Abadie (2005) begins his paper on semi-parametric difference-in-differences (DiD) estimators by quoting Daniel Mcfadden:

> A good way to do econometrics is to look for good natural experiments and use statistical methods that can tidy up the confounding factors that nature has not controlled for us.

Abadie notes that a fundamental assumption behind DiD estimators is that the average outcomes for the treated and control groups would have followed parallel paths over time [1]. This assumption is untenable, and Abadie offers a semi-parametric method to clean up variation induced by non-parallel paths. Suppose, further, that the paths are non-parallel *and* the lag structures in the treatment and control groups are variable and not perfectly matched through time. That is, suppose that there is relative stretching in the outcome time series, and that this stretching is exhibited at the time of treatment. The lag structure will be relegated to the residual variance, potentially inducing bias in the estimated treatment effect. We propose in this short section a way to similarly clean up confounding factors by examining the patterns in the error structures across the treatment and control groups — a sort of information theoretic approach to identification.

Dynamic time warping (DTW) is a method commonly used in time series classification. The DTW method iteratively searches for non-linear alignments in two series, based on short-term patterns that are not necessarily in phase. Standard DiD estimators rely on perfectly vertical Euclidean distance to calculate the difference between the two outcome time series. This definition of distance between the series is severely restrictive, and can potentially introduce bias in the presence of non-linear lag structures. If the general patterns are similar between the treatment and control groups, we may be able to better align the series for a more correct identification of the treatment effect.

Consider, as an illustration, the following data generating process:

$$y_{it} = \beta_i + \gamma D_{it} + \sin(t/\alpha) + \epsilon_{it}, \tag{1}$$

with $t = 1, 2, \ldots, 100$; D_{it} an indicator of treatment; and $\epsilon_i \sim N(0, 1/4)$. The parameter α is inversely related to the frequency of the oscillations in y_i. Define the composite error to be $u_{it} = \sin(t/\alpha) + \epsilon_{it}$, and note that as long as $\mathbb{E}(D_{it}u_{it}) = 0$, then we can consistently estimate the treatment effect γ with a DiD estimator. Specifically, define G_i to be a group indicator of individuals in the treatment group, and define P_i to be a binary indicator of the post-treatment period. The DiD estimator of the treatment effect can be recovered with the following regression:

$$y_{it} = \beta_0 + \beta_1 G_i + \beta_2 P_t + \gamma(G_i \cdot P_t) + u_{it} \tag{2}$$

If the systematic time structure in the error is the same for the treatment and control groups, then the DiD estimator will be consistent.

We can observe this in code, which may help solidify the intuition. The following function creates a `data.frame` object with random errors, which will be useful in simulations for data preparation. Note that we set $\beta_0 = 1$, $\beta_1 = 3$, $\alpha = 5$, and $\gamma = 1/2$ from Equation (1), purely for illustration.

```
random.data <- function(T = 100, alpha = 5) {
  t <- 1:T
  e0 <- rnorm(T, sd=0.25); e1 <- rnorm(T, sd=0.25)
  time.error <- sin(t/alpha)
  P <- ifelse(t > T/2, 1, 0)

  y0 <- 1 + time.error + e0
  y1 <- 3 + time.error + 0.5*P + e1
```

```
  data <- data.frame(rbind(cbind(y0, t, 0, P), cbind(y1, t, 1, P)))
  names(data) <- c("y", "t", "G", "P")
  return(data)
}
```

The function yields a 200×4 data frame, indexed and ready for DiD estimation:

```
tail(random.data())
```

```
           y   t G P
195 3.520060  95 1 1
196 3.864511  96 1 1
197 4.276846  97 1 1
198 4.021589  98 1 1
199 4.121730  99 1 1
200 4.609750 100 1 1
```

Then define a function to extract the DiD treatment estimator:

```
treatment.est <- function(df) {
  m <- lm(y ~ 1 + P + G + P*G, data=df)
  return(m$coefficients[["P:G"]])
}
```

We then run a simulation with 1,000 iterations to check to see if the estimator seems unbiased. The results of the following simulation are graphed in Figure 1:

```
treatment <- sapply(1:1000, function(x) {treatment.est(random.data())})
```

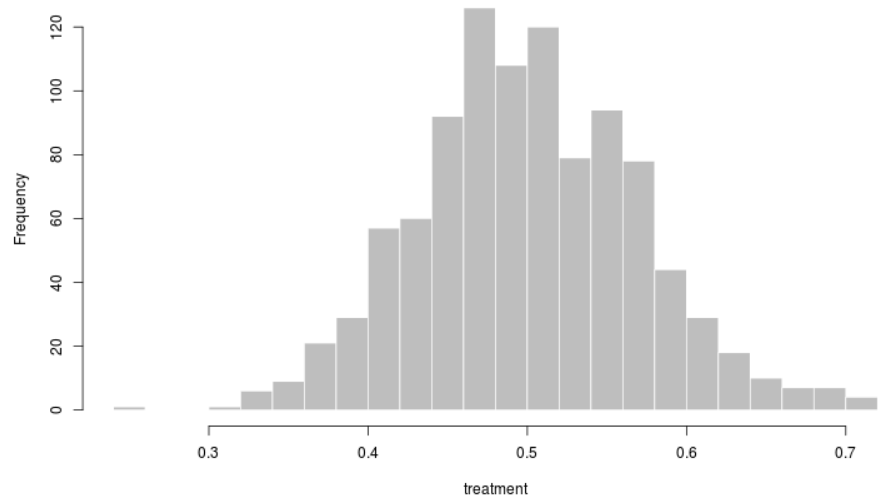

Figure 1: Frequency of treatment estimates for a simulation with 1,000 iterations

Suppose, now, that the time lag structure is different between the treatment and control groups. Moreover, assume that the lag structure within the treatment group changes over time. We can model this type of behavior using a random walk, which defines the lag structure in each period. Consider a cumulative binomial process which has been normalized and scaled:

```
slow.factor <- cumsum(rbinom(100, 1, p=0.1))
```

This process in non-decreasing and remains in the closed interval $[0, 2]$. If we add this to the α parameter for the treatment group, then the sinusoidal error will slow throughout the time interval.

```
alpha.treatment <- alpha.treatment + 2 * slow.factor/max(slow.factor)
```

The time component of the error term for the treatment group will become stretched, relative to the time component of the error term for the control group. Figure 2 shows the new outcome variables for the treatment and control groups over time. The treatment time series progressively drifts further from the control time series. If we run the simulation on the new data generating process, then the mean of the treatment effect distribution is 0.607 with a standard deviation of 0.145 for 1,000 iterations (shown in Figure 3). The DiD method *overestimates* the treatment effect because the error processes are out of phase.

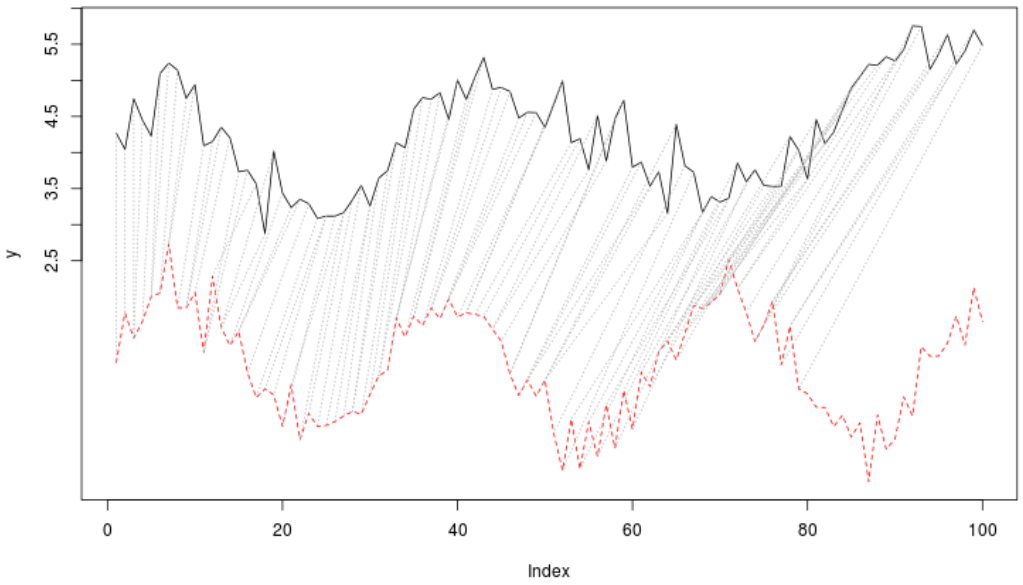

Figure 2: Outcomes for **treatment** and **control groups**, where treatment error drifts; grey lines indicate the match from the dynamic time warping algorithm

I won't go into the detail about the time warping algorithm right now; but essentially the DTW method aligns the two sequences by (1) building a distance matrix and (2) searching for a path through the matrix that minimizes cumulative distance. The normalized, total distance between the two sequences is usually the value of primary interest. With this, the target sequence can be classified based on an array of reference sequences. This is similar to the way that Siri processes speech. Siri will identify the word s-t-a-t-s, whether I say "stats" or "staaaaats" — likely using some variant of the DTW algorithm. In the context of identifying the treatment effect, and specifically for the warped DiD estimator, we reconstruct the treatment sequence based on the matched values in the control sequence. As an illustration, consider the match lines in Figure {fig:dtw}, which identify similar patterns in the unexplained variance of the outcome. If we run the same simulation as before, with 1,000 iterations, the estimated treatment effect is 0.484 with a standard deviation of 0.172 — much closer to the true parameter. The standard DiD estimator constrains the match to perfectly vertical lines, which will ignore the variable lag structure, thereby biasing the estimate of the treatment effect.

As with any empirical method, DTW only works in certain circumstances; and if it is misused, it may induce bias in the parameter estimate or balloon the standard errors. But it seems to work in certain circumstances — which I will describe later, if useful.

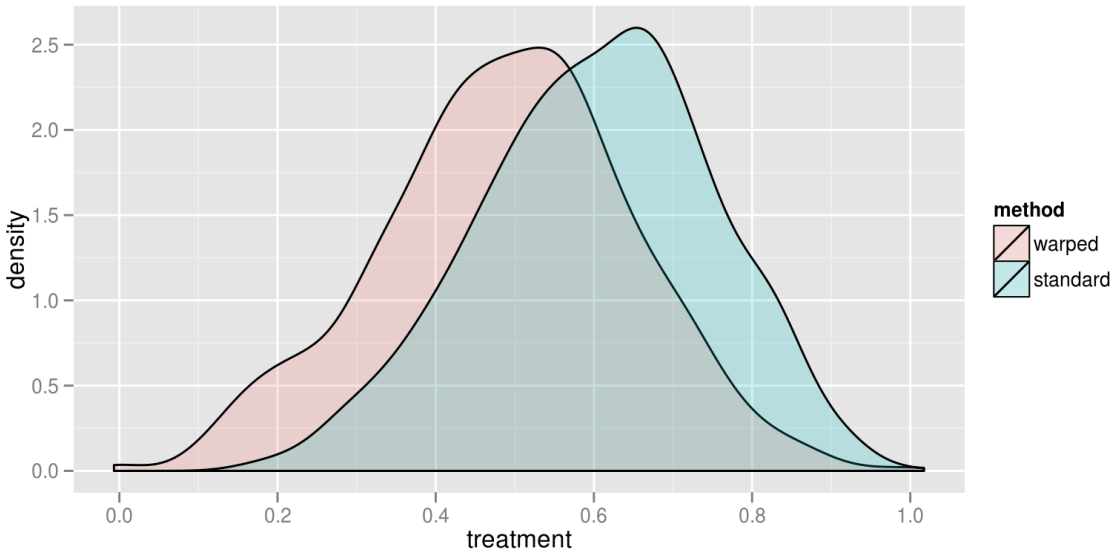

Figure 3: Treatment effects from simulated data

References

[1] A. Abadie. Semiparametric difference-in-differences estimators. *Review of Economic Studies*, 72(1):1–19, 2005.