When Do Prediction Market Makers Make Profits?

Po-An Chen*, Yiling Chen†, Chi-Jen Lu‡, Chuang-Chieh Lin§

IIM, National Yang-Ming Chiao-Tung University*
SEAS, Harvard†
IIS, Academia Sinica‡
CSIE, Tamkang University§

Speaker: Chuang-Chieh Lin (Joseph)

AAAC'21, Section A2

23 October 2021

Outline

- Introduction: Prediction Markets
- Preliminaries
  - Designing the Cost Function via Conjugate Duality.
  - Equivalence between Market Making and Online Learning
  - Predictable Sequences
- Profitable Market Making via Online Learning
  - Optimistic Lazy-Update Online Mirror Descents with Predictable Sequences
  - Expert Setting: Modified Double Multiplicative Updates with Predictable Sequences
- Future Work

- A prediction market can be viewed as
  - information aggregation of the traders
  - traders' collective estimate of how likely the associated event will occur.
- Prediction markets focus on information aggregation.

A. Mann: The power of prediction markets. *Nature* 538, 308–310 (2016).

Prediction Markets

- E.g., Arrow-Debreu securities:
  - Certain security pays $1 if a particular state of the world is reached.
  - A risk-neutral trader believing that the prob. of event occurring is p
  - Buying/selling this security if the price is below/above p
  - E.g., insurance contracts, options, futures, and many other financial derivatives
- The information of the trader can be capitalized.

(Automated) Market Makers

- An automated market maker: an institution *adaptively setting prices* for each security and accepting trades at these prices all the time.
- A common goal: *the worst-case loss is minimized*.
- A **complete** securities market: offering a finite number of linearly independent securities over a set $\mathcal{O}$ of mutually exclusive and exhaustive outcomes [Arrow 1964].
  - A trader may bet on any combination of the securities in a complete securities market.
- **Our goal**: profit guarantee in automated market making
  - in terms of negative regrets under appropriate *patterns of trade sequences*.

Market making Using Cost Function + Conjugate Duality (1/2)

[Abernethy, Chen & Vaughan TEAC 2012].

- Outcome space $\mathcal{O}$.
- N securities by a payoff function $\rho : \mathcal{O} \to \mathbb{R}^N_{\geq 0}$.
  - $\rho_i : (\mathbf{o}) \in \{0, 1\}$ for $\mathbf{o} \in \mathbf{O}$ and security i.
- Convex compact price space $\Pi = \mathcal{H}(\rho(\mathcal{O}))$.

Market making Using Cost Function + Conjugate Duality (1/2)

[Abernethy, Chen & Vaughan TEAC 2012].

- Outcome space $\mathcal{O}$.
- N securities by a payoff function $\rho : \mathcal{O} \to \mathbb{R}^N_{\geq 0}$.
  - $\rho_i : (\mathbf{o}) \in \{0, 1\}$ for $\mathbf{o} \in \mathbf{O}$ and security i.
- Convex compact price space $\Pi = \mathcal{H}(\rho(\mathcal{O}))$.
- Cost function $C : \mathbb{R}^N \mapsto \mathbb{R}$ with

$$C(\mathbf{q}) = \max_{\mathbf{x} \in \Pi} \mathbf{x} \cdot \mathbf{q} - R(\mathbf{x})$$

  - A convex R with $\text{relint}(\Pi) \subseteq dom(R)$.
  - A quantity vector $\mathbf{q}_t = \mathbf{q}_{t-1} + \mathbf{p}_t$ for security bundle purchases $\mathbf{p}_t$.

Market making Using Cost Function + Conjugate Duality (1/2)

[Abernethy, Chen & Vaughan TEAC 2012].

- Outcome space $\mathcal{O}$.
- N securities by a payoff function $\rho : \mathcal{O} \to \mathbb{R}^N_{\geq 0}$.
    - $\rho_i : (\mathbf{o}) \in \{0, 1\}$ for $\mathbf{o} \in \mathbf{O}$ and security i.
- Convex compact price space $\Pi = \mathcal{H}(\rho(\mathcal{O}))$.
- Cost function $C : \mathbb{R}^N \mapsto \mathbb{R}$ with

$$C(\mathbf{q}) = \max_{\mathbf{x} \in \Pi} \mathbf{x} \cdot \mathbf{q} - R(\mathbf{x})$$

    - A convex R with relint$(\Pi) \subseteq dom(R)$.
    - A quantity vector $\mathbf{q}_t = \mathbf{q}_{t-1} + \mathbf{p}_t$ for security bundle purchases $\mathbf{p}_t$.
- Set the price $\mathbf{x}_t = \arg\max_{\mathbf{x} \in \Pi} \mathbf{x} \cdot \mathbf{q}_t - R(\mathbf{x})$.

Market making Using Cost Function + Conjugate Duality (1/2)

[Abernethy, Chen & Vaughan TEAC 2012].

- Outcome space $\mathcal{O}$.
- N securities by a payoff function $\rho : \mathcal{O} \to \mathbb{R}^N_{\geq 0}$.
  - $\rho_i : (\mathbf{o}) \in \{0, 1\}$ for $\mathbf{o} \in \mathbf{O}$ and security i.
- Convex compact price space $\Pi = \mathcal{H}(\rho(\mathcal{O}))$.
- Cost function $C : \mathbb{R}^N \mapsto \mathbb{R}$ with

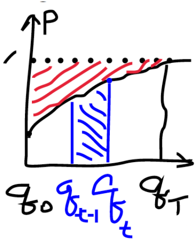
$$C(\mathbf{q}) = \max_{\mathbf{x} \in \Pi} \mathbf{x} \cdot \mathbf{q} - R(\mathbf{x})$$

  - A convex R with relint$(\Pi) \subseteq dom(R)$.
  - A quantity vector $\mathbf{q}_t = \mathbf{q}_{t-1} + \mathbf{p}_t$ for security bundle purchases $\mathbf{p}_t$.
- Set the price $\mathbf{x}_t = \arg\max_{\mathbf{x} \in \Pi} \mathbf{x} \cdot \mathbf{q}_t - R(\mathbf{x})$.
- The worst-case lost: $-(C(\mathbf{q}_T) - C(\mathbf{q}_0) - \max_{\mathbf{x} \in \Pi} x \cdot \mathbf{q}_T)$.

Market making Using Cost Function + Conjugate Duality (2/2)

$$
\begin{aligned}
C(\mathbf{q}_T) - C(\mathbf{q}_0) &= \sum_{t=1}^{T} C(\mathbf{q}_t) - C(\mathbf{q}_{t-1}) \\
&\approx \sum_{t=1}^{T} \nabla C(\mathbf{q}_{t-1}) \cdot (\mathbf{q}_t - \mathbf{q}_{t-1}) = \sum_{t=1}^{T} \mathbf{x}_t \cdot \mathbf{p}_t
\end{aligned}
$$

- The instantaneous price $x_t = \nabla C(\mathbf{q}_{t-1})$.
- The worst-case loss of the market maker $\approx \max_{\mathbf{x} \in \Pi} \mathbf{x} \cdot \mathbf{q}_T - \sum_{t=1}^{T} \mathbf{x}_t \cdot \mathbf{p}_t$.



Correspondence b/w Market Making & Online Learning

- Considering security bundles $\mathbf{p}_t$ as the negative loss vectors ℓ_t,
  $\Rightarrow$ **the duality-based cost function market maker becomes FTRL [Chen & Vaughan EC'10, Abernethy, Chen & Vaughan TEAC 2012].**
- Online linear optimization problem using FTRL:
  1. The learner given access to a fixed space of weights $\mathcal{K} \subseteq \mathbb{R}^N$
  2. Select a weight vector $\mathbf{x}_t \in \mathcal{K}$.
  3. Uses a convex regularizer $\mathcal{R}(\cdot)$, a parameter of FTRL.
  4. Receives loss vectors $f_t(\mathbf{x}_t) = \langle \ell_t, \mathbf{x} \rangle$. A cumulative loss vector L_t updated according to $L_t = L_{t-1} + \ell_t$
  5. FTRL selects the weights by solving $\mathbf{x}_t = \arg\min_{\mathbf{x}\in\mathcal{K}} [\sum_{s=1}^{t-1} f_s(\mathbf{x}) + \frac{1}{\eta}\mathcal{R}(\mathbf{x})]$.
  6. The learner's regret: $\sum_{t=1} \mathbf{x}_t \cdot f_t - \min_{x\in\mathcal{K}} \mathbf{x} \cdot L_T$

Predictable Sequences (1/2)

- "Regular" or "predictable" sequences [Rakhlin & Sridharan COLT'13]: For $t \in \{1, \ldots, T\}$, $M_t : \mathcal{F}^{t-1} \mapsto \mathcal{F}$:

$$M_1, M_2(\boldsymbol{\ell}_1), \ldots, M_T(\boldsymbol{\ell}_1, \ldots, \boldsymbol{\ell}_{T-1}).$$

$\Rightarrow \quad M_t(\boldsymbol{\ell}_1, \ldots, \boldsymbol{\ell}_{t-1}) = \boldsymbol{\ell}_t$ for all t.

  - $\{\boldsymbol{\ell}_t\}_{t \geq 1}$: a noiseless time series.

Predictable Sequences (1/2)

- "Regular" or "predictable" sequences [Rakhlin & Sridharan COLT'13]: For $t \in \{1, \ldots, T\}$, $M_t : \mathcal{F}^{t-1} \mapsto \mathcal{F}$:

$$M_1, M_2(\ell_1), \ldots, M_T(\ell_1, \ldots, \ell_{T-1}).$$

$\Rightarrow \quad M_t(\ell_1, \ldots, \ell_{t-1}) = \ell_t$ for all t.

  - $\{\ell_t\}_{t \geq 1}$: a noiseless time series.

This suffers no regret.

Predictable Sequences (1/2)

- "Regular" or "predictable" sequences [Rakhlin & Sridharan COLT'13]: For $t \in \{1, \ldots, T\}$, $M_t : \mathcal{F}^{t-1} \mapsto \mathcal{F}$:

$$M_1, M_2(\ell_1), \ldots, M_T(\ell_1, \ldots, \ell_{T-1}).$$

$\Rightarrow \quad M_t(\ell_1, \ldots, \ell_{t-1}) = \ell_t$ for all t.

  - $\{\ell_t\}_{t \geq 1}$: a noiseless time series.

  This suffers no regret.

- Consider $M_t(\ell_1, \ldots, \ell_{t-1}) \approx \ell_t$, for all t.

Predictable Sequences (1/2)

- "Regular" or "predictable" sequences [Rakhlin & Sridharan COLT'13]: For $t \in \{1, \dots, T\}$, $M_t : \mathcal{F}^{t-1} \mapsto \mathcal{F}$:

$$M_1, M_2(\ell_1), \dots, M_T(\ell_1, \dots, \ell_{T-1}).$$

$\Rightarrow \quad M_t(\ell_1, \dots, \ell_{t-1}) = \ell_t$ for all t.

  - $\{\ell_t\}_{t \geq 1}$: a noiseless time series.

  This suffers no regret.

- Consider $M_t(\ell_1, \dots, \ell_{t-1}) \approx \ell_t$, for all t.
  - $\{\ell_t\}_{t \geq 1}$: a predictable sequence plus adversarial noise.

Predictable Sequences (2/2)

- L_p-deviation for the loss functions $\{f_t\}$:

$$D_p = \sum_{t=1}^{T} \max_{\mathbf{x} \in \mathcal{X}} \|\nabla f_t(\mathbf{x}) - \nabla f_{t-1}(\mathbf{x})\|_p^2.$$

- Low deviation $\Rightarrow \boldsymbol{\ell}_t \approx \boldsymbol{\ell}_{t-1}$, so we focus on $M_t(\boldsymbol{\ell}_1, \ldots, \boldsymbol{\ell}_{t-1}) = \boldsymbol{\ell}_{t-1}$.

Theorems [Chiang *et al.* COLT'12]

When the L_2-deviation is D_2, the regret of the algorithm is $O(\sqrt{D_2})$.
When the L_∞-deviation is D_∞, the regret of the algorithm is $O(\sqrt{D_\infty \ln N})$.

Optimistic Lazy-Update Online Mirror Descents (1/3)

- **Inspiration:** the double online mirror descent [Chiang *et al.* COLT'12].
- We propose the optimistic lazy-update online mirror descent algorithm (OLU-OMD) for predictable sequences.
  - We focus on the case that $M_t(\ell_1, \ldots, \ell_{t-1}) = \ell_{t-1}$.
  - Double lazy-update online mirror descent.

Algorithm OLU-OMD

1. Let $\mathbf{x}_1 = \hat{\mathbf{x}}_1 = \mathbf{y}_1 = (\frac{1}{N}, \ldots, \frac{1}{N})^\top$.
2. **for** $t \in [T]$ **do**
   - Receive f_t and compute $\ell_t = \nabla f_t(\hat{\mathbf{x}}_t)$.
   - Update
     $\mathbf{x}_{t+1} = \arg\min_{\mathbf{x}\in\mathcal{X}} \mathcal{B}^{\mathcal{R}}(\mathbf{x}, \mathbf{y}_{t+1})$ for $\nabla\mathcal{R}(\mathbf{y}_{t+1}) = \nabla\mathcal{R}(\mathbf{y}_t) - \eta_t \ell_t$,
     $\hat{\mathbf{x}}_{t+1} = \arg\min_{\hat{\mathbf{x}}\in\mathcal{X}} \mathcal{B}^{\mathcal{R}}(\hat{\mathbf{x}}, \hat{\mathbf{y}}_{t+1})$ for $\nabla\mathcal{R}(\hat{\mathbf{y}}_{t+1}) = \nabla\mathcal{R}(\mathbf{y}_{t+1}) - \eta_t M_{t+1}$.

Optimistic Lazy-Update Online Mirror Descents (2/3)

Regard OLU-OMD as "Be The Regularized Leader with Predictors".

Lemma 1

According to the first update of OLU-OMD, $\nabla R_t(\mathbf{y}_{t+1}) = \nabla R_t(\mathbf{y}_t) - \eta_t \boldsymbol{\ell}_t$ and $\mathbf{x}_{t+1} = \mathcal{B}^{R_t}(\mathbf{x}, \mathbf{y}_{t+1})$, we have

$$\mathbf{x}_{t+1} = \arg\min_{\mathbf{x}\in\mathcal{K}} \left(\eta_t \left\langle \sum_{s=1}^{t} \boldsymbol{\ell}_s, \mathbf{x} \right\rangle + R(\mathbf{x}) \right).$$

For the second update, $\nabla R(\hat{\mathbf{y}}_{t+1}) = \nabla R(\mathbf{y}_{t+1}) - \eta_t M_{t+1}$ and $\hat{\mathbf{x}}_{t+1} = \mathcal{B}^{R_t}(\hat{\mathbf{x}}, \mathbf{y}_{t+1})$, we have

$$\hat{\mathbf{x}}_{t+1} = \arg\min_{\mathbf{x}\in\mathcal{K}} \left(\eta_t \left\langle \sum_{s=1}^{t} \boldsymbol{\ell}_s + M_{t+1}, \mathbf{x} \right\rangle + R(\mathbf{x}) \right).$$

Optimistic Lazy-Update Online Mirror Descents (3/3)

Regard OLU-OMD as "**Be** The Regularized Leader (BTRL) + Predictors".

Lemma 1

According to the first update of OLU-OMD, $\nabla R_t(\mathbf{y}_{t+1}) = \nabla R_t(\mathbf{y}_t) - \eta_t \boldsymbol{\ell}_t$ and $\mathbf{x}_{t+1} = \mathcal{B}^{R_t}(\mathbf{x}, \mathbf{y}_{t+1})$, we have

$$\mathbf{x}_{t+1} = \arg\min_{\mathbf{x} \in \mathcal{K}} \left(\eta_t \left\langle \sum_{s=1}^{t} \boldsymbol{\ell}_s, \mathbf{x} \right\rangle + R(\mathbf{x}) \right).$$

For the second update with $M_{t+1} = \boldsymbol{\ell}_t$, $\nabla R(\hat{\mathbf{y}}_{t+1}) = \nabla R(\mathbf{y}_{t+1}) - \eta_t \boldsymbol{\ell}_t$ and $\hat{\mathbf{x}}_{t+1} = \mathcal{B}^{R_t}(\hat{\mathbf{x}}, \mathbf{y}_{t+1})$, we have

$$\hat{\mathbf{x}}_{t+1} = \arg\min_{\mathbf{x} \in \mathcal{K}} \left(\eta_t \left\langle \sum_{s=1}^{t} \boldsymbol{\ell}_s + \boldsymbol{\ell}_t, \mathbf{x} \right\rangle + R(\mathbf{x}) \right).$$

Implementation of BTRL

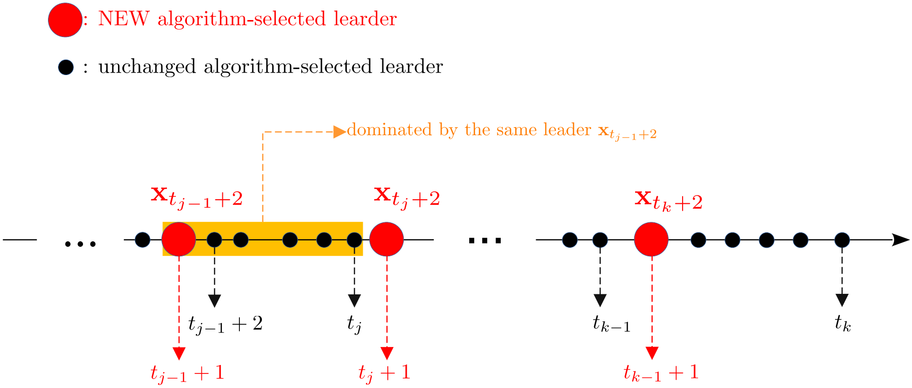
- The cumulative loss of BTRL can be obtained by evaluating

$$\sum_{t=1}^{T} \langle \boldsymbol{\ell}_t, \mathbf{x}_{t+1} \rangle, \quad \text{where } \mathbf{x}_{t+1} \text{ is from FTRL.}$$

- Let $L_t := \langle \sum_{s=1}^{t} \boldsymbol{\ell}_s, \mathbf{x} \rangle$, for all t.
- Let $\mathbf{x}_1 := \arg\min_{\mathbf{x} \in \mathcal{K}} R(\mathbf{x})$ and

$$\mathbf{x}_{t_j+1} \in \arg\min_{\mathbf{x} \in \mathcal{K}} \left\{ L_{t_j(\mathbf{x})} + \frac{R(\mathbf{x})}{\eta} \right\}.$$

Implementation of BTRL + Changes of Leaders



Condition 1: Frequent Changes of Strong Leaders

Main Theorem 1

If $M_t(\boldsymbol{\ell}_1, \ldots, \boldsymbol{\ell}_{t-1}) = \boldsymbol{\ell}_{t-1}$ and for all $t \in \{1, \ldots, T\}$, with $L_t(\mathbf{x}) - L_t(\mathbf{x}_{t+1}) \geq \delta$ for all $\mathbf{x} \notin \arg\min_{\mathbf{x} \in \mathcal{K}} L_t(\mathbf{x})$ over k changes of leaders, then the regret of OLU-OMD is

$$\sum_{t=1}^{T} \langle \boldsymbol{\ell}_t, \hat{\mathbf{x}}_t \rangle - L_T(\mathbf{x}^*) = O\left(\sqrt{\sum_{t=1}^{T} \|\boldsymbol{\ell}_t\|_2 \|\boldsymbol{\ell}_t - \boldsymbol{\ell}_{t-1}\|_2}\right) - \delta k,$$

where $\eta_t = \frac{1}{\sqrt{\sum_{s=1}^{t-1} \|\boldsymbol{\ell}_s\|_2 \|\boldsymbol{\ell}_s - \boldsymbol{\ell}_{s-1}\|_2}}$.

Condition 2: Periodic Changes of DOMINANT Experts (1/2)

- To ease the discussion, we consider the *reward maximizing* setting of two experts.
  - $\Rightarrow$ Modified Optimistic Multiplicative Updates (OMU) [Chiang, Lee, & Lu 2011] for predictable sequences.
- A reward maximization model with experts of $\{0,1\}$-rewards, each $r_i^{(t)} \in \{0,1\}$ for $i \in \{1,2\}$: a dominant expert gives a reward of 1.

Condition 2: Periodic Changes of DOMINANT Experts (1/2)

- To ease the discussion, we consider the *reward maximizing* setting of two experts.
  - $\Rightarrow$ Modified Optimistic Multiplicative Updates (OMU) [Chiang, Lee, & Lu 2011] for predictable sequences.
- A reward maximization model with experts of $\{0, 1\}$-rewards, each $r_i^{(t)} \in \{0, 1\}$ for $i \in \{1, 2\}$: a dominant expert gives a reward of 1.
- **The Tricky Cae**: Only few changes of dominant experts.
  - The BTRL algorithm cannot change leaders promptly enough.

Condition 2: Periodic Changes of DOMINANT Experts (2/2)

Try to catch up with the changes of dominant experts quickly enough!

Modified Optimistic Multiplicative Updates [MOMU] [Chiang, Lee & Lu 2011]

1. Let $\bar{\mathbf{x}}^{(1)} = \mathbf{x}^{(1)} = \hat{\mathbf{x}}^{(1)} = (\frac{1}{N}, \ldots, \frac{1}{N})^\top$
2. **for** $t \in [T]$ **do**
   Receive f_t and compute $\mathbf{r}_t = \nabla f_t(\hat{\mathbf{x}}^t)$.
   Update for $i \in \{1, \ldots, N\}$:
   $$\bar{\mathbf{x}}^{(i)}_{t+1} = \mathbf{x}^{(i)}_t e^{\eta \mathbf{r}^{(i)}_t} / \bar{Z}_{t+1} \text{ with } \bar{Z}_{t+1} = \sum_j \mathbf{x}^{(j)}_t e^{\eta \mathbf{r}^{(j)}_t},$$
   $$\mathbf{x}^{(i)}_{t+1} = (1-\beta)\bar{\mathbf{x}}^{(i)}_{t+1} + \beta \frac{1}{N},$$
   $$\hat{\mathbf{x}}^{(i)}_{t+1} = \mathbf{x}^{(i)}_{t+1} e^{\eta \mathbf{r}^{(i)}_t} / \hat{Z}_{t+1} \text{ with } \hat{Z}_{t+1} = \sum_j \mathbf{x}^{(j)}_{t+1} e^{\eta \mathbf{r}^{(j)}_t}.$$

- $\beta = \frac{1}{t}$
- $M_{t+1} = \ell_t$

Few Changes of Dominant Experts

- We have k periods of time, each defined by an dominant expert.
  - The dominant expert is alternating $\Rightarrow D_\infty = k$.
- **Claim:** After the end of any period, it takes at most $O(\log(T/k)/\eta)$ time steps for the ratio of the previous dominant expert's probability to the probability of the dominant expert in current period to achieve at most 1 again.

Main Theorem 2

The regret of MOMU is $O(-T/2 + (k-1)\log(T/k)/\eta)$. With $\eta = O(1/\sqrt{D_\infty}) = O(1/\sqrt{k})$, the regret is

$$O(-T/2 - \sqrt{k}\log(T/k) + k^{\frac{3}{2}}\log(T/k)).$$

Future Work

- Complementing our finding with experimental results.
- A "passive" market maker makes profit of a security by placing *limit orders* as *bids and asks* to reap the bid-ask spreads on the orderbook, yet takes exposure risk of the inventory.
  - A promising direction of market making design to try is to combine the merits of both the designs.

Thanks for your attention.