawn from the Panel Study of Income Dynamics (PSID). This is observational data. The PSID control sample is not a proper stand-in for a real randomized control group: the people in this data did not sign up to participate in a job-training randomized trial, and so are qualitatively different the NSW treated sample. Most formal research refers to the PSID as a “comparison” group for just this reason.

The full data set (file nsw_psid.csv) combines these two: the treat indicator is one for the 185 treated men and zero for the 2490 men from the PSID.

(a) Using the PSID control sample as though it were the control group for a randomized trial, estimate the average treatment effect. That is, compute $b_T$ just as above. Compare your results to what we found in lecture 5.

(b) Does the PSID sample appear to be a good control group for this purpose? What characteristics of the men help answer this question? Provide data-based evidence and discussion for your conclusion, either way you decide. How does this shed light on your finding in (a)?
(c) Using the above analysis to guide you, build a regression that attempts to “control” for any sources of nonrandomization. Does using the partial $F$-test help you further select/remove variables? Does your regression-based treatment effect estimate recover the experimental benchmark treatment effect estimate? Discuss the uncertainty of your regression-based estimate.

The regression you ran in part (c) follows the lessons we learned in lecture 5. Now we will push beyond our understanding a little bit, and use a more sophisticated [Note that I didn’t say better!] approach to using observational data in an attempt to recover the experimental benchmark causal effect.

The key insight we exploit here is that we don’t have to generate a perfect control sample, we just need a good stand-in for $\bar{Y}_{T=0}$, which is an average. More precisely, the number we need is the average income the 185 men would have had if they were instead assigned to the control group.

In the following steps we will re-weight the PSID data so that it acts like a good control sample on average using the information in the $X$ variables. We will use a technique called weighted least squares. Multiple linear regression solves (from lectures 1 and 3):

$$\min_{b_0, b_1, \ldots, b_k} \frac{1}{n} \sum (Y_i - b_0 - b_1 X_{1,i} - b_2 X_{2,i} - \cdots - b_k X_{k,i})^2.$$  

If each observation has a weight $W_i$ attached, then we can use weighted multiple linear regression, which solves

$$\min_{b_0, b_1, \ldots, b_k} \frac{1}{n} \sum W_i (Y_i - b_0 - b_1 X_{1,i} - b_2 X_{2,i} - \cdots - b_k X_{k,i})^2.$$  

Intuitively, we let some observations contribute more or less to fitting the line. In class, each point was treated equally, so that every point influenced the least squares line according to its own leverage. Now, even beyond different points having different leverage (which, remember, depended only on the $X$ values), points can have different weights altogether.

Weighted analyses are very common, and very useful any time you know that some observations are more important than others. For example, it is often true that some consumers are more important than others (more popular than ever with social media “taste-makers”); some markets are more important to a business than others (e.g. Pepsi probably does not care about its sales in Atlanta, the headquarters of Coca-Cola), and so forth.

(d) Estimate $P[T = 1 | X]$ (called the “propensity score”) using logistic regression on the full data. Justify your choice of control variables. You may only use pre-treatment variables, that is, information known before treatment is assigned and carried out, and so in particular you cannot use the outcome income.after to predict treatment.

Notice here how we don’t necessarily care how each $X$ variable is associated with the decision $T = 1$ vs. $T = 0$. In the language of our class, we are not interested in relationship/inference type questions here, this is a prediction problem. With binary (or categorical) outcomes, this is a classification problem: we want to build the best prediction model for $T$, i.e. the best classifier for grouping the observations into the two categories $T = 1$ vs. $T = 0$ based on their characteristics.

(e) Define weights for each observation as

$$W_i = \frac{1}{1 - \hat{P}[T = 1 | X]}.$$  

and run weighted least squares on the PSID data. (Look at the weights option in lm.)

(f) Use your regression to predict, for each treated observation, what their income would have been had they been assigned to control (this is called the “counterfactual” outcome).
Create the final treatment effect estimate by averaging your predictions and compare it to the experimental benchmark. \textit{(Do not worry about obtaining confidence intervals, it’s not as easy.)}

2 Predicting Restaurant Ratings

From the website Yelp we have star ratings of 10,668 restaurants (file \texttt{yelp.csv}), total number of reviews, and various characteristics of the restaurant taken from the reviews themselves (e.g. type of parking, types of meals the place is good for, etc). The goal is to predict how many stars a restaurant will get based on its characteristics. The outcome variable is \texttt{stars}.

(a) Decide which linear model or generalized linear model (GLM) discussed in class is best suited to model \texttt{stars}, potentially after transforming \texttt{stars} in some way (that is, the raw variable \texttt{stars} may not fit in any GLM, and you may have to transform it to fit). State your reasoning.

(b) Using the (G)LM you selected, model \texttt{stars}, or your transformation of it, as a function of all the characteristics of the restaurant using the appropriate generalized linear model, both with and without \texttt{review_count} (which gives the total number of reviews) as a control variable. Comment on any differences in the conclusions from these two models.

(c) Using your favorite model selection technique construct a model to predict star ratings (or your transformation), considering main effects, interactions, nonlinear terms, dummies, etc. Interpret your findings. (Use \texttt{review_count} if you want to.)

(d) A Model Building Contest!

\textit{(This is ungraded of course, and if your group doesn’t want to, or can’t, participate, that’s fine.)}

Now let’s see how good your model really is! We will have a little in-class competition to see which group’s model gives the best predictions. At the beginning of class in week 9, I will use data on 5,000 \textit{new} restaurants to test each group’s final model from part (c). These restaurants come from the same original data as \texttt{yelp.csv}; I held out 5,000 for this testing, and given you the above 10,668 for training.

The goal is to minimize the out-of-sample prediction MSE for this new data. To make this fair the MSE will be based on the actual star ratings, not any transformation. The group with the lowest MSE wins!

To be able to run your model in class, we need code that:

- Creates the new outcome by doing whatever transformation of \texttt{stars} you settled on in part (a), if any, and creates any new variables such as dummies, transformations, etc, needed for your model. For example:
  
  \begin{verbatim}
  yelp$Expensive <- (yelp$Price.Range==4)
  yelp$log.reviews <-log(yelp$review_count)
  mean.stars <- mean(yelp$stars)
  sd.stars <- sd(yelp$stars)
  yelp$stars2 <- (yelp$stars - mean.stars)/sd.stars
  yelp$stars <- NULL
  \end{verbatim}

- Estimates your final model, such as
your.model <- glm(stars2 ~ . + log.reviews*(Delivery + Good.for.Kids),
                 family="gaussian", data=yelp)

• Gets predictions from the model, and puts these predictions back on the scale for stars, if necessary, i.e. undoing the transformation from part (a). Continuing the above example:
  
  your.pred <- predict(your.model, newdata=yelp.contest.data, type="response")
  predicted.stars <- your.pred*sd.stars + mean.stars

The day before class, please email me a separate code file with all of the above that will run from start to finish on the file yelp.csv. An example of a complete code file is below.

#Example code for model building contest
#read in original/training data
  yelp <- read.csv("yelp.csv")
#new variables
  yelp$Expensive <- (yelp$Price.Range==4)
  yelp$log.reviews <- log(yelp$review_count)
  mean.stars <- mean(yelp$stars)
  sd.stars <- sd(yelp$stars)
  yelp$stars2 <- (yelp$stars - mean.stars)/sd.stars
  yelp$stars <- NULL

#estimate the model on the training data
  your.model <- glm(stars2 ~ . + log.reviews*(Delivery + Good.for.Kids),
                    family="gaussian", data=yelp)

#Get predictions on the contest data
#create the same new variables
  yelp.contest.data <- read.csv("yelp.contest.data.csv")
  yelp.contest.data$Expensive <- (yelp.contest.data$Price.Range==4)
  yelp.contest.data$log.reviews <- log(yelp.contest.data$review_count)

#get the predictions
  your.pred <- predict(your.model, newdata=yelp.contest.data, type="response")
#undo the transformation
  predicted.stars <- your.pred*sd.stars + mean.stars

#Get predictions on the contest data
  your.MSE <- mean((predicted.stars - yelp.contest.data$stars)^2)