An Initial Assessment of a Right to Counsel

Patrick Power **Shomik Ghosh** **Markus Schwedeler**

Boston University

Github Repository

Abstract

As Economists, we're interested in understanding how policies scale. Specifically, whether there are unintended consequences that that might negate or work against the positive effects that we see in small scale, randomized control settings. In this paper, we assess whether the right to counsel initiative (already adopted in 14 cities and 2 states) exhibits the unintended consequence at scale that some have voiced concern over. Exploiting the staggered roll-out of a Right to Counsel within the state of Connecticut, we examine the extent to which the policy may have actually hurt those with the greatest housing instability. Using deep learning estimation techniques, we find little evidence to suggest that such a policy has noticeable unintended consequences.

Contents

1 Context

1.1 Why is this Important?

- “Pandemic has highlighted the importance of housing . . . eviction was a problem before the pandemic”HUD
- "By increasing landlords’ costs of doing business, legal services attorneys may enrich their clients at the expense of all other similarly situated poor tenants."

1.2 Right to Counsel

1.2.1 Cost to Landlords

- Took thirty percent longer to reach final disposition than cases in which tenants represented themselves and contested their evictions.

1.3 Rapid Rehousing

1.3.1 Overview

- An intervention that helps homeless families exit shelters and get back into permanent housing quickly, provides short-term help with housing expenses (e.g., rent arrears, ongoing rent assistance, moving costs) and case management focused on housing stability.
- Rather than providing services to get families ready for housing before permanent placement, a philosophy many transitional housing programs use, rapid re-housing services are designed with a housing first approach to get families in permanent housing and keep them stable once they are there.
- With the following three components: Housing Identification Services, Financial Assistance for housing-related expenses, Case management services

1.3.2 Benchmarks for Success (HUD)

1. Households should move into permanent housing in an average of 30 days or less
2. At least 80 percent of households that exit a rapid rehousing program should exit to permanent housing
3. At least 85 percent of households that exit a rapid rehousing program to permanent housing should not become homeless again within one year.

1.3.3 Right Population?

- It is imperative that any lease agreement provides the tenant with **the same rights and responsibilities as a typical lease holder** and that the financial terms of the lease are such that the household has a reasonable ability to assume rental costs once financial support ends (keeping in mind that in the majority of cases, even households with no income at move-in retain their housing)"
- The target populations for these programs were households who faced barriers to housing, but who were not likely to need long-term assistance"
- Rapid re-housing is intended to serve people experiencing homelessness with no preconditions such as employment, income, absence of criminal record, or sobriety

1.4 This Project

1.4.1 Treatment

- In Connecticut, landlords will be required to give information about the program when they provide notice of eviction. Courts will also send information about the program along with notices of hearings.

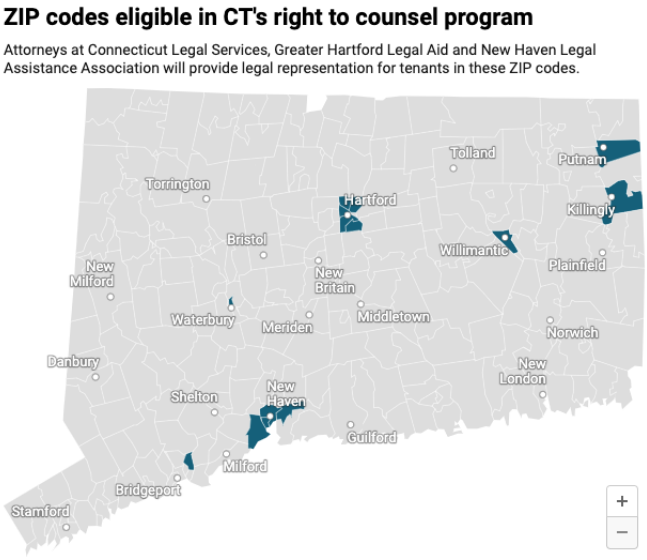
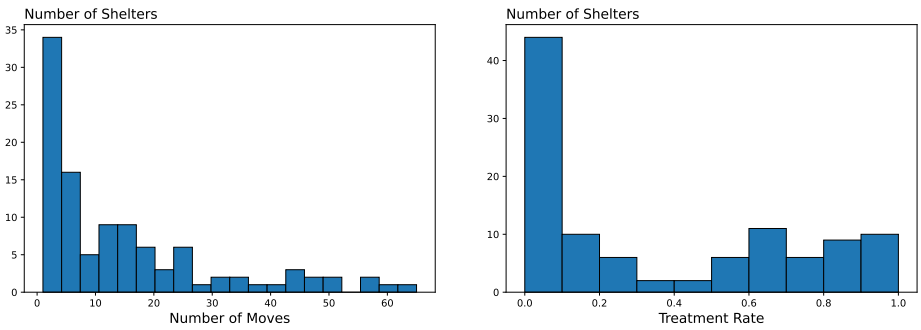


Figure 1: Produced by Ginny Monk and Hearst Connecticut Media Group – Here

- Eligible tenants are those who make 80% or less than the area median income and live in certain ZIP codes
- Expected to assist 2,000 households in the first phase, in a state where 20,000 residents faced eviction in Connecticut each year[1]
- Housing and neighborhood quality

1.4.2 The Need for Clustering



(a) Known Move-in Date: > 2019 (b) Known Move-in Date: > 2022

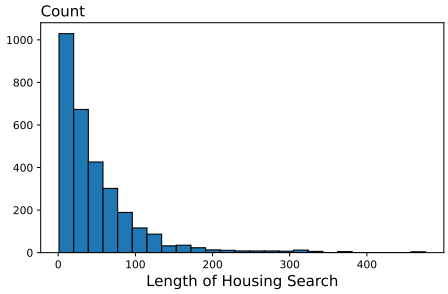
Figure 2: The Need for Clustering: Reproduced Here

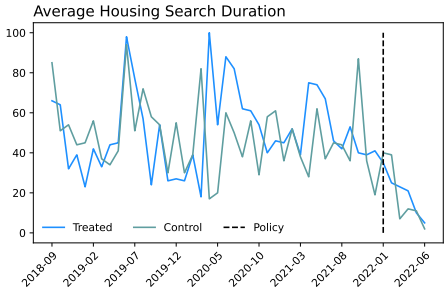
- Further, no clear model has emerged. Rapid re-housing, with all its local variations, is more of an approach then a specific model, which makes replication and scaling difficult, as it is most certain that implementation matters. Further, it is not clear which components of rapid re-housing are critical to achieving success.

[1] Some residents facing eviction could now be eligible for free legal aid - "It's not really just about how much money is available. But how long does it take to get enough attorneys hired and trained in order to provide the representation," Wagner said.

We potentially underestimate the effect on this subpopulation as most families (especially those with children) move within one year following the program.[2]

1.4.3 Connecticut





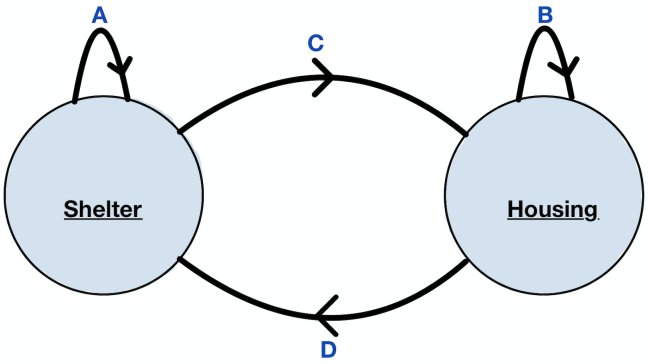
1.5 Limitations

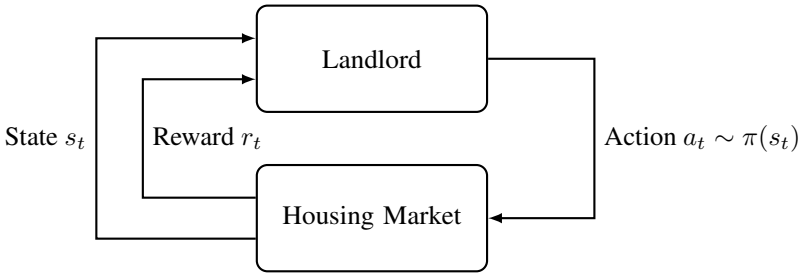
- The saliency of the policy
- The macroeconomic effects
- The sample size

[2] Urban Institute

2 Model

We model[3] housing stability via a stochastic process with a state space consisting of shelter and housing.[4] As referenced earlier, prior work has focused on the costs and benefits associated with transitions B and D. We complement this work by also assessing transitions A and C.





$$\underset{\text{sell,price,repairs,screening}}{\text{maximize}} \; \mathbb{E}\Big[\sum_{t=1}^{\infty} U\big(\text{Revenue} - \text{Cost}\big)\Big]$$

[3] This is a strong simplication of the housing search process. For instance, it omits details about whether a person was evicted (which might make it harder to get future housing) as well as whether the person was allowed to stay in their house while they looked for new housing – the name a few omitted, but important details.

[4] **Notation**:

- Probability Space: $\big(\Omega, \mathcal{F}_\Omega, \mathbb{P}\big)$
- Index Set: T
- State Space: $\big(S, \mathcal{F}_S\big)$
- Stochastic Process: $\{X_t : \Omega \to S\}_t$

3 Methodology

3.1 Notation

We begin by defining our probability space of interest.

$$
\begin{aligned}
&\big(\Omega_1, \mathcal{F}_1, \mu_1\big), \quad \omega \in \Omega_1 \longmapsto Y_{po}(\omega) \in f\big(\{-1, 0, 1\}, \mathbb{N}\big) \\
&\big(\Omega_2, \mathcal{F}_2, \mu_2\big), \quad \omega \in \Omega_2 \longmapsto V(\omega) \in \{-1, 0, 1\} \\
&\big(\Omega, \mathcal{F}, \mu\big) = \big(\Omega_1 \times \Omega_2, \sigma(\mathcal{F}_1 \times \mathcal{F}_2), \mu_1 \times \mu_2\big) \\
&\omega = (\omega_1, \omega_2) \in \Omega_1 \times \Omega_2 \longmapsto Y_{po}(\omega_1)(D(\omega_2)) = Y(\omega) \in \mathbb{R}
\end{aligned}
$$

We will assume that

$$\{Y_{po}(1) - Y_{po}(-1), Y_{po}(0) - Y_{po}(-1)\} \perp D | \; D \in \{0, 1\}$$

This assumption provides justification for interpreting β_1 as the causal effect:

$$
\begin{aligned}
Y_{1,i} - Y_{0,i} &= \beta_0 + \beta_1 D_i + \varepsilon_i \\
Y_{1,i} &= \beta_0 + \beta_1 D_i + Y_{0,i} + \varepsilon_i
\end{aligned}
$$

$$\hat{Y}_{i,c,t} = \beta_0 T + \beta_1 D_i \times T_i + g_c(X_i)$$

3.2 Overview

Table 1: Learning Frameworks

Framework	Top Model	Feature Map	Cluster Map	Trainable Params
OLS	Linear	Identity	Identity	False
Kernel Methods	Linear	Non-Identity	Identity	False
Supervised Learning	Linear	Non-Identity	Identity	True
Cluster Supervised Learning	Linear	Non-Identity	Non-Identity	True

$$Y_i = \beta_0 + \beta_1 D_i + g(X_i) + \varepsilon_i$$

If you have non-parametric identification, and if you are motivated to use this partially linear model, then the question becomes what's the "right" functional prior on g? Within this general approach to causal inference, we introduce a method that enhances the causal flavor as it generalizes OLS, flexibly accounts for the cluster nature of data, and is inherently compositional.

3.3 Overview

$$\mathcal{L}(\text{Model}, \text{Data}) = \mathcal{E}(\text{Model}, \text{Data}) + \mathcal{R}(\text{Model}, \text{Data})$$

- **Generalizing OLS**: OLS can be refactored into the composition of identity functions. Our method simply replaces these identity functions with suitable alternatives, thereby producing reasonable estimates on benchmark examples where the OLS model breaks down
- **Flexible Clustering Effects**: This highlights that the cluster map, this attempt to correct for cluster sampling, can be thought of as a gradient correction

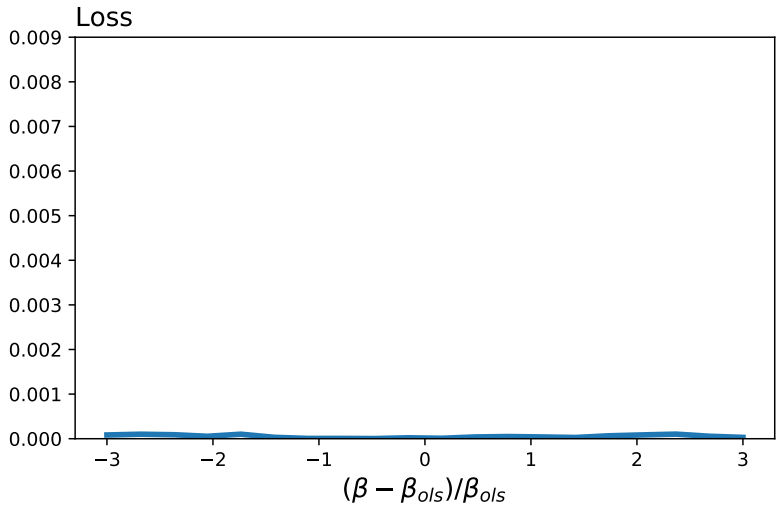


Figure 3: Reproduced Here

- **Inherently Compositional**: Even with the data-dependent form of regularization that we make use of our method remains compositional, which means that conceptually, the model remains simple.[5][6]

$$
\begin{aligned}
&\text{linearModel data}\\
&\text{linearModel} \circ \text{ identityMap data}\\
&\text{linearModel} \circ \big(\text{featureMap data}\big) \text{ params}\\
&\text{linearModel} \circ \big(\text{featureMap data}\big) \circ \text{identityMap params}\\
&\text{linearModel} \circ \big(\text{featureMap data}\big) \circ \big(\text{clusterMap data}\big)\text{params}\\
&\text{linearModel} >\!\!=\!\!> \big(\text{featureMap data}\big) >\!\!=\!\!> \big(\text{clusterMap data}\big)\text{params}
\end{aligned}
$$

$$
\begin{aligned}
&\text{regNeuralODE} :: \text{Data} \to \text{Params} \to \big(\text{Data}, \text{Float}\big)\\
&\text{regNeuralODE} \_\ x\ \_\ \theta := x + \int f(t, x(t), \theta)dt, \quad \int \left\| \frac{\partial^k}{dt^k} f(t, x(t), \theta) \right\| dt\\
&\qquad\qquad \text{where} \quad x(0) = x
\end{aligned}
$$

$$
\begin{aligned}
&\text{regMAML} :: \text{Data} \to \text{Params} \to \big(\text{Params}, \text{Float}\big)\\
&\text{regMAML data } \theta := \Big(\text{Update}_m \circ \text{Update}_{m-1} \cdots \circ \text{Update}_1\Big)\ \theta, \quad \mathcal{L}_c(\text{data}, \theta)\\
&\qquad\qquad \text{where} \quad \text{Update}_t\ \theta = \theta - \alpha_t \nabla \mathcal{L}_c(\text{data}, \theta)
\end{aligned}
$$

[5]The composition of partially evaluated functions

[6]Perhaps what we mean by this is that it can be shown to be compositional which provides us with a simple way to understand the model. The fact that something can be expressed in a compositional manner is a relatively weak statement on its own. Consider how one might define 3 in agda[7]

$$
\begin{aligned}
&\text{three} : \mathbb{N}\\
&\text{three} := \text{suc (suc (suc zero))}
\end{aligned}
$$

3.4 Kleisil Category

- From a programmer's perspective, computing the empirical loss of our model stays within the category of types and functions[8]
- To add regularzation to our model, we have to
  1. Introduce a new type constructor[9]
  2. Augment the output of our functions [10]
  3. Redefine Composition[11]
- Doing so generates a new category (the Kleisli Category)where , but where
  1. Objects are types (just as they were before)
  2. Morphisms between A, B are now functions from A to Kleisi B
  3. Composition is defined using the $>=>$ operator

3.5 Feature Map

1. We can implement any p.d kernel as the inner product in some high dimensional space

$$k(x, z) = \langle \phi(x), \phi(z) \rangle_{\mathcal{H}}$$

2. Kernel Methods can "rationalize" linear models in higher dimensions[12]

$$f(\theta, w, x) = w^T \phi_\theta(x) = \sum_i \alpha_i k(x, x_i)$$

- Sets: Ω
- Vector Space (of Functions): $F(\Omega, \mathbb{R})$
  - RKHS: $\mathcal{H} \subset \mathcal{F}$ whose Evaluation Functional are continuous (bounded!)

$$E_x : \mathcal{H} \to \mathbb{R}, \quad f \longmapsto f(x)$$

- Group
  - Preserve structure. This structure can be as simple the set structure
  - When we one group can be thought of as a relabeling of another group we say that the groups are isomorphic.

$$f : G_1 \to G_2, \quad \text{s.t} \quad f(g_{1a} \circ_1 g_{1b}) = f(g_{2a}) \circ_2 f(g_{2b})$$

    With a symmetry group, the isomorphism would be consider a high order function.
  - In contrast to the symmetries of a set, the automorphisms of a group are the bijective maps that preserve the group structure.

---

[8] In Haskell with a bit of handwaving, we can consider this category to be SET

[9]

type Kleisi a = (a, Float)

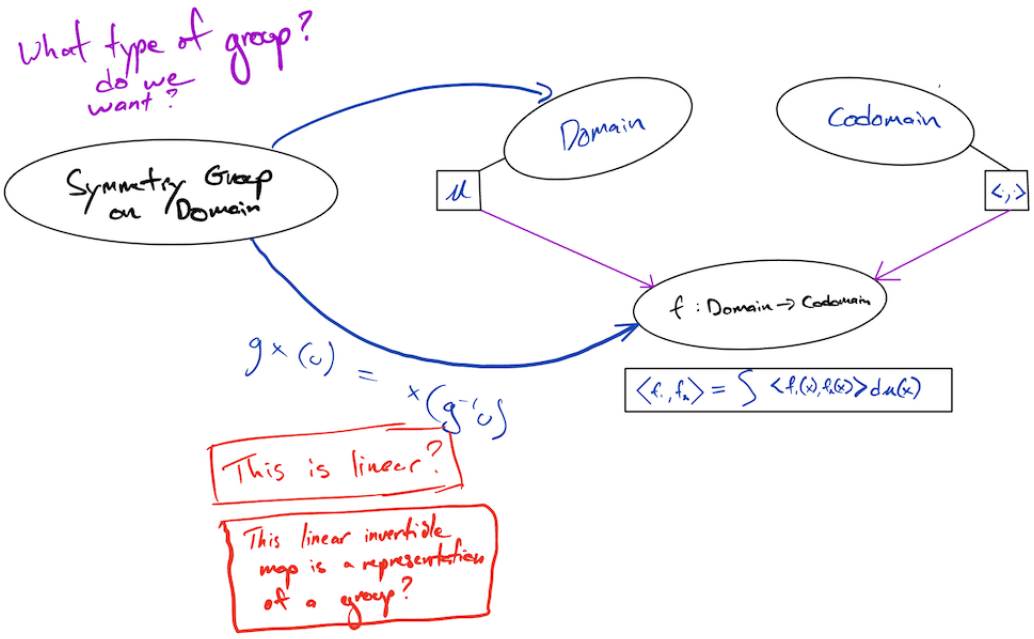
[10]

f :: a $\to$ Kleisi b

[11]

$$\begin{aligned} f >=> g = \ & x \to \\ & \text{let } (a, a_1) = f\ x \\ & \quad (b, b_1) = g\ a \\ & \text{in } (b, a_1 + b_1) \end{aligned}$$

[12] What is the reference for this?



  - Automorphisms are a group (are they a subgroup of the symmetry group?)
  - We don't want to fit automorphisms (what group structure do we want to preserve?)
- Symmetry Group: a group whose elements are transformations:[13]

$$g : \Omega \to \Omega$$

- Group Action: essentially just function application
- Signal: a function whose co-domain is a vector space

$$\mathcal{X}(\Omega, \mathcal{C}) = \{x : \Omega \to \mathcal{C}\}$$

- Given an inner product on C and a measure on Ω, we can construct an inner product on this function space

$$\langle x, y \rangle = \int_\Omega \langle x(u), y(u) \rangle_\mathcal{C} d\mu$$

What structure on $\mathbb{R}^n$ should we preserve?

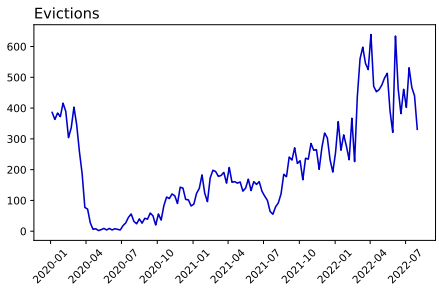
- Diffeomorphisms?

$$g : \text{bijections} \Rightarrow \mathbb{R}^n \to \mathbb{R}^n \equiv \text{Sym}(\mathbb{R}^n)$$

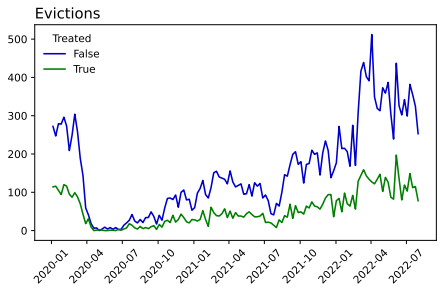
[13] Geometric Deep Learning

4 Results

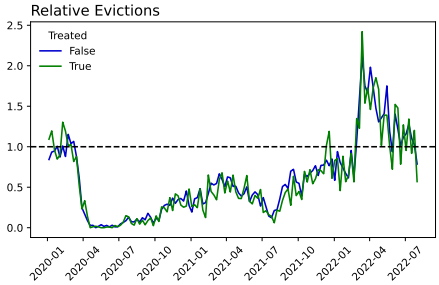
4.1 Direct Effects of Policy



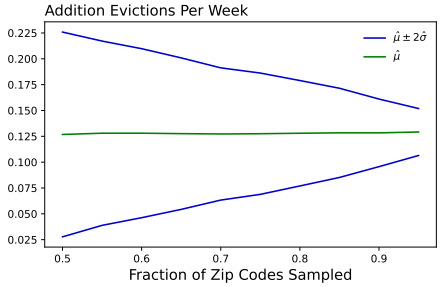
(a) State of Connecticut



(b) By Treatment Status



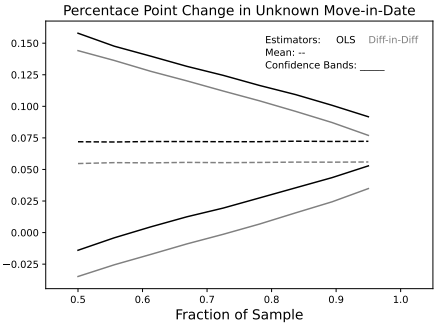
(c) Evictions Relative to Seasonality



(d) Estimated Policy Effect

Figure 4: Reproduced Here

4.2 Raw Results



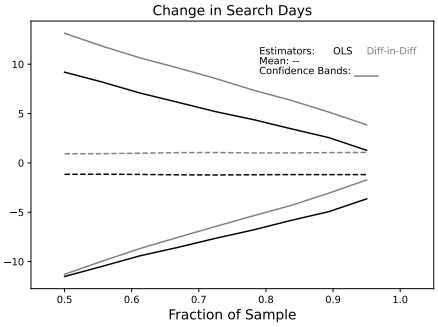


Figure 5: Initial Regression Results: Reproduced Here

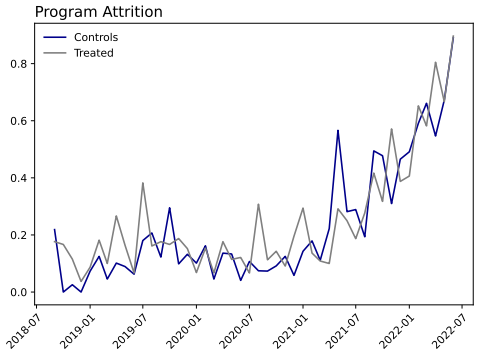


Figure 6: Average Attrition Rate by Month: Reproduced Here

5 Appendix

5.1 Limitations of OLS

Partial Non-parametric Identification

$$Y_i(D_i) \perp\!\!\!\perp D_i \mid X_i$$

Linear Approximation

$$\hat{Y}_i = \hat{\beta}_0 + \hat{\beta}_1 D_i + \hat{\beta}_2 X_i$$

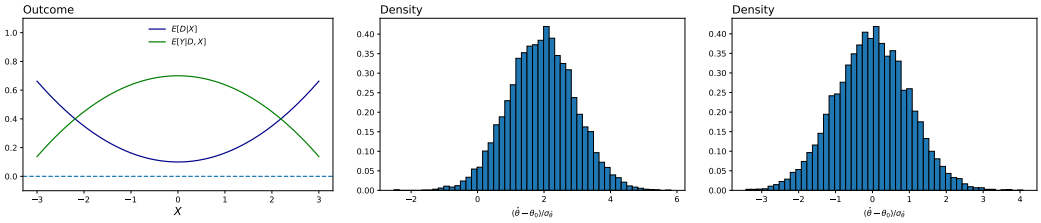
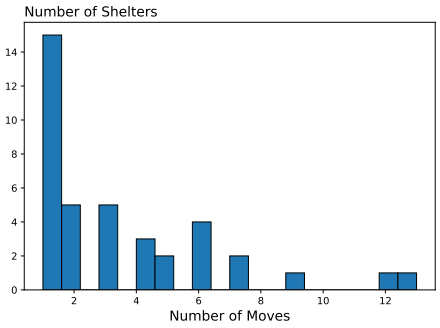


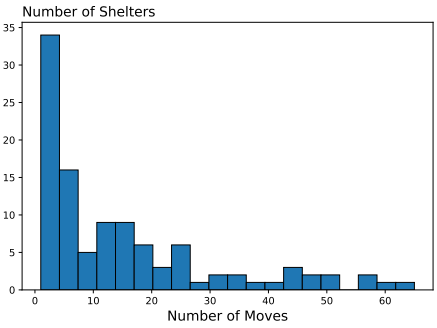
Figure 7: Reproduced Here

5.2 Program Intensity

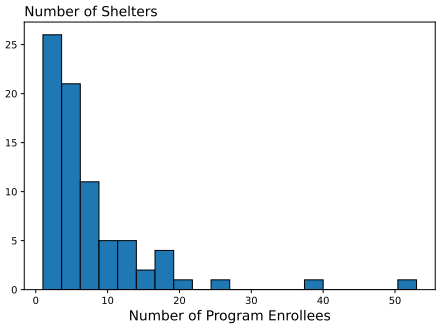
5.3 Treatment Intensity



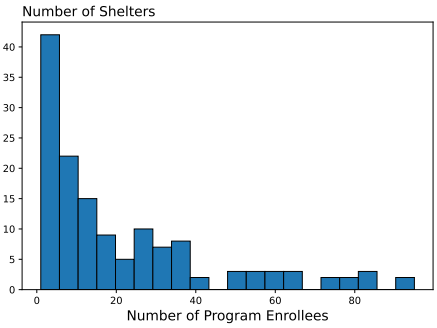
(a) Known Move-in Date: $>$ 2019



(b) Known Move-in Date: $>$ 2022

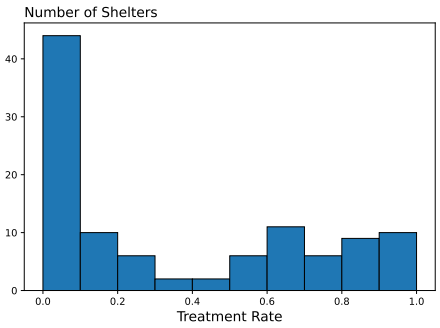


(c) Known Zipcode: $>$ 2019

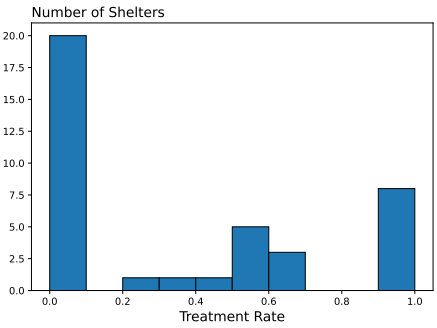


(d) Known Zipcode: $>$ 2022

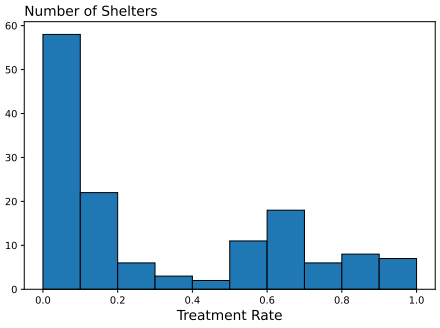
Figure 8: Figures to capture the relative intensity of the program across shelters: Reproduced Here



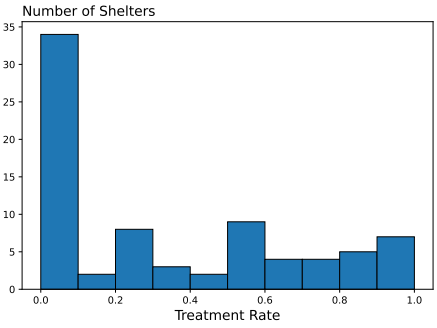
(a) Known Move-in Date: $>$ 2019



(b) Known Move-in Date: $>$ 2022



(c) Known Zipcode: $>$ 2019



(d) Known Zipcode: $>$ 2022

Figure 9: Figures to capture the relative intensity of the program across shelters: Reproduced Here

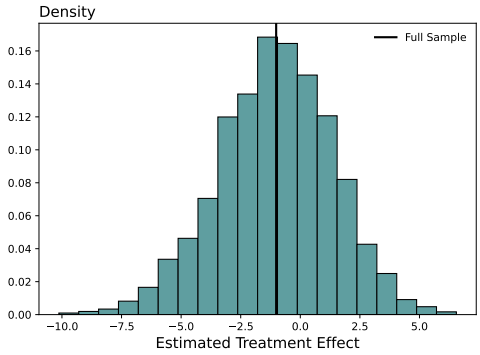


Figure 10: Bootstrapped Difference-in-Difference Estimate Ignoring Attrition & Clustering –> Reproduced Here

5.4 Category Explanation Continued

Our belief is that if we can show you how each submodel of our model can be thought of as a morphism in a category where the objects are types then either our model is simple, or we have provided a simple way to conceptualize our model.[14]

Of these transformations (is that the right word), perhaps the most interesting is regularzation which can be thought of as a notion of computation and defines a monad.[15]

I think we have more structure than just a monad, because the function retains the same output when its a morphsim in the category SET

- Our model is a function from A to $T(B)$.

$$\text{A} \rightarrow \text{T(B)}$$

- We can think of this though as a morphism from A to B

$$\text{A} \rightsquigarrow \text{B}$$

- We can define composition of arrows as $>=>$
- Then $(\mathcal{A}, \rightsquigarrow, >=>)$[16]
- Arrows represent the partial evaluation of functions

Below we capture the above points by writing the model in pseudo Haskell[17] code.[18]

[14] You are of course free to choose either of these two definitions or to disagree, and we would be happy to hear which you choose

[15] A type constructor and an extendor

[16] If the category of T-programs is constructed using a Kleisli triple, as depicted on the next slide, then T defines a monad. – She says this should be attributed to Main?

[17] the feature and cluster maps both take as inputs data and parameters. The dollar sign denotes partial application (i.e. currying) of a function. $\circ$ denotes composition

[18] A reader familiar with Haskell/Category theory will recognize that the act of adding regularization to our model conceptually moves us from the category of Sets to the Kleisi Category. A full description of this change following the layout of Category Theory for Programmers is described in the appendix

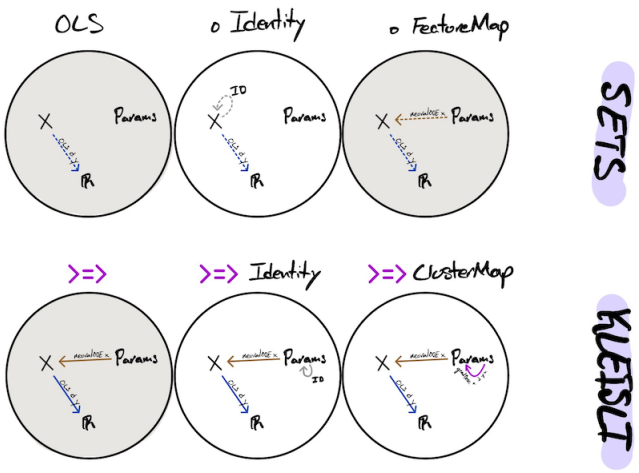
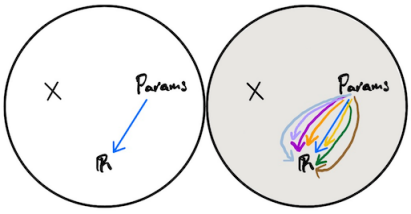


Figure 11: This can be seen as a complete refactoring of OLS by introducing the identity functor (this part is not reflected in the refactor arrow). If we look at this diagram vertically, we see that we are workign with two categories, and our "model" is attained by selecting the appropriate morphism between these two categories (where the morphisms are functors)



6 Probability Theory Set-up

- Feature Map
  - We would like the function space perspective, but we'll settle for an input space perspective in contrast to regularizing the parameter space
- Cluster Effects
  - Gradient Correction that favors early stopping at the cluster level
  - Parametrics: We want to be able to control the influence of certain variables (like cluster indicators)
  - Nonparametrics: We want to partial out the cluster effects! (We want to use the cluster level information in the training set to produce a cluster-free model) AKA generalizing across clusters
- Defining the problem (need to add the clustering part!)
- Defining conditional independence
- Defining the linear approximation

6.1 The Problem We Would Like to Solve

- "At the level of this book, the theory would be more 'elegant' if we regarded a random variable as an equivalence class of measurable functions on the sample space, two functions belonging to the same equivalence class if and only if they are equal almost everywhere"[19]

[19] Williams

- We begin by defining our probability space of interest as follows

$$\begin{aligned}
&\big(\Omega_1, \mathcal{F}_1, \mu_1\big), \quad \omega \in \Omega_1 \longmapsto Y_{po}(\omega) \in C(\mathbb{R}) \\
&\big(\Omega_2, \mathcal{F}_2, \mu_2\big), \quad \omega \in \Omega_2 \longmapsto D(\omega) \in \mathbb{R} \\
&\big(\Omega, \mathcal{F}, \mu\big) = \big(\Omega_1 \times \Omega_2, (\mathcal{F}_1 \times \mathcal{F}_2), \mu_1 \times \mu_2\big) \\
&\omega = (\omega_1, \omega_2) \in \Omega_1 \times \Omega_2 \longmapsto Y_{po}(\omega_1)(D(\omega_2)) = Y(\omega) \in \mathbb{R}
\end{aligned}$$

- And our parameter of interest (where H is the set of of functions. In the appendix we walk through this derivation in more detail.

Is this the set of affine, $\sigma(D)$ measurable, square integrable functions?

$$\begin{aligned}
\underset{g \in \mathcal{H}}{\text{argmin}}\, \mathbb{E}_\mu\big[(Y - g_\theta(D))^2\big] &= \underset{g \in \mathcal{H}}{\text{argmin}}\, \mathbb{E}_\mu\big[(E(Y|D) - g_\theta(D))^2\big] \\
&= \underset{g \in \mathcal{H}}{\text{argmin}}\, \mathbb{E}_{\mu_2}\big[(E(Y|D) - g_\theta(D))^2\big] \\
&= \underset{g \in \mathcal{H}}{\text{argmin}}\, \mathbb{E}_{\mu_2}\big[(E(Y_D) - g_\theta(D))^2\big]
\end{aligned}$$

6.2 The Problem We Are Dealt (1)

6.2.1 Treatment Assignment Mechanism

As we'll now explain, we don't have access to realizations from the product measure[20]. Instead our measure v can be constructed as follows.[21]

$$\mu \ll v, \quad \mu \overset{T}{\longmapsto} v; \quad v(A) = \int_A f d\mu \quad \forall A \in \mathcal{F}$$

With the additional condition that for all

$$\omega_1 \longmapsto \mathbb{E}[f(\omega_1, \omega_2)] \overset{a.s.}{=} \omega_1 \longmapsto 1$$

As highlighted below

$$\begin{aligned}
v(A_1 \times \Omega_2) &= \int_{A_1 \times \Omega_2} f d\mu \quad \text{by Radon Nykodym} \\
&= \int_{A_1} \int_{\Omega_2} f(\omega_1, \omega_2) d\mu_2 d\mu_1 \quad \text{by Tonelli's theorem} \\
&= \int_{A_1} \mathbb{E}[f(\omega_1, \omega_2)] d\mu_1 \quad \text{by definition} \\
&= \mu(A_1)
\end{aligned}$$

This is an important condition because it tells us that we just need to work to get the following:

We know that

$$E_v(Y_D|X) = E_\mu(Y_D|X)$$

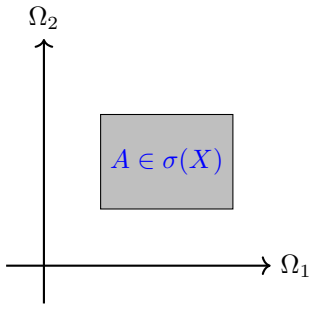
What we have now is

$$E_v\big[E_v(Y_D|X)\big] = E_\mu\big[E_\mu(Y_D|X)\big]$$

- **Clusters**:
  - In what follows, we assume that the number of clusters is known ex ante

[20] We need these measures to by σ -finite

[21] We model the treatment assignment mechanism as a higher order function on the set of measures



- Consider
  1. Conditional on X treatment is randomly assigned, but the conditional distribution of treatment might not match the desired one!
  2. Are we defining equality of random variables the right way? See the appendix

  By SUTVA

$$\int_A E(Y|D,X)dv = \int_A E(Y_D|D,X)dv \quad \forall A \in \sigma(D,X)$$

  By Conditional Independence Assumption

$$= \int_A E(Y_D|X)dv \quad \forall A \in \sigma(D,X)$$

  By the restriction on f

$$D \longmapsto \int_{\Omega_1} E(Y|D,X)dv \iff D \longmapsto \underbrace{\int_{\Omega_1} E(Y_D|X)d\mu_1}_{=\mathbb{E}_{\mu_1}[Y_D]}$$

  As

$$\int_{\Omega_1} E(Y|D,X)dv = \int_{\Omega_1} E(Y|D,X)fd\mu_2\mu_1$$

  Therefore, we can define an objective function which we observe who shares the same minimum as our desired objective function:

$$= \underset{g\in\mathcal{H}}{\text{argmin}} \int_{\Omega_2} \Big(\int_{\Omega_1} E[Y|D,X]d\mu_2 - g_\theta(D) \Big)^2 d\mu_1$$
$$= \underset{g\in\mathcal{H}}{\text{argmin}} \int_{\Omega_2} \Big(E[Y_D] - g_\theta(D) \Big)^2 d\mu_1$$

and defining my objective function in terms of this function.

$$\theta^* = \underset{\theta}{\text{argmin}} \int_{\Omega_2} \big(h_X \circ D - g(D) \big)^2 d\mu_2$$

$$E[1_U 1_V|X] = E[1_U|X]E[1_V|X] \quad \forall U \in \sigma(Y_D), V \in \sigma(D)$$

- We start by introducing our probability space of interest $\big(\Omega, \mathcal{F}, \mathbb{P}\big)$ and our working probability space: $\big(\Omega, \mathcal{F}, v\big)$ with the following random variables, Y_i, X_i, D_i, C_i, defined on the measurable space $\big(\Omega, \mathcal{F}\big)$.
- Parameter of Interest
  - Let $W_i = (Y_i, D_i)$ be a continuous random variable on $\big(\Omega, \mathcal{F}, \mathbb{P}\big)$ with pdf f_W.

- Let $g_\theta(W_i) = (Y_i - \theta_0 - \theta_1 D_i)^2$, be an $\mathcal{F}$-measurable function.[22]

$$\theta^*(\mathbb{P}) = \underset{\theta}{\text{argmin}}\, \mathbb{E}[g_\theta \circ W] = \underset{\theta}{\text{argmin}} \int g_\theta d\mathbb{P}_W = \underset{\theta}{\text{argmin}} \int g_\theta f_W d\lambda$$

- **Clustered Data**:

$$\begin{aligned} C : \mathcal{F} &\to \{\mathbb{P}\} \to \{\mathbb{P}\} \\ \mathbb{P}, A &\longmapsto \mathbb{P}_A \\ \text{where} \quad \mathbb{P}_A(B) &= \frac{\mathbb{P}(A \cap B)}{\mathbb{P}(A)} \end{aligned}$$

We define the event of interest, A_m, as follows (where m denotes the number of clusters)[23]

$$\begin{aligned} H &: \Omega \to \mathbb{N} \\ \omega &\longmapsto H = \lim_{n\to\infty} \text{numUnique}(C_{i\leq n}) \\ A_m &\in \mathcal{F} := \{\omega \in \Omega | H(\omega) = m\} \end{aligned}$$

- Treatment Assignment Mechanism:

$$T : \{\mathbb{P}\} \to \{\mathbb{P}\}$$

$$v = (C\ A_m) \circ T\ \mathbb{P}$$

- Identification

$$Y_i(D_i) \perp\!\!\!\perp D_i \mid X_i$$

- **Conditional Expectation**
  - TL;DR[24] (a version of)[25]the conditional expectation of a random variable X with respect to another random variable, Y, is (a/the) random variable, $E_Y(X)$, whose integral over any element in $\sigma(Y)$ is equal to that of X. That is

$$\int_A E_Y(X) d\mathbb{P} = \int_A X d\mathbb{P} \quad \forall A \in \sigma(Y)$$

Moreover, we can see that any $\sigma(Y)$-measurable and integrable function $g \circ Y$

$$\begin{aligned} & \int_A (g \circ Y) E_Y(X) d\mathbb{P} = \int_A (g \circ Y) X d\mathbb{P} \quad \forall A \in \sigma(Y) \\ \iff \quad & \int_A (g \circ Y)\big(E_Y(X) - X\big) d\mathbb{P} = 0 \end{aligned}$$

  - The **Law of Iterated Expectation** is just a special case of the above statement

$$\begin{aligned} \mathbb{E}\big[E_Y(X)\big] &:= \int_\Omega E_Y(X) d\mathbb{P} \\ &= \int_\Omega X d\mathbb{P} \\ &= \mathbb{E}\big[X\big] \end{aligned}$$

[22] We don't need to assume that it is absolutely integrable because it is a nonnegative function – ref

[23] In haskell, we might define this function as done in LearnYouaGoodHaskell:

$$\begin{aligned} &\text{numUniques} :: (\text{Eq a}) => [a] -> \text{Int} \\ &\text{numUniques} = \text{length} \,.\, \text{nub} \end{aligned}$$

[24] The conditional expectation of X with respect to either Y or Z will be the same random variable if $\sigma(Y) = \sigma(Z)$

[25] https://web.ma.utexas.edu/users/gordanz/notes/conditional$_e xpectation.pdf$

- What I want I think integrating over X-measurable D-constant subsets seems what we want to do (ish)

$$E_\mu[Y|D,X] = E_{\mathbb{P}}[Y_D|X]$$

- Conditional Probability
  - Partially evaluated, this is a continuous linear function[26]
  - The atoms of a countably generated sigma algebra is measurable

$$E :: S(\mathcal{F}) \to L^1(\Omega,\mathcal{F},\mathbb{P}) \to L^1(\Omega,\mathcal{F},\mathbb{P})$$

such that

$$\int E_{\mathcal{A}}(f)d\mathbb{P} = \int_A f d\mathbb{P} \quad \forall A \in \mathcal{A}$$

[26] pf. 177

6.3 Measure Theory Notes

- Alternative Notation for of Parameter of Interest

$$\theta = \underset{\theta}{\text{argmin}} \int_{\Omega_2} \int_{\Omega_1} f_\theta(\omega_1, \omega_2)^2 d\mu_1 d\mu_2$$
$$\text{where} \quad f_\theta(\omega_1, \omega_2) := Y(\omega) - \theta_0 - \theta_1 D(\omega_2)$$

- 1st Equality Sign

$$\begin{aligned}\mathbb{E}\big[(Y - g(D))^2\big] &= \mathbb{E}\big[(Y - E[Y|D] + E[Y|D] - g(D))^2\big] \\ &= \mathbb{E}\big[(Y - E[Y|D])^2\big] + \mathbb{E}\big[(E[Y|D] - g(D))^2\big] \\ &\quad + \overbrace{\mathbb{E}\big[(Y - E[Y|D])\underbrace{(E[Y|D] - g(D))}_{\phi(D)}\big]}^{=0}\end{aligned}$$

- The second equality sign is related to the following:[27]

$$\int_A E(Y|D) d\mu = \int_A Y d\mu \quad \forall A \in \sigma(D)$$

$$\begin{aligned}A_{\omega_2} &= \{\omega_1 \in \Omega_1 : (\omega_1, \omega_2) \in A \subset \Omega\} \subset \Omega_1 \in \mathcal{F}_1 = \Omega_1 \\ \mu(A) &= \int_{\Omega_2} \mu_1(A_{\omega_2}) d\mu_2 \\ &= \int_{\Omega_2} \mu_1(\Omega_1) d\mu_2 \\ &= \int_{\Omega_2} d\mu_2\end{aligned}$$

- The fourth equality sign is related to the following

$$\int_A E(Y|D) d\mu_2 = \int_A E[Y_D] d\mu_2 \quad \forall A \in \sigma(D)$$

- A measureable function is integrable if

$$\int |f| d\mu < \infty$$

- Space of Integrable Functions

$$L^1(\Omega, \mathcal{F}, \mathbb{P}) := \{f : \Omega \to \mathbb{R} | f \text{ is measurable}, \int |f| d\mathbb{P} < \infty\}$$

- **Question**: If we have two random variable f and g whose integration matches on $\mathcal{F}$ then we can replace one for the other in our objective function?
- Proof by contradiction: Assume the following holds[28]

$$\int_A f d\mu = \int_A g d\mu \quad \forall A \in \mathcal{F}$$
$$\int F(f) d\mu \neq \int F(g) d\mu$$

By definition:

$$\int F(f) d\mu = \underset{h \in S(F(f))}{\text{argmax}} \sum_{i=1}^{n} a_i \mu(A_i)$$

[27] helpful lecture
[28] Helpful proof description