Deep Architecture and Consumer Heterogeneity

Sanjog Misra (University of Chicago Booth School of Business)

{Chris Hansen, Max Farell, Tengyuan Liang} + {J.P. Dube} + {Gunter Hitsch}.

DSE 2019

Connected Research

Deep Neural Networks for Estimation and Inference: Application to Causal Effects and Other Semiparametric Estimands (with Farell and Liang) https://arxiv.org/abs/1809.09953

Targeted Undersmoothing (with Hansen)
https://arxiv.org/abs/1706.07328

Heterogeneous Treatment Effects and Optimal Targeting Policy Evaluation (with Hitsch)
https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3111957

Scalable Price Targeting (with Dube)
https://papers.ssrn.com/sol3/papers.cfm?abstract_id=2992257

Consumer Heterogeneity

- A fundamental building block of marketing decisions is the construct of consumer heterogeneity.
- Accounting for such heterogeneity is relevant for a number of tasks
  - Accuracy (Bias)
  - Inference (Variance)
  - Policy Design and Evalauation
    - Targeting
    - Segmentation
    - Personalization
- This paper is about the *practice* of estimating and using heterogeneity measures.

Thinking about heterogeneity

- Consider some (log) likelihood

$$\sum_i \ell(\mathbb{D}_i; \theta_i)$$

- Since we don't know θ_i we have to figure something out
- We can assume the problem aways and set $\theta_i = \theta$
- If we had panel data with could try to estimate θ_i
  - In most applications (e.g. B2B, eCommerce,...) this is not the case
- Even if it were, simply estimating θ_i may not be useful.
- For example, we need to know *types* (θ_i) for new customers.

Predicting Types

- We assume that we can project customer types onto a known vector of consumer characteristics x_i

$$\theta_i = \theta(X_i)$$

- In low-dimensional cases with f linear this is simply an "interactions" model.
  - Think $\theta_i = X\theta$, so that

$$Y_i = a + (X_i\theta) \cdot T_i + \epsilon_i$$

Approximating Types

- As we get more data about a customer it is plausible to assume that some subset of these high dimensional data will be relevant in articulating/approximating consumer types with some reasonable accuracy.
- Assumption: There exist $\theta(X_i)$ such that for some X_i

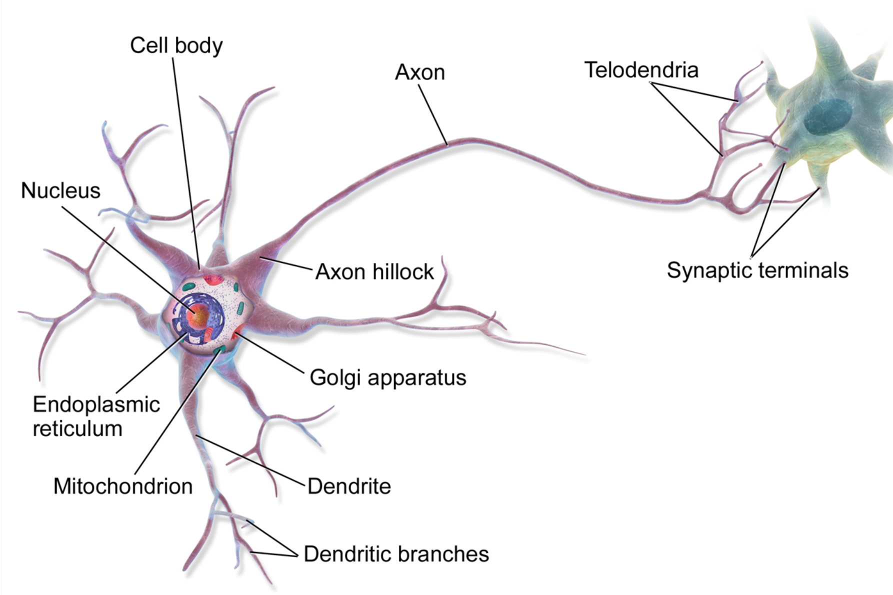
$$\|\theta_i - \theta(X_i)\| \to 0$$

- **Couple of issues**:
  1. **We dont know which X_i matter.**
  2. **We dont know the function $\theta(X)$.**

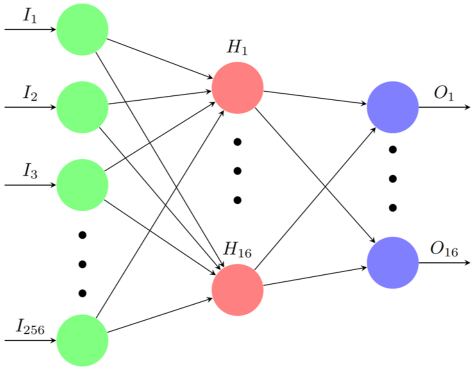
DeepNets Architecture

- We propose assuming that $\theta(X_i)$ is β-smooth and can be approximated by some Deepnet or **Deep Neural Network** (DNN).
- That is
$$\|\theta(X_i) - f_{\text{DNN}}(X_i; \Theta)\| \to 0$$
- What are Deepnets?
- Why Deepnets?

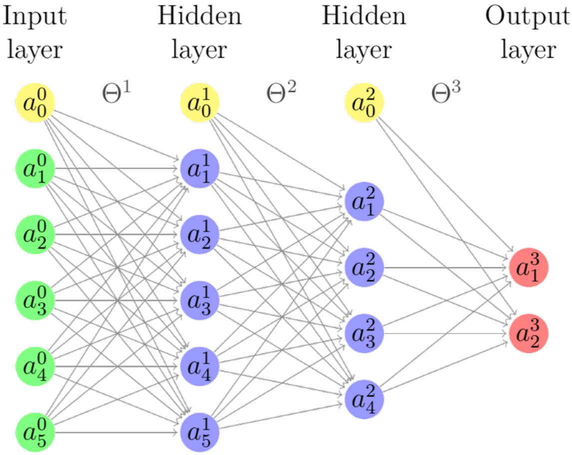
Neural Networks



Artificial Neural Nets



Deepnets



Input layer
Hidden layer
Hidden layer
Output layer
Θ^1
Θ^2
Θ^3
a_0^0
a_1^0
a_2^0
a_3^0
a_4^0
a_5^0
a_0^1
a_1^1
a_2^1
a_3^1
a_4^1
a_5^1
a_0^2
a_1^2
a_2^2
a_3^2
a_4^2
a_1^3
a_2^3

Deepnets

- Units are arranged into layers...
  - According to a directed, acyclic graph
  - Number of layers L (**depth**)
  - Unit is in layer l if it has a predecessor in $l-1$ and none for any $l' \geq l$
  - Unit receives some $\tilde{\mathbf{x}}_l'\mathbf{w} + b_l$, returns $\tilde{\mathbf{x}}_{l+1} = \sigma(\tilde{\mathbf{x}}_l'\mathbf{w}_l + b_l)$
  - σ is an activation function.
  - Dimension of $\tilde{\mathbf{x}}_l$ is **width**
  - Final layer is $\hat{y}_{\mathrm{MLP}}(\mathbf{x}) = \Phi(\tilde{\mathbf{x}}_L(\mathbf{x})'\mathbf{w}_L + b_L)$

Why DeepNets?

- Simple. Scalable.
- They are **universal approximation** theorems.
- Loosely speaking, for continuous functions there will always exists some DNN that approximates it arbitrarily well.
- We could use some ML (say Random forests or Lasso)
  - Many appraoches (such as the Lasso) require us to know the true basis functions.
  - Other more flexible ones (say Forests) may not have the ability to maintain structural assumptions.
  - For example we may want $g(x_i)$ in

$$q_i = a + g(x_i) \cdot p_i + \epsilon_i$$

    to be a random forest.

Why DeepNets?

- Wemay be able to construct an ML based estimator on a case by case basis
- DNNs offer the potential for a standardized approach for any **structural** model we may write down, so that for some Θ

$$\|\theta(X_i) - f(X_i;\Theta)\| \to 0$$

and

$$\left\|\sum_i \ell(\mathbb{D};\theta(X_i)) - \sum_i \ell(\mathbb{D};f(X_i;\Theta))\right\| \to 0$$

Current applications of DNNs

- Typically ...

$$Y = f_{\mathrm{DNN}}(X)$$

- A tad more interesting is...

$$W = f_{\mathrm{DNN}}(X)$$

$$Y = m(Z, \gamma, \widehat{f}_{\mathrm{DNN}}(X))$$

- examples: DNN Text analysis on Mturk, Plugin estimation using other data.
- Our **theory** applies in both cases (under some verifiable conditions)

Agenda

- Heterogenous Treatment Effects (NP)
- Embedding DeepNets in Structural models
  - Heterogenous Choice Models
  - *Heterogeneous Discount Factors*

Causal Effects

- In a variety of contexts we might be interested in how individual subjects respond to a discrete (binary) treatment.
  - Health responses to a medical treatment
  - Voter tunrounout as a response to some intervention
  - Unemployment as a function of a change in minimum wage
  - **Customer spending in response to some targeted marketing campaign**
- This example is about the use of DNNs for the purposes of estimating heterogeneous treatment effects and policy evaluation.

Potential Outcomes Framework

- Population of units (customers) indexed by i
- Binary treatment $t_i \in \{0, 1\}$
  - $t_i = 1$ if customer is treated
  - $t_i = 0$ otherwise
- Potential outcomes: $Y_i(t_i = 0)$ and $Y_i(t_i = 1)$ or

The object of interest.

- Conditional Average Treatment Effects (CATE)

$$\tau(x) = \mathbb{E}(Y|X = x, t = 1) - \mathbb{E}(Y|X = x, t = 0)$$

- Useful object to construct average treatment effects (ATE)
- For policy evaluation
- For policy design

Data

- The data vector is $\mathcal{D}_i = \{Y_i, X_i, t_i\}$ for all i
- Outcomes are observed conditional on treatment,

$$Y_i = \begin{cases} Y_i(0) & \text{if } t_i = 0 \\ Y_i(1) & \text{if } t_i = 1 \end{cases}$$

- **Problem**: Counterfactuals are unobserved $\implies$ fundamental problem of causal inference

Response functions

- Recall that the CATE is defined as

$$\tau(x) = \mathbb{E}(Y|X = x, t = 1) - \mathbb{E}(Y|X = x, t = 0)$$

- We can write this as

$$\tau(x) = \mu(x, 1) - \mu(x, 0)$$

- where $\mu(x, t) = \mathbb{E}(Y|X = x, t = 1)$
- Note that the $\mu(x, t)$ are simply conditional expectations.

Difference Estimators

- A natural estimator is

$$\widehat{\tau}(x) = \widehat{\mu}(x, 1) - \widehat{\mu}(x, 0)$$

- Now if we obtain estimators for $\mu(x, t)$ we can construct an estimator for $\tau(x)$
- We can use any "regression" framework to obtain $\widehat{\mu}(x, t)$
  - Linear Models (or GLMs)
  - Regularized Models (e.g. Lasso)
  - Trees and Random Forests
  - Gradient Boosted Machines
  - Deep Nets
  - Kernels
  - ...

Direct Estimators

- The above estiamtors use the MSE $\sum(y-\widehat{y})^2$ to estiamte parameters
- Ideally, one should minimize $\sum(\tau-\widehat{\tau}(x))^2$
- Seems impossible (since we dont observe τ).

Direct Estimators: Transformed Outcomes

- Consider the following...

$$Y_i^* = t_i \frac{Y_i(1)}{e(X_i)} - (1 - t_i) \frac{Y_i(0)}{1 - e(X_i)}$$

- where $e(X_i) = \Pr(t_i = 1 | X_i)$ (**Propensity Score**)
- If unconfoundedness holds, then

$$\mathbb{E}[Y_i^* | X_i = x] = \tau(x)$$

- Hence Y_i^* is a proxy for the CATE: $Y_i^* = \tau(X_i) + \nu_i$
- $\mathbb{E}[\nu_i | X_i] = 0$ and ν_i is orthogonal to any function of X_i
- This approach is used (in some way) by
  - Causal forests
  - Causal k-NN
  - ...

Interactions Estimator

- We can define $\tau(x)$ implicitly by

$$y = \mu(x, 0) + \tau(x)t$$

- Then use an interactions approach...
- Easy to do with linear models.

$$y_i = \alpha' x_i + (\beta' x_i)t_i + \varepsilon_i$$

  - Can replace x_i with other basis functions $\varphi(x)$
  - Can add regularization

Causal Deepnets

- We will use an interactions approach:

$$y_i = \alpha(x_i) + \beta(x_i) \cdot t_i$$

- where by definition $\tau(x) = \beta(x)$
- Define

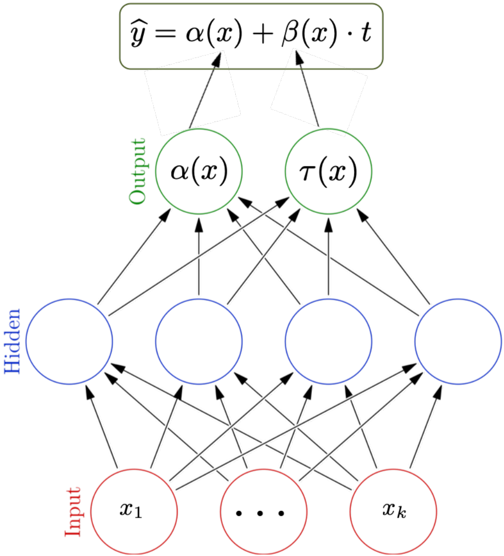
$$e(X_i) = \Pr(t_i = 1|X_i)$$

  and write

$$\theta(x) = \{\alpha(x), \beta(x), e(x)\}$$

- We assume that $\theta(x)$ can be approximated by a DNN.
- Note: In our application $e(x) = e =$ known constant $= \frac{2}{3}$

Causal Deepnets



Implementation

- The implementation of these models is simple but not trivial.
  - DNN's can be implement with standard software (Keras,h2o, mxnet ...)
  - These are too restrictive for our purposes.
- We use **Tensorflow** for all computations.
  - One could use **Theaono**,**CNTK**,**(Py)Torch** etc. as well.
  - Any software that allows for designing DNNs with *custom* architecture and loss functions.

Some (new) theory...

- No current convergence rates for DNNs!
- To do inference we need some work.
  - We need to show that a deepnet that gets at the truth exists
  - And that we can estimate this deepnet at a fast enough rate.
- Start with the ususal Bias-Variance set-up

$$\left\|\hat{f} - f_*\right\|^2_{L_2(X)} \lesssim (\mathbb{E} - \mathbb{E}_n)\left[\ell(\hat{f}, \boldsymbol{z}) - \ell(f_*, \boldsymbol{z})\right] + \mathbb{E}_n\left[\ell(f_n, \boldsymbol{z}) - \ell(f_*, \boldsymbol{z})\right]$$

- Use novel localization methods along with new approximation results to bound this quantity.

Some (new) theory...

- We show that

$$\left\| \widehat{f}_{\mathbf{MLP}} - f_* \right\|_n^2 \leq C \cdot \left\{ n^{-\frac{\beta}{\beta+d}} \log^8 n + \frac{\log \log n}{n} \right\}$$

- Comments:
  - These are the first results on rates of convergence for DNNs
  - Uses standard architecture MLP+ReLU
  - Fast enough rates for semiparametric inference
  - ... but not minimax optimal.

When is this trivial

- Consider
$$\|\theta(X_i) - f_{\mathrm{DNN}}(X_i; \Theta_k)\| \approx 0$$
- where k is the number of parameters and n is the number of observations.
$$\frac{k}{\sqrt{n}} \to 0$$

Semiprametric Objects

- Recall that
$$\theta(X_i) = \{\alpha(X_i), \beta(X_i), e(X_i)\}$$
- Define the following...

$$\hat{\psi}_t = \frac{\mathcal{I}\{t_i = t\}(y_i - \hat{\mu}_t(x_i))}{t_i e(x_i) + (1 - t_i)(1 - e(x_i))} + \hat{\mu}_t(x_i)$$

then
$$\widehat{\tau} = \mathbb{E}_n\left[\hat{\psi}_1 - \hat{\psi}_0\right]$$
- Further, the estimator is **Doubly Robust**
  - i.e. it is consistent even if one component is misspecified.

Semiprametric Objects

- More generally for some $s(x)$

$$\sqrt{n}\,\mathbb{E}_n\left[s(x_i)\hat{\psi}_t - s(x_i)\psi_t\right] = o_P(1)$$

$$\frac{\mathbb{E}_n[(s(x_i)\hat{\psi}_t)^2]}{\mathbb{E}_n[(s(x_i)\psi_t)^2]} = o_P(1)$$

- Could use these to construct other objects
- Profits: $\mathbb{E}_n\left[s\left(\boldsymbol{x}_i\right)\hat{\psi}_1\left(\boldsymbol{z}_i\right) + \left(1 - s\left(\boldsymbol{x}_i\right)\right)\hat{\psi}_0\left(\boldsymbol{z}_i\right)\right]$
- Profit Differences:

$$\mathbb{E}_n\left[\left[s_a\left(\boldsymbol{x}_i\right) - s_b\left(\boldsymbol{x}_i\right)\right]\hat{\psi}_1\left(\boldsymbol{z}_i\right) - \left[s_a\left(\boldsymbol{x}_i\right) - s_b\left(\boldsymbol{x}_i\right)\right]\hat{\psi}_0\left(\boldsymbol{z}_i\right)\right]$$

Empirical Application

Direct Mail Marketing

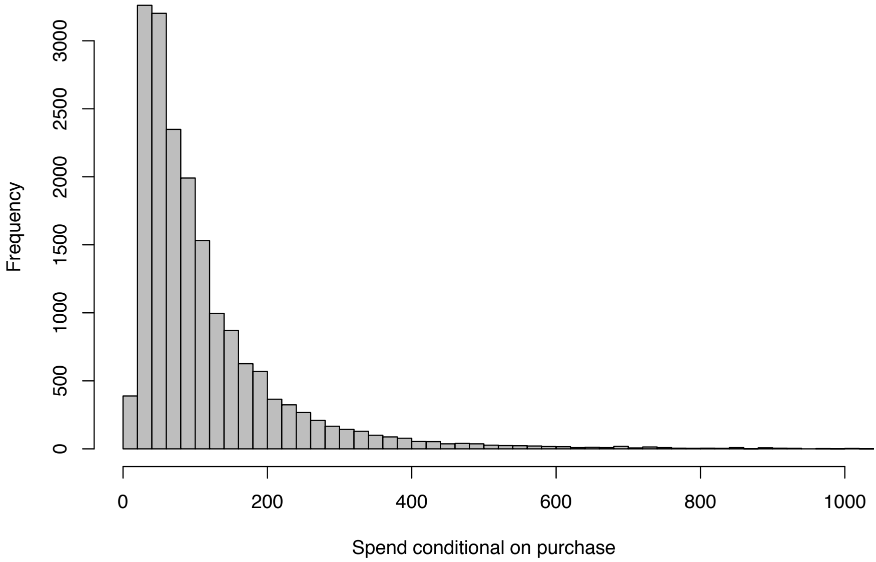
- Data from a large consumer-goods retailer
  - Direct to customers only
- Treatment is **catalog mailing**
- Outcome is consumer spending
- Questions:
  - Does getting a catalog increase spending (ATE)?
  - Who should be mailed a catalog (profit)?

The Data

- $n = 292,657$
- Randomized: $\mathbb{P}[T = 1 \mid X] = 2/3$
- ≈ 150 covariates
- $Y =$ sales from all channels

	Mean	SD
Purchase	0.062	0.24
Spend	7.311	43.55
Spend | Purchase	117.730	132.44
Treatment	0.669	0.47

Distribution of Spending



Deep Network Architectures

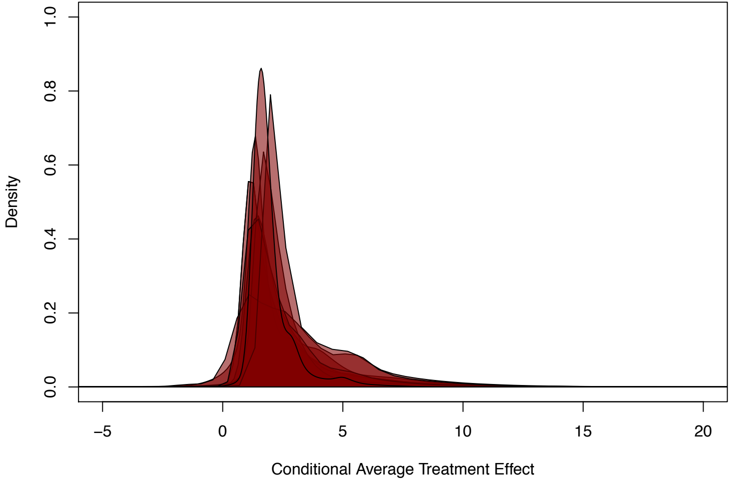
- Single layer networks must be quite wide $\Rightarrow$ many parameters
- Experimented with different/no regularization
- All use SGD, ReLU activation
- Joint estimation was **much** better:

$$\begin{pmatrix} \hat{\mu}_0(\boldsymbol{x}) \\ \hat{\tau}(\boldsymbol{x}) = \hat{\mu}_1(\boldsymbol{x}) - \hat{\mu}_0(\boldsymbol{x}) \end{pmatrix} := \arg\min_{\tilde{\mu}_0, \tilde{\tau}} \sum_{i=1}^{n} \frac{1}{2} \Big(y_i - \tilde{\mu}_0(\boldsymbol{x}_i) - \tilde{\tau}(\boldsymbol{x}_i) t_i \Big)^2$$

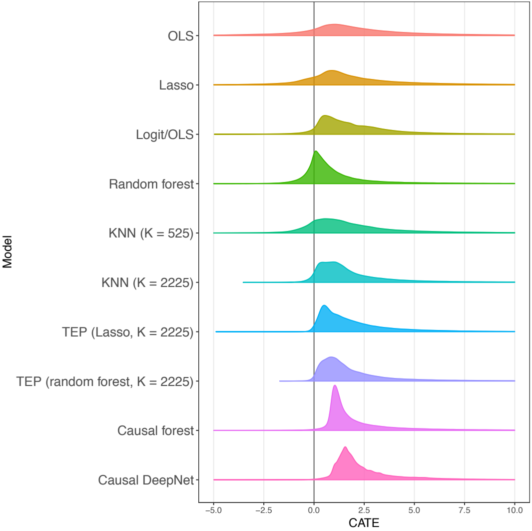
Learning Rate	Widths $[H_1, H_2, ...]$	Dropout $[H_1, H_2, ...]$	Total Parameters	Validation Loss	Training Loss
0.0003	[60]	[0.5]	8702	1405.62	1748.91
0.0003	[100]	[0.5]	14502	1406.48	1751.87
0.0001	[30, 20]	[0.5, 0]	4952	1408.22	1751.20
0.0009	[30, 10]	[0.3, 0.1]	4622	1408.56	1751.62
0.0003	[30, 30]	[0, 0]	5282	1403.57	1738.59
0.0003	[30, 30]	[0.5, 0]	5282	1408.57	1755.28
0.0003	[100, 30, 20]	[0.5, 0.5, 0]	17992	1408.62	1751.52
0.00005	[80, 30, 20]	[0.5, 0.5, 0]	14532	1413.70	1756.93

One Goodness of Fit Measure

- Getting a catalog shouldn't make you buy less
- Plot $\hat{\tau}(\boldsymbol{x}_i)$



Comparison



OLS
Lasso
Logit/OLS
Random forest
KNN (K = 525)
KNN (K = 2225)
TEP (Lasso, K = 2225)
TEP (random forest, K = 2225)
Causal forest
Causal DeepNet
Model
−5.0
−2.5
0.0
2.5
5.0
7.5
10.0
CATE

Semiparametric Results

- Networks gave similar results
- Lines up closely with difference in means (RCT)
- Study ATE and profits from different mailing strategies
  - Loyalty: $s(\boldsymbol{x}_i) = 1$ if purchased in prior calendar year

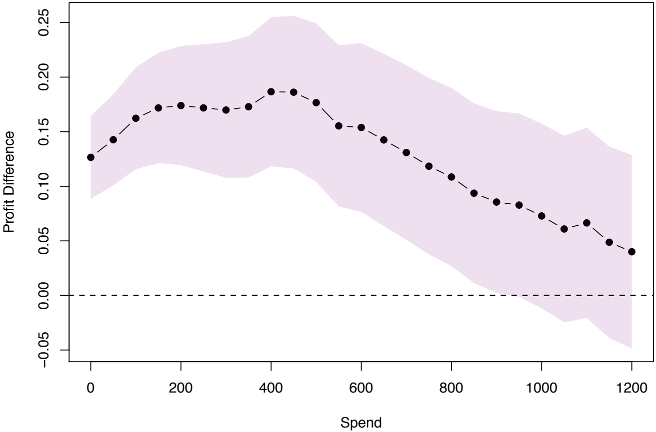
From Network 3:

Estimand:	$\hat{\boldsymbol{\pi}}(\boldsymbol{s})$	**95% CI**
ATE	2.547	[2.223 , 2.872]
π (never treat)	2.027	[1.934 , 2.120]
π (always treat)	2.224	[2.152 , 2.296]
π (loyalty policy)	2.358	[2.283 , 2.434]

Targeting Strategy

Should we target bigger spenders?

- Study $\left\{\pi\left(\text{spend} > \bar{y}\right) - \pi\left(\text{always treat}\right)\right\}$
- *Pointwise* 95% confidence band
- Profits increase in $\bar{y}$ until roughly \$500, then too few are targeted



General Discrete Choice Models

- Essentially these are (semiparametric) interactions models
- with possibly continuous treatments
- and a specified *structure*.

$$\Pr(y_{ij} = 1|X_i) = \frac{\exp(\theta(X_i)\widetilde{Z}_i)}{1 + \sum_j \exp(\theta(X_i)\widetilde{Z}_j)}$$

- Whis is structure important?
  - Information from theory (Shape Restrictions)
  - Guards against overfitting

Theory

- What we dont have theory for
  - IV
  - Fixed Effects (True unobserved heterogeneity)
- What our theory is ok (kind of) for
  - If you want to do flexible control functions.
  - If you assume the the deepnet is a **parametric** model.
  - If you assume the DeepNet converges fast enough.

Deep Architecture for Choice Models

- Our working framework is now

$$\Pr(y_{ij} = 1 | X_i) = \frac{\exp(f_{\mathrm{DNN}}(X_i; \Theta)\widetilde{Z}_i)}{1 + \sum_j \exp(f_{\mathrm{DNN}}(X_i; \Theta)\widetilde{Z}_j)}$$

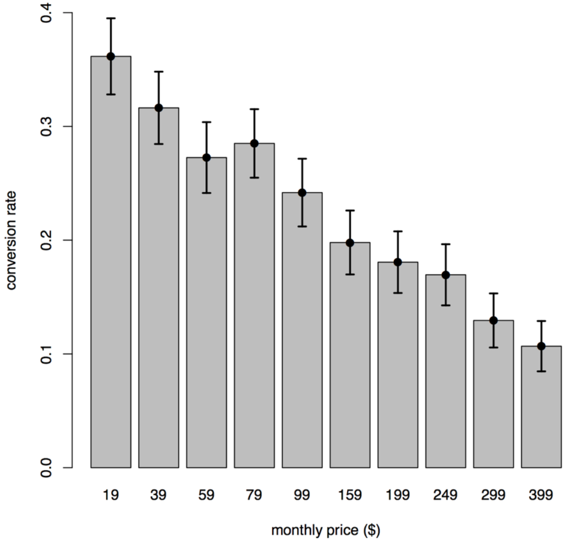
- where

$$f_{\mathrm{DNN}}(X_i; \Theta)\widetilde{Z}_j = \alpha_j(X_i) + \sum_k \beta_k(X_i) \cdot Z_{ik}$$

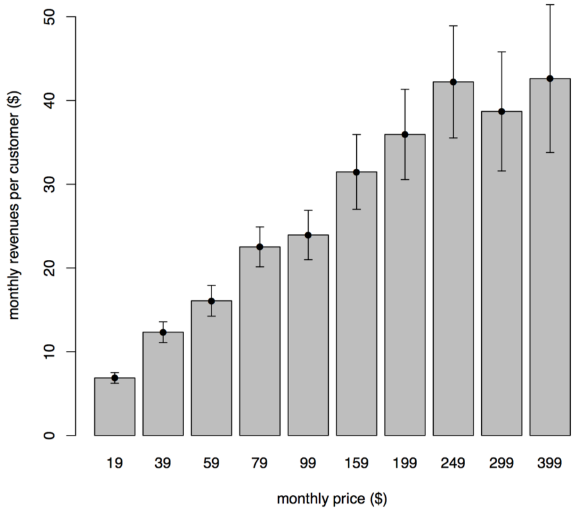
Personalized Pricing

- Ziprecruiter.com: online recruiting platform to match jobseekers and potential employees
- Customers are potential employers that pay a monthly subscription rate to access a stream of matched resumes for posted jobs
- Base price for “starter” firm (small business < 50 employees) was $99/month
- Customers required to register details about firm, job descriptions etc before they can reach paywall
- i.e. we obtain set of features, x_{i}, for each new firm i
- Goal: use methods described to improve pricing at ziprecruiter.com

Ziprecruiter Experiment Data



Ziprecruiter Experiment Data



Example (Ziprecruiter)

- Choice model structure is standard

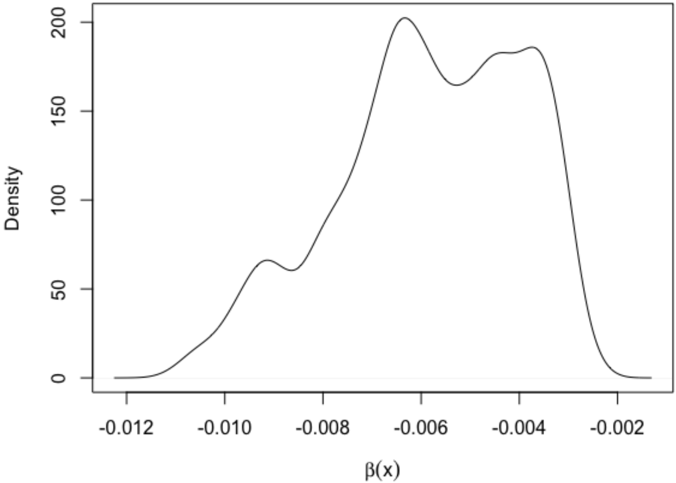
$$\mathbb{P}\left(p_i; x_i\right) = \frac{\exp\left(\alpha\left(x_i; \theta_\alpha\right) + \beta\left(x_i; \theta_\beta\right) p_i\right)}{1 + \exp\left(\alpha\left(x_i; \theta_\alpha\right) + \beta\left(x_i; \theta_\beta\right) p_i\right)}$$

- Let $f(x_i) = \{\alpha(X_i), \beta(X_i)\}$ and $\widetilde{p}_i = [1 \;\; p_i]$ then we can write

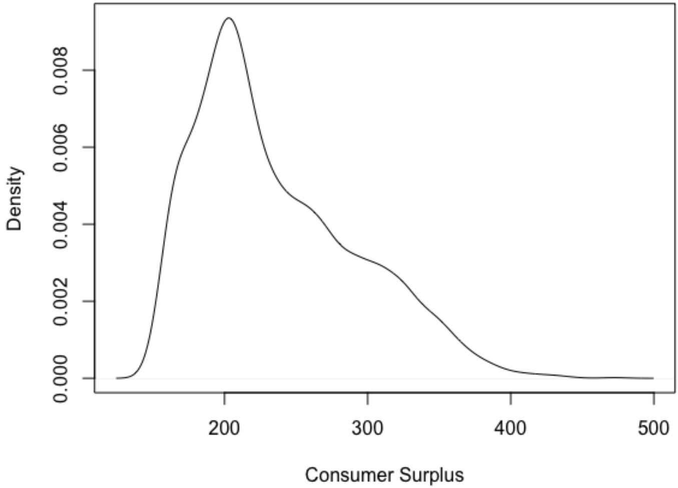
$$\mathbb{P}\left(p_i; x_i\right) = \frac{1}{1 + \exp(-f_{\mathrm{DNN}}(x_i; \Theta)\widetilde{p}_i)}$$

- Estimate and use for optimal uniform and targeted pricing.

Results



Results



Density
0.000
0.002
0.004
0.006
0.008
200
300
400
500
Consumer Surplus

Verification

Pricing Structure	Conversion Rate		Profit per Lead ($)	
	Mean	95% CI	Mean	95% CI
Control	0.26	(0.24,0.29)	25.76	(23.74,28.5)
Implemented Uniform	0.16	(0.13,0.19)	40.05	(32.97,47.5)
Targeted	0.16	(0.13,0.18)	44.49	(35.12,53.71)

(a) Expected Outcomes

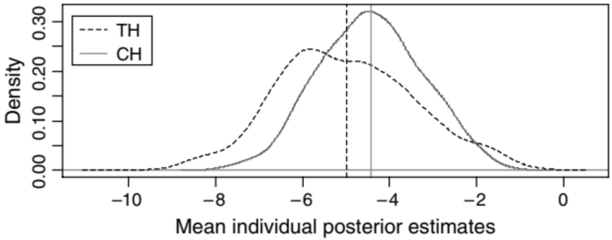
Pricing Structure	Conversion Rate		Profit per Customer ($)	
	Mean	95% CI	Mean	95% CI
Control	0.23	(0.21,0.25)	22.55	(20.75,24.39)
Implemented Uniform	0.15	(0.14,0.17)	37.73	(33.78,41.79)
Targeted	0.15	(0.14,0.16)	41.48	(38.15,45.07)

(b) Realized Outcomes

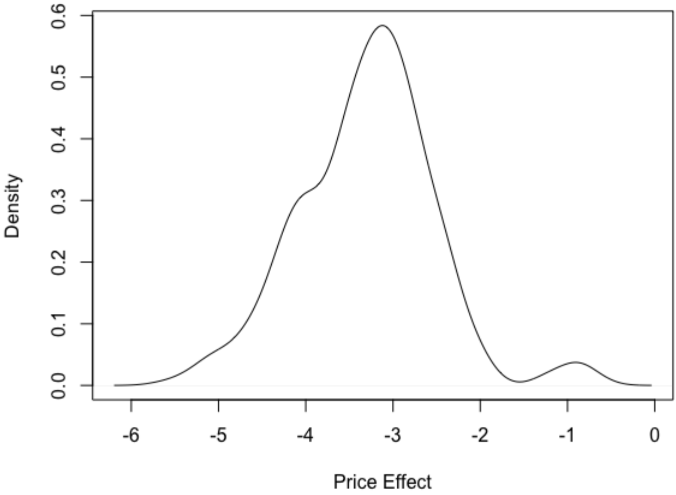
A standard brand choice model

- Data for toothpaste choices
- 7 brands
  - Aim, A&H, Aquafrsh, Colgate, Crest, Mentadent, Pepsodent
- Standard scanner panel data with price and display.
- We estimate choice model with DNN architechture for heterogeneity.
- Use all available consumer characteristics (demo)
- Results comparable to random coefficients.

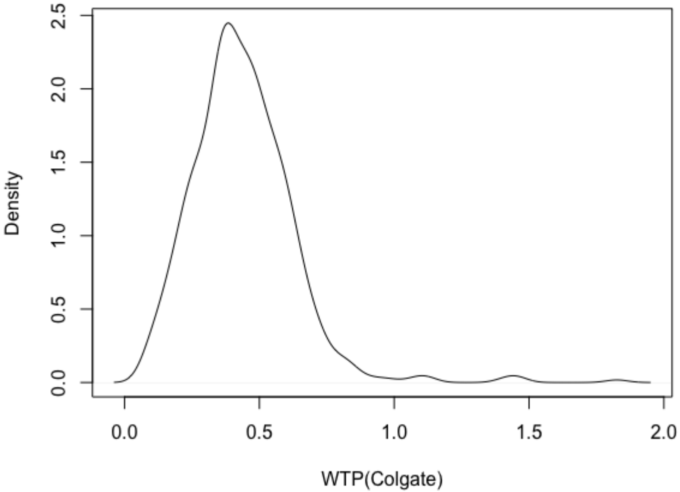
Toothpaste Heterogenity



Toothpaste Heterogenity



Toothpaste Heterogenity



Takeaways

- Deepnets can be used for a flexible approach to heterogenity.
- The approach is practical and feasible.
- Need more case studies and examples.
- They can be embedded in structural models.
- There are avenues for inference.
- Challenges:
  - Architecture is still human driven. Requires search.
  - Leaves out informnation (panel e.g.)
  - Inference is limited.

More generally. . .

- Need to think about Deepnets
- Useful tool in a number of contexts where NP estimators are needed. . .
  - Games
  - Dynamics
  - Selection
  - Causal Inference
  - . . .
- As a prediction too DNNs are useful.
- We need to think about what it is we want to predict.

Discussion: ML in Dynamics

- Estimating NP Objects
- Function Spaces
- Heterogeneity
- Optimization
- Tricks and Treats
- New Ideas
- Challenges: Theory, Inference, Data