Lecture 05 - The Logic of Matching and Logistic Regression

Samuel DeWitt

The Logic of Matching and Logistic Regression

In place of an RCT (randomized control trial, for those who do not remember), researchers have sometimes justified the use of **matching** to pair units similar in characteristics that are assumed to have an impact on both selection into treatment and subsequent outcomes.

Matching procedures can range from the very complicated - e.g., inverse probability weighting - to the very simple - e.g., individual matching on select variables.

The *logic* of matching requires some review before we get into the procedures generating a *good* match (and, for that matter, how we define a **good match**).

The Logic of Matching

As a mental exercise, imagine we have a sample of 100 units and can randomly select half of those units to receive treatment ($D = 1$) and the other half to be in the control group ($D = 0$).

Let's also say that we know some characteristics of these units can have a profound impact on their reaction to treatment. In this example, treatment is twice as effective for units where $X_1 = 1$ as compared to units where $X_1 = 0$.

Under random assignment, we assume that the groups will be approximately evenly distributed with respect to X_1, so the treatment having a different effect across groups of units with different values of X_1 is expected to be inconsequential for estimates of the impact of treatment.

The Logic of Matching

Let's suppose, however, that this treatment is a prison sentence and, as such, we may not randomly assign individuals to go to prison. Let's also suppose that there are no other options for exploiting randomization in who goes to prison and who does not.

Is the distribution of X_1 across these groups more consequential now than under random assignment?

The Logic of Matching

Let's suppose, however, that this treatment is a prison sentence and, as such, we may not randomly assign individuals to go to prison. Let's also suppose that there are no other options for exploiting randomization in who goes to prison and who does not.

Is the distribution of X_1 across these groups more consequential now than under random assignment?

If you said yes, you are correct. If that characteristic can influence both selection into being treated (i.e., going to prison) and subsequent outcomes (e.g., recidivism), any departure from an even distribution between the groups could bias an estimate of the true treatment effect.

The Logic of Matching

One way to mitigate this issue is to use some form of matching.

Matching comes in various shapes and sizes - the most rudimentary of which would be a simple 1-unit to 1-unit match on X_1.

In simple terms, we select a matching unit in the control group with $X_1 = 1$ for every unit in the treatment group where $X_1 = 1$ until we exhaust either A) control group units where $X_1 = 1$ or B) treatment group units where $X_1 = 1$.

This will continue for units where $X_1 = 0$ until we run out of units to match.

The Logic of Matching

Once we have paired each treatment unit with a control unit, we then estimate the treatment effect as the average difference $(Y_1 - Y_0)$ across each pair.

$$\text{Effect of } D = \frac{\sum(Y_1 - Y_0)}{n_{pairs}}$$

We can compare this estimate to what we would have obtained without matching in order to see how much our inference would have been affected by bias had we done a simple comparison of group averages.

The Logic of Matching

Matching on a *very* small number of simple variables can be plausible in some scenarios where you have a large enough sample size.

However, matching on a larger number of even simple variables, or a smaller number of complicated variables, can create significant issues.

These types of issues are more generally called **curse of dimensionality** problems, where the additional complexities of a more variables (or more complicated variables) creates more sparse distributions of potential cases for matching.

Simple Matching and the Curse of Dimensionality

Simple case: One variable with two values - Gender (Male = 0, Female = 1)

n=100, 50 in treatment, 50 in control; evenly split by gender

We then have 25 pairs of men and 25 pairs of women if we want to match on gender.

Simple Matching and the Curse of Dimensionality

Now, let's complicate matters: One more matching variable with five values - Age (20-29 = 1; 30-39 = 2; 40 - 49 = 3; 50 - 59 = 4; 60+ = 5).

n=100, 50 in treatment, 50 in control, evenly split by gender and age category.

How many gender-age categories do we now have?

Simple Matching and the Curse of Dimensionality

Now, let's complicate matters: One more matching variable with five values - Age (20-29 = 1; 30-39 = 2; 40 - 49 = 3; 50 - 59 = 4; 60+ = 5).

n=100, 50 in treatment, 50 in control, evenly split by gender and age category.

How many gender-age categories do we now have?

2*5 = 10 total categories, one category for each gender-age combination.

Simple Matching and the Curse of Dimensionality

Hopefully this illustrates the nature of the problem - simple matching becomes more difficult very quickly even when you have two relatively simple variables.

As a general rule for all types of analyses, the more variables you have, the larger the sample you will need to properly estimate that variable's effect (or to find enough matching pairs to estimate its effect).

Simply stated, simple matching is fine when the problem is of low dimensionality - i.e., few uncomplicated variables influence treatment - but quickly becomes impractical as the process being analysed increases in complexity (dimensionality).

How Do We Match Instead?

One other suitable idea for matching has to do with matching individuals on the *probability* they would be in some group that has been **treated**.

In an observational study, this amounts to collecting data on some dichotomous **treatment** variable and also on a host of variables we think to influence whether someone has a 1 on that variable (i.e., $D = 1$) or a 0 (i.e., $D = 0$).

We then need to implement a model that produces a predicted probability that $D = 1$ given what we know about the part of our sample where D is actually 1 (i.e., by looking at their values on the other variables we think are correlated with being **treated**).

The Logistic Regression Model

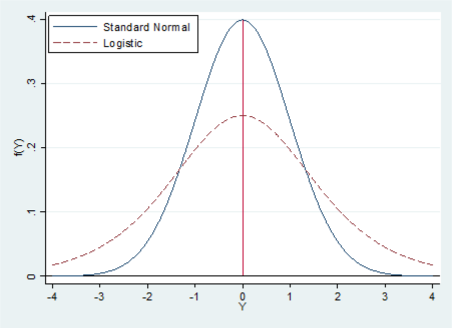
This brings us to the logistic regression model.

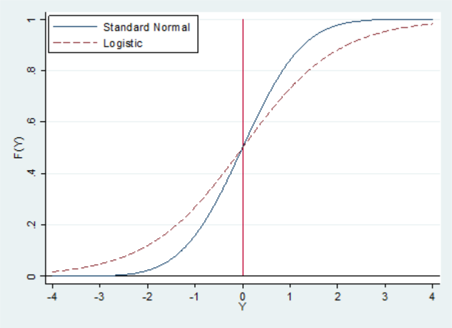
A logistic regression model is a special form of an OLS model that is specifically designed to accommodate dichotomous outcomes variables.

Although OLS (multivariate regression) can be estimated using a dichotomous outcome (this is known as a *linear probability model*) in practice it is rarely used to do so because of some well known violations of regression assumptions.

Comparing OLS and Logistic Regression

The distribution plots are a bit different - on the left is the **standard normal** and a *transformed* **logistic** distribution. On the right is a *transformed* **normal distribution** (now a *probit*) and the regular form of the **logistic** distribution.





Comparing OLS and Logistic Regression

Equation for OLS:

$$Y = \alpha + X\beta + \epsilon$$

Equation for Logistic Regression:

$$ln\left(\frac{p}{1 = p}\right) = \alpha + X\beta_1$$

Where the $ln\left(\frac{p}{1-p}\right)$ term is referred to as a log odds, or, the logarithm of the odds ratio. p is the probability that a unit belongs to the **treated** group and $1 - p$ is the probability that unit belongs to the **untreated** group.

Variables on the right-hand side of this equation help us to predict the **log odds** that a given unit is **treated** or not given the association between these variables and actually observed units that are treated.

Comparing OLS and Logistic Regression

We interpret the outcome variable a little differently between the OLS and logistic models, also.

The outcome in a logistic regression - the log odds of belonging to the treatment group, is not actually **observable** as it is in OLS - that is, we cannot observe the probability that a person belongs to this group, only if they do $(D = 1)$ or do not $(D = 0)$.

In other words, we consider this probability as a **latent variable** underlying the assignment process - a variable we **do not observe** but that we expect to influence whether someone becomes **treated** or not.

Predictor (or independent) variables included in the logistic model influence this **latent probability**, adjusting the **log odds** of being in the treatment group based upon one-unit changes (similar to their interpretation in OLS).

An Applied Example - Predicting Delinquency

The NLSY97 data we have been using so far this semester contains a few examples we could use to demonstrate how a logistic regression works.

For the purposes of this example, I will be using a version of the NLSY97 data set.

We want to predict the log odds that a youth will report engaging in at least one type of delinquent activity since the date of the last interview.

I computed this variable by simply running an **ifelse()** command to the number of crimes a youth reports since the last interview. If that value is greater than 0, the dummy variable prefixed by **delinq** has a value of 1 and 0 otherwise.

An Applied Example - Predicting Delinquency

To employ logistic regression in R we need to use the glm() function, which stands for **generalized linear model**

What this term means is that logistic regression is a **generalization** of OLS for dichotomous outcomes.

The parameters (estimates) from the model are interpreted similarly - we still assume there's a **linear relationship** between our independent and dependent variables.

However, the linear relationship in a logistic regression is not assumed to be linear in the **latent probability** of belonging to the **treated** group. This probability is assumed to be continuous, but we may only observe its two different outcomes in practice.

An Applied Example - Predicting Delinquency

```
logit1<-glm(delinq99~log_num_crimes+varscore98+
            bad_peers+age, data=NLSY97HW1,
            family=binomial)
```

Notice that the command is quite similar to a linear model (lm()) but I have to specify family=binomial as an option.

This tells R that the distribution function for the outcome variable is binomial - the outcome has just two values (0/1) and, therefore, to estimate a logistic regression.

Other family options are also possible, including Gaussian (normally distributed) or Poisson (count variable), to name just two.

```r
summary(logit1)
```

```
##
## Call:
## glm(formula = delinq99 ~ log_num_crimes + varscore98 + bad_peers +
##     age, family = binomial, data = NLSY97HW1)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.5330  -0.6452  -0.5727  -0.5041   2.1341
##
## Coefficients:
##                 Estimate Std. Error z value Pr(>|z|)
## (Intercept)     0.337864   0.300424   1.125   0.2607
## log_num_crimes  0.311676   0.043907   7.099 1.26e-12 ***
## varscore98      0.404286   0.039250  10.300  < 2e-16 ***
## bad_peers       0.013268   0.006298   2.107   0.0351 *
## age            -0.143690   0.021779  -6.598 4.18e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 9679.3  on 8983  degrees of freedom
## Residual deviance: 8665.9  on 8979  degrees of freedom
## AIC: 8675.9
##
## Number of Fisher Scoring iterations: 4
```

An Applied Example - Predicting Delinquency

How do we interpret those coefficients?

Well, it's similar to how we interpret coefficients from an OLS model with one important caveat - the influence of the predictor variable is on the **log odds* of the dependent variable being equal to 1.

So, for the *varscore98* variable, a one unit increase in the different types of crimes a youth commits increases the **log odds** of being delinquent at the next wave by 0.404286.

The **log odds** is still a bit unclear though, as it is quite literally the *logarithm of the odds ratio*, which is difficult to interpret naturally.

An Applied Example - Predicting Delinquency

To make interpretations a bit easier, we can *exponentiate* the **log odds**, which will return the original odds ratio (which has a more intuitive interpretation).

We can do so by using the **coefficients** list in the stored logistic regression model:

```
odds<-exp(logit1$coefficients)
odds
```

```
##    (Intercept) log_num_crimes     varscore98      bad_peers            age
##      1.4019503      1.3657120      1.4982319      1.0133563      0.8661559
```

An Applied Example - Predicting Delinquency

This is how we would interpret those coefficients:

Logged Number of Crimes

- A youth who is one unit higher on the *logged number of crimes* is 1.3657120 times as likely to be delinquent in the following year than a youth who is one unit lower on the *logged number of crimes*
  - Stated differently, a youth who is one unit higher on the *logged number of crimes* is 36.57120% **more likely** to be delinquent in the next year than a youth who is one unit lower on the *logged number of crimes*.

Age (in years)

- A youth who is one year older is 0.8661559 times as likely to be delinquent in the following year than a youth who is one year younger.
  - Stated differently, a youth who is one year older is 13.38441% less likely to be delinquent in the next year than a youth who is one year younger.

An Applied Example - Predicting Delinquency

If you take a look at the fitted values from the logisitic regression you will notice they all lie between 0 and 1 - values indicate the **latent probability** that a particular case has a 1 on the outcome given their values on the predictor variables in the regression model.

```r
summary(logit1$fitted.values)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.1014  0.1465  0.1710  0.2295  0.2364  0.9622
```

An Applied Example - Predicting Delinquency

The basic logic of propensity score matching rests upon there being overlap in these probabilities across youth who are and are not delinquent, but each share an approximately similar probability of being delinquent.

Logically, we can then **pair** these observations to obtain a suitable **counterfactual** comparison. The suitability of the **counterfactual** will be directly related to the strength of the logistic regression which produces the **latent probability**,

We will discuss propensity score models in more detail in a following lecture.

The End

What are you still doing here?