# Consumer Quality Search and Platform Algorithm Design: A Structural Apporach

Working in Progress

Guo Zhang [1]

This Version: November 25, 2017

[1]WISE, Xiamen University. Email: zhangguo@stu.xmu.edu.cn. All codes can be found here: https://github.com/xmucpp/online_quality_search/blob/master/consumer_quality_search.ipynb

# Contents

# Motivation

- A design or improvement of algorithms and mechanisms can produce revenue of millions of dollars for online platforms.
- An efficient and well-organized method for a certain situation is valuable for business leaders and managers to make data-driven decisions.
- The underlying processes that makes them work have not been clarified with structural frameworks for many cases.

# Theme

- Structurally Modeling consumer search based on prices and historical sales, dynamic pricing of sellers and market equilibrium on the environment of ranking algorithms based on prices and historical sales on the search engine of Tmall[2]with consideration set approach.
- Looking for the optimal weighting of prices and historical sales for ranking algorithms to maximize consumer/seller/total welfare with the structural-form model.

[2]Example:
https://list.tmall.com/search_product.htm?q=%BD%B4%D3%CD

# Overview

- Background of online consumer search and seller pricing
- Model of consumer search, seller pricing and market equilibrium
- Estimation of model parameters (doing)
- Simulation to find optimal weighting of ranking algorithms (to do)

# Literatures Review

- Search friction and price dispersion
- Limited consumer search
- Search obfuscation
- Reputation Mechanism

# Literatures Review - Framework

- Levin et.al. (2014): structuring consumer search and platform design for online retailers
- Fan et.al. (2013): cross-period return to reputation and reputation management for sellers

# Literatures Review - Search Friction and Price Dispersion

- Baye et.al. (2014): price dispersion is persistent and is greater in the market with smaller number of products on online markets
    - Reinganum (1979): positive search costs
    - Spulber (1995): private information
    - Stahl (1989): heterogeneous consumers with different search costs

# Literatures Review - Limited Consumer Search

- Goeree (2008): limited consumer information and advertising in PC industry with consideration set approach
- Levin et.al. (2014): search engine instead

## Literatures Review - Search Obfusaction

- Ellison et.al. (2009): search obfuscation on the environment of search engine

# The Organization of Markets with Online Platforms

- Overcome the limitation of geographic and social barriers
- Increase the search costs of users by crowding a large amount of information
  - Platforms usually desgin algorithms to display products with higher quality and higher relevance for users in practice.

# Platform Algorithm Design

- Ranking: centralized, often all the results
- Recommendation: personalized, often a limited amount of results

# Different Approaches of Ranking Designs

- Sort by sales
- Sort by prices
- Sort by date
- Best Match (mixed methods)

## Dimensions of Consumer Online Search

- Step 1: search relevant products
  - **User query** or **keyword search**
  - Advertising (i.e. home page, social media, video platforms)
  - Recommendation (i.e. home page, item page)
- Step 2: search products with attractive **prices** and **qualities** (which we will focus in this paper)

# Consideration Set Approach

- Ranking algorithms provide consideration set from all available products
- Consumers make decision in the consideration set

# Stategies and Decisions of Sellers

- Make strategies on prices, quality, relevance, reputation, etc.
- Search obfuscation

# Consumer Utility

- Utility of consumer j for product i at period t:

$$u_{ijt} = \alpha_0 + \alpha_p p_{jt} + \alpha_q ln(q_{j,t-1}) + \varepsilon_{ijt}$$

- Choice probability given consideration set by logit discrete choice model:

$$P(j|j \in L_{it}) = \frac{\exp(\alpha_0 + \alpha_1 p_{jt} + \alpha_q ln(q_{j,t-1}))}{\sum_{k \in L} \exp(\alpha_0 + \alpha_p p_{kt} + \alpha_q ln(q_{k,t-1}))}$$

# Consideration Set I

- Sampling weight for consumer j at period t with Wallenius' non-central hypergeometric distribution:

$$
\begin{aligned}
w_{jt} = \exp[-\gamma_p(\frac{p_{jt} - \min_{k \in J_t}(p_{kt})}{std_{k \in J_t}}) \\
+ \gamma_q(\frac{q_{jt} - \min_{k \in J_t}(ln(q_{kt}))}{std_{k \in J_t}(ln(q_{kt}))}) \\
]
\end{aligned}
$$

# Consideration Set II

- The probability of product $j$ in the consideration set at period t:

$$P(j \in L_t | j \in J_t) = 1 - \Pi_{k=1}^{l}(1 - p_{kt})$$

where

$$P(j_{lt} \in L_t | j_{1t} \in J_t) = \frac{w_{j_l t}}{\sum_{k \in J_t}(w_{kt})}$$

# Market Demand

- Choice probability of product j at period t:

$$D(p_{jt}) = \sum_{L_{it}:L_{it} \in J_t} P(j|j \in L_{it}) P(j \in L_{it}|j \in J_t)$$

# Seller Pricing and Equilibrium

- Profit of seller/product j in each period:

$$\pi_{jt} = (p_{jt} - c_j) D_{jt}(p_{jt})$$

- One-Period:

$$\max_{p_j} \pi_{jt}$$

- Muti-Period:

$$\max_{p_{jt}} \Pi_j = \max_{p_j t} \sum_{t=0}^{\infty} \beta^t \pi_{jt}$$

# Data Source I

- Source: results given keywords on the product search engine of Tmall collected by China's Prices Project[3]
- Time Period: daily data from May 2016 - Present
- Variables: price, monthly sales, comment number, name of goods, URL of product page, name of shop, name of shop page, rank method, scrape time, category information, page number, rank order, etc.
- Observations: more than 1 billion

[3] China's Prices Project is an academic initiative and technology firm working on economic and business big data found by Guo Zhang. A data warehouse of price data from the search page of Tmall and JD is constructing with big data technology.

# Data Sample

- First 3 pages(117 results) on search result pages displayed on APPs of Tmall and JD from Jan 2017 to now
- One day (2017-04-01) of data with keyword "apple" is selected in this version

# Summarized Statistics

| **Listings Data** | **Values** |
|---|---|
| Number of Listings Number of Sellers | 117 |
| % of Sellers with $> 1$ | 25.64% |
| Listing Mean List Price | 57.83 |
| Median List Price | 46.8 |
| Standard Deviation of List Price | 44.86 |

Table: : Summarized Statistics

# Correlations

| func | dep | indep | corr | p-value |
|---|---|---|---|---|
| pearsonr | price | rank | 0.048838 | 0.601044 |
| pearsonr | sales_last | rank | -0.418686 | 0.000003*** |
| pearsonr | comments_last | rank | -0.385328 | 0.000018*** |
| spearmanr | price | rank | 0.064618 | 0.488828 |
| spearmanr | sales_last | rank | -0.571240 | 0.000000*** |
| spearmanr | comments_last | rank | -0.454517 | 0.000000*** |
| kendalltau | price | rank | 0.040969 | 0.515410 |
| kendalltau | sales_last | rank | -0.416305 | 0.000000*** |
| kendalltau | comments_last | rank | -0.326366 | 0.000000*** |

Table: : Correlations

# Steps of Estimation

1. Demand parameters: parameters of consumer utility $\alpha$
2. Platform parameters: weighting of the ranking algorithm $\gamma$
3. Seller parameters: marginal costs of products $c$

## Note for This Version

- Accumulated comment number is also considered as a proxy of product quality, which will be excluded for the poor performance in the describe statistics and model estimation.
- The scale of sales and comment is set as log form to make the estimation work.

# Estimation of Demand Parameters

- Observed consideration set defined as the first 10 products by ranking algorithms
- MLE directly by mutinomial logit discrete choice model

$$u_{ij} = 3.29718207^{-6} - 1.31976519^{-2} price_t + 8.79789875^{-1} log(sales_{t-1}) + -8.01401981^{-2} log(comment_{t-1})$$

# Estimation of Platform Parameters

- All available products defined as all scraped observations
- Maximum simulated likelihood estimation

$$
\begin{aligned}
w_{jt} = \exp[&-1.20910197(\frac{p_{jt} - \min_{k \in J_t}(p_{kt})}{std_{k \in J_t}}) \\
&+ 0.39808008(\frac{q_{jt} - \min_{k \in J_t}(log(q_{kt}))}{std_{k \in J_t}(log(q_{kt}))}) \\
&+ 0.59927273(\frac{m_{jt} - \min_{k \in J_t}(log(m_{kt}))}{std_{k \in J_t}(log(m_{kt}))})]
\end{aligned}
$$

# Estimation of Seller Parameters

- Solving the optimization of maximizing profits of seller:

$$c_j = p_j + \frac{D_{jt}(p_j)}{D'_{jt}}$$

# Simulation Methods

- Calculate the total surplus as a function of algorithm parameters given other estimated parameters
- Change $\gamma$, find a solution that maximize total surplus

# Next Step

- Extend the model from static to dynamic
- Improve the sample data
- Relax the assumption that qualities of goods are given / Allow entry and exit of sellers

# Further Work

- Competition for product with horizontal differentation on one search results
- Obfuscation on search keywords by sellers
- Environment of recommender system

# Contents