CS 6840: Algorithmic Game Theory	**Spring 2017**

Lecture 31: April 17

Lecturer: Éva Tardos *Scribe: Anna Yesypenko*

31.1 Generalized Second Price Auctions

We review some terminology covered in the last lecture for advertisers bidding on slots to place their ads.

- Slot i has click rate α_i. Ad j has relevance γ_j. Therefore if ad j has slot i, probability of a click is $\alpha_i \gamma_j$.
- Bids $b_1, \dots b_n$, where we sort bids such that $b_1\gamma_1 \geq \cdots \geq b_n\gamma_n$.

Advertiser j has valuation v_j and pays $p_j(b_j)$ per-click, depending on the bid. If he gets slot i, then advertiser j has utility $(v_j - p_j(b_j))\alpha_i\gamma_j$.

Let $x(b_j)$ be the click probability of person j given bid b_j. We can plot $x(b_j)$ against possible bids b_j, where click probability for the top slot is $\gamma_j\alpha_1$.

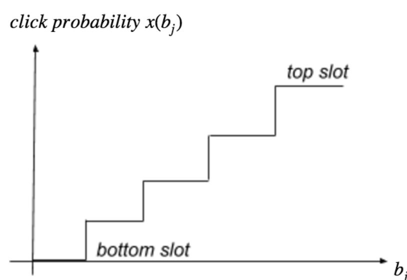

Figure 31.1: Step Function $x(b)$

Let $P(b_j) = p_j(b_j)\alpha_i\gamma_j$, which is the total payment on bid b_j. We can also plot $P(b_j)$ against possible bids b_j. The plot is analogous to the one pictured above, with step-function behavior.

If person j getting slot i is pure strategy Nash, then j prefers slot i to every other slot. In the last lecture, we tried to infer the valuations v_j. Because person j doesn't prefer a slot $k > i$, we have a lower bound on v_j, and because person j similarly doesn't prefer a slot $k < i$, we have an upper bound on v_j. We can infer a range for the values v_j in this manner.

31.2 Bayesian Nash Assumption

Instead of playing a full-information game, we make the assumption that the game is Bayesian to better reflect real-world situations. We still have the same notation as in the full-info game. When person j submits bid b_j, the parameters α_i for click-rate are known and fixed, but his quality factor γ_j (and other people's quality factors) are Bayesian and drawn from a known distribution. This model reflects what occurs in real-life situations because Google/Microsoft recompute compute quality factors each ad, and then report values in a Bayesian sense because they are hard to know exactly.

31.2.1 Empirical Fact

When we look at real data, the plots $x(b)$ and $P(b)$ are not stepwise as we drew them for the full-information game. In fact, the plots are smooth. Therefore, in our analysis, we can assume the functions $x(b)$ and $P(b)$ are monotone, nondecreasing, and differentiable.

Since the game is Bayesian-Nash, we can find bid b for player j that maximizes utility

$$\max_b v_j x(b) - P(b).$$

Since $x(b)$ and $P(b)$ are differentiable, we can take the derivative and solve for the maximum:

$$v_j x'(b) = P'(b) \qquad \Rightarrow \qquad v_j = \frac{P'(b)}{x'(b)}.$$

But should we believe that we can infer values this way? Every few seconds the inferred valuation will change because it depends on the bid and opportunities that happened that second. So what went wrong?

- If $x'(b)$ is small because the plot $x(b)$ is flat, then value v_j inferred is uncertain.
- The Bayesian Nash assumption is strong. Bidders do not necessarily perfectly optimize to maximize their utility.

31.3 Small-Regret Learning Assumption

Maybe we cannot think about auctions as a Bayesian Nash game, but perhaps the bidders are learning. Instead of picking a single bid, we take a sequence of bids and assume that over the sequence, bidders have little or no regret. Assume b_j^t is this sequence. Let X^0 be number of clicks won over the interval and P^0 be the total person j paid. Then the utility of person j over the interval is $v_j X^0 - P^0$. We would like to infer v_j.

Let $X(b)$ be the expected number of clicks over interval with fixed bid b. Let $P(b)$ be the expected cost over the interval with fixed bid b. Both of these functions can be computed from data, assuming α's and γ's are given.

Since we assume small regret,

$$v_j X^0 - P^0 \geq v_j X(b) - P(b) - Reg \qquad \Rightarrow \qquad v_j(X^0 - X(b)) + Reg \geq P^0 - P(b).$$

Whether this inequality gives a lower or upper bound depends on whether $X_0 - X(b)$ is positive or negative. We have two unknowns v_j and Reg and produce the following plot. If $X(b) > X^0$, we have line (1) as a contraint, and if $X(b) < X^0$, we have line (2) as a constraint. For real data, the curve is smooth, as shown. The possible (v_j, Reg) points are in the region to the right of the curve.

However, with this plot, no matter the value v_j, the regret can never be 0. This means that the bidders are not learning. Ideally, we want to translate the curve to the left such that it crosses the y-axis. This would

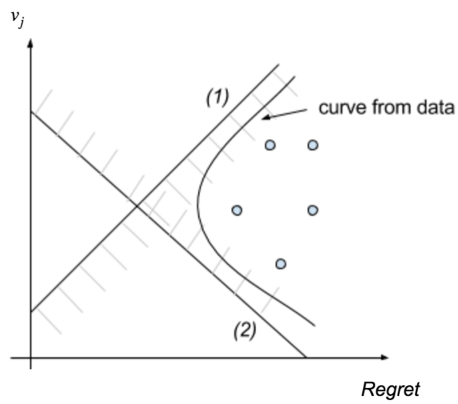

Figure 31.2: Plot of possible (v_j, Reg) pairs

allow for players to have 0 or negative regret (in the negative case, the player did something better than a fixed alternative).

So if there are values for which the player has low regret, which is most likely her value? In the paper "Econometrics for Learning Agents" (2015), Tardos et al. pick v_j such that

$$v_j(X^0 - X(b) \geq (1-\epsilon)(P^0 - P(b))$$

for smallest $\epsilon > 0$ possible. Instead of having an additive error in the inequality, the paper uses multiplicative error.

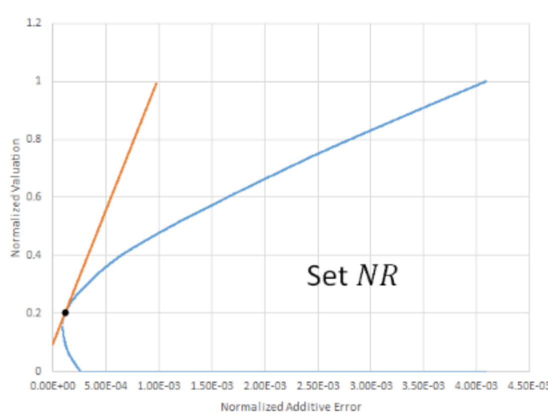

Figure 31.3: Paper result - inferring v_j while minimizing multiplicative error

Among the possible v_j to choose, the paper chooses the point indicated by the tangent line, inferring higher v_j to minimize multiplicative error. Inferring v_j is a difficult problem, and the inferred value cannot be verified against real advertiser data.

An alternate, and possible better option would be to say that we suggest the value v_j that allows the smallest regret Reg in the possible area. The multiplicative rule above appears to favor larger values of v_j, while this does not.

For an interesting paper on experimentally evaluating these ideas, you may want to look at the paper An

Experimental Evaluation of Regret-Based Econometrics by Noam Nisan, Gali Noti, in WWW'17, see also at `https://arxiv.org/abs/1605.03838`.