A Two-Stage Model of Consideration Set and Choice

Srikanth Jagabathula

Joint with: Paat Rusmevichientong, USC Marshall

Broad research focus: choice models for operational decision making

Choice models: demand models that account for substitution

Broad research focus: choice models for operational decision making

Choice models: demand models that account for substitution

Focus: develop choice models customized for operations

- estimate model parameters from choice data and establish fit
- study operational decision problems using new structures

Broad research focus: choice models for operational decision making

Choice models: demand models that account for substitution

Focus: develop choice models customized for operations

- estimate model parameters from choice data and establish fit
- study operational decision problems using new structures

Sample operational decision problems with choice models

- assortment optimization: offer set to max revenue/sales
- price optimization: price vector to max revenue/sales
- space-aware opt: “best” offer set subject to space constraints

Broad research focus: choice models for operational decision making

Choice models: demand models that account for substitution

Focus: develop choice models customized for operations

- estimate model parameters from choice data and establish fit
- study operational decision problems using new structures

Sample operational decision problems with choice models

- assortment optimization: offer set to max revenue/sales
- price optimization: price vector to max revenue/sales
- space-aware opt: “best” offer set subject to space constraints

Focus of the talk:
challenge of isolating substitution due to price and availability

Existing approaches offer either flexibility or simultaneity, but not both

Demand models in product space: regress demand on prices (double-log model, Cobb-Douglas model, CES model)

- the good: flexible in capturing price-elasticity patterns
- the bad: too many parameters, doesn't account for unavailability

Existing approaches offer either flexibility or simultaneity, but not both

Demand models in product space: regress demand on prices (double-log model, Cobb-Douglas model, CES model)

- the good: flexible in capturing price-elasticity patterns
- the bad: too many parameters, doesn’t account for unavailability

Parametric choice models

- the good: parsimony, capture *both* price and availability effects
- the bad: restricted structures

Existing approaches offer either flexibility or simultaneity, but not both

Demand models in product space: regress demand on prices (double-log model, Cobb-Douglas model, CES model)

- the good: flexible in capturing price-elasticity patterns
- the bad: too many parameters, doesn't account for unavailability

Parametric choice models

- the good: parsimony, capture *both* price and availability effects
- the bad: restricted structures

Our contribution:
semi-parametric choice-based model for pricing

Our focus: demand and revenue share predictions as a function of price

Our contributions

- class of two-stage choice models: flexible and high performance
- methods for estimating these models from transaction data
- statistical and computational guarantees for our methods

There is rich literature on pricing and empirical justification to two-stage models

New stream of choice models in operations:

Farias, Jagabathula, Shah'13; Blanchett, Gallego, Goyal '13; Alptekinoglu, Semple '13

Characterized by focus on predictive (as opposed to explanatory) modeling

There is rich literature on pricing and empirical justification to two-stage models

New stream of choice models in operations:

Farias, Jagabathula, Shah'13; Blanchett, Gallego, Goyal '13; Alptekinoglu, Semple '13

Pricing under specific choice models

- Multinomial logit model

  expected revenues in MNL are not concave in prices (Hanson and Martin '96) but concave in market shares (Song and Xu '97, Dong et al '09)

- Two-level nested logit model

  Li and Huh '11: assume nest specific price sensitivity

  Gallego and Wang '11: Extend to product specific sensitivity

Product line design (joint assortment and pricing)

Hauser and Simmie '81; McBride and Zufryden '88; Dobson and Kalish '93; Yano and Dobson '98

Empirical justification of two stage-models

Gilbride and Allenby '04; Hauser and Wernerfelt '90; Alba and Chattopadhyay ' 85; Howard and Sheth '69; Parkinson and Reilly '79

Pricing under two stage models: Outline of the talk

1. Model

   assumptions, properties, data for estimation

2. Numerical results

   real-world data on durable goods, revenue predictions

3. Model estimation and revenue prediction

   ML estimation, sample and computational complexity

4. Summary/Conclusions

   takeaway messages

Pricing under two stage models: Outline of the talk

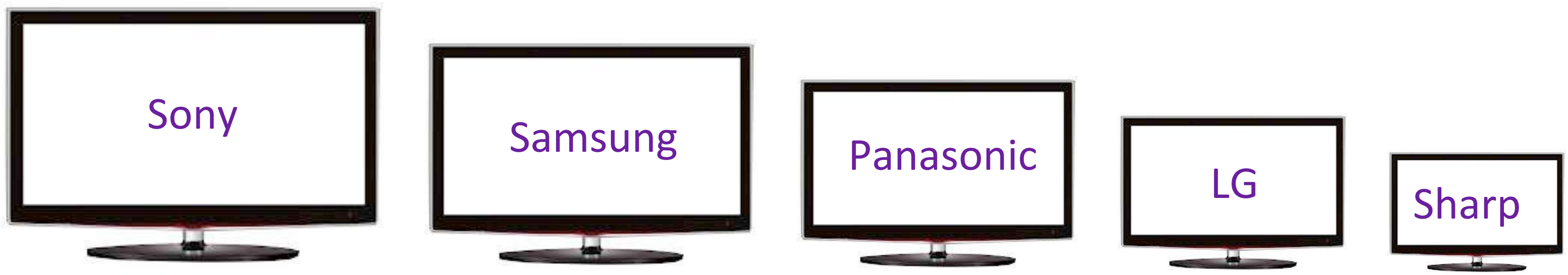
1. Model
   assumptions, properties, data for estimation

2. Numerical results
   real-world data on durable goods, revenue predictions

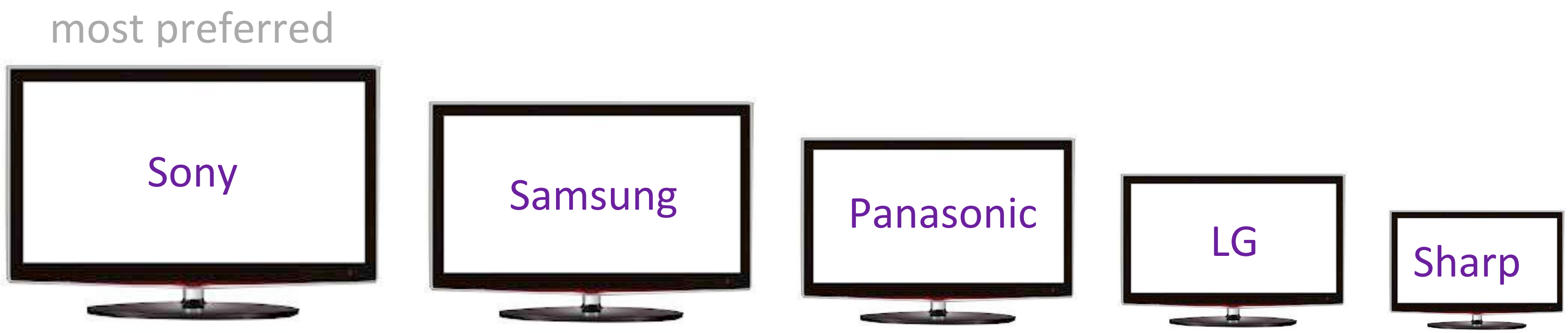
3. Model estimation and revenue prediction
   ML estimation, sample and computational complexity

4. Summary/Conclusions
   takeaway messages

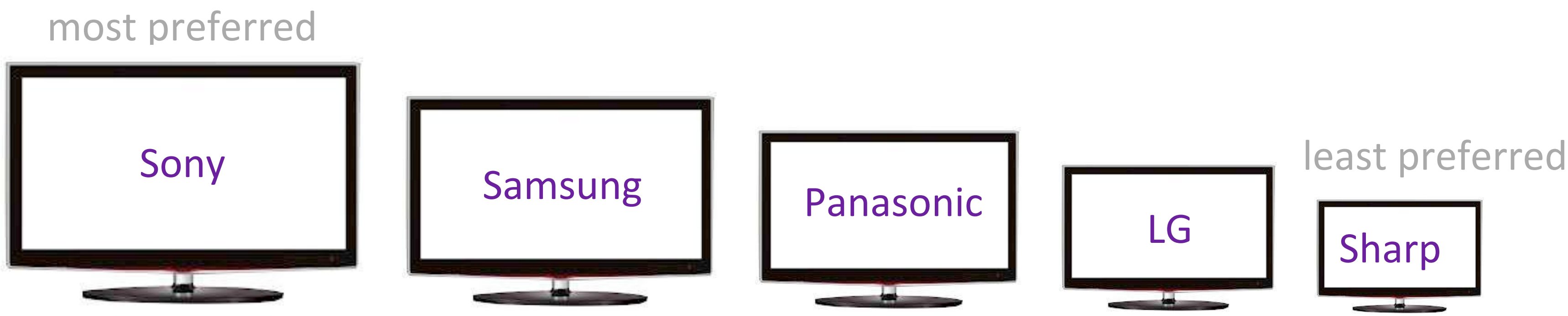
Two-stage model: choose best product with price less than willingness-to-pay



Two-stage model: choose best product with price less than willingness-to-pay



Two-stage model: choose best product with price less than willingness-to-pay



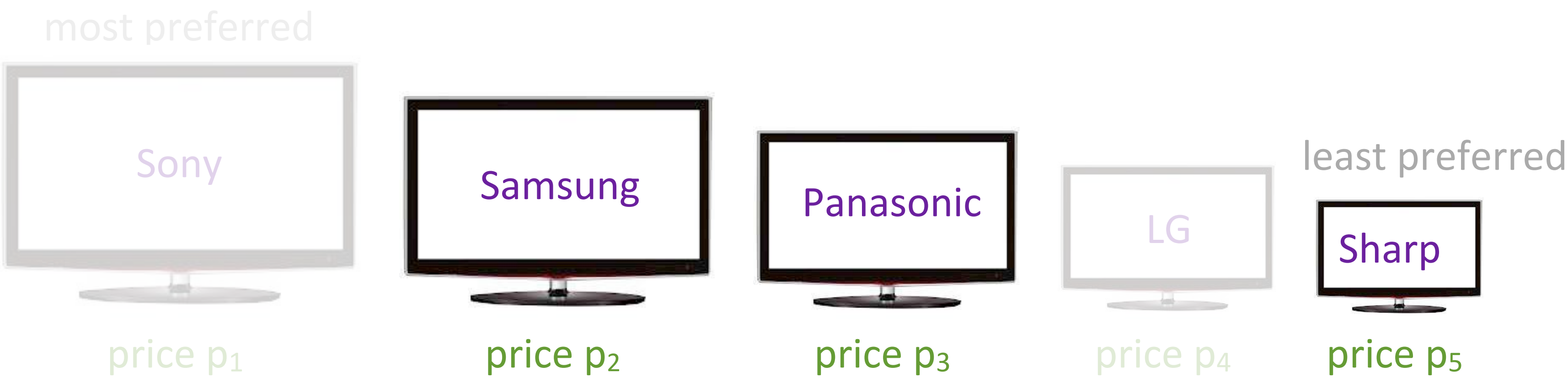
Two-stage model: choose best product with price less than willingness-to-pay

most preferred

least preferred

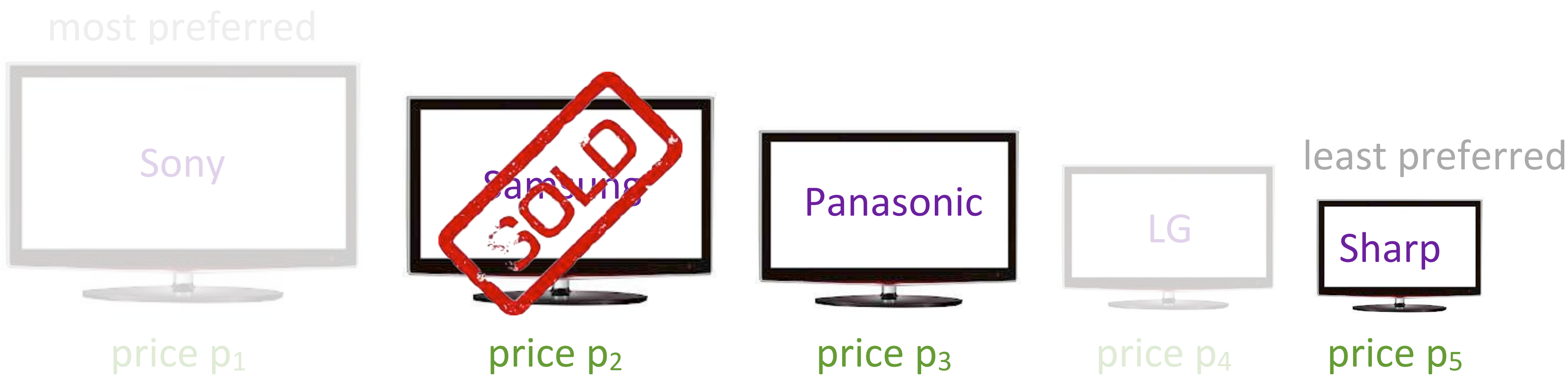
price p_1 price p_2 price p_3 price p_4 price p_5

Two-stage model: choose best product with price less than willingness-to-pay



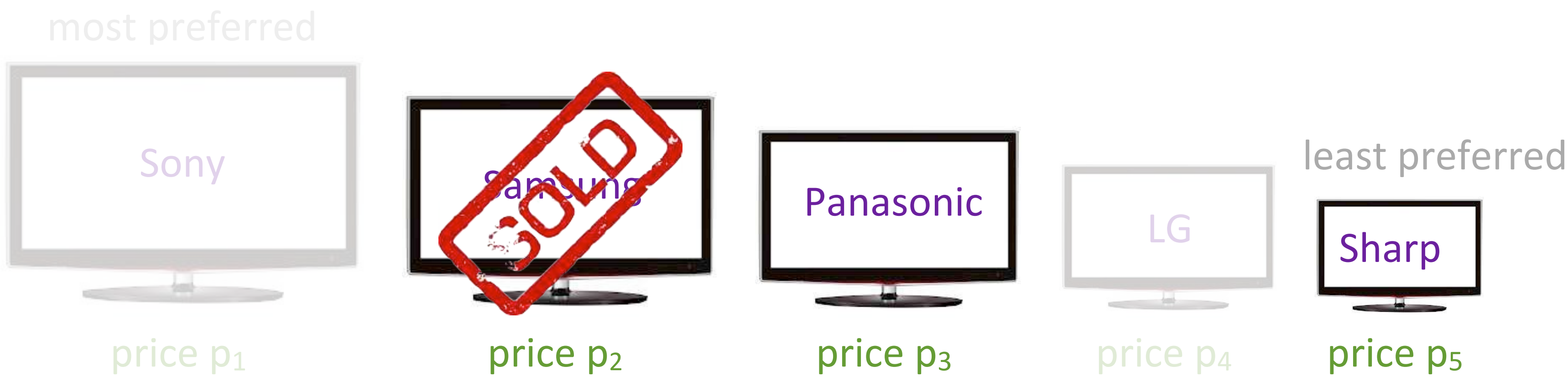
prices p_2, p_3, p_5 $<$ willingness to pay (WTP)

Two-stage model: choose best product with price less than willingness-to-pay



prices p_2, p_3, p_5 $<$ willingness to pay (WTP)

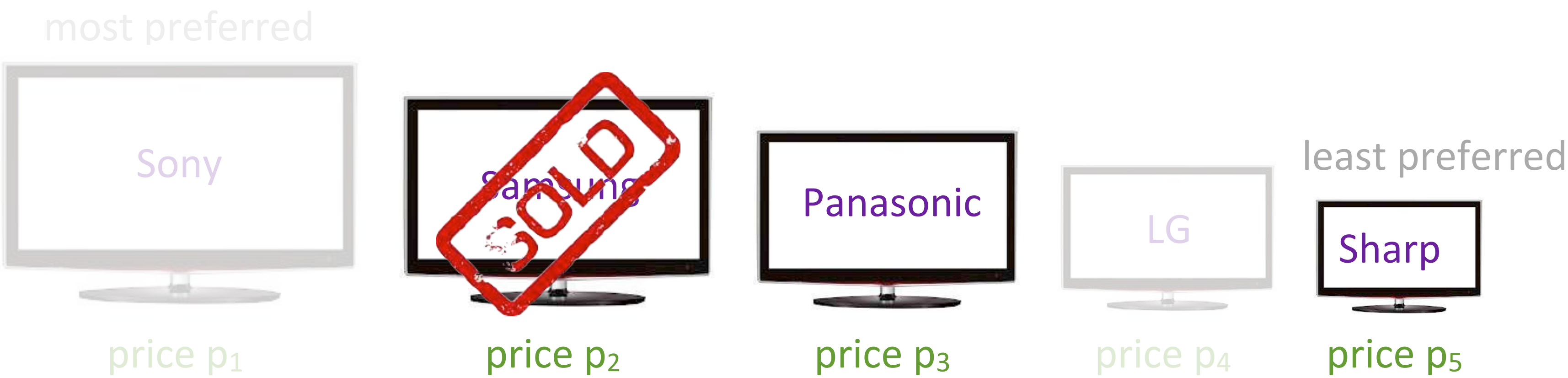
Two-stage model: choose best product with price less than willingness-to-pay



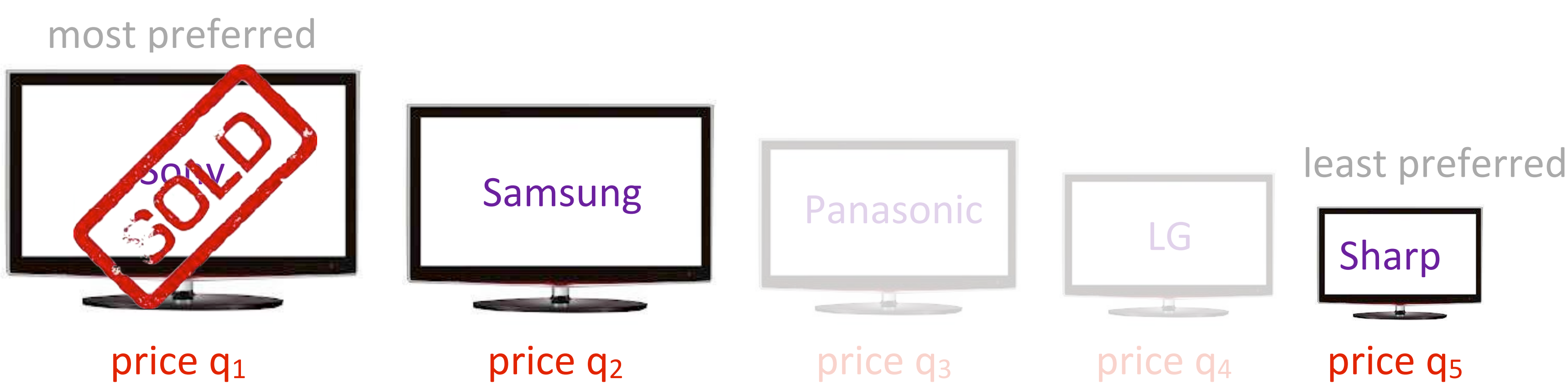
prices p_2, p_3, p_5 $<$ willingness to pay (WTP)

Two-stage model: choose best product with price less than willingness-to-pay

Two-stage model: choose best product with price less than willingness-to-pay



prices p_2, p_3, p_5 < willingness to pay (WTP)



prices q_1, q_2, q_5 < willingness to pay (WTP)

Two-stage model: choose best product with price less than willingness-to-pay

Universe of products $N = \{1, 2, \ldots, n\}$

Products offered at price vector $(p_1, p_2, \ldots, p_n)$

Two-stage model: choose best product with price less than willingness-to-pay

Universe of products $N = \{1, 2, \ldots, n\}$

Products offered at price vector $(p_1, p_2, \ldots, p_n)$

Arriving customer samples

preference ordering

willingness-to-pay (WTP)

chooses most preferred product with price < WTP

Two-stage model: choose best product with price less than willingness-to-pay

Universe of products $N = \{1, 2, \ldots, n\}$

Products offered at price vector $(p_1, p_2, \ldots, p_n)$

Arriving customer samples

preference ordering

willingness-to-pay (WTP)

model parameters

chooses most preferred product with price < WTP

Two-stage model: choose best product with price less than willingness-to-pay

Universe of products $N = \{1, 2, \ldots, n\}$

Products offered at price vector $(p_1, p_2, \ldots, p_n)$

Arriving customer samples | model parameters
preference ordering | preference order distribution
willingness-to-pay (WTP)

chooses most preferred product with price < WTP

Two-stage model: choose best product with price less than willingness-to-pay

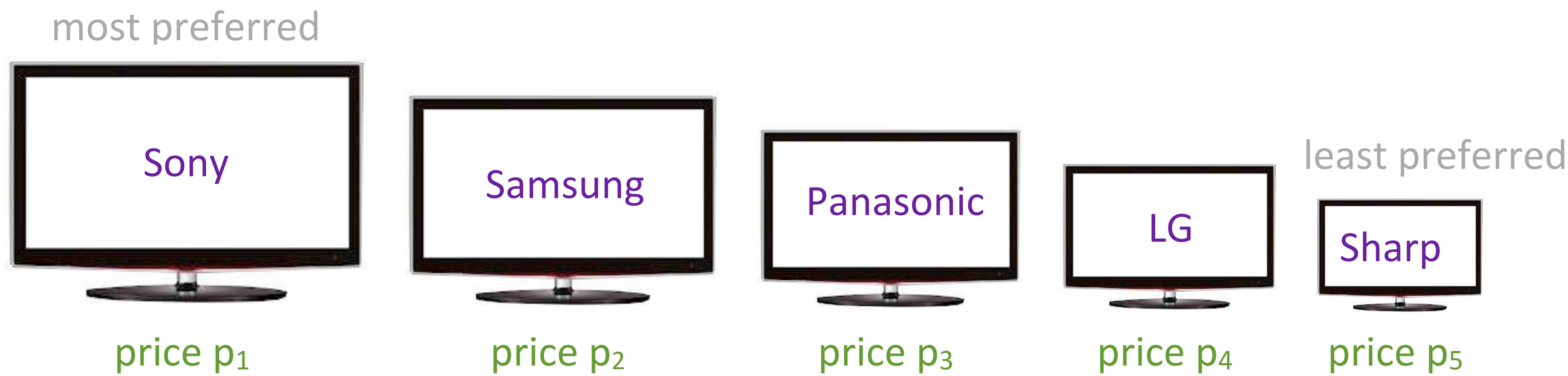
Universe of products $N = \{1, 2, \ldots, n\}$

Products offered at price vector $(p_1, p_2, \ldots, p_n)$

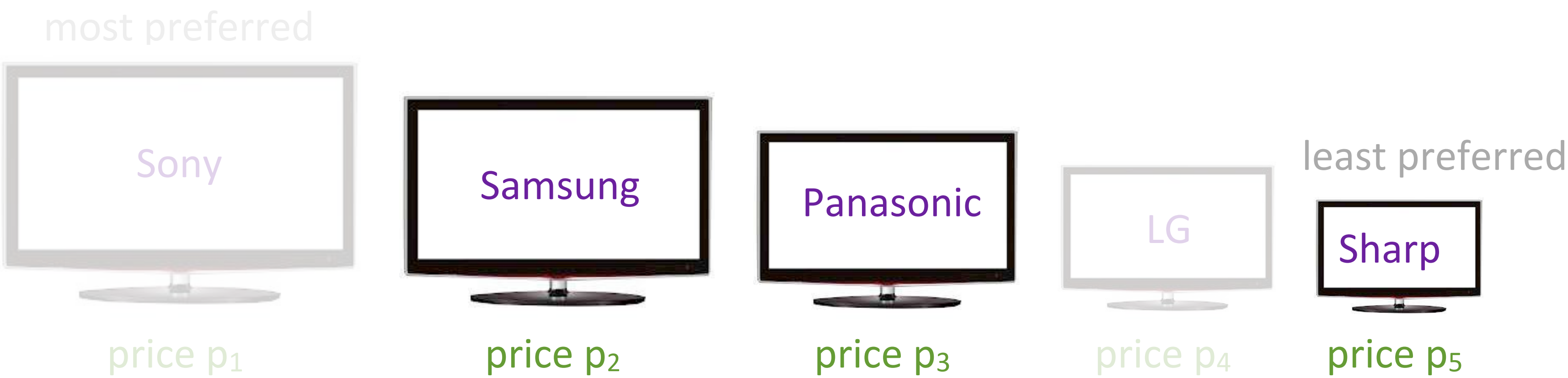
Arriving customer samples	model parameters
preference ordering	preference order distribution
willingness-to-pay (WTP)	distribution over WTP

chooses most preferred product with price < WTP

Rich model class obtained by stapling different distributions over consideration sets and preferences

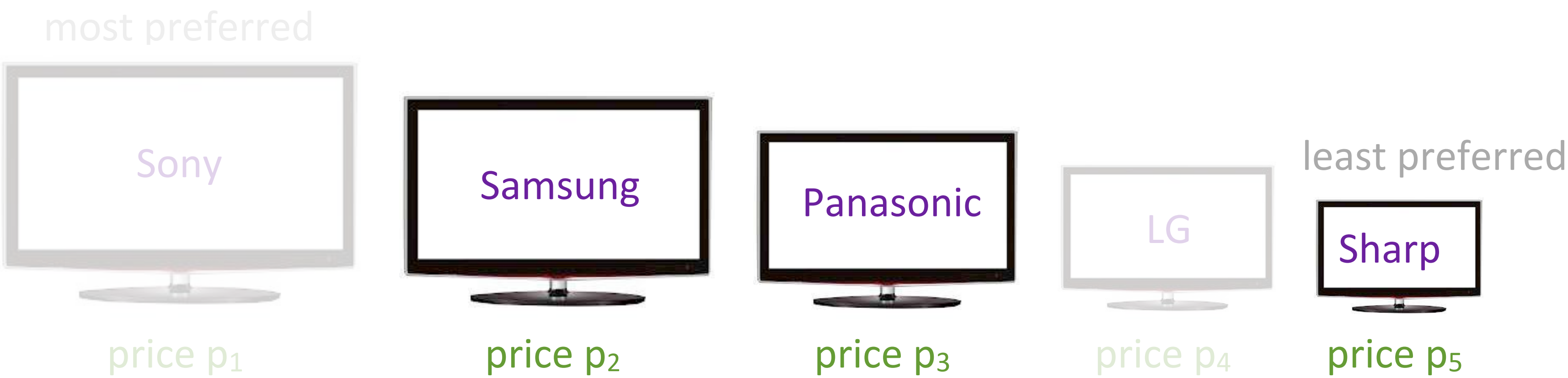


Rich model class obtained by stapling different distributions over consideration sets and preferences



prices p_2, p_3, p_5 $<$ willingness to pay (WTP)

Rich model class obtained by stapling different distributions over consideration sets and preferences



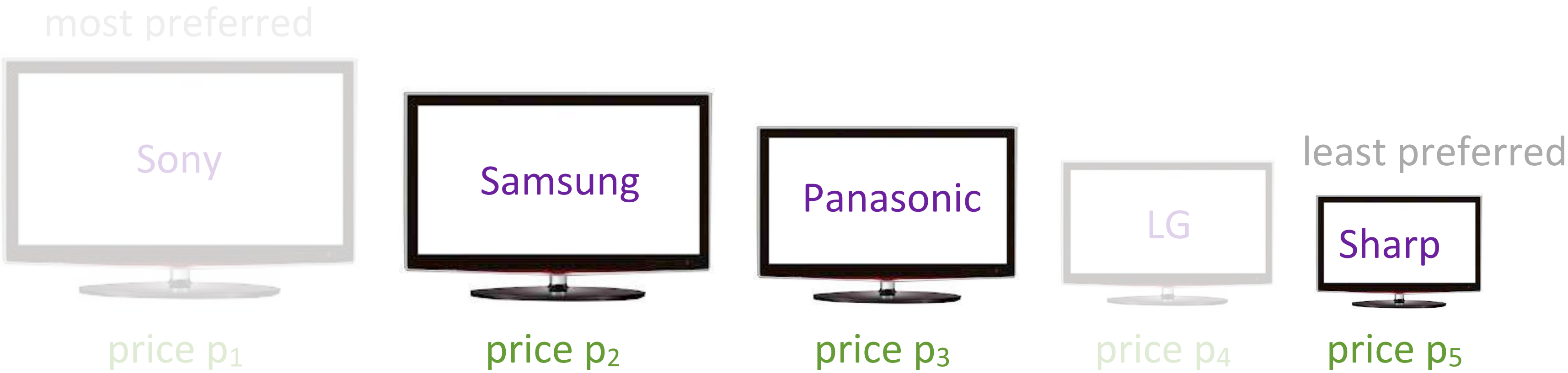
prices p_2, p_3, p_5 < willingness to pay (WTP)

Consideration set: smaller set of relevant products formed before choosing

Models other than WTP-based possible for consideration sets. Examples:

- set to maximize expected utility net of search costs [Hauser and Wernerfelt' 90]
- K-cheapest products...

Rich model class obtained by stapling different distributions over consideration sets and preferences



prices p_2, p_3, p_5 < willingness to pay (WTP)

Consideration set: smaller set of relevant products formed before choosing

Models other than WTP-based possible for consideration sets. Examples:

- set to maximize expected utility net of search costs [Hauser and Wernerfelt' 90]
- K-cheapest products...

Empirical support from Gilbride and Allenby (2004)

We focus on two specific instances of the general model class with WTP model for consideration sets

Two models for the preference list distribution

1. Parametric: multinomial logit (MNL) model
2. Semi-parametric: general distribution over preference lists

MNL-WTP model: WTP model for consideration sets and MNL model for preferences

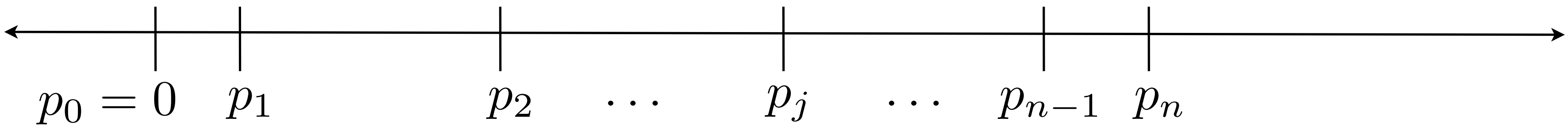
WTP distributed according to $H_\theta(p) = \Pr(w \geq p)$

θ is parameter of the distribution

MNL-WTP model: WTP model for consideration sets and MNL model for preferences

WTP distributed according to $H_\theta(p) = \Pr(w \geq p)$

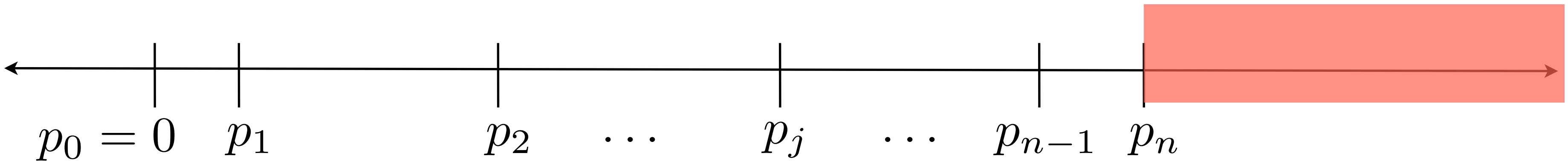
θ is parameter of the distribution



MNL-WTP model: WTP model for consideration sets and MNL model for preferences

WTP distributed according to $H_\theta(p) = \Pr(w \geq p)$

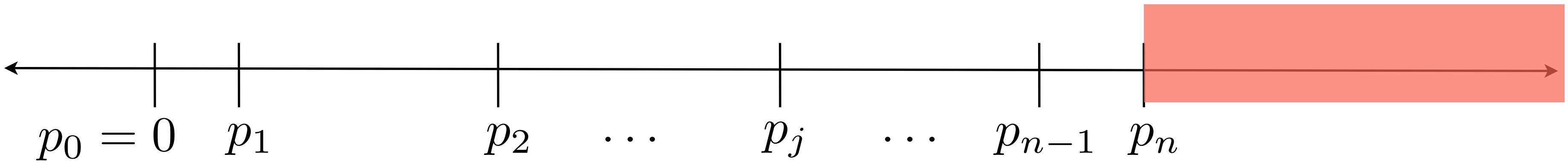
θ is parameter of the distribution



MNL-WTP model: WTP model for consideration sets and MNL model for preferences

WTP distributed according to $H_\theta(p) = \Pr(w \geq p)$

θ is parameter of the distribution

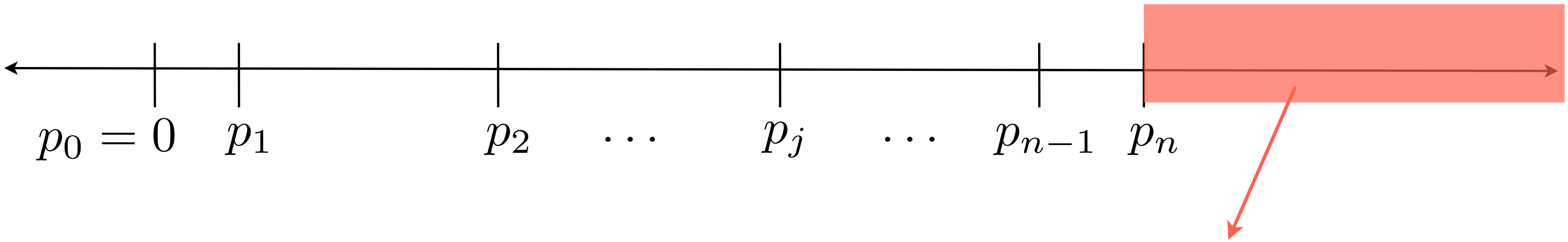


fraction with WTP >= p_n and who buy j

MNL-WTP model: WTP model for consideration sets and MNL model for preferences

WTP distributed according to $H_\theta(p) = \Pr(w \geq p)$

θ is parameter of the distribution

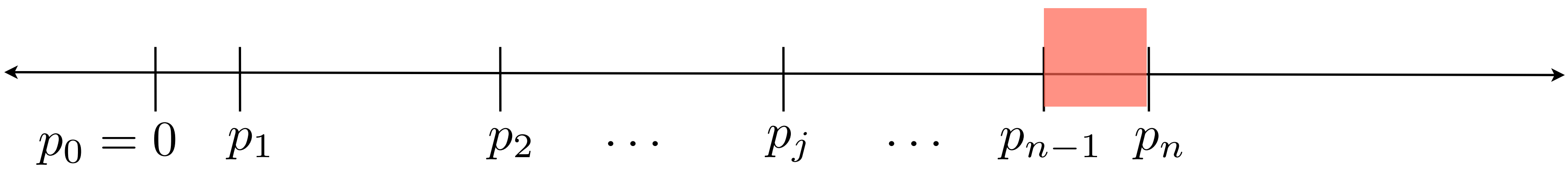


fraction with WTP >= p_n and who buy j $\quad H_\theta(p_n)\dfrac{v_j}{v_0 + v_1 + \cdots + v_n}$

MNL-WTP model: WTP model for consideration sets and MNL model for preferences

WTP distributed according to $H_\theta(p) = \Pr(w \geq p)$

θ is parameter of the distribution



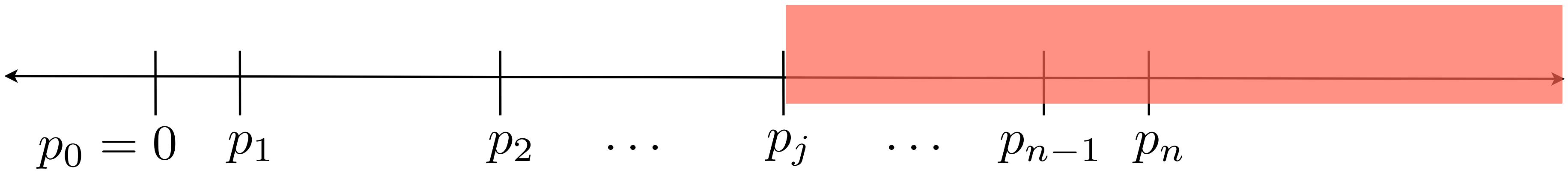
fraction with WTP >= p_n, < p_{n-1} and who buy j

$$(H_\theta(p_{n-1}) - H_\theta(p_n)) \frac{v_j}{v_0 + v_1 + \cdots + v_{n-1}}$$

MNL-WTP model: WTP model for consideration sets and MNL model for preferences

WTP distributed according to $H_\theta(p) = \Pr(w \geq p)$

θ is parameter of the distribution



Only customers with WTP >= p_j buy product j

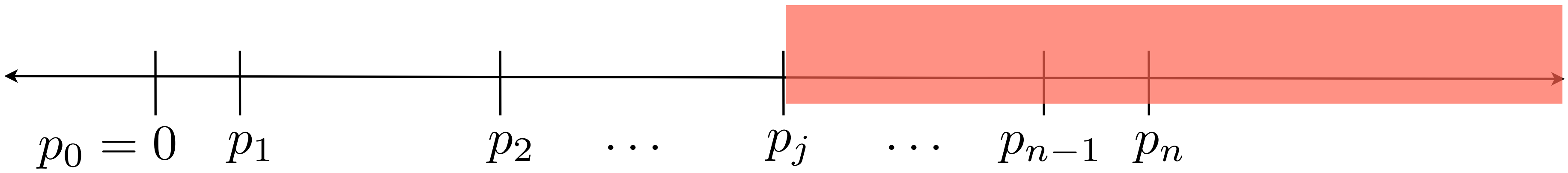
$$\mathbb{P}(j) = \sum_{i=j}^{n} \left(H_\theta(p_i) - H_\theta(p_{i+1})\right) \frac{v_j}{v_0 + \cdots + v_i}$$

we set $p_{n+1} = \infty$

Semi-parametric model: WTP model for consideration sets and general distribution over preference lists

WTP distributed according to $H_\theta(p) = \Pr(w \geq p)$

θ is parameter of the distribution



Only customers with WTP >= p_j buy product j

$$\mathbb{P}(j) = \sum_{i=j}^{n} \left(H_\theta(p_i) - H_\theta(p_{i+1})\right) \mathbb{P}_\lambda(j|\{0, 1, 2, \ldots, i\})$$

we set $p_{n+1} = \infty$

Our goal: estimate models and predict revenues using historical transaction data

Data:

observed sales (market shares) of products at different price vectors

Our goal: estimate models and predict revenues using historical transaction data

Data:

observed sales (market shares) of products at different price vectors

product 1 fraction of sales at price p_1 y_{p1}

Our goal: estimate models and predict revenues using historical transaction data

Data:

observed sales (market shares) of products at different price vectors

product 1	fraction of sales at price	p_1	y_{p1}
product 2	fraction of sales at price	p_2	y_{p2}

Our goal: estimate models and predict revenues using historical transaction data

Data:

observed sales (market shares) of products at different price vectors

product 1	fraction of sales at price	p_1	y_{p1}
product 2	fraction of sales at price	p_2	y_{p2}
$\vdots$			
product n	fraction of sales at price	p_n	y_{pn}

Our goal: estimate models and predict revenues using historical transaction data

Data:

observed sales (market shares) of products at different price vectors

product 1	fraction of sales at price	p_1	y_{p1}
product 2	fraction of sales at price	p_2	y_{p2}
	⋮		
product n	fraction of sales at price	p_n	y_{pn}

at a set of training price vectors $p \in \mathcal{P}$

Our goal: estimate models and predict revenues using historical transaction data

Data:

observed sales (market shares) of products at different price vectors

product 1	fraction of sales at price	p_1	y_{p1}
product 2	fraction of sales at price	p_2	y_{p2}
⋮			
product n	fraction of sales at price	p_n	y_{pn}

at a set of training price vectors $p \in \mathcal{P}$

[if a product is not offered, we set its price to infinity]

Our goal: estimate models and predict revenues using historical transaction data

We accomplish two tasks:

Our goal: estimate models and predict revenues using historical transaction data

We accomplish two tasks:

1. Procedure for estimating model parameters
   - yields point identification if possible, set identification o.w.
2. Procedure for predicting expected revenues at new prices
   - useful when only set identification of parameters possible

Our goal: estimate models and predict revenues using historical transaction data

We accomplish two tasks:

1. Procedure for estimating model parameters
   - yields point identification if possible, set identification o.w.
2. Procedure for predicting expected revenues at new prices
   - useful when only set identification of parameters possible

Guarantees:

1. Statistical
   - sample complexity and identification of model
2. Computational
   - when are our algorithms efficient

Pricing under two stage models: Outline of the talk

1. Model
   assumptions, properties, data for estimation

2. Numerical results
   real-world data on durable goods, revenue predictions

3. Model estimation and revenue prediction
   ML estimation, sample and computational complexity

4. Summary/Conclusions
   takeaway messages

We ran simulations with ground-truth model as double-log demand model

Simple demand model for generating training data

$$\log D_j = \alpha_j + \sum_{i=1}^{n} \beta_{ij} \log p_j, \quad 1 \le j \le n$$

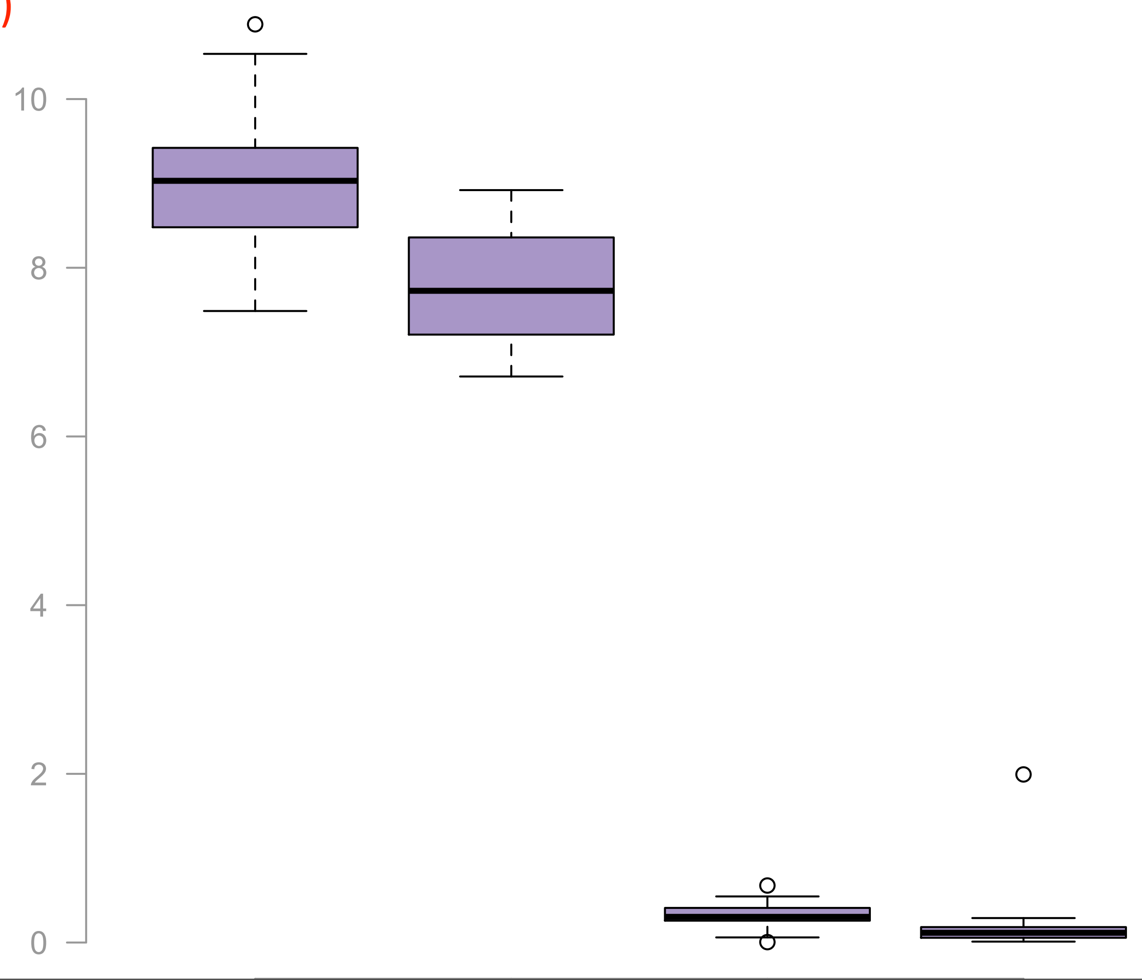
Training data

- 20 price vectors over 10 products generated randomly
- two sets: same and random price ordering

Test MNL and Semiparam methods on 30 test price vectors

Semi-parametric method manages to find good approx's to models not based on utilities

	MNL-Random	MNL-Same	SP-Random	SP-Same
Mean relative error (lower is better)	903.1%	778.8%	32.61%	18.95%



Relative errors in predicting expected revenues at 30 test price vectors

We ran experiments on real-world dataset on sales of durable goods

ISMS dataset 1

- panel data on purchase, return of durable goods
- 19,936 households; 292 product categories; 173,262 transactions
- over time period Dec 1998 to Nov 2004

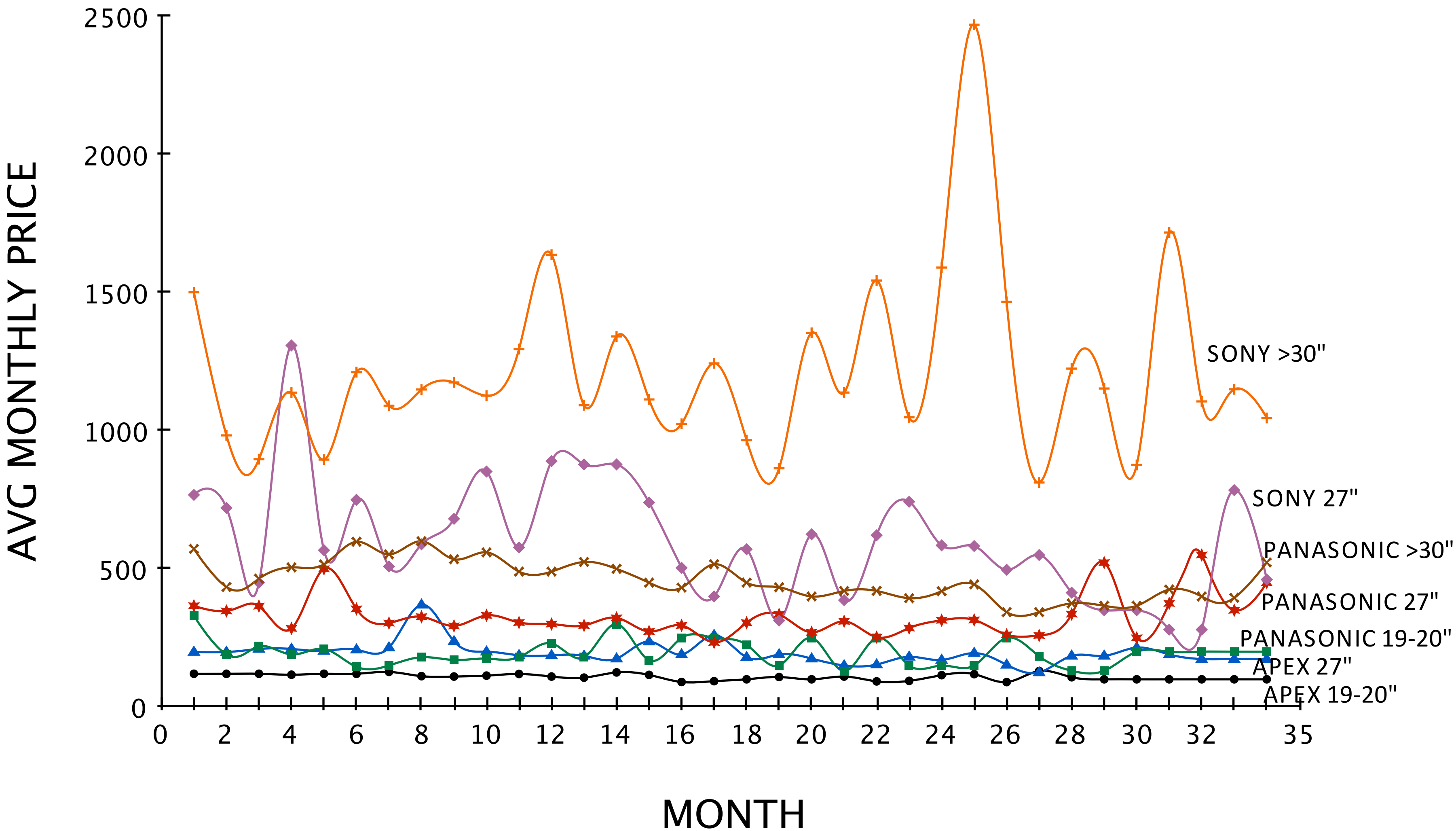
We focused on TV sales from Feb ’02 to Nov ’04 (34 months)

- product is defined by (brand, size) combination: total 7 products
- we aggregate sales by month
- price in a month is fixed at average selling price during the month

Data used for experiments

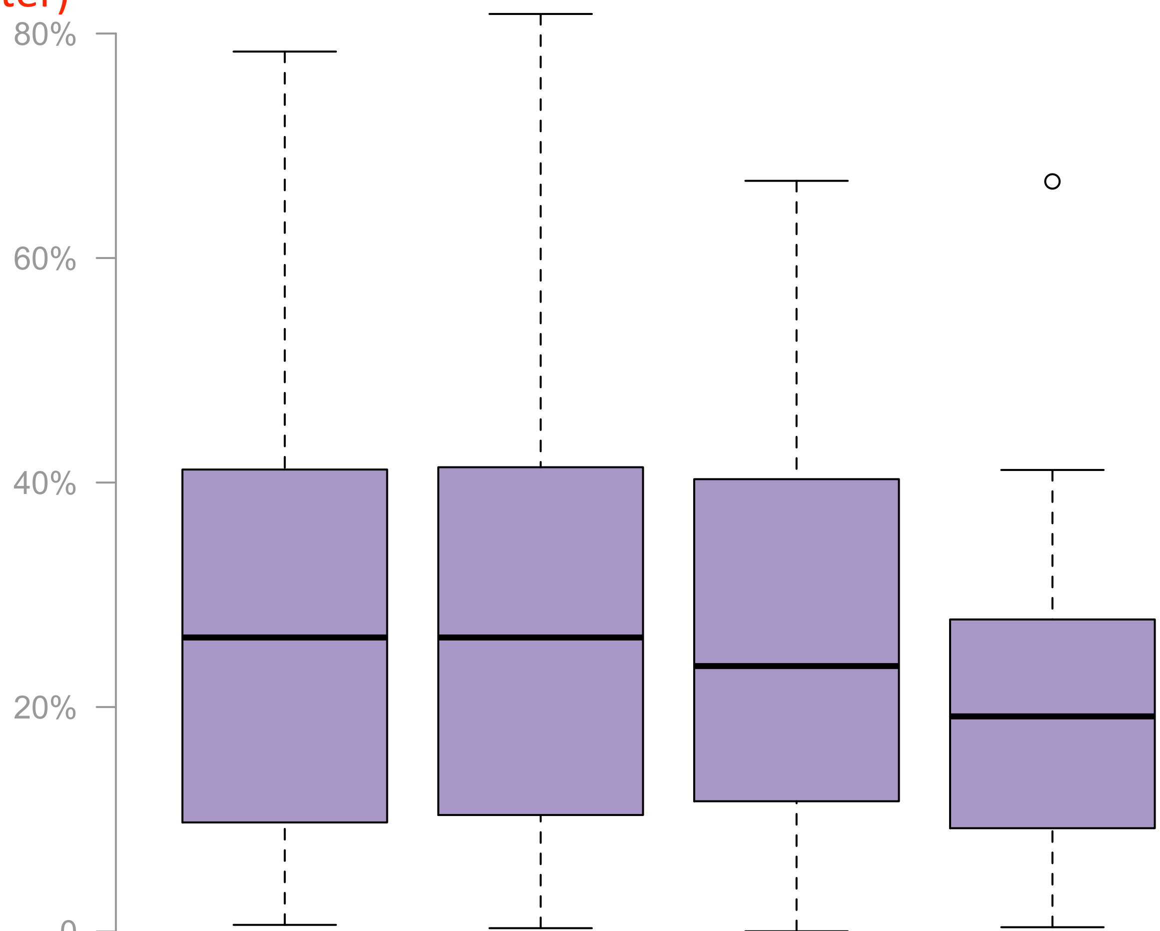
- 7 products
- offered over 32 selling periods at different prices
- aggregate sales for each of the products during selling period

Monthly price variation of the seven products: price ordering changes but only by a little



Semi-parametric method outperforms benchmark methods by 30% in prediction accuracy

	Naive-MNL	MNL	MNL-WTP	Semiparam
Mean relative error (lower is better)	28.81%	29.33%	26.96%	20.18%



Relative errors in predicting expected revenues in 2-fold cross-validation

Pricing under two stage models: Outline of the talk

1. Model
   assumptions, properties, data for estimation
2. Numerical results
   real-world data on durable goods, revenue predictions
3. Model estimation and revenue prediction
   ML estimation, sample and computational complexity
4. Summary/Conclusions
   takeaway messages

We use the method of maximum likelihood estimation to estimate model parameters

Popular method of maximum likelihood (ML) estimation

- selects parameters that maximize log-likelihood
- estimates have good asymptotic properties
- but limited when there are multiple ML estimates

$$\hat{\theta}, \hat{\lambda} = \arg\max_{\theta,\lambda} \sum_{p\in\mathcal{P}} \sum_{j=0}^{n} y_{pj} \log \mathbb{P}_{\theta,\lambda}(j;p)$$

We use the method of maximum likelihood estimation to estimate model parameters

Popular method of maximum likelihood (ML) estimation

- selects parameters that maximize log-likelihood
- estimates have good asymptotic properties
- but limited when there are multiple ML estimates

$$\hat{\theta}, \hat{\lambda} = \arg\max_{\theta,\lambda} \sum_{p \in \mathcal{P}} \sum_{j=0}^{n} y_{pj} \log \mathbb{P}_{\theta,\lambda}(j; p)$$

Theorem: When transaction data is generated according to a ground-truth two-stage model and observed probabilities are exact (no finite sample errors), then every ML estimate must satisfy the following moment equations

$$\mathbb{P}_{\hat{\theta},\hat{\lambda}}(j; p) = y_{pj}\,, \;\forall\; j \in N \cup \{0\}, p \in \mathcal{P}$$

Semi-parametric model: in general, only set identification possible

Any parameters satisfying following moment conditions is an ML estimate

$$\mathbb{P}_{\hat{\theta},\hat{\lambda}}(j;p) = y_{pj},\ \forall\ j \in N \cup \{0\}, p \in \mathcal{P}$$

Semi-parametric model: in general, only set identification possible

Any parameters satisfying following moment conditions is an ML estimate

$$\mathbb{P}_{\hat{\theta},\hat{\lambda}}(j;p) = y_{pj}, \; \forall\, j \in N \cup \{0\}, p \in \mathcal{P}$$

Number of moment equations: $(n+1)|\mathcal{P}|$

Number of unknowns: $|\theta| + (n+1)!$

Semi-parametric model: in general, only set identification possible

Any parameters satisfying following moment conditions is an ML estimate

$$\mathbb{P}_{\hat{\theta},\hat{\lambda}}(j;p) = y_{pj}, \; \forall \; j \in N \cup \{0\}, p \in \mathcal{P}$$

Number of moment equations: $(n+1)|\mathcal{P}|$

Number of unknowns: $|\theta| + (n+1)!$

Reasonable restrictions allow point identification of θ

Semi-parametric model: in general, only set identification possible

Any parameters satisfying following moment conditions is an ML estimate

$$\mathbb{P}_{\hat{\theta},\hat{\lambda}}(j;p) = y_{pj}, \; \forall \, j \in N \cup \{0\}, p \in \mathcal{P}$$

Number of moment equations: $(n+1)|\mathcal{P}|$

Number of unknowns: $|\theta| + (n+1)!$

Reasonable restrictions allow point identification of θ

Theorem [Sher, Fox, Kim, Bajari,'11]: When there are at least four alternatives, distribution over preference lists is not identified from choice probabilities

Semi-parametric model: in general, only set identification possible

Any parameters satisfying following moment conditions is an ML estimate

$$\mathbb{P}_{\hat{\theta},\hat{\lambda}}(j;p) = y_{pj}, \; \forall \, j \in N \cup \{0\}, p \in \mathcal{P}$$

Number of moment equations: $(n+1)|\mathcal{P}|$

Number of unknowns: $|\theta| + (n+1)!$

Reasonable restrictions allow point identification of θ

Theorem [Sher, Fox, Kim, Bajari,'11]: When there are at least four alternatives, distribution over preference lists is not identified from choice probabilities

# of choice probabilities $n2^n$

# of unknowns $(n+1)! \approx 2^{n \log n}$

Semi-parametric model: estimation of WTP distribution

Moment equations corresponding to highest priced product

$$y_{pt_p} = H_\theta(p_{t_p})\mathbb{P}_\lambda(t_p|N),\ p \in \mathcal{P}$$

Semi-parametric model: estimation of WTP distribution

Moment equations corresponding to highest priced product

$$y_{pt_p} = H_\theta(p_{t_p})\mathbb{P}_\lambda(t_p|N),\ p \in \mathcal{P}$$

highest priced product at price vector *p*

Semi-parametric model: estimation of WTP distribution

Moment equations corresponding to highest priced product

$$y_{pt_p} = H_\theta(p_{t_p})\mathbb{P}_\lambda(t_p|N), \; p \in \mathcal{P}$$

Number of moment equations: $|\mathcal{P}|$ (assumes all top prices are distinct)

Number of unknowns: $|\theta| + |T|$

Semi-parametric model: estimation of WTP distribution

Moment equations corresponding to highest priced product

$$y_{pt_p} = H_\theta(p_{t_p})\mathbb{P}_\lambda(t_p|N), \; p \in \mathcal{P}$$

Number of moment equations: $|\mathcal{P}|$ (assumes all top prices are distinct)

Number of unknowns: $|\theta| + |T|$

$$T = \{t_p \colon p \in \mathcal{P}\} \subset N$$

Semi-parametric model: estimation of WTP distribution

Moment equations corresponding to highest priced product

$$y_{pt_p} = H_\theta(p_{t_p})\mathbb{P}_\lambda(t_p|N), \ p \in \mathcal{P}$$

Number of moment equations: $|\mathcal{P}|$ (assumes all top prices are distinct)

Number of unknowns: $|\theta| + |T|$

Semi-parametric model: estimation of WTP distribution

Moment equations corresponding to highest priced product

$$y_{pt_p} = H_\theta(p_{t_p})\mathbb{P}_\lambda(t_p|N),\ p \in \mathcal{P}$$

Number of moment equations: $|\mathcal{P}|$ (assumes all top prices are distinct)

Number of unknowns: $|\theta| + |T|$

In the presence of noise, we solve the following optimization problem

$$\min_{\theta,x} \sum_{p\in\mathcal{P}} \left(\frac{H_\theta(p_{t_p})}{y_{pt_p}} - x_{t_p} \right)^2 \text{ s. t. } x_j \geq 1, \forall j \in N$$

Semi-parametric model: estimation of WTP distribution

A general family for WTP distributions is mixture of exponentials (MEXP)

$$H_\theta(w) = \sum_{\ell=1}^{L} \gamma_\ell \exp(-w\phi_\ell)$$

Mixture of exponential distributions are popular

- in survival analysis
- to model long-tail distributions

MEXP is reasonable for WTP distributions because they

- are right censored (only lower bound available as in survival analysis)
- have a long-tail

Semi-parametric model: estimation of distribution over preference lists

Given θ, we obtain polytope identification of λ

$$\sum_{i=0}^{n} \left(H_\theta(p_i) - H_\theta(p_{i+1})\right) \mathbb{P}_\lambda(j | \{0, 1, 2, \ldots, i\} = y_{pj}, \forall p \in \mathcal{P}$$

Semi-parametric model: estimation of distribution over preference lists

Given θ, we obtain polytope identification of λ

$$\sum_{i=0}^{n} (H_\theta(p_i) - H_\theta(p_{i+1}))\, \mathbb{P}_\lambda(j|\{0, 1, 2, \ldots, i\} = y_{pj}, \forall p \in \mathcal{P}$$

linear function of λ

Semi-parametric model: estimation of distribution over preference lists

Given θ, we obtain polytope identification of λ

$$\sum_{i=0}^{n} (H_\theta(p_i) - H_\theta(p_{i+1}))\, \mathbb{P}_\lambda(j|\{0,1,2,\ldots,i\} = y_{pj}, \forall p \in \mathcal{P}$$

linear function of λ

Such polytope identification can be used in many ways.

Semi-parametric model: estimation of distribution over preference lists

Given θ, we obtain polytope identification of λ

$$\sum_{i=0}^{n} (H_\theta(p_i) - H_\theta(p_{i+1}))\, \mathbb{P}_\lambda(j|\{0,1,2,\ldots,i\} = y_{pj}, \forall p \in \mathcal{P}$$

linear function of λ

Such polytope identification can be used in many ways.

For instance, we can predict expected revenues at a new price vector q

Semi-parametric model: estimation of distribution over preference lists

Given θ, we obtain polytope identification of λ

$$\sum_{i=0}^{n} \left(H_\theta(p_i) - H_\theta(p_{i+1})\right) \mathbb{P}_\lambda(j|\{0,1,2,\ldots,i\} = y_{pj}, \forall p \in \mathcal{P}$$

linear function of λ

Such polytope identification can be used in many ways.

For instance, we can predict expected revenues at a new price vector q

$$\min_{\lambda} \; R(q) = \sum_{j=1}^{n} q_j \mathbb{P}_{\lambda,\theta}(j)$$

subject to $\sum_{i=0}^{n} \left(H_\theta(p_i) - H_\theta(p_{i+1})\right) \mathbb{P}_\lambda(j|\{0,1,2,\ldots,i\} = y_{pj}, \forall p \in \mathcal{P}$

Semi-parametric model:
we raise computational and statistical questions

polytope identification of λ

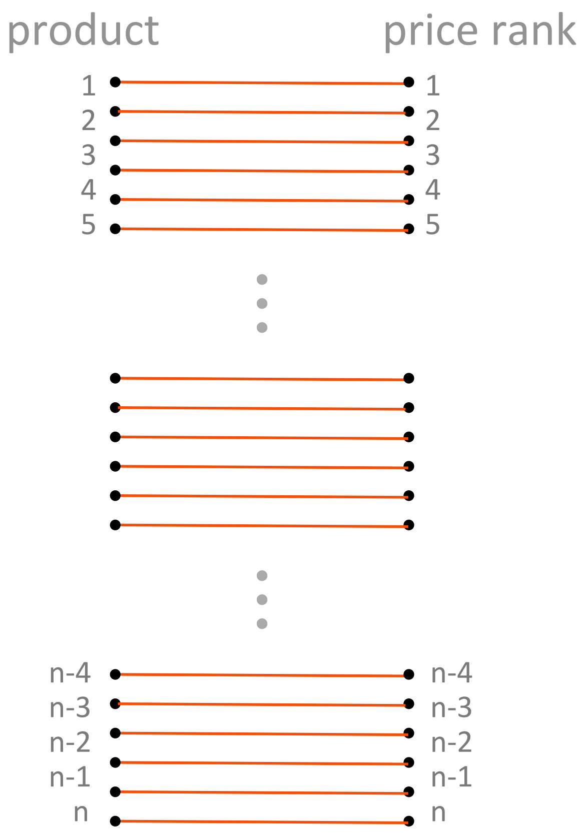
$$\sum_{i=0}^{n} (H_\theta(p_i) - H_\theta(p_{i+1}))\, \mathbb{P}_\lambda(j|\{0,1,2,\ldots,i\} = y_{pj}, \forall p \in \mathcal{P}$$

We raise two questions:

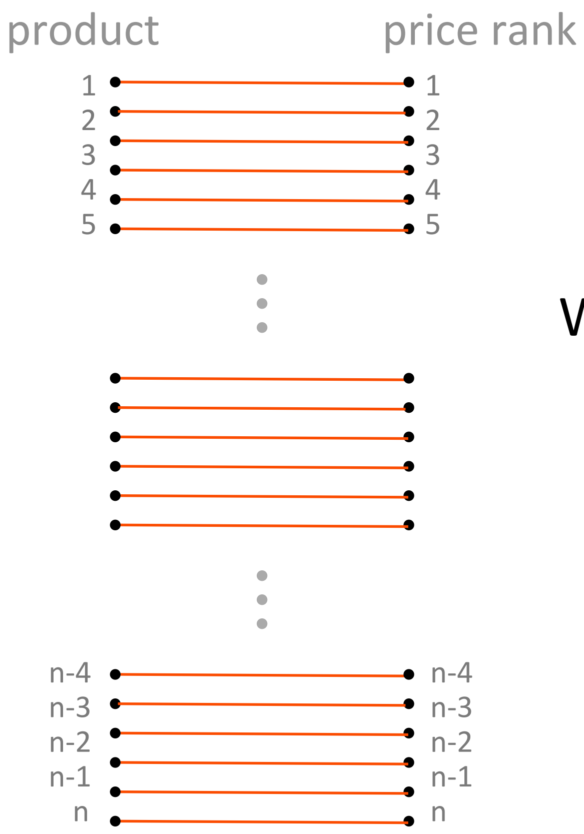
- computational: is there an efficient description for feasible solutions
- statistical: “how much” data is needed for point identification?

We answer both questions by restricting the domain of price vectors

We restrict price vectors to have "approximately" the same order



We restrict price vectors to have "approximately" the same order

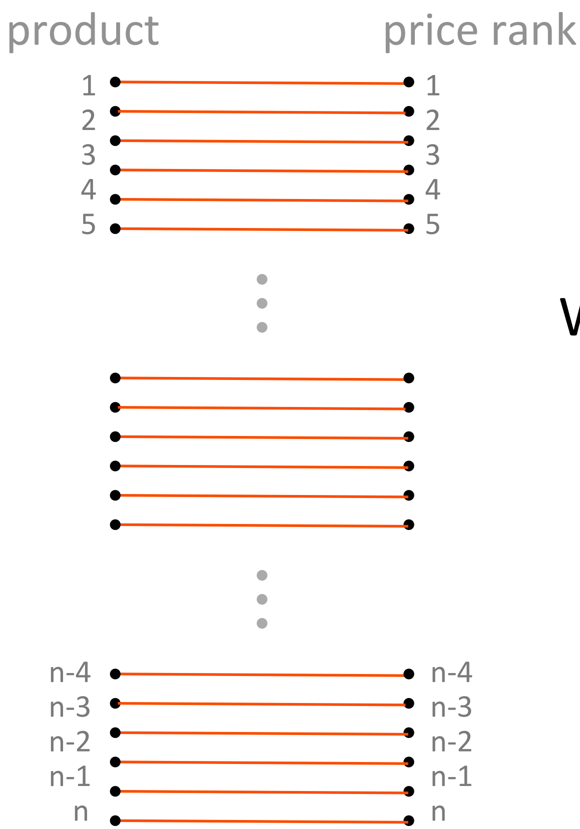


When all prices have the same (identity) ordering

- this is called 'price ladder' constraint
- this is known to result in tractable problems

[Rusmevichientong, Van Roy, Glynn], [Aggarwal, Feder, Motwani, Zhu]

We restrict price vectors to have "approximately" the same order



When all prices have the same (identity) ordering

- this is called 'price ladder' constraint
- this is known to result in tractable problems

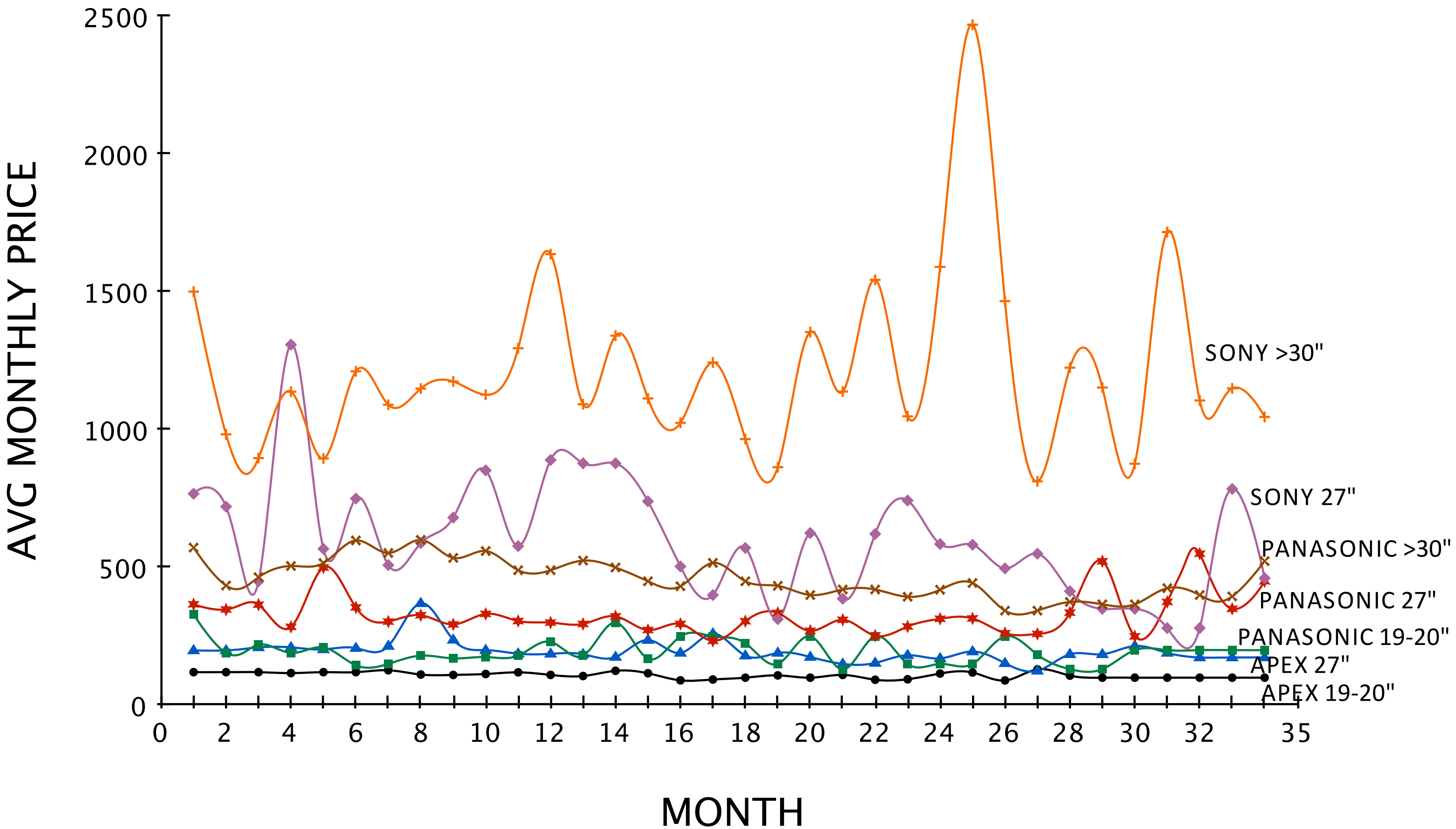
[Rusmevichientong, Van Roy, Glynn], [Aggarwal, Feder, Motwani, Zhu]

Only *n* possible consideration sets

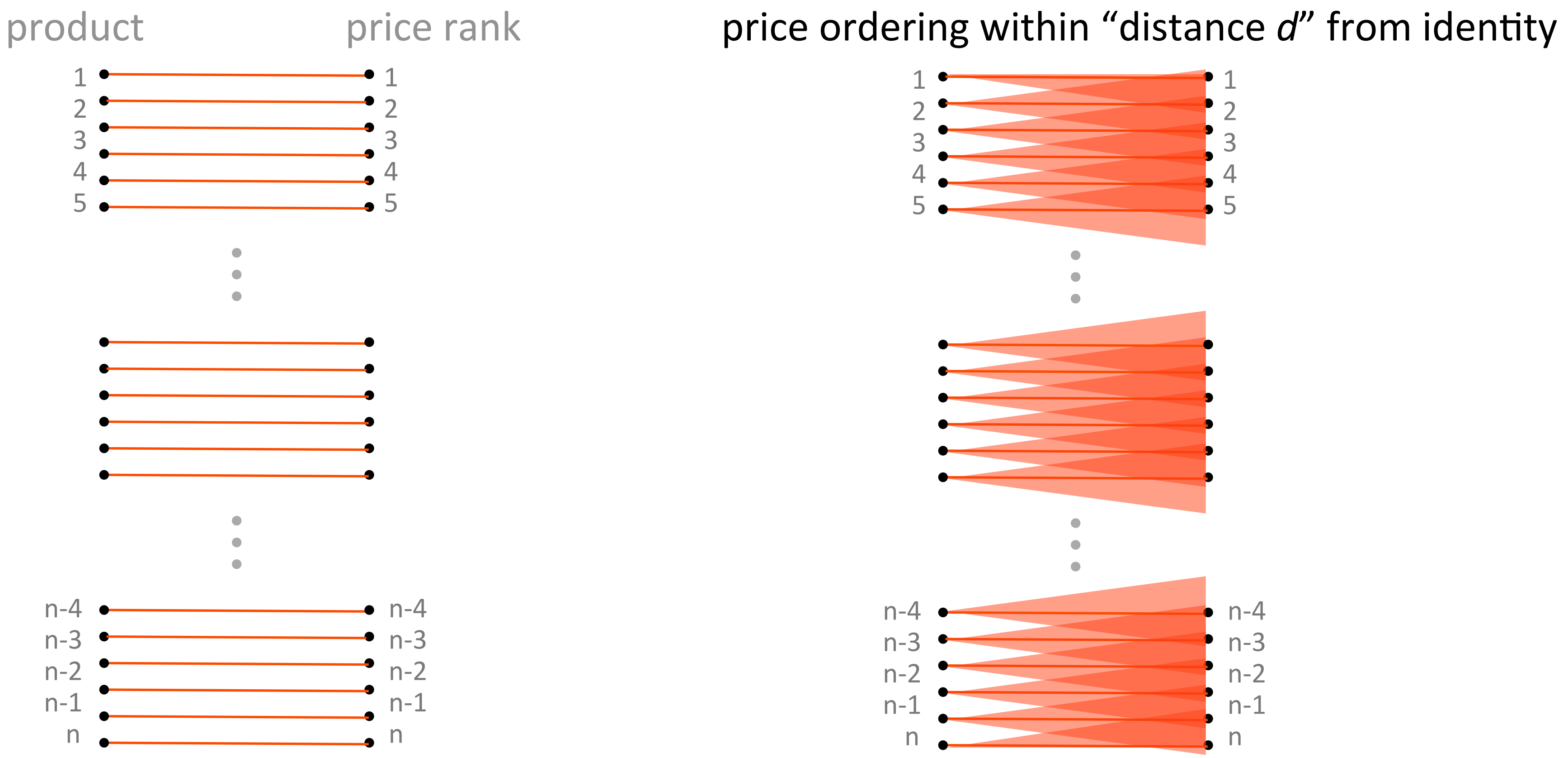
$$\{1\}, \{1,2\}, \ldots, \{1,2,\ldots,n\}$$

under the assumption $p_1 \leq p_2 \leq \cdots \leq p_n$

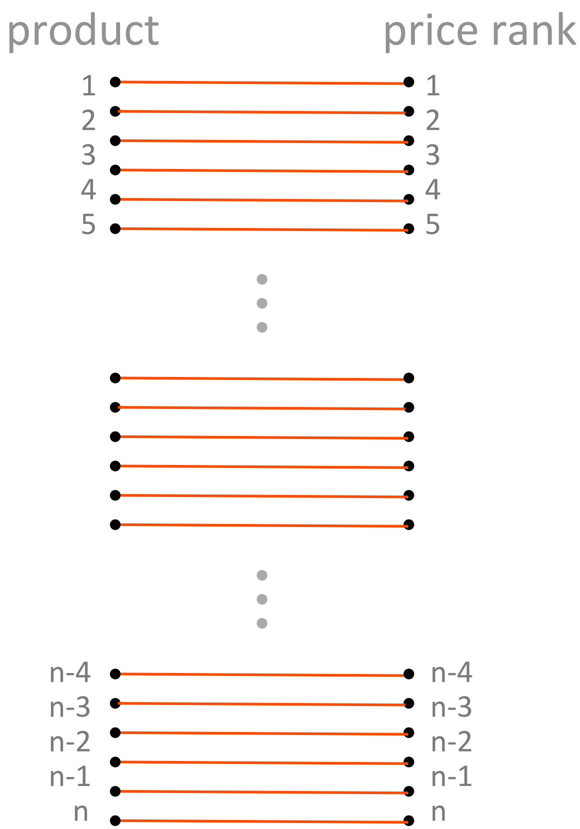
Monthly price variation of the seven products: price ordering changes but only by a little



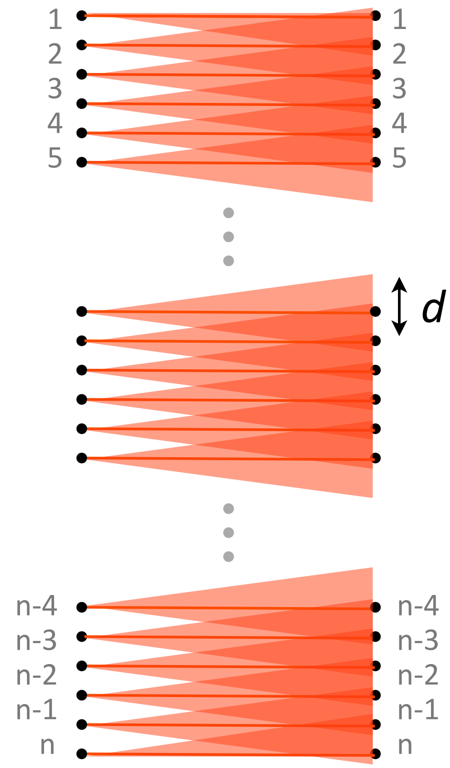
We restrict price vectors to have "approximately" the same order



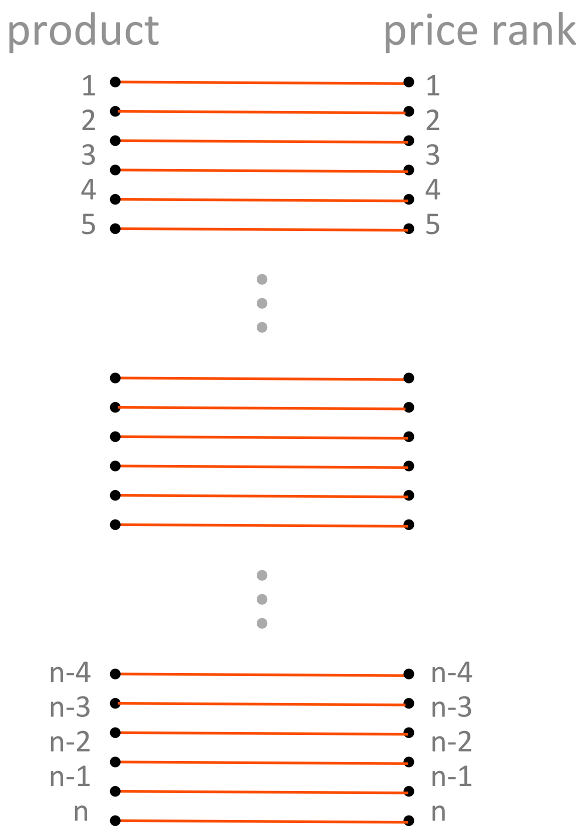
We restrict price vectors to have "approximately" the same order



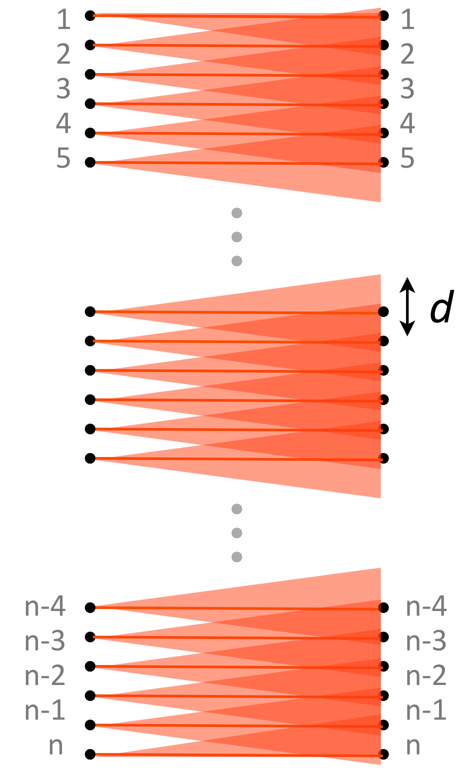
price ordering within "distance *d*" from identity



We restrict price vectors to have “approximately” the same order



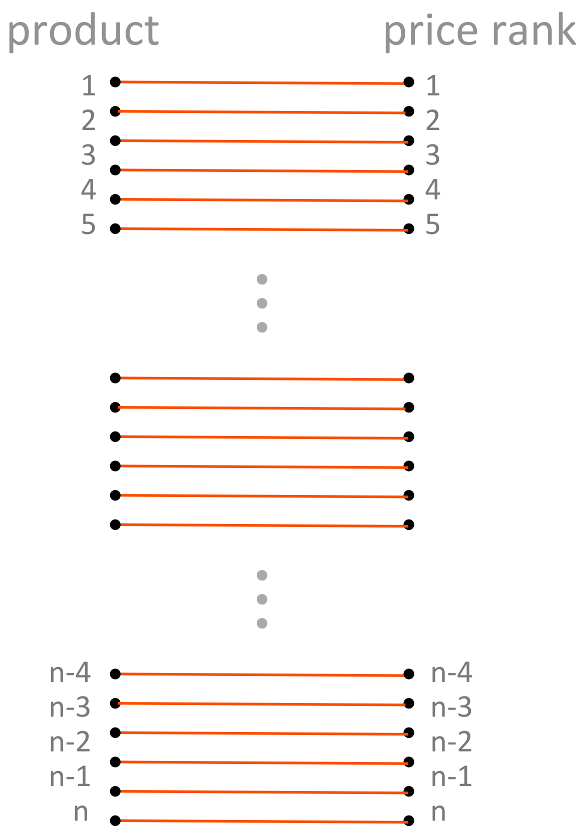
price ordering within “distance *d*” from identity



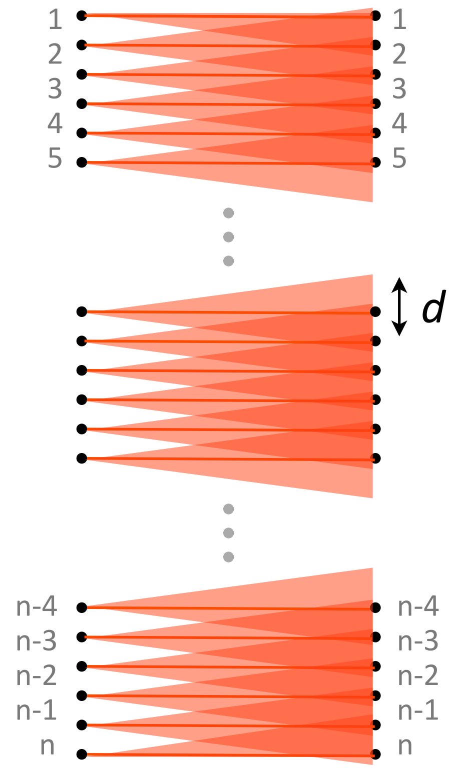
$\mathcal{P}_d$

class of price vectors with ordering within distance d from identity

We restrict price vectors to have "approximately" the same order



price ordering within "distance *d*" from identity



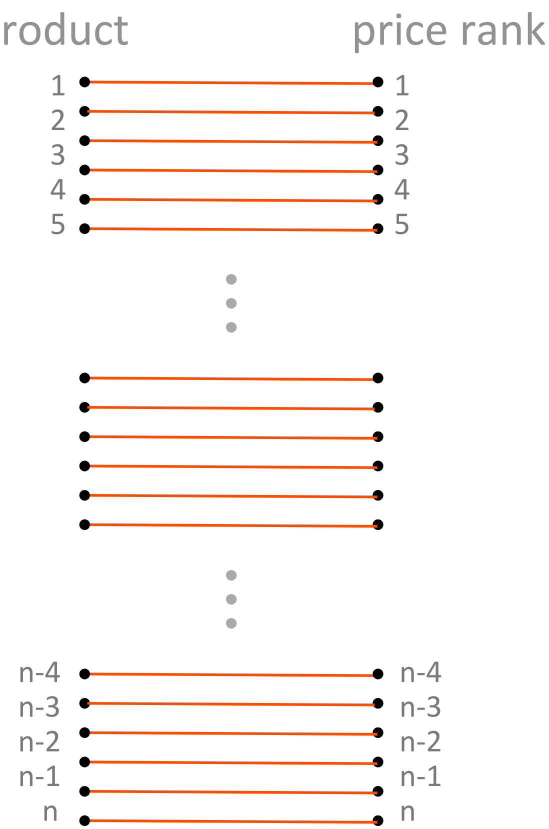
$\mathcal{P}_d$

class of price vectors with ordering within distance d from identity

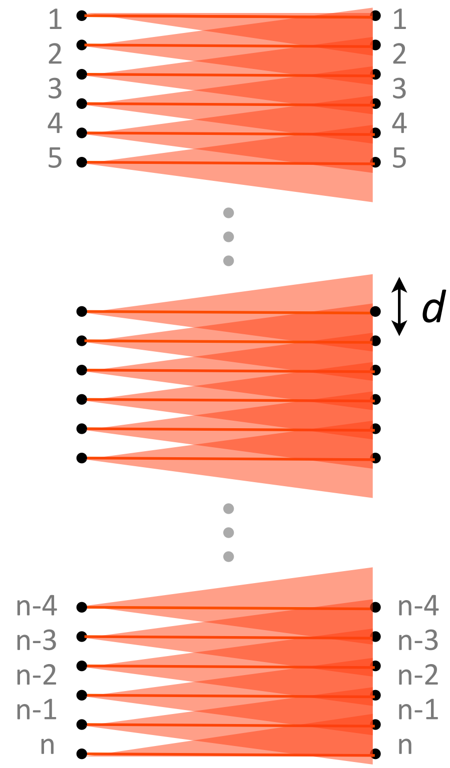
As *d* increases, we get an increasingly complex class of prices

$$\mathcal{P}_1 \subset \mathcal{P}_2 \subset \cdots \subset \mathcal{P}_n = \mathbb{R}^n_+$$

We restrict price vectors to have "approximately" the same order



price ordering within "distance *d*" from identity



$\mathcal{P}_d$

class of price vectors with ordering within distance d from identity

As *d* increases, we get an increasingly complex class of prices

$$\mathcal{P}_1 \subset \mathcal{P}_2 \subset \cdots \subset \mathcal{P}_n = \mathbb{R}^n_+$$

"small" *d* is reasonable in many applications: price orderings don't change too much

Computational complexity: for small *d* we have efficient description for dual polytope

$$\max_{\sigma} \sum_{p\in\mathcal{P}, j\in N\cup\{0\}} \alpha_{pj}\left(H_\theta(p_j) - H_\theta(p_{\sigma,j})\right)^+ - \sum_{j\in N\cup\{0\}} q_j\left(H_\theta(q_j) - H_{q_{\sigma,j}}\right)^+$$

Theorem: If all price vectors are restricted to have orderings "within distance *d*" then the above feasibility problem can be solved with computational complexity $O(n^4|\mathcal{P}|2^{4d})$

Computational complexity: for small *d* we have efficient description for dual polytope

$$\max_{\sigma} \sum_{p\in\mathcal{P}, j\in N\cup\{0\}} \alpha_{pj} \left(H_\theta(p_j) - H_\theta(p_{\sigma,j})\right)^+ - \sum_{j\in N\cup\{0\}} q_j \left(H_\theta(q_j) - H_{q_{\sigma,j}}\right)^+$$

Theorem: If all price vectors are restricted to have orderings "within distance *d*" then the above feasibility problem can be solved with computational complexity $O(n^4|\mathcal{P}|2^{4d})$

We cast the above problem as a DP and show it has polynomially (in *n*) large state space

Sample complexity: for small *d* , we need polynomially many price vectors for point identification

We only need to identify "relevant" choice probabilities

$$\Lambda(\mathcal{S}_d) = \{\mathbb{P}_\lambda(j|S) \colon j \in N \cup \{0\}, S \in \mathcal{S}_d\}$$

Sample complexity: for small *d* , we need polynomially many price vectors for point identification

We only need to identify “relevant” choice probabilities

$$\Lambda(\mathcal{S}_d) = \{\mathbb{P}_\lambda(j|S) \colon j \in N \cup \{0\}, S \in \mathcal{S}_d\}$$

class of consideration sets from restriction of price vectors to $\mathcal{P}_d$

Sample complexity: for small *d* , we need polynomially many price vectors for point identification

We only need to identify “relevant” choice probabilities

$$\Lambda(\mathcal{S}_d) = \{\mathbb{P}_\lambda(j|S) \colon j \in N \cup \{0\}, S \in \mathcal{S}_d\}$$

class of consideration sets from restriction of price vectors to $\mathcal{P}_d$

Theorem: There exist $O(n(2d \text{ choose } d))$ price vectors that allow for point identification of all relevant choice probabilities

Sample complexity: for small *d* , we need polynomially many price vectors for point identification

We only need to identify "relevant" choice probabilities

$$\Lambda(\mathcal{S}_d) = \{\mathbb{P}_\lambda(j|S) \colon j \in N \cup \{0\}, S \in \mathcal{S}_d\}$$

class of consideration sets from restriction of price vectors to $\mathcal{P}_d$

Theorem: There exist $O(n(2d \text{ choose } d))$ price vectors that allow for point identification of all relevant choice probabilities

We prove the above theorem by establishing the following about $\mathcal{P}_d$

number of price orderings $(d!)^{n/d}$

number of consideration sets $4^d n$

Pricing under two stage models: Outline of the talk

1. Model
   assumptions, properties, data for estimation
2. Numerical results
   real-world data on durable goods, revenue predictions
3. Model estimation and revenue prediction
   ML estimation, sample and computational complexity
4. Summary/Conclusions
   takeaway messages

Summary and conclusions

There is a need for data-driven pricing models that account for

- substitution due to price variation
- substitution due to non-availability

Our contributions: demand share prediction as a function of price

- new two-stage model of choice
- methods for estimating these models from transaction data
- statistical and computational guarantees on the methods

Main takeaway

- two-stage models are rich and flexible
- can be extended in a tractable way to capture various effects

Class of two-stage models is dense in Random Utility Max (RUM) model class

Class of two-stage models is dense in Random Utility Max (RUM) model class

RUM model

Class of two-stage models is dense in Random Utility Max (RUM) model class

RUM model

$u_1 - p_1$

Class of two-stage models is dense in Random Utility Max (RUM) model class

RUM model

$u_1 - p_1$

$u_2 - p_2$

Class of two-stage models is dense in Random Utility Max (RUM) model class

RUM model

$u_1 - p_1$

$u_2 - p_2$

⋮

$u_i - p_i$

⋮

$u_n - p_n$

Class of two-stage models is dense in Random Utility Max (RUM) model class

RUM model

$u_1 - p_1$

$u_2 - p_2$

⋮

$u_i - p_i$

⋮

$u_n - p_n$

choose item with max net utility

$i = \arg\max \ (u_j - p_j)$

Class of two-stage models is dense in Random Utility Max (RUM) model class

RUM model	Two-stage model
$u_1 - p_1$	u_1
$u_2 - p_2$	u_2
⋮	⋮
$u_i - p_i$	u_i
⋮	⋮
$u_n - p_n$	u_n

choose item with max net utility

$i = \arg\max (u_j - p_j)$

Class of two-stage models is dense in Random Utility Max (RUM) model class

RUM model	Two-stage model
$u_1 - p_1$	u_1
$u_2 - p_2$	u_2
⋮	⋮
$u_i - p_i$	u_i
⋮	⋮
$u_n - p_n$	u_n

price > WTP

choose item with max net utility

$i = \arg\max (u_j - p_j)$

Class of two-stage models is dense in Random Utility Max (RUM) model class

RUM model

$u_1 - p_1$

$u_2 - p_2$

⋮

$u_i - p_i$

⋮

$u_n - p_n$

choose item with max net utility

$i = \arg\max \, (u_j - p_j)$

Two-stage model

u_1, u_2, ⋮ — price > WTP

u_i, ⋮, u_n — price < WTP

Class of two-stage models is dense in Random Utility Max (RUM) model class

RUM model

$u_1 - p_1$

$u_2 - p_2$

⋮

$u_i - p_i$

⋮

$u_n - p_n$

choose item with max net utility

$i = \arg\max\ (u_j - p_j)$

Two-stage model

u_1

u_2

⋮

} price > WTP

u_i

⋮

u_n

} price < WTP

Class of two-stage models is dense in Random Utility Max (RUM) model class

RUM model

$u_1 - p_1$

$u_2 - p_2$

⋮

$u_i - p_i$

⋮

$u_n - p_n$

choose item with max net utility

$i = \arg\max \, (u_j - p_j)$

Two-stage model

u_1

u_2

⋮

price > WTP

u_i

⋮

u_n

price < WTP

choose item with max utility and price < WTP

Class of two-stage models is dense in Random Utility Max (RUM) model class

RUM model

$u_1 - p_1$

$u_2 - p_2$

$\vdots$

$u_i - p_i$

$\vdots$

$u_n - p_n$

choose item with max net utility

$i = \arg\max\ (u_j - p_j)$

Two-stage model

u_1

u_2

$\vdots$

price > WTP

u_i

$\vdots$

u_n

price < WTP

choose item with max utility and price < WTP

Theorem: For any RUM model, there exists a joint distribution of preference lists and WTP such that

Prob (choosing i under RUM model) = Prob (choosing i under two-stage model)

Class of two-stage models is dense in Random Utility Max (RUM) model class

RUM model

$u_1 - p_1$

$u_2 - p_2$

⋮

$u_i - p_i$

⋮

$u_n - p_n$

choose item with max net utility

$i = \arg\max (u_j - p_j)$

Two-stage model

u_1, u_2, ⋮ — price > WTP

u_i, ⋮, u_n — price < WTP

choose item with max utility and price < WTP

Theorem: For any RUM model, there exists a joint distribution of preference lists and WTP such that

Prob (choosing i under RUM model) = Prob (choosing i under two-stage model)

Class of two-stage models is dense in Random Utility Max (RUM) model class

Class of two-stage models is dense in Random Utility Max (RUM) model class

Joint distribution of preference lists and WTP

Class of two-stage models is dense in Random Utility Max (RUM) model class

Joint distribution of preference lists and WTP

Prob(pref list, WTP) = Prob(pref list) x Prob(WTP | pref list)

Class of two-stage models is dense in Random Utility Max (RUM) model class

Joint distribution of preference lists and WTP

Prob(pref list, WTP) = Prob(pref list) x Prob(WTP | pref list)

obtained from *any* choice model

Class of two-stage models is dense in Random Utility Max (RUM) model class

Joint distribution of preference lists and WTP

too many degrees of freedom; need restriction!

Prob(pref list, WTP) = Prob(pref list) x Prob(WTP | pref list)

obtained from *any* choice model

Class of two-stage models is dense in Random Utility Max (RUM) model class

Joint distribution of preference lists and WTP

too many degrees of freedom; need restriction!

Prob(pref list, WTP) = Prob(pref list) x Prob(WTP | pref list)

obtained from *any* choice model

Restrict dependence of WTP to one/few central preference list(s)

Class of two-stage models is dense in Random Utility Max (RUM) model class

Joint distribution of preference lists and WTP

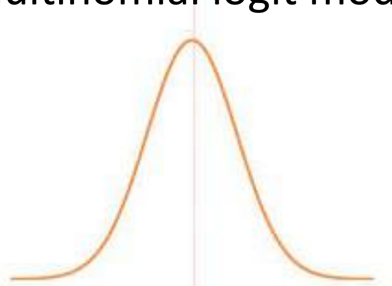
too many degrees of freedom; need restriction!

Prob(pref list, WTP) = Prob(pref list) x Prob(WTP | pref list)

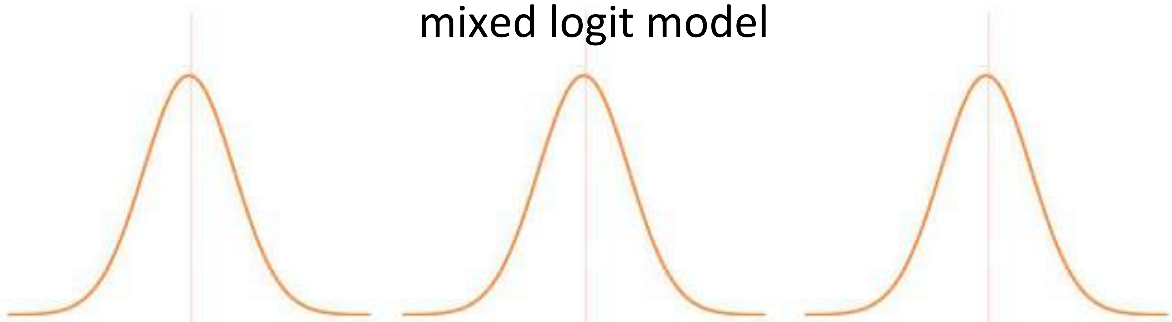
obtained from *any* choice model

Restrict dependence of WTP to one/few central preference list(s)

multinomial logit model



mixed logit model



Practical choice models concentrate around one/few central preference list

Hence, restriction of WTP distribution provides good approximations!