Minimax Solutions, Random Playouts, and Perturbations

Jacob Abernethy

University of Michigan
Department of Computer Science and Engineering

December 13, 2014

Prediction with Expert Advice Game

We have n experts. One expert will make no more than k errors. Let $\mathbf{C} \in \mathbb{N}^n$ be the cumulative number of losses on the experts. Let $Loss_{\text{alg}}$ be the loss of the algorithm.

Prediction with Expert Advice Game

We have n experts. One expert will make no more than k errors. Let $\mathbf{C} \in \mathbb{N}^n$ be the cumulative number of losses on the experts. Let $Loss_{\text{alg}}$ be the loss of the algorithm.

While $\min_i C_i \leq k$:

1. Algorithm selects weights $\mathbf{w} \in \Delta_n$
2. Adversary selects $\ell \in \{0, 1\}^n$
3. Algorithm total cost: $Loss_{\text{alg}} \leftarrow Loss_{\text{alg}} + \mathbf{w}^\top \ell$
4. Experts' costs: $\mathbf{C} \leftarrow \mathbf{C} + \ell$

Prediction with Expert Advice Game

We have n experts. One expert will make no more than k errors. Let $\mathbf{C} \in \mathbb{N}^n$ be the cumulative number of losses on the experts. Let $Loss_{\text{alg}}$ be the loss of the algorithm.

While $\min_i C_i \leq k$:

1. Algorithm selects weights $\mathbf{w} \in \Delta_n$
2. Adversary selects $\ell \in \{0, 1\}^n$
3. Algorithm total cost: $Loss_{\text{alg}} \leftarrow Loss_{\text{alg}} + \mathbf{w}^\top \ell$
4. Experts' costs: $\mathbf{C} \leftarrow \mathbf{C} + \ell$

This is a zero-sum game:

loss to learner = gain to adversary = $Loss_{\text{alg}}$.

Can we solve this game?

Prediction with Expert Advice Game

While $\min_i C_i \leq k$:

1. Algorithm selects weights $\mathbf{w} \in \Delta_n$
2. Adversary selects $\ell \in \{0, 1\}^n$
3. Algorithm total cost: $Loss_{\text{alg}} \leftarrow Loss_{\text{alg}} + \mathbf{w}^\top \ell$
4. Experts' costs: $\mathbf{C} \leftarrow \mathbf{C} + \ell$

Prediction with Expert Advice Game

While $\min_i C_i \leq k$:

1. Algorithm selects weights $\mathbf{w} \in \Delta_n$
2. Adversary selects $\ell \in \{0,1\}^n$
3. Algorithm total cost: $Loss_{\text{alg}} \leftarrow Loss_{\text{alg}} + \mathbf{w}^\top \ell$
4. Experts' costs: $\mathbf{C} \leftarrow \mathbf{C} + \ell$

Question 1: Given some "state" $\mathbf{C}$, least-achievable $L_{\text{alg}}(\mathbf{C})$?

Prediction with Expert Advice Game

While $\min_i C_i \leq k$:

1. Algorithm selects weights $\mathbf{w} \in \Delta_n$
2. Adversary selects $\ell \in \{0, 1\}^n$
3. Algorithm total cost: $Loss_{\text{alg}} \leftarrow Loss_{\text{alg}} + \mathbf{w}^\top \ell$
4. Experts' costs: $\mathbf{C} \leftarrow \mathbf{C} + \ell$

Question 1: Given some "state" $\mathbf{C}$, least-achievable $L_{\text{alg}}(\mathbf{C})$?

Question 2: Given some "state" $\mathbf{C}$, what is $\mathbf{w}^*(\mathbf{C})$?

Solution

Given state $\mathbf{C}$, define a random process $\hat{\mathbf{C}}^t$:

$\hat{\mathbf{C}}^0 = \mathbf{C}$ and $\hat{\mathbf{C}}^{t+1} = \hat{\mathbf{C}}^t + \mathbf{e}_I$ where $I \sim [n]$ u.a.r.

(That is, $\hat{\mathbf{C}}^t$ generated by randomly assigning t expert losses.)

Solution

Given state $\mathbf{C}$, define a random process $\hat{\mathbf{C}}^t$:

$\hat{\mathbf{C}}^0 = \mathbf{C}$ and $\hat{\mathbf{C}}^{t+1} = \hat{\mathbf{C}}^t + \mathbf{e}_I$ where $I \sim [n]$ u.a.r.

(That is, $\hat{\mathbf{C}}^t$ generated by randomly assigning t expert losses.)

$$Loss_{\text{alg}}(\mathbf{C}) \quad =$$

Solution

Given state $\mathbf{C}$, define a random process $\hat{\mathbf{C}}^t$:

$\hat{\mathbf{C}}^0 = \mathbf{C}$ and $\hat{\mathbf{C}}^{t+1} = \hat{\mathbf{C}}^t + \mathbf{e}_I$ where $I \sim [n]$ u.a.r.

(That is, $\hat{\mathbf{C}}^t$ generated by randomly assigning t expert losses.)

$$Loss_{\text{alg}}(\mathbf{C}) \quad = \quad \frac{1}{n}\mathbb{E}[\text{time } t \text{ until } \hat{\mathbf{C}}^t \text{ ``dead''}]$$

Solution

Given state $\mathbf{C}$, define a random process $\hat{\mathbf{C}}^t$:

$\hat{\mathbf{C}}^0 = \mathbf{C}$ and $\hat{\mathbf{C}}^{t+1} = \hat{\mathbf{C}}^t + \mathbf{e}_I$ where $I \sim [n]$ u.a.r.

(That is, $\hat{\mathbf{C}}^t$ generated by randomly assigning t expert losses.)

$$\begin{aligned} Loss_{\text{alg}}(\mathbf{C}) &= \frac{1}{n}\mathbb{E}[\text{time } t \text{ until } \hat{\mathbf{C}}^t \text{ ``dead''}] \\ &= \frac{1}{n}\mathbb{E}[\min\{t : \hat{C}_i^t \geq k \ \forall i\}] \end{aligned}$$

Solution

Given state $\mathbf{C}$, define a random process $\hat{\mathbf{C}}^t$:

$\hat{\mathbf{C}}^0 = \mathbf{C}$ and $\hat{\mathbf{C}}^{t+1} = \hat{\mathbf{C}}^t + \mathbf{e}_I$ where $I \sim [n]$ u.a.r.

(That is, $\hat{\mathbf{C}}^t$ generated by randomly assigning t expert losses.)

$$\begin{aligned} Loss_{\text{alg}}(\mathbf{C}) &= \frac{1}{n}\mathbb{E}[\text{time } t \text{ until } \hat{\mathbf{C}}^t \text{ ``dead''}] \\ &= \frac{1}{n}\mathbb{E}[\min\{t : \hat{C}_i^t \geq k \ \forall i\}] \end{aligned}$$

$$\mathbf{w}^*(\mathbf{C}) =$$

Solution

Given state $\mathbf{C}$, define a random process $\hat{\mathbf{C}}^t$:
$\hat{\mathbf{C}}^0 = \mathbf{C}$ and $\hat{\mathbf{C}}^{t+1} = \hat{\mathbf{C}}^t + \mathbf{e}_I$ where $I \sim [n]$ u.a.r.
(That is, $\hat{\mathbf{C}}^t$ generated by randomly assigning t expert losses.)

$$\begin{aligned} Loss_{\text{alg}}(\mathbf{C}) &= \frac{1}{n}\mathbb{E}[\text{time } t \text{ until } \hat{\mathbf{C}}^t \text{ ``dead''}] \\ &= \frac{1}{n}\mathbb{E}[\min\{t : \hat{C}_i^t \geq k \ \forall i\}] \end{aligned}$$

$$\mathbf{w}^*(\mathbf{C}) = \mathbb{E}[\text{last expert to die}]$$

Solution

Given state $\mathbf{C}$, define a random process $\hat{\mathbf{C}}^t$:

$\hat{\mathbf{C}}^0 = \mathbf{C}$ and $\hat{\mathbf{C}}^{t+1} = \hat{\mathbf{C}}^t + \mathbf{e}_I$ where $I \sim [n]$ u.a.r.

(That is, $\hat{\mathbf{C}}^t$ generated by randomly assigning t expert losses.)

$$\begin{aligned} Loss_{\text{alg}}(\mathbf{C}) &= \frac{1}{n}\mathbb{E}[\text{time } t \text{ until } \hat{\mathbf{C}}^t \text{ ``dead''}] \\ &= \frac{1}{n}\mathbb{E}[\min\{t : \hat{C}_i^t \geq k \ \forall i\}] \end{aligned}$$

$$\begin{aligned} \mathbf{w}^*(\mathbf{C}) &= \mathbb{E}[\text{last expert to die}] \\ &= [\Pr(\exists t \text{ s.t. } \hat{C}_i^t < k \leq \hat{C}_j^t \ \forall j \neq i)]_{i=1\ldots n} \end{aligned}$$

[Abernethy and Warmuth, 2010, Abernethy, Warmuth, and Yellin, 2008]

Random Playouts: An Online Decision Template

The previous example gives us a nice template for designing online decision algorithms.

1. Take your current state S defined by the history of moves thus far
2. Add to the history a sequence of random moves, "guesses" of the adversary's strategy
3. Train an offline algorithm on the full sequence (history and guessed future)
4. On current round, play according to the optimal offline algorithm

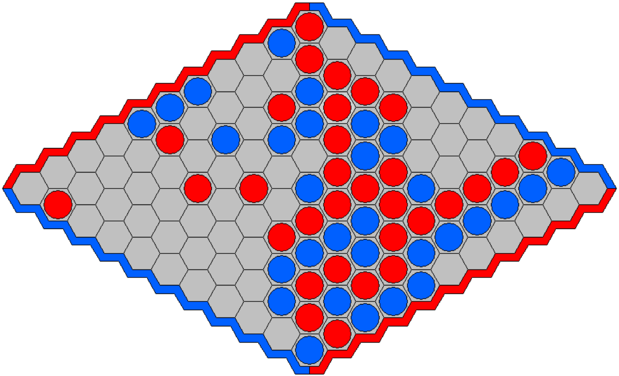
Random Playouts: An Online Decision Template

The previous example gives us a nice template for designing online decision algorithms.

1. Take your current state S defined by the history of moves thus far
2. Add to the history a sequence of random moves, "guesses" of the adversary's strategy
3. Train an offline algorithm on the full sequence (history and guessed future)
4. On current round, play according to the optimal offline algorithm

This is *minimax optimal* in a number of cases!

Random-Turn Variant of Hex



Like regular Hex, but on each round a coin is tossed to select which player goes next.

The Typical Regret-minimization Framework

We imagine an online game between Nature and Learner. Learner has a (typically convex) *decision set* $\mathcal{X} \subset \mathbb{R}^d$, and Nature has an action set $\mathcal{Z}$, and there is a loss function $\ell : \mathcal{X} \times \mathcal{Z} \to \mathbb{R}$ defined in advance.

Online Convex Optimization

For $t = 1, \ldots, T$:

- Learner chooses $x_t \in \mathcal{X}$
- Nature chooses $z_t \in \mathcal{Z}$
- Learner suffers $\ell(x_t, z_t)$

Learner is concerned with the *regret*:

$$\sum_{t=1}^{T} \ell(x_t, z_t) - \min_{x \in \mathcal{X}} \sum_{t=1}^{T} \ell(x, z_t)$$

This talk we assume ℓ is *linear* in x; WLOG $\ell(x_t, z_t) = x^\top z_t$.

A Bad Algorithm

Follow the Leader (FTPL)

for $t = 1 \ldots T$,

$$x_t \longleftarrow \arg\min_{x \in \mathcal{X}} \left(\sum_{s=1}^{t-1} x^\top l_s \right)$$

A Bad Algorithm

Follow the Leader (FTPL)

for $t = 1 \ldots T$,

$$x_t \longleftarrow \arg\min_{x \in \mathcal{X}} \left(\sum_{s=1}^{t-1} x^\top l_s \right)$$

Why is this a bad algorithm?

A Bad Algorithm

Follow the Leader (FTPL)

for $t = 1 \ldots T$,

$$x_t \longleftarrow \arg\min_{x \in \mathcal{X}} \left(\sum_{s=1}^{t-1} x^\top l_s \right)$$

Why is this a bad algorithm?
Instability!

Follow the _____ Leader

Follow the **Regularized** Leader (FTRL)

Input: learning rate $\eta > 0$, regularizer $R : \mathcal{X} \to \mathbb{R}$

for $t = 1 \ldots T$, $\quad x_t \longleftarrow \arg\min_{x \in \mathcal{X}} \left(R(x) + \eta \sum_{s=1}^{t-1} x^\top l_s \right).$

Follow the _____ Leader

Follow the **Regularized** Leader (FTRL)

Input: learning rate $\eta > 0$, regularizer $R : \mathcal{X} \to \mathbb{R}$

for $t = 1 \ldots T$, $\quad x_t \longleftarrow \arg\min_{x \in \mathcal{X}} \left(R(x) + \eta \sum_{s=1}^{t-1} x^\top l_s \right).$

Follow the **Perturbed** Leader (FTPL)

Input: A perturbation distribution $\mathcal{D} \in \Delta(\mathbb{R}^d)$.

for $t = 1 \ldots T$,

Sample $Z \sim \mathcal{D}$, $\quad x_t \longleftarrow \arg\min_{x \in \mathcal{X}} \left(x^\top Z + \sum_{s=1}^{t-1} x^\top l_s \right)$

Follow the _____ Leader

Follow the **Regularized** Leader (FTRL)

Input: learning rate $\eta > 0$, regularizer $R : \mathcal{X} \to \mathbb{R}$

$$\textbf{for } t = 1 \ldots T, \quad x_t \longleftarrow \arg\min_{x \in \mathcal{X}} \left(R(x) + \eta \sum_{s=1}^{t-1} x^\top l_s \right).$$

Follow the **Perturbed** Leader (FTPL)

Input: A perturbation distribution $\mathcal{D} \in \Delta(\mathbb{R}^d)$.

for $t = 1 \ldots T$,

$$\text{Sample } Z \sim \mathcal{D}, \quad x_t \longleftarrow \arg\min_{x \in \mathcal{X}} \left(x^\top Z + \textstyle\sum_{s=1}^{t-1} x^\top l_s \right)$$

This COLT: FTPL is (in expectation) just a special case of FTRL [Abernethy, Lee, Sinha, and Tewari, 2014]

Regret Bounds EASY for FTRL

Theorem (now classical)

Let $l_1, \ldots, l_T$ be an arbitrary sequence of vectors, and let $L_t := l_1 + \ldots l_t$. Assume $R(x_0) = 0$. Then

$$\begin{aligned} \text{Regret}_T &\leq \frac{R(x^*)}{\eta} + \sum_{t=1}^{T} D_R(x_t, x_{t+1}) \\ &\leq \frac{R(x^*)}{\eta} + \eta \sum_{t=1}^{T} (x_t - x_{t+1})^\top l_t \\ \implies \text{Regret}_T &\leq O\left(\sqrt{\textstyle\sum_{t=1}^{T} \|l_t\|^2}\right) \end{aligned}$$

where $D_R(\cdot, \cdot)$ is the *Bregman divergence* w.r.t. R, and the last line follows from tuning η and assuming some curvature properties of R.

Regret Bounds NOT SO EASY with FTPL

- Kalai and Vempala (2005)

The exponential density from which $p_1[i]$ is chosen, namely $\varepsilon e^{-\varepsilon x}$, has the following property:

$$\begin{aligned} P[p_1[i] > v + c \mid p_1[i] \geqslant v] &= \frac{\int_{v+c}^{\infty} \varepsilon e^{-\varepsilon x}\, dx}{\int_{v}^{\infty} \varepsilon e^{-\varepsilon x}\, dx} \\ &= e^{-\varepsilon c} \\ &\geqslant 1 - \varepsilon c. \end{aligned}$$

- Devroye et al. (2013)

$$\begin{aligned} \mathbb{P}[|A_t| = 1] &= \sum_{k=-t}^{t} \sum_{j=1}^{N} p_t(k) \mathbb{P}\left[\min_{i \neq j} \{L_{i,t-1} + Z_{i,t}\} \geq L_{j,t-1} + \frac{k}{2} + 2\right] \\ &\geq \sum_{k=-t}^{t-4} \sum_{j=1}^{N} p_t(k+4) \mathbb{P}\left[\min_{i \neq j} \{L_{i,t-1} + Z_{i,t}\} \geq L_{j,t-1} + \frac{k+4}{2}\right] \frac{p_t(k)}{p_t(k+4)} \\ &= \sum_{k=-t+4}^{t} \sum_{j=1}^{N} p_t(k) \mathbb{P}\left[\min_{i \neq j} \{L_{i,t-1} + Z_{i,t}\} \geq L_{j,t-1} + \frac{k}{2}\right] \frac{p_t(k-4)}{p_t(k)} . \end{aligned}$$

Before proceeding, we need to make two observations. First of all,

$$\begin{aligned} \sum_{j=1}^{N} p_t(k) \mathbb{P}\left[\min_{i \neq j} \{L_{i,t-1} + Z_{i,t}\} \geq L_{j,t-1} + \frac{k}{2}\right] &\geq \mathbb{P}\left[\exists j \in S_t : Z_{j,t} = \frac{k}{2}\right] \\ &\geq \mathbb{P}\left[\min_{j \in S_t} Z_{j,t} = \frac{k}{2}\right], \end{aligned}$$

[more math omitted]

- Van Evran et al. (2014)

$$\begin{aligned} &\Pr(\mathcal{A}_t \mid M = m, C = c) \\ &\quad = \Pr(V = m-1, W > m)\frac{c}{c+1} + \Pr(V = m-1, W = m)\frac{c+1}{c+2} \\ &\quad + \Pr(V = m, W > m)\frac{1}{c+1} + \Pr(V = m, W = m)\frac{1}{c+2} \\ &\quad + \Big(\Pr(V = W-1, W < m) + \Pr(V = W, W < m)\Big)\frac{1}{2}, \\ &\Pr(\mathcal{A}_{t+1} \mid M = m, C = c) \\ &\quad = \Pr(V = m-1, W+X > m)\frac{c}{c+1} + \Pr(V = m-1, W+X = m)\frac{c+1}{c+2} \\ &\quad + \Pr(V = m, W+X > m)\frac{1}{c+1} + \Pr(V = m, W+X = m)\frac{1}{c+2} \\ &\quad + \Big(\Pr(V = W+X-1, W+X < m) + \Pr(V = W+X, W+X < m)\Big)\frac{1}{2} \end{aligned}$$

for any m and c. Thus

$$\begin{aligned} &\Pr(\mathcal{A}_{t+1} \mid M = m, C = c) - \Pr(\mathcal{A}_t \mid M = m, C = c) \\ &\quad = \alpha\Big(\Pr(\mathcal{A}_{t+1} \mid M = m, C = c, X = 0) - \Pr(\mathcal{A}_t \mid M = m, C = c, X = 0)\Big) \\ &\quad + (1-\alpha)\Big(\Pr(\mathcal{A}_{t+1} \mid M = m, C = c, X = 1) - \Pr(\mathcal{A}_t \mid M = m, C = c, X = 1)\Big) \\ &\quad = (1-\alpha)\Big(\Pr(\mathcal{A}_{t+1} \mid M = m, C = c, X = 1) - \Pr(\mathcal{A}_t \mid M = m, C = c)\Big), \qquad (13) \end{aligned}$$

+ more than 10 pages

Fenchel Duality: A Primer

Definition of the Fenchel Conjugate

Given a convex $f : \mathbb{R}^d \to \mathbb{R}$, the *Fenchel Conjugate* of f is

$$f^*(\theta) := \sup_{x \in \mathrm{dom}(f)} x^\top \theta - f(x)$$

Fenchel Duality: A Primer

Definition of the Fenchel Conjugate

Given a convex $f : \mathbb{R}^d \to \mathbb{R}$, the *Fenchel Conjugate* of f is

$$f^*(\theta) := \sup_{x \in \mathrm{dom}(f)} x^\top \theta - f(x)$$

Lemma

The solution to

$$\arg \max_{x \in \mathrm{dom}(f)} x^\top \theta - f(x)$$

is given by the gradient $\nabla f^*(\theta)$.

FTRL $\longleftrightarrow$ FTPL

1. Let us switch from "losses" to "gains".
2. Let $\theta_t := -l_t$, and let $\Theta_t := \sum_{s=1}^{t} \theta_s$.
3. For simplicity, let us look in one dimension $x \in [0, 1]$

FTRL $\longleftrightarrow$ FTPL

1. Let us switch from "losses" to "gains".
2. Let $\theta_t := -l_t$, and let $\Theta_t := \sum_{s=1}^{t} \theta_s$.
3. For simplicity, let us look in one dimension $x \in [0, 1]$

$$\text{FTRL:} \qquad x_t = \arg\max_{x \in [0,1]} x\Theta_{t-1} - R(x)$$

FTRL $\longleftrightarrow$ FTPL

1. Let us switch from "losses" to "gains".
2. Let $\theta_t := -l_t$, and let $\Theta_t := \sum_{s=1}^{t} \theta_s$.
3. For simplicity, let us look in one dimension $x \in [0, 1]$

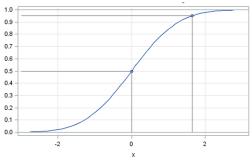
$$\text{FTRL:} \qquad x_t = \arg\max_{x \in [0,1]} x\Theta_{t-1} - R(x)$$
$$= R^{*\prime}(\Theta_{t-1})$$

FTRL $\longleftrightarrow$ FTPL

1. Let us switch from "losses" to "gains".
2. Let $\theta_t := -l_t$, and let $\Theta_t := \sum_{s=1}^{t} \theta_s$.
3. For simplicity, let us look in one dimension $x \in [0,1]$

$$\text{FTRL:} \qquad x_t = \arg\max_{x\in[0,1]} x\Theta_{t-1} - R(x) = R^{*\prime}(\Theta_{t-1})$$

Notice that $R^{*\prime}$ is an increasing function with range in $[0,1]$. Maybe looks something like this:

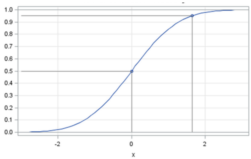


FTRL $\longleftrightarrow$ FTPL

1. Let us switch from "losses" to "gains".
2. Let $\theta_t := -l_t$, and let $\Theta_t := \sum_{s=1}^{t} \theta_s$.
3. For simplicity, let us look in one dimension $x \in [0, 1]$

$$\begin{aligned} \text{FTRL:} \qquad x_t &= \arg\max_{x \in [0,1]} x\Theta_{t-1} - R(x) \\ &= R^{*\prime}(\Theta_{t-1}) \end{aligned}$$

Notice that $R^{*\prime}$ is an increasing function with range in $[0, 1]$. Maybe looks something like this:



Hmmm.... That looks a lot like a CDF of a distribution!

FTRL $\longleftrightarrow$ FTPL

Let's define a distribution $\mathcal{D}$ with CDF $R^{*\prime}$. Then:

$$\text{FTRL:} \quad x_t = R^{*\prime}(\Theta_{t-1})$$

FTRL $\longleftrightarrow$ FTPL

Let's define a distribution $\mathcal{D}$ with CDF $R^{*\prime}$. Then:

$$\begin{aligned} \text{FTRL:} \quad x_t &= R^{*\prime}(\Theta_{t-1}) \\ &= \Pr_{Z\sim\mathcal{D}}[Z \le \Theta_{t-1}] \end{aligned}$$

FTRL $\longleftrightarrow$ FTPL

Let's define a distribution $\mathcal{D}$ with CDF $R^{*\prime}$. Then:

$$
\begin{aligned}
\text{FTRL:} \quad x_t &= R^{*\prime}(\Theta_{t-1}) \\
&= \Pr_{Z\sim\mathcal{D}}[Z \leq \Theta_{t-1}] \\
&= \mathbb{E}_{Z\sim\mathcal{D}}\left[\arg\max_{x\in[0,1]} x^\top(\Theta_{t-1} - Z)\right]
\end{aligned}
$$

FTRL $\longleftrightarrow$ FTPL

Let's define a distribution $\mathcal{D}$ with CDF $R^{*\prime}$. Then:

$$\begin{aligned} \text{FTRL:} \quad x_t &= R^{*\prime}(\Theta_{t-1}) \\ &= \Pr_{Z\sim\mathcal{D}}[Z \leq \Theta_{t-1}] \\ &= \mathbb{E}_{Z\sim\mathcal{D}}\left[\arg\max_{x\in[0,1]} x^\top(\Theta_{t-1} - Z)\right] \\ &\equiv \text{FTPL algorithm} \end{aligned}$$

That is: we have just "replicated" the FTRL algorithm (in one dimension) with FTPL via a particular perturbation.

FTRL $\longleftrightarrow$ FTPL

Let's define a distribution $\mathcal{D}$ with CDF $R^{*\prime}$. Then:

$$\begin{aligned}
\text{FTRL:} \quad x_t &= R^{*\prime}(\Theta_{t-1}) \\
&= \Pr_{Z\sim\mathcal{D}}[Z \leq \Theta_{t-1}] \\
&= \mathbb{E}_{Z\sim\mathcal{D}}\left[\arg\max_{x\in[0,1]} x^\top(\Theta_{t-1} - Z)\right] \\
&\equiv \text{FTPL algorithm}
\end{aligned}$$

That is: we have just "replicated" the FTRL algorithm (in one dimension) with FTPL via a particular perturbation.

Does this equivalence work in general?

FTPL is reducible to FTRL (but not vice versa)

1. Assume we have an arbitrary online linear optimization problem with domain $\mathcal{X}$.
2. Let $\Phi_0(\Theta) := \max_{x \in \mathcal{X}} x^\top \Theta$.
3. Notice: $\nabla \Phi_0(\Theta) = \arg\max_{x \in \mathcal{X}} x^\top \Theta$
4. Let $\mathcal{D}$ be some smooth perturbation distribution on $\mathbb{R}^d$

FTPL is reducible to FTRL (but not vice versa)

1. Assume we have an arbitrary online linear optimization problem with domain $\mathcal{X}$.
2. Let $\Phi_0(\Theta) := \max_{x \in \mathcal{X}} x^\top \Theta$.
3. Notice: $\nabla \Phi_0(\Theta) = \arg\max_{x \in \mathcal{X}} x^\top \Theta$
4. Let $\mathcal{D}$ be some smooth perturbation distribution on $\mathbb{R}^d$

In *expectation*, Follow the Perturbed Leader described as:

$$\text{FTPL:} \quad x_t \; = \; \mathbb{E}_{Z \sim \mathcal{D}}[\arg\max_{x \in \mathcal{X}} \; x^\top (\Theta + Z)]$$

FTPL is reducible to FTRL (but not vice versa)

1. Assume we have an arbitrary online linear optimization problem with domain $\mathcal{X}$.
2. Let $\Phi_0(\Theta) := \max_{x \in \mathcal{X}} x^\top \Theta$.
3. Notice: $\nabla \Phi_0(\Theta) = \arg\max_{x \in \mathcal{X}} x^\top \Theta$
4. Let $\mathcal{D}$ be some smooth perturbation distribution on $\mathbb{R}^d$

In *expectation*, Follow the Perturbed Leader described as:

$$\text{FTPL:} \quad x_t = \mathbb{E}_{Z \sim \mathcal{D}}[\arg\max_{x \in \mathcal{X}} \; x^\top (\Theta + Z)] = \mathbb{E}_{Z \sim \mathcal{D}}[\nabla \Phi_0(\Theta + Z)]$$

FTPL is reducible to FTRL (but not vice versa)

1. Assume we have an arbitrary online linear optimization problem with domain $\mathcal{X}$.
2. Let $\Phi_0(\Theta) := \max_{x \in \mathcal{X}} x^\top \Theta$.
3. Notice: $\nabla \Phi_0(\Theta) = \arg\max_{x \in \mathcal{X}} x^\top \Theta$
4. Let $\mathcal{D}$ be some smooth perturbation distribution on $\mathbb{R}^d$

In *expectation*, Follow the Perturbed Leader described as:

$$
\begin{aligned}
\text{FTPL:} \quad x_t &= \mathbb{E}_{Z \sim \mathcal{D}}[\arg\max_{x \in \mathcal{X}} \; x^\top (\Theta + Z)] \\
&= \mathbb{E}_{Z \sim \mathcal{D}}[\nabla \Phi_0(\Theta + Z)] \\
\text{(usually)} \quad &= \nabla \underbrace{\mathbb{E}_{Z \sim \mathcal{D}}[\Phi_0(\Theta + Z)]}_{\text{define as } \Phi_{\mathcal{D}}}
\end{aligned}
$$

FTPL is reducible to FTRL (but not vice versa)

1. Assume we have an arbitrary online linear optimization problem with domain $\mathcal{X}$.
2. Let $\Phi_0(\Theta) := \max_{x \in \mathcal{X}} x^\top \Theta$.
3. Notice: $\nabla \Phi_0(\Theta) = \arg\max_{x \in \mathcal{X}} x^\top \Theta$
4. Let $\mathcal{D}$ be some smooth perturbation distribution on $\mathbb{R}^d$

In *expectation*, Follow the Perturbed Leader described as:

$$\begin{array}{lcll} \text{FTPL:} \quad x_t & = & \mathbb{E}_{Z \sim \mathcal{D}}[\arg\max_{x \in \mathcal{X}} \; x^\top (\Theta + Z)] & \\ & = & \mathbb{E}_{Z \sim \mathcal{D}}[\nabla \Phi_0(\Theta + Z)] & \\ \text{(usually)} & = & \nabla \underbrace{\mathbb{E}_{Z \sim \mathcal{D}}[\Phi_0(\Theta + Z)]}_{\text{define as } \Phi_{\mathcal{D}}} & = \quad \nabla \Phi_{\mathcal{D}}(\Theta) \end{array}$$

FTPL is reducible to FTRL (but not vice versa)

1. Assume we have an arbitrary online linear optimization problem with domain $\mathcal{X}$.
2. Let $\Phi_0(\Theta) := \max_{x \in \mathcal{X}} x^\top \Theta$.
3. Notice: $\nabla \Phi_0(\Theta) = \arg\max_{x \in \mathcal{X}} x^\top \Theta$
4. Let $\mathcal{D}$ be some smooth perturbation distribution on $\mathbb{R}^d$

In *expectation*, Follow the Perturbed Leader described as:

$$\begin{aligned}
\text{FTPL:} \quad x_t &= \mathbb{E}_{Z \sim \mathcal{D}}[\arg\max_{x \in \mathcal{X}} \; x^\top (\Theta + Z)] \\
&= \mathbb{E}_{Z \sim \mathcal{D}}[\nabla \Phi_0(\Theta + Z)] \\
\text{(usually)} \quad &= \nabla \underbrace{\mathbb{E}_{Z \sim \mathcal{D}}[\Phi_0(\Theta + Z)]}_{\text{define as } \Phi_{\mathcal{D}}} \quad = \quad \nabla \Phi_{\mathcal{D}}(\Theta) \\
&= \arg\max_{x \in \mathcal{X}} \; x^\top \Theta - \Phi^*_{\mathcal{D}}(x)
\end{aligned}$$

In short, given dist $\mathcal{D}$, we can *replicate* FTPL by regularizing with Fenchel conjugate of $\Phi_{\mathcal{D}}(\Theta) = \mathbb{E}_{Z \sim \mathcal{D}}[\Phi_0(\Theta + Z)]$

Perturbing with the Gaussian is Cool!

It turns out that you get special properties when you perturb with a Gaussian. That is, letting $\mathcal{D} := N(\mathbf{0}, I)$ gives an "optimal algorithm" in a couple of cases.

The important lemma is this one:

Gaussian smoothing

For any differentiable function f we have

$$\mathbb{E}_{Z \sim N(0,1)}[\nabla f(Z) - Z^\top f(Z)] = 0$$

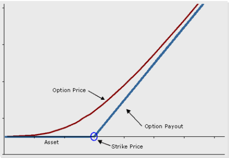
Derivative Hedging and a Minimax View of Black-Scholes

- AAPL will take a sequence of future price (multiplicative) fluctuations $\alpha_1, \ldots, \alpha_T \in (-1, \infty)$.

Derivative Hedging and a Minimax View of Black-Scholes

- AAPL will take a sequence of future price (multiplicative) fluctuations $\alpha_1, \ldots, \alpha_T \in (-1, \infty)$.
- You (investor) have sold a *derivative* on AAPL whose payoff is a function of the price fluct's, $g(\alpha_1, \ldots, \alpha_T)$. E.g., for a European Call Option:

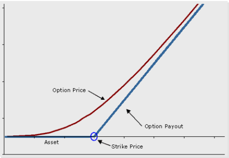
$$g(\alpha_1, \ldots, \alpha_T) = C \max(0, (1+\alpha_1) \times \cdots \times (1+\alpha_T) - D)$$



Derivative Hedging and a Minimax View of Black-Scholes

- AAPL will take a sequence of future price (multiplicative) fluctuations $\alpha_1, \ldots, \alpha_T \in (-1, \infty)$.
- You (investor) have sold a *derivative* on AAPL whose payoff is a function of the price fluct's, $g(\alpha_1, \ldots, \alpha_T)$. E.g., for a European Call Option:

$$g(\alpha_1, \ldots, \alpha_T) = C \max(0, (1+\alpha_1) \times \cdots \times (1+\alpha_T) - D)$$



- Can I *hedge* my exposure to this option?
- In finance terms: exists a trading strategy (on AAPL stock) which can "super replicate" the option?

Minimax Hedging

A *hedging strategy* is an online algorithm that selects a sequence of share purchases $\delta_1, \ldots, \delta_T \in \mathbb{R}$ (neg. means a short sale) with the goal of minimizing

$$g(\alpha_1, \ldots, \alpha_T) - \sum_{t=1}^{T} \delta_t \alpha_t$$

Minimax Hedging

A *hedging strategy* is an online algorithm that selects a sequence of share purchases $\delta_1, \ldots, \delta_T \in \mathbb{R}$ (neg. means a short sale) with the goal of minimizing

$$g(\alpha_1, \ldots, \alpha_T) - \sum_{t=1}^{T} \delta_t \alpha_t \quad \equiv \quad \text{HedgingRegret.}$$

The HedgingRegret is the gap in return between the option contract and the strategy.

Minimax Hedging

A *hedging strategy* is an online algorithm that selects a sequence of share purchases $\delta_1, \ldots, \delta_T \in \mathbb{R}$ (neg. means a short sale) with the goal of minimizing

$$g(\alpha_1, \ldots, \alpha_T) - \sum_{t=1}^{T} \delta_t \alpha_t \quad \equiv \quad \text{HedgingRegret}.$$

The HedgingRegret is the gap in return between the option contract and the strategy.

$$\text{Minimax Option Price} \equiv \inf_{\text{Hedge Algs}} \sup_{\alpha_{1:T} \in \mathcal{Z}} \text{HedgingRegret}$$

Relationship to Black-Scholes

In the 1970s, Black and Scholes utilized techniques from stochastic calculus to develop a theory of pricing options. Required assumption: the price moves according to *geometric Brownian motion*.

Relationship to Black-Scholes

In the 1970s, Black and Scholes utilized techniques from stochastic calculus to develop a theory of pricing options. Required assumption: the price moves according to *geometric Brownian motion*. In this case,

$$\text{B-S Option Price} = \mathop{\mathbb{E}}_{X \sim N(-\sigma/2, \sigma^2)} [g(\exp(X))]$$

Relationship to Black-Scholes

In the 1970s, Black and Scholes utilized techniques from stochastic calculus to develop a theory of pricing options. Required assumption: the price moves according to *geometric Brownian motion*. In this case,

$$\text{B-S Option Price} = \mathop{\mathbb{E}}_{X \sim N(-\sigma/2, \sigma^2)} [g(\exp(X))]$$

Alternatively:

Theorem [Abernethy, Frongillo, and Wibisono, 2012] [Abernethy, Bartlett, Frongillo, and Wibisono, 2013]

$$\text{Minimax Option Price} \to \mathop{\mathbb{E}}_{X \sim N(-\sigma/2, \sigma^2)} [g(\exp(X))]$$

as T (the hedging frequency) tends to ∞, and under *certain bounds on the price fluctuations*.

Black-Scholes as Random Playout?

In the Black-Scholes pricing formulation, the price of an option is determine according to a potential function $\Phi(S, t)$ where S is current price and t is time.

$$\Phi(S, t) := \mathop{\mathbb{E}}_{X \sim N(-\frac{1}{2}(T-t), T-t)} [g(S \exp(X))]$$

Black-Scholes as Random Playout?

In the Black-Scholes pricing formulation, the price of an option is determine according to a potential function $\Phi(S, t)$ where S is current price and t is time.

$$\Phi(S,t) := \underset{X \sim N(-\frac{1}{2}(T-t), T-t)}{\mathbb{E}} [g(S \exp(X))]$$

The (δ-)hedging strategy: buy $\frac{\partial \Phi(S,t)}{\partial S}$ shares of asset. (This is indeed asymptotically optimal)

Black-Scholes as Random Playout?

In the Black-Scholes pricing formulation, the price of an option is determine according to a potential function $\Phi(S, t)$ where S is current price and t is time.

$$\Phi(S,t) := \mathop{\mathbb{E}}_{X \sim N(-\frac{1}{2}(T-t), T-t)} [g(S \exp(X))]$$

The (δ-)hedging strategy: buy $\frac{\partial \Phi(S,t)}{\partial S}$ shares of asset. (This is indeed asymptotically optimal)

Random Playout Formulation:

1. Current price of asset is S
2. Sample random price future $X \sim N(-\frac{1}{2}(T-t), T-t)$
3. If "guessed" final price $S \exp(X)$ is above the strike price then *hedge* by buying 1 share, otherwise no hedge.

In other words: δ-hedging is random playout!

THANK YOU

Jacob Abernethy and Manfred K Warmuth. Repeated games against budgeted adversaries. In *(NIPS) Advances in Neural Information Processing Systems*, pages 1–9, 2010.

Jacob Abernethy, Manfred K Warmuth, and Joel Yellin. Optimal strategies from random walks. In *(COLT) Conference on Learning Theory*, 2008.

Jacob Abernethy, Rafael M Frongillo, and Andre Wibisono. Minimax option pricing meets black-scholes in the limit. In *(STOC) Proceedings of the 44th Symposium on Theory of Computing*, pages 1029–1040. ACM, 2012.

Jacob Abernethy, Peter Bartlett, Rafael Frongillo, and Andre Wibisono. How to hedge an option against an adversary: Black-scholes pricing is minimax optimal. In *(NIPS) Advances in Neural Information Processing Systems*, pages 2346–2354, 2013.

Jacob Abernethy, Chansoo Lee, Abhinav Sinha, and Ambuj Tewari. Online linear optimization via smoothing. In *(COLT) Conference on Learning Theory*, 2014.