Q-Learning in Auctions

Pranjal Rawat

May 15, 2023

Georgetown University

Introduction

Can Q-learning bots learn to suppress bids in auctions? We consider the following:

- Reward: First price vs Second Price
- Feedback: Instant vs Delayed (English, Dutch)
- Algorithms: Deep Q-learning

Constructing a real world scenarios: data from display advertising.

Repeated Static Auctions

Stage Game

Two bidders compete in an auction:

- Bid: $b \in \{0, 0.1, 0.2...1.0\}$
- Common Valuation: $v = 1$
- Profit:
  - First Price: $v - b'$ if $b \geq b'$, $0.5(v - b')$ if $b = b'$, else 0.
  - Second Price: $v - b$ if $b \geq b'$, $0.5(v - b)$ if $b = b'$, else 0.
- State: s is opponents' past bid b'_{t-1}
- Policy: $\sigma(s)$ is a strategy of choosing bids given states
- Discount factor: γ

Rewards

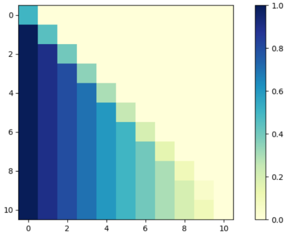


Figure 1: Second Price Rewards

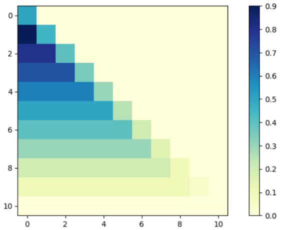


Figure 2: First Price Rewards

Dynamics

For policy σ, every state s has a **expected return**, assuming the opponents plays by σ'.

$$V_{\sigma,\sigma'}(s) = \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t \pi(b_t, b'_t) \mid s_0 = s, \sigma, \sigma'\right]$$

$$Q_{\sigma,\sigma'}(s, b) = \pi(b, b') + \mathbb{E}\left[\sum_{t=1}^{\infty} \gamma^t \pi(b_t, b'_t) \mid \sigma, \sigma'\right]$$

Q-Learning

"Experience based equilibrium" is $(\bar{\sigma}, \bar{\sigma'})$ arrived at by an iterative process. One such process is Multi-agent Q-learning:

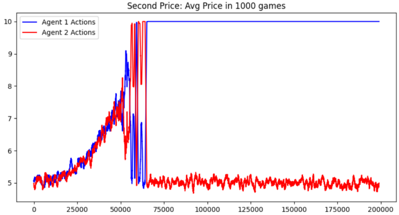
- Guess $Q_0(s, b)$
- at t, do:
  - observe s_t
  - take action b_t from **exploratory strategy**
  - collect reward π_t
  - observe transition s_{t+1}
  - update Q at point (s_t, b_t):

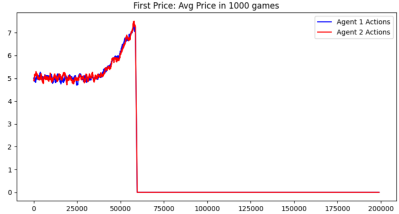
$$Q_{t+1}(s_t, b_t) = (1-\alpha)Q_t(s_t, b_t) + \alpha(\pi_t + \gamma max_{b'} Q_t(s_{t+1}, b'))$$

As bots' Q tables stabilize, $\bar{\sigma}(s) = \text{argmax}_p Q(s, b)$ is going to be optimal against $\bar{\sigma'}(s)$ and both will be best responses to each other.

Simulation 1

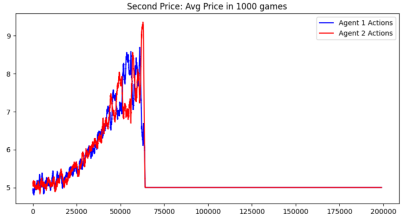
No state or discounting $\gamma = 0$, $\alpha = 0.5$. First price auction leads to collusion, while second price auction remains robust.

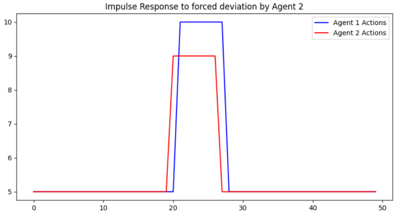




Simulation 2

State b'_{t-1} and discounting $\gamma = 95$, $\alpha = 0.5$. With help of a memory, second price auction shows collusion.





Enhancements

Deep Q-Learning Networks (DQN)

In practice firms use Deep Networks to represent Q as a function.

- Initialize: $Q(s, a; \theta)$.
- At t
  - Choose b_t from exploratory strategy
  - Observe s_{t+1}, π_t
  - Draw a random sample $(\bar{s}, \bar{s}', \bar{r}, \bar{b})$ from history
  - Bellman Error: $e = Q(\bar{s}, \bar{b}) - (\bar{r} + \gamma E[max_{b'} Q(\bar{s}', b')])$.
  - Loss: $L(\theta) = e'e$.
  - Update Parameters: $\theta = \theta - \delta L'(\theta)$.
  - End when loss does not decrease anymore

English/eBay Auction

Two bidders compete in an auction that lasts T periods:

- Bid: $b_t \in \{0, 0.1, 0.2 ... 1.0\}$
- State: s is opponents' largest bid b'_{t-1} and rounds left $T - t$
- First price auction with delayed rewards
- Since reward comes at end, have to remember path and attribute correctly

Research Framework

Research Question

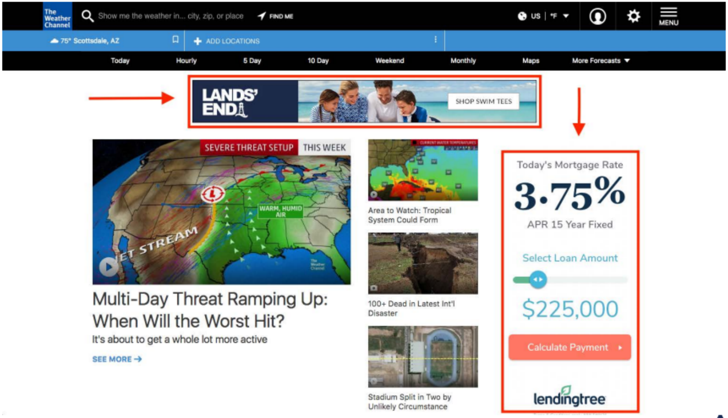
Can reinforcement learning algorithms reduce competition? What market design elements can restore competition?

Gaps:

1. All papers have considered highly idealized settings so far.
2. Firm will first train bots to behave in different scenarios before deployment

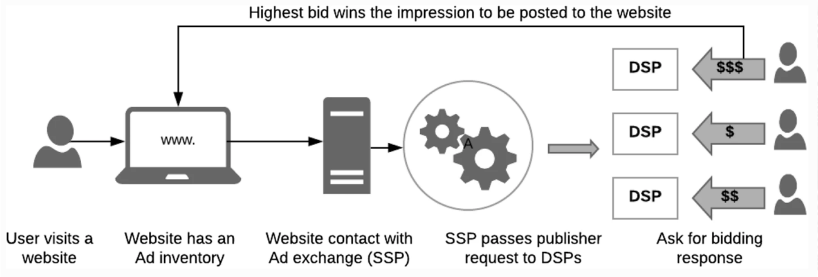
Display Advertising

Two types of digital advertising - sponsered search and display ads.



Real Time Auctions

Modern advertising uses high frequency auctions to sell banner ads to advertisers.



Players

- Customer: Click on ads if they could lead to purchase, and then purchase.
- Ad Exchange/Demand Side Platform (e.g. iPinYou): wants to maximize revenue and/or thicken market
- Advertiser: Estimate “value” of a bid-request, exhaust budget over time to maximise CTR or conversion.

Data

iPinYou DSP conducted a competition in 2014, data included

- Advertiser metadata: data on advertiser industry, group and sub-group of product
- Bidding logs: each bid from each advertiser on each ad-impression. Contains some information about consumer which made impression.
- Impression, click, conversion logs for winning bidder's ad

Q-learning in Display Advertising

- Use the data to approximate real world settings.
- Each bot needs a demand model - predicts expected CTR/Conversion given customer demographics and ad details.
- Each bot has to have a "bidding landscape function" which predicts winning bid using historical data.
- Each bot has a different budget that replenishes over time.
- Deep Q-learning used to decide bid amount.

Analysis

- One firm takes lead and impliments a RL bot - using historical data to train it - and deploys it on holdout data. This measures the unilateral adoption gain.
- All firms impliment RL bots - using historical data to train it - and deploy on holdout data. This measures the post-adoption gain.
- Market design elements - information revelation?
- How do we connect this to a structural model of auctions?