Relaxing the I.I.D. Assumption

Adaptively Minimax Optimal Regret via Root-Entropic Regularization

Blair Bilodeau[*,†,1], Jeffrey Negrea[*,1], Daniel M. Roy[†,1]

March 30, 2021

UCLA Computer Science Weekly Seminar

[*] Equal Contribution

[†] Presenting

[1] University of Toronto and Vector Institute

Background

A Simple Example

Trading Stocks

A Simple Example

Trading Stocks

- You need to invest your money into a stock portfolio.

A Simple Example

Trading Stocks

- You need to invest your money into a stock portfolio.
- You have access to several market experts that give you advice.

A Simple Example

Trading Stocks

- You need to invest your money into a stock portfolio.
- You have access to several market experts that give you advice.
- You *regret* not having always followed the *post hoc* best expert's advice

A Simple Example

Trading Stocks

- You need to invest your money into a stock portfolio.
- You have access to several market experts that give you advice.
- You *regret* not having always followed the *post hoc* best expert's advice

What assumptions should we make?

A Simple Example

Trading Stocks

- You need to invest your money into a stock portfolio.
- You have access to several market experts that give you advice.
- You *regret* not having always followed the *post hoc* best expert's advice

What assumptions should we make?

A simplifying assumption is that the data are I.I.D. (e.g., Black–Scholes–Merton)

A Simple Example

Trading Stocks

- You need to invest your money into a stock portfolio.
- You have access to several market experts that give you advice.
- You *regret* not having always followed the *post hoc* best expert's advice

What assumptions should we make?

A simplifying assumption is that the data are I.I.D. (e.g., Black–Scholes–Merton)

In real life, market is driven in part by non-stochastic forces.

A Simple Example

Trading Stocks

- You need to invest your money into a stock portfolio.
- You have access to several market experts that give you advice.
- You *regret* not having always followed the *post hoc* best expert's advice

What assumptions should we make?

A simplifying assumption is that the data are I.I.D. (e.g., Black–Scholes–Merton)

In real life, market is driven in part by non-stochastic forces.

Is assuming adversarial data too pessimistic?

A Simple Example

Trading Stocks

- You need to invest your money into a stock portfolio.
- You have access to several market experts that give you advice.
- You *regret* not having always followed the *post hoc* best expert's advice

What assumptions should we make?

A simplifying assumption is that the data are I.I.D. (e.g., Black–Scholes–Merton)

In real life, market is driven in part by non-stochastic forces.

Is assuming adversarial data too pessimistic?

Is the departure from I.I.D.-ness benign? How can we quantify that?

A Simple Example

Trading Stocks

- You need to invest your money into a stock portfolio.
- You have access to several market experts that give you advice.
- You *regret* not having always followed the *post hoc* best expert's advice

What assumptions should we make?

A simplifying assumption is that the data are I.I.D. (e.g., Black–Scholes–Merton)

In real life, market is driven in part by non-stochastic forces.

Is assuming adversarial data too pessimistic?

Is the departure from I.I.D.-ness benign? How can we quantify that?
Influence of non-stochastic forces "small" $\Rightarrow$ maybe.

A Simple Example

Trading Stocks

- You need to invest your money into a stock portfolio.
- You have access to several market experts that give you advice.
- You *regret* not having always followed the *post hoc* best expert's advice

What assumptions should we make?

A simplifying assumption is that the data are I.I.D. (e.g., Black–Scholes–Merton)

In real life, market is driven in part by non-stochastic forces.

Is assuming adversarial data too pessimistic?

Is the departure from I.I.D.-ness benign? How can we quantify that?
  Influence of non-stochastic forces "small" $\Rightarrow$ maybe.
  Meaning of "small" TBD.

A Simple Example

Trading Stocks

- You need to invest your money into a stock portfolio.
- You have access to several market experts that give you advice.
- You *regret* not having always followed the *post hoc* best expert's advice

What assumptions should we make?

A simplifying assumption is that the data are I.I.D. (e.g., Black–Scholes–Merton)

In real life, market is driven in part by non-stochastic forces.

Is assuming adversarial data too pessimistic?

Is the departure from I.I.D.-ness benign? How can we quantify that?
Influence of non-stochastic forces "small" $\Rightarrow$ maybe.
Meaning of "small" TBD.

Want to maximize profit without having to know what drives the market.

A Simple Example

Trading Stocks

- You need to invest your money into a stock portfolio.
- You have access to several market experts that give you advice.
- You *regret* not having always followed the *post hoc* best expert's advice

What assumptions should we make?

A simplifying assumption is that the data are I.I.D. (e.g., Black–Scholes–Merton)

In real life, market is driven in part by non-stochastic forces.

Is assuming adversarial data too pessimistic?

Is the departure from I.I.D.-ness benign? How can we quantify that?
Influence of non-stochastic forces "small" $\Rightarrow$ maybe.
Meaning of "small" TBD.

Want to maximize profit without having to know what drives the market.

Sequential Prediction with Expert Advice

Sequential Prediction a.k.a. Online Learning

Sequential Prediction with Expert Advice

Sequential Prediction a.k.a. Online Learning

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

Sequential Prediction with Expert Advice

Sequential Prediction a.k.a. Online Learning

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

- Predict $\hat{y}(t) \in \hat{\mathcal{Y}}$ based on historical data before time t

Sequential Prediction with Expert Advice

Sequential Prediction a.k.a. Online Learning

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

- Predict $\hat{y}(t) \in \hat{\mathcal{Y}}$ based on historical data before time t
- Observe $y(t) \in \mathcal{Y}$ from the environment

Sequential Prediction with Expert Advice

Sequential Prediction a.k.a. Online Learning

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

- Predict $\hat{y}(t) \in \hat{\mathcal{Y}}$ based on historical data before time t
- Observe $y(t) \in \mathcal{Y}$ from the environment
- Incur loss $\ell(\hat{y}(t), y(t))$

Sequential Prediction with Expert Advice

Sequential Prediction with Expert Advice

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

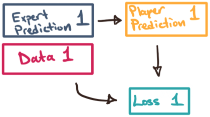
- Receive $x(t) = (x_1(t), \ldots, x_N(t)) \subseteq \hat{\mathcal{Y}}$ expert predictions
- Predict $\hat{y}(t) \in \hat{\mathcal{Y}}$ based on historical data before time t and expert predictions
- Observe $y(t) \in \mathcal{Y}$ from the environment
- Incur loss $\ell(\hat{y}(t), y(t))$

Sequential Prediction with Expert Advice

Sequential Prediction with Expert Advice

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

- Receive $x(t) = (x_1(t), \ldots, x_N(t)) \subseteq \hat{\mathcal{Y}}$ expert predictions
- Predict $\hat{y}(t) \in \hat{\mathcal{Y}}$ based on historical data before time t and expert predictions
- Observe $y(t) \in \mathcal{Y}$ from the environment
- Incur loss $\ell(\hat{y}(t), y(t))$

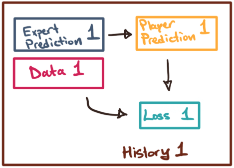


Sequential Prediction with Expert Advice

Sequential Prediction with Expert Advice

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

- Receive $x(t) = (x_1(t), \ldots, x_N(t)) \subseteq \hat{\mathcal{Y}}$ expert predictions
- Predict $\hat{y}(t) \in \hat{\mathcal{Y}}$ based on historical data before time t and expert predictions
- Observe $y(t) \in \mathcal{Y}$ from the environment
- Incur loss $\ell(\hat{y}(t), y(t))$

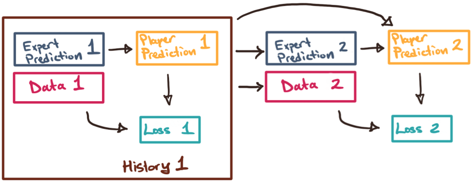


Sequential Prediction with Expert Advice

Sequential Prediction with Expert Advice

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

- Receive $x(t) = (x_1(t), \ldots, x_N(t)) \subseteq \hat{\mathcal{Y}}$ expert predictions
- Predict $\hat{y}(t) \in \hat{\mathcal{Y}}$ based on historical data before time t and expert predictions
- Observe $y(t) \in \mathcal{Y}$ from the environment
- Incur loss $\ell(\hat{y}(t), y(t))$

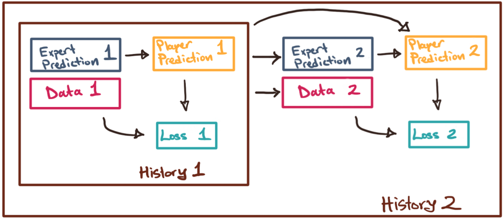


Sequential Prediction with Expert Advice

Sequential Prediction with Expert Advice

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

- Receive $x(t) = (x_1(t), \ldots, x_N(t)) \subseteq \hat{\mathcal{Y}}$ expert predictions
- Predict $\hat{y}(t) \in \hat{\mathcal{Y}}$ based on historical data before time t and expert predictions
- Observe $y(t) \in \mathcal{Y}$ from the environment
- Incur loss $\ell(\hat{y}(t), y(t))$

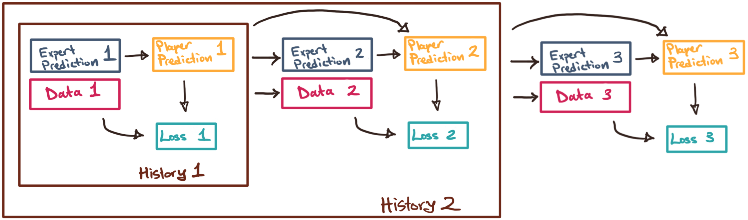


Sequential Prediction with Expert Advice

Sequential Prediction with Expert Advice

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

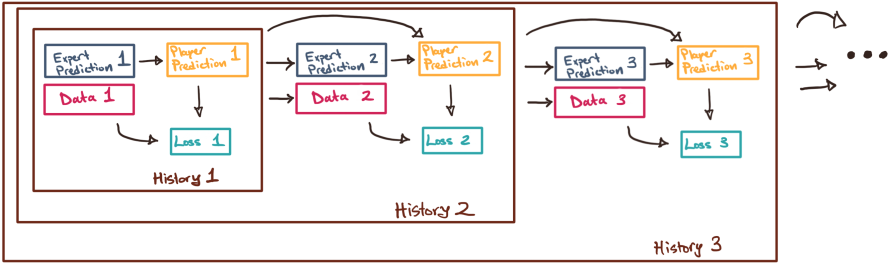
- Receive $x(t) = (x_1(t), \ldots, x_N(t)) \subseteq \hat{\mathcal{Y}}$ expert predictions
- Predict $\hat{y}(t) \in \hat{\mathcal{Y}}$ based on historical data before time t and expert predictions
- Observe $y(t) \in \mathcal{Y}$ from the environment
- Incur loss $\ell(\hat{y}(t), y(t))$



Sequential Prediction with Expert Advice

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

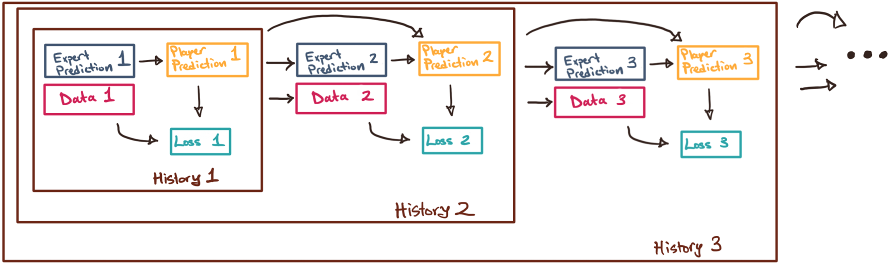
- Receive $x(t) = (x_1(t), \ldots, x_N(t)) \subseteq \hat{\mathcal{Y}}$ expert predictions
- Predict $\hat{y}(t) \in \hat{\mathcal{Y}}$ based on historical data before time t and expert predictions
- Observe $y(t) \in \mathcal{Y}$ from the environment
- Incur loss $\ell(\hat{y}(t), y(t))$



Sequential Prediction with Expert Advice

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

- Receive $x(t) = (x_1(t), \ldots, x_N(t)) \subseteq \hat{\mathcal{Y}}$ expert predictions
- Predict $\hat{y}(t) \in \hat{\mathcal{Y}}$ based on historical data before time t and expert predictions
- Observe $y(t) \in \mathcal{Y}$ from the environment
- Incur loss $\ell(\hat{y}(t), y(t))$



Measuring Performance

The measure of the player's performance is...

Measuring Performance

The measure of the player's performance is...

- Relative to the class of N reference *experts*;

Measuring Performance

The measure of the player's performance is...

- Relative to the class of N reference *experts*;
- Given by the excess cumulative loss of the player over the best expert;

Measuring Performance

The measure of the player's performance is...

- Relative to the class of N reference *experts*;
- Given by the excess cumulative loss of the player over the best expert;

$$\textbf{Regret:} \quad R(T) = \sum_{t=1}^{T} \ell(\hat{y}(t), y(t)) - \min_{i \in [N]} \sum_{t=1}^{T} \ell(x_i(t), y(t))$$

Measuring Performance

The measure of the player's performance is...

- Relative to the class of N reference *experts*;
- Given by the excess cumulative loss of the player over the best expert;

$$\textbf{Regret:} \quad R(T) = \sum_{t=1}^{T} \ell(\hat{y}(t), y(t)) - \min_{i \in [N]} \sum_{t=1}^{T} \ell(x_i(t), y(t))$$

The prediction problem is *online learnable* if a player can incur sub-linear regret:

$$\mathbb{E}R(T) \in o(T).$$

Measuring Performance

The measure of the player's performance is...

- Relative to the class of N reference *experts*;
- Given by the excess cumulative loss of the player over the best expert;

$$\textbf{Regret:} \quad R(T) = \sum_{t=1}^{T} \ell(\hat{y}(t), y(t)) - \min_{i \in [N]} \sum_{t=1}^{T} \ell(x_i(t), y(t))$$

The prediction problem is *online learnable* if a player can incur sub-linear regret:

$$\mathbb{E}R(T) \in o(T).$$

Where the $\mathbb{E}$ is taken with respect to the randomness in the player's and expert's predictions, and the data-generating mechanism for $(y(t))_{t \in \mathbb{N}}$.

Measuring Performance

The measure of the player's performance is...

- Relative to the class of N reference *experts*;
- Given by the excess cumulative loss of the player over the best expert;

$$\textbf{Regret:} \quad R(T) = \sum_{t=1}^{T} \ell(\hat{y}(t), y(t)) - \min_{i \in [N]} \sum_{t=1}^{T} \ell(x_i(t), y(t))$$

The prediction problem is *online learnable* if a player can incur sub-linear regret:

$$\mathbb{E} R(T) \in o(T).$$

Where the $\mathbb{E}$ is taken with respect to the randomness in the player's and expert's predictions, and the data-generating mechanism for $(y(t))_{t \in \mathbb{N}}$.

(The $\mathbb{E}$ may be under a complicated, non-I.I.D. measure.)

Measuring Performance

The measure of the player's performance is...

- Relative to the class of N reference *experts*;
- Given by the excess cumulative loss of the player over the best expert;

$$\textbf{Regret:} \quad R(T) = \sum_{t=1}^{T} \ell(\hat{y}(t), y(t)) - \min_{i \in [N]} \sum_{t=1}^{T} \ell(x_i(t), y(t))$$

The prediction problem is *online learnable* if a player can incur sub-linear regret:

$$\mathbb{E}R(T) \in o(T).$$

Where the $\mathbb{E}$ is taken with respect to the randomness in the player's and expert's predictions, and the data-generating mechanism for $(y(t))_{t \in \mathbb{N}}$.

(The $\mathbb{E}$ may be under a complicated, non-I.I.D. measure.)

Optimality in the Stochastic and Adversarial Regimes

Optimality in the Stochastic and Adversarial Regimes

Stochastic-with-a-Gap

Optimality in the Stochastic and Adversarial Regimes

Stochastic-with-a-Gap

- Expert predictions and data are I.I.D. over time from some distribution.
- There is an expert whose mean loss is Δ smaller than the others.

Optimality in the Stochastic and Adversarial Regimes

Stochastic-with-a-Gap

- Expert predictions and data are I.I.D. over time from some distribution.
- There is an expert whose mean loss is Δ smaller than the others.

Theorem (Gaillard et al. 2014 + Mourtada and Gaïffas 2019)

A constructive algorithm achieves the minimax regret:

$\mathbb{E}R(T) \asymp \frac{\log N}{\Delta}$, uniformly bounded in T.

Optimality in the Stochastic and Adversarial Regimes

Stochastic-with-a-Gap

- Expert predictions and data are I.I.D. over time from some distribution.
- There is an expert whose mean loss is Δ smaller than the others.

Theorem (Gaillard et al. 2014 + Mourtada and Gaïffas 2019)

A constructive algorithm achieves the minimax regret:

$\mathbb{E}R(T) \asymp \frac{\log N}{\Delta}$, uniformly bounded in T.

Adversarial

- Compete against expert predictions and data that maximize $R(T)$.

Optimality in the Stochastic and Adversarial Regimes

Stochastic-with-a-Gap

- Expert predictions and data are I.I.D. over time from some distribution.
- There is an expert whose mean loss is Δ smaller than the others.

Theorem (Gaillard et al. 2014 + Mourtada and Gaïffas 2019)

A constructive algorithm achieves the minimax regret:

$\mathbb{E}R(T) \asymp \frac{\log N}{\Delta}$, uniformly bounded in T.

Adversarial

- Compete against expert predictions and data that maximize $R(T)$.

Theorem (Vovk 1998, see also [FS97; CL06])

A constructive algorithm achieves the minimax regret:

$\mathbb{E}R(T) \asymp \sqrt{T \log N}$ for all T.

Optimality in the Stochastic and Adversarial Regimes

Stochastic-with-a-Gap

- Expert predictions and data are I.I.D. over time from some distribution.
- There is an expert whose mean loss is Δ smaller than the others.

Theorem (Gaillard et al. 2014 + Mourtada and Gaïffas 2019)

A constructive algorithm achieves the minimax regret:

$\mathbb{E}R(T) \asymp \frac{\log N}{\Delta}$, uniformly bounded in T.

Adversarial

- Compete against expert predictions and data that maximize $R(T)$.

Theorem (Vovk 1998, see also [FS97; CL06])

A constructive algorithm achieves the minimax regret:

$\mathbb{E}R(T) \asymp \sqrt{T \log N}$ for all T.

Can a single algorithm be optimal in both settings simultaneously?

Optimality in the Stochastic and Adversarial Regimes

Stochastic-with-a-Gap

- Expert predictions and data are I.I.D. over time from some distribution.
- There is an expert whose mean loss is Δ smaller than the others.

Theorem (Gaillard et al. 2014 + Mourtada and Gaïffas 2019)

A constructive algorithm achieves the minimax regret:

$\mathbb{E}R(T) \asymp \frac{\log N}{\Delta}$, uniformly bounded in T.

Adversarial

- Compete against expert predictions and data that maximize $R(T)$.

Theorem (Vovk 1998, see also [FS97; CL06])

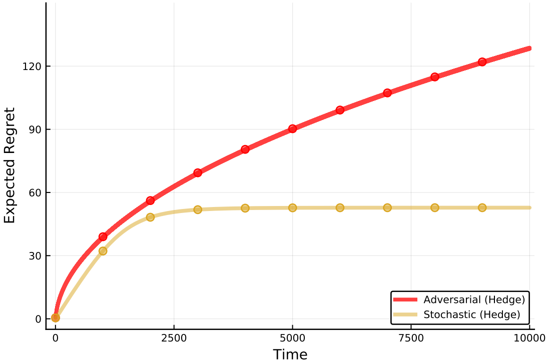
A constructive algorithm achieves the minimax regret:

$\mathbb{E}R(T) \asymp \sqrt{T \log N}$ for all T.

Can a single algorithm be optimal in both settings simultaneously?

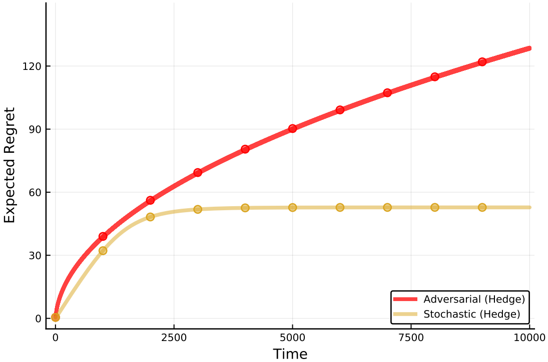
Simultaneous Optimality of Hedge

Can a single algorithm be optimal in both settings simultaneously? Yes!



Simultaneous Optimality of Hedge

Can a single algorithm be optimal in both settings simultaneously? Yes!



Proposition: [MG19]

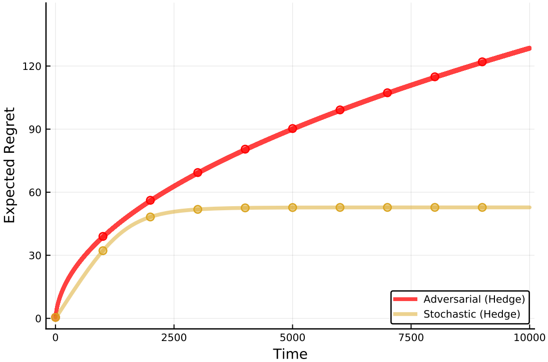
Hedge is optimal in both the stochastic and adversarial settings.

Stochastic: $\mathbb{E}R(T) \asymp (\log N)/\Delta$ uniformly in T.

Adversarial: $\mathbb{E}R(T) \asymp \sqrt{T \log N}$.

Simultaneous Optimality of Hedge

Can a single algorithm be optimal in both settings simultaneously? Yes!



Proposition: [MG19]

Hedge is optimal in both the stochastic and adversarial settings.

Stochastic: $\mathbb{E}R(T) \asymp (\log N)/\Delta$ uniformly in T.

Adversarial: $\mathbb{E}R(T) \asymp \sqrt{T \log N}$.

Beyond Stochastic and Adversarial

Real data $\neq$ stochastic.

Beyond Stochastic and Adversarial

Real data $\neq$ stochastic. $\leftarrow$ Too optimistic.

Beyond Stochastic and Adversarial

Real data $\neq$ stochastic. $\leftarrow$ Too optimistic.

Real data $\neq$ adversarial.

Beyond Stochastic and Adversarial

Real data $\neq$ stochastic. $\leftarrow$ Too optimistic.

Real data $\neq$ adversarial. $\leftarrow$ Too pessimistic.

Beyond Stochastic and Adversarial

Real data $\neq$ stochastic. $\leftarrow$ Too optimistic.

Real data $\neq$ adversarial. $\leftarrow$ Too pessimistic.

Building upon [RST11], we study a spectrum between stochastic and adversarial.

Beyond Stochastic and Adversarial

Real data $\neq$ stochastic. $\leftarrow$ Too optimistic.

Real data $\neq$ adversarial. $\leftarrow$ Too pessimistic.

Building upon [RST11], we study a spectrum between stochastic and adversarial.

Intuitively, fix a "neighbourhood" of distributions.

Beyond Stochastic and Adversarial

Real data $\neq$ stochastic. $\leftarrow$ Too optimistic.

Real data $\neq$ adversarial. $\leftarrow$ Too pessimistic.

Building upon [RST11], we study a spectrum between stochastic and adversarial.

Intuitively, fix a "neighbourhood" of distributions.

Each data point drawn from an arbitrary distribution in "neighbourhood".

Beyond Stochastic and Adversarial

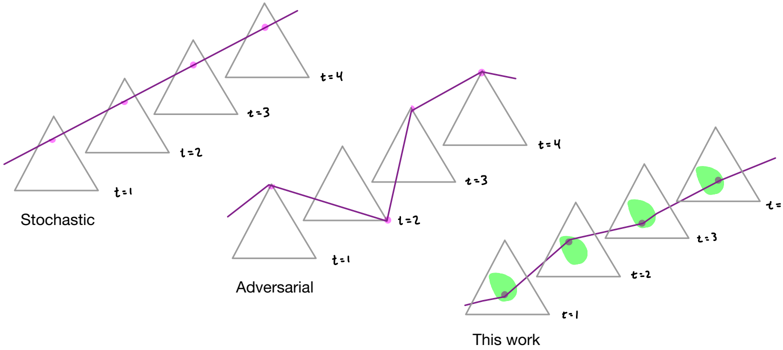
Real data $\neq$ stochastic. $\leftarrow$ Too optimistic.

Real data $\neq$ adversarial. $\leftarrow$ Too pessimistic.

Building upon [RST11], we study a spectrum between stochastic and adversarial.

Intuitively, fix a "neighbourhood" of distributions.

Each data point drawn from an arbitrary distribution in "neighbourhood".



Beyond Stochastic and Adversarial

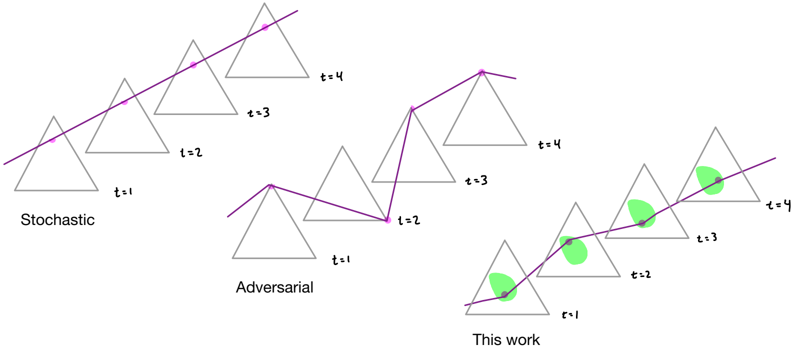
Real data $\neq$ stochastic. $\leftarrow$ Too optimistic.

Real data $\neq$ adversarial. $\leftarrow$ Too pessimistic.

Building upon [RST11], we study a spectrum between stochastic and adversarial.

Intuitively, fix a "neighbourhood" of distributions.

Each data point drawn from an arbitrary distribution in "neighbourhood".



Adaptively Minimax Optimal Algorithms

Algorithms should be robust to a spectrum of data-generating mechanisms.

Adaptively Minimax Optimal Algorithms

Algorithms should be robust to a spectrum of data-generating mechanisms.

Definition BNR20

An algorithm is **adaptively minimax optimal** for a spectrum of settings if:

- it achieves the minimax optimal performance in each setting; and
- it does not require knowledge of the true setting in advance.

Adaptively Minimax Optimal Algorithms

Algorithms should be robust to a spectrum of data-generating mechanisms.

Definition BNR20

An algorithm is **adaptively minimax optimal** for a spectrum of settings if:

- it achieves the minimax optimal performance in each setting; and
- it does not require knowledge of the true setting in advance.

How to formalize this?

Adaptively Minimax Optimal Algorithms

Algorithms should be robust to a spectrum of data-generating mechanisms.

Definition BNR20

An algorithm is **adaptively minimax optimal** for a spectrum of settings if:

- it achieves the minimax optimal performance in each setting; and
- it does not require knowledge of the true setting in advance.

How to formalize this?

- Given a spectrum of settings Λ...

Adaptively Minimax Optimal Algorithms

Algorithms should be robust to a spectrum of data-generating mechanisms.

Definition BNR20

An algorithm is **adaptively minimax optimal** for a spectrum of settings if:

- it achieves the minimax optimal performance in each setting; and
- it does not require knowledge of the true setting in advance.

How to formalize this?

- Given a spectrum of settings Λ...
- If we know $\theta \in \Lambda$ in advance, the best achievable performance is $R^*_\theta(T)$.

Adaptively Minimax Optimal Algorithms

Algorithms should be robust to a spectrum of data-generating mechanisms.

Definition BNR20

An algorithm is **adaptively minimax optimal** for a spectrum of settings if:

- it achieves the minimax optimal performance in each setting; and
- it does not require knowledge of the true setting in advance.

How to formalize this?

- Given a spectrum of settings Λ...
- If we know $\theta \in \Lambda$ in advance, the best achievable performance is $R^*_\theta(T)$.
- We want an algorithm that does as well as possible without knowing θ:

$$R_\theta(T) \leq C\, R^*_\theta(T) \text{ uniformly in } \theta \text{ for large enough } T.$$

Adaptively Minimax Optimal Algorithms

Algorithms should be robust to a spectrum of data-generating mechanisms.

Definition BNR20

An algorithm is **adaptively minimax optimal** for a spectrum of settings if:

- it achieves the minimax optimal performance in each setting; and
- it does not require knowledge of the true setting in advance.

How to formalize this?

- Given a spectrum of settings Λ...
- If we know $\theta \in \Lambda$ in advance, the best achievable performance is $R^*_\theta(T)$.
- We want an algorithm that does as well as possible without knowing θ:

$$R_\theta(T) \leq C\, R^*_\theta(T) \text{ uniformly in } \theta \text{ for large enough } T.$$

Overview of Our Work

Overview of Our Work

We show Hedge is suboptimal between Stochastic and Adversarial.

Overview of Our Work

We show Hedge is suboptimal between Stochastic and Adversarial.

This was surprising for us.

Overview of Our Work

We show Hedge is suboptimal between Stochastic and Adversarial.

- This was surprising for us.
- We initially set out hoping to prove that Hedge was adaptive to all scenarios.

Overview of Our Work

We show Hedge is suboptimal between Stochastic and Adversarial.

This was surprising for us.

We initially set out hoping to prove that Hedge was adaptive to all scenarios.

Theorem BNR20

Without oracle knowledge to tune the learning rate, Hedge is not simultaneously minimax optimal at all settings between stochastic-with-a-gap and adversarial.

Overview of Our Work

We show Hedge is suboptimal between Stochastic and Adversarial.

This was surprising for us.

We initially set out hoping to prove that Hedge was adaptive to all scenarios.

Theorem BNR20

Without oracle knowledge to tune the learning rate, Hedge is not simultaneously minimax optimal at all settings between stochastic-with-a-gap and adversarial.

We provide a new algorithm that achieves the minimax rate in all settings...

Overview of Our Work

We show Hedge is suboptimal between Stochastic and Adversarial.

This was surprising for us.

We initially set out hoping to prove that Hedge was adaptive to all scenarios.

Theorem BNR20

Without oracle knowledge to tune the learning rate, Hedge is not simultaneously minimax optimal at all settings between stochastic-with-a-gap and adversarial.

We provide a new algorithm that achieves the minimax rate in all settings...

...without knowledge of which setting prevails!

Overview of Our Work

We show Hedge is suboptimal between Stochastic and Adversarial.

This was surprising for us.

We initially set out hoping to prove that Hedge was adaptive to all scenarios.

Theorem BNR20

Without oracle knowledge to tune the learning rate, Hedge is not simultaneously minimax optimal at all settings between stochastic-with-a-gap and adversarial.

We provide a new algorithm that achieves the minimax rate in all settings...

...without knowledge of which setting prevails!

Theorem BNR20

There is an adaptively minimax optimal algorithm: Meta-CARE.

Overview of Our Work

We show Hedge is suboptimal between Stochastic and Adversarial.

This was surprising for us.

We initially set out hoping to prove that Hedge was adaptive to all scenarios.

Theorem BNR20

Without oracle knowledge to tune the learning rate, Hedge is not simultaneously minimax optimal at all settings between stochastic-with-a-gap and adversarial.

We provide a new algorithm that achieves the minimax rate in all settings...

...without knowledge of which setting prevails!

Theorem BNR20

There is an adaptively minimax optimal algorithm: Meta-CARE.

Main Result

Motivating Intuition

Main Result

Motivating Intuition

- In the adversarial case the minimax optimal regret is $\Theta(\sqrt{T \log N})$

Main Result

Motivating Intuition

- In the adversarial case the minimax optimal regret is $\Theta(\sqrt{T \log N})$

- If we know only N_0 of the experts can ever be "the best", and which ones, ...

Main Result

Motivating Intuition

- In the adversarial case the minimax optimal regret is $\Theta(\sqrt{T \log N})$
- If we know only N_0 of the experts can ever be "the best", and which ones, ...
  - we could restrict an adversarially optimal algorithm to the "best experts"

Main Result

Motivating Intuition

- In the adversarial case the minimax optimal regret is $\Theta(\sqrt{T \log N})$
- If we know only N_0 of the experts can ever be "the best", and which ones, ...
  - we could restrict an adversarially optimal algorithm to the "best experts"
  - so we might strive to have regret $\Theta(\sqrt{T \log N_0})$ in (T, N_0)

Main Result

Motivating Intuition

- In the adversarial case the minimax optimal regret is $\Theta(\sqrt{T \log N})$
- If we know only N_0 of the experts can ever be "the best", and which ones, ...
  - we could restrict an adversarially optimal algorithm to the "best experts"
  - so we might strive to have regret $\Theta(\sqrt{T \log N_0})$ in (T, N_0)
- If we know *one* expert is better than the rest by Δ_0, but not which it is...

Main Result

Motivating Intuition

- In the adversarial case the minimax optimal regret is $\Theta(\sqrt{T \log N})$
- If we know only N_0 of the experts can ever be "the best", and which ones, ...
    - we could restrict an adversarially optimal algorithm to the "best experts"
    - so we might strive to have regret $\Theta(\sqrt{T \log N_0})$ in (T, N_0)
- If we know *one* expert is better than the rest by Δ_0, but not which it is...
    - then we are *almost* in the stochastic-with-a-gap case

Main Result

Motivating Intuition

- In the adversarial case the minimax optimal regret is $\Theta(\sqrt{T \log N})$
- If we know only N_0 of the experts can ever be "the best", and which ones, …
  - we could restrict an adversarially optimal algorithm to the "best experts"
  - so we might strive to have regret $\Theta(\sqrt{T \log N_0})$ in (T, N_0)
- If we know *one* expert is better than the rest by Δ_0, but not which it is…
  - then we are *almost* in the stochastic-with-a-gap case
  - so we might hope for regret $\Theta((\log N)/\Delta_0)$

Main Result

Motivating Intuition

- In the adversarial case the minimax optimal regret is $\Theta(\sqrt{T \log N})$
- If we know only N_0 of the experts can ever be "the best", and which ones, ...
  - we could restrict an adversarially optimal algorithm to the "best experts"
  - so we might strive to have regret $\Theta(\sqrt{T \log N_0})$ in (T, N_0)
- If we know *one* expert is better than the rest by Δ_0, but not which it is...
  - then we are *almost* in the stochastic-with-a-gap case
  - so we might hope for regret $\Theta((\log N)/\Delta_0)$

Theorem BNR20

The adaptively minimax optimal rate of regret, which Meta-CARE achieves, is

$$\mathbb{E}R(T) \asymp \begin{cases} \sqrt{T \log N_0} & N_0 \geq 2 \\ (\log N)/\Delta_0 & N_0 = 1. \end{cases}$$

Main Result

Motivating Intuition

- In the adversarial case the minimax optimal regret is $\Theta(\sqrt{T \log N})$
- If we know only N_0 of the experts can ever be "the best", and which ones, ...
  - we could restrict an adversarially optimal algorithm to the "best experts"
  - so we might strive to have regret $\Theta(\sqrt{T \log N_0})$ in (T, N_0)
- If we know *one* expert is better than the rest by Δ_0, but not which it is...
  - then we are *almost* in the stochastic-with-a-gap case
  - so we might hope for regret $\Theta((\log N)/\Delta_0)$

Theorem BNR20

The adaptively minimax optimal rate of regret, which Meta-CARE achieves, is

$$\mathbb{E}R(T) \asymp \begin{cases} \sqrt{T \log N_0} & N_0 \geq 2 \\ (\log N)/\Delta_0 & N_0 = 1. \end{cases}$$

Relaxing the I.I.D. Assumption

Our Setting: Time-Homogeneous Convex Constraints

Our Setting: Time-Homogeneous Convex Constraints

Intuition

Experts and environment may collude.

Our Setting: Time-Homogeneous Convex Constraints

Intuition

Experts and environment may collude.

Realizations $(x(t), y(t))$ are sampled from an adversarial conditional distribution.

Our Setting: Time-Homogeneous Convex Constraints

Intuition

Experts and environment may collude.

Realizations $(x(t), y(t))$ are sampled from an adversarial conditional distribution.

Formal Framework

- Fix a convex set of distributions $\mathcal{D} \subseteq \mathcal{M}(\hat{\mathcal{Y}}^N \times \mathcal{Y})$.

Our Setting: Time-Homogeneous Convex Constraints

Intuition

Experts and environment may collude.

Realizations $(x(t), y(t))$ are sampled from an adversarial conditional distribution.

Formal Framework

- Fix a convex set of distributions $\mathcal{D} \subseteq \mathcal{M}(\hat{\mathcal{Y}}^N \times \mathcal{Y})$.
- $(x(t), y(t))$ drawn from an element of $\mathcal{D}$ given the history prior to t.
  - The choice of distribution is made based on outcomes of the previous rounds.

Our Setting: Time-Homogeneous Convex Constraints

Intuition

Experts and environment may collude.

Realizations $(x(t), y(t))$ are sampled from an adversarial conditional distribution.

Formal Framework

- Fix a convex set of distributions $\mathcal{D} \subseteq \mathcal{M}(\hat{\mathcal{Y}}^N \times \mathcal{Y})$.
- $(x(t), y(t))$ drawn from an element of $\mathcal{D}$ given the history prior to t.
  - The choice of distribution is made based on outcomes of the previous rounds.

More Details

- Time-Homogeneous: $\mathcal{D}$ does not depend on t.

Our Setting: Time-Homogeneous Convex Constraints

Intuition

Experts and environment may collude.

Realizations $(x(t), y(t))$ are sampled from an adversarial conditional distribution.

Formal Framework

- Fix a convex set of distributions $\mathcal{D} \subseteq \mathcal{M}(\hat{\mathcal{Y}}^N \times \mathcal{Y})$.
- $(x(t), y(t))$ drawn from an element of $\mathcal{D}$ given the history prior to t.
  - The choice of distribution is made based on outcomes of the previous rounds.

More Details

- Time-Homogeneous: $\mathcal{D}$ does not depend on t.
- Convex: environment can flip a coin to select between basic elements of $\mathcal{D}$.

Our Setting: Time-Homogeneous Convex Constraints

Intuition

Experts and environment may collude.

Realizations $(x(t), y(t))$ are sampled from an adversarial conditional distribution.

Formal Framework

- Fix a convex set of distributions $\mathcal{D} \subseteq \mathcal{M}(\hat{\mathcal{Y}}^N \times \mathcal{Y})$.
- $(x(t), y(t))$ drawn from an element of $\mathcal{D}$ given the history prior to t.
  - The choice of distribution is made based on outcomes of the previous rounds.

More Details

- Time-Homogeneous: $\mathcal{D}$ does not depend on t.
- Convex: environment can flip a coin to select between basic elements of $\mathcal{D}$.
- Environment may aim to maximize regret subject to the constraint.

Our Setting: Time-Homogeneous Convex Constraints

Intuition

Experts and environment may collude.

Realizations $(x(t), y(t))$ are sampled from an adversarial conditional distribution.

Formal Framework

- Fix a convex set of distributions $\mathcal{D} \subseteq \mathcal{M}(\hat{\mathcal{Y}}^N \times \mathcal{Y})$.
- $(x(t), y(t))$ drawn from an element of $\mathcal{D}$ given the history prior to t.
  - The choice of distribution is made based on outcomes of the previous rounds.

More Details

- Time-Homogeneous: $\mathcal{D}$ does not depend on t.
- Convex: environment can flip a coin to select between basic elements of $\mathcal{D}$.
- Environment may aim to maximize regret subject to the constraint.

Constraint-Characterizing Quantities

The set $\mathcal{D}$ may be complex and difficult to describe for a particular application.

Constraint-Characterizing Quantities

The set $\mathcal{D}$ may be complex and difficult to describe for a particular application.

We want to characterize the constraint using quantities that:

Constraint-Characterizing Quantities

The set $\mathcal{D}$ may be complex and difficult to describe for a particular application.

We want to characterize the constraint using quantities that:

- simplify the abstract complexity of the constraint;

Constraint-Characterizing Quantities

The set $\mathcal{D}$ may be complex and difficult to describe for a particular application.

We want to characterize the constraint using quantities that:

- simplify the abstract complexity of the constraint;
- differentiate whether the data is "easy" or not, independent of algorithms;

Constraint-Characterizing Quantities

The set $\mathcal{D}$ may be complex and difficult to describe for a particular application.

We want to characterize the constraint using quantities that:

- simplify the abstract complexity of the constraint;
- differentiate whether the data is "easy" or not, independent of algorithms;
- yield matching lower and upper bounds on regret.

Constraint-Characterizing Quantities

The set $\mathcal{D}$ may be complex and difficult to describe for a particular application.

We want to characterize the constraint using quantities that:

- simplify the abstract complexity of the constraint;
- differentiate whether the data is "easy" or not, independent of algorithms;
- yield matching lower and upper bounds on regret.

Effective Experts

Constraint-Characterizing Quantities

The set $\mathcal{D}$ may be complex and difficult to describe for a particular application.

We want to characterize the constraint using quantities that:

- simplify the abstract complexity of the constraint;
- differentiate whether the data is "easy" or not, independent of algorithms;
- yield matching lower and upper bounds on regret.

Effective Experts

$$\mathcal{I}_0 = \{\text{experts that are optimal in } \mathbb{E} \text{ for some } \mu \in \mathcal{D}\}$$

$$N_0 = |\mathcal{I}_0|$$

Constraint-Characterizing Quantities

The set $\mathcal{D}$ may be complex and difficult to describe for a particular application.

We want to characterize the constraint using quantities that:

- simplify the abstract complexity of the constraint;
- differentiate whether the data is "easy" or not, independent of algorithms;
- yield matching lower and upper bounds on regret.

Effective Experts

$$\mathcal{I}_0 = \{\text{experts that are optimal in } \mathbb{E} \text{ for some } \mu \in \mathcal{D}\}$$

$$N_0 = |\mathcal{I}_0|$$

Analogous to the single best expert in the stochastic-with-a-gap setting.

Constraint-Characterizing Quantities

The set $\mathcal{D}$ may be complex and difficult to describe for a particular application.

We want to characterize the constraint using quantities that:

- simplify the abstract complexity of the constraint;
- differentiate whether the data is "easy" or not, independent of algorithms;
- yield matching lower and upper bounds on regret.

Effective Experts

$$\mathcal{I}_0 = \{\text{experts that are optimal in } \mathbb{E} \text{ for some } \mu \in \mathcal{D}\}$$

$$N_0 = |\mathcal{I}_0|$$

Analogous to the single best expert in the stochastic-with-a-gap setting.

Effective Stochastic Gap

Constraint-Characterizing Quantities

The set $\mathcal{D}$ may be complex and difficult to describe for a particular application.

We want to characterize the constraint using quantities that:

- simplify the abstract complexity of the constraint;
- differentiate whether the data is "easy" or not, independent of algorithms;
- yield matching lower and upper bounds on regret.

Effective Experts

$$\mathcal{I}_0 = \{\text{experts that are optimal in } \mathbb{E} \text{ for some } \mu \in \mathcal{D}\}$$

$$N_0 = |\mathcal{I}_0|$$

Analogous to the single best expert in the stochastic-with-a-gap setting.

Effective Stochastic Gap

$$\Delta_0 = \inf_{\mu \in \mathcal{D}} \{\mu\text{-expected difference in loss of best expert and best expert not in } \mathcal{I}_0\}$$

Constraint-Characterizing Quantities

The set $\mathcal{D}$ may be complex and difficult to describe for a particular application.

We want to characterize the constraint using quantities that:

- simplify the abstract complexity of the constraint;
- differentiate whether the data is "easy" or not, independent of algorithms;
- yield matching lower and upper bounds on regret.

Effective Experts

$$\mathcal{I}_0 = \{\text{experts that are optimal in } \mathbb{E} \text{ for some } \mu \in \mathcal{D}\}$$
$$N_0 = |\mathcal{I}_0|$$

Analogous to the single best expert in the stochastic-with-a-gap setting.

Effective Stochastic Gap

$$\Delta_0 = \inf_{\mu \in \mathcal{D}} \{\mu\text{-expected difference in loss of best expert and best expert not in } \mathcal{I}_0\}$$

Analogous to the gap in the stochastic-with-a-gap setting.

Constraint-Characterizing Quantities

The set $\mathcal{D}$ may be complex and difficult to describe for a particular application.

We want to characterize the constraint using quantities that:

- simplify the abstract complexity of the constraint;
- differentiate whether the data is "easy" or not, independent of algorithms;
- yield matching lower and upper bounds on regret.

Effective Experts

$$\mathcal{I}_0 = \{\text{experts that are optimal in } \mathbb{E} \text{ for some } \mu \in \mathcal{D}\}$$

$$N_0 = |\mathcal{I}_0|$$

Analogous to the single best expert in the stochastic-with-a-gap setting.

Effective Stochastic Gap

$$\Delta_0 = \inf_{\mu \in \mathcal{D}} \{\mu\text{-expected difference in loss of best expert and best expert not in } \mathcal{I}_0\}$$

Analogous to the gap in the stochastic-with-a-gap setting.

Constraint Example

$\mathcal{I}_0 = \{$experts that are optimal for some $\mu \in \mathcal{D}\}$ $\qquad$ $N_0 = |\mathcal{I}_0|$

$\Delta_0 = \inf_{\mu \in \mathcal{D}} \{\mu$-expected difference in loss of best expert and best expert not in $\mathcal{I}_0\}$

Constraint Example

$\mathcal{I}_0 = \{$experts that are optimal for some $\mu \in \mathcal{D}\}$ $\quad\quad N_0 = |\mathcal{I}_0|$

$\Delta_0 = \inf_{\mu \in \mathcal{D}} \{\mu$-expected difference in loss of best expert and best expert not in $\mathcal{I}_0\}$

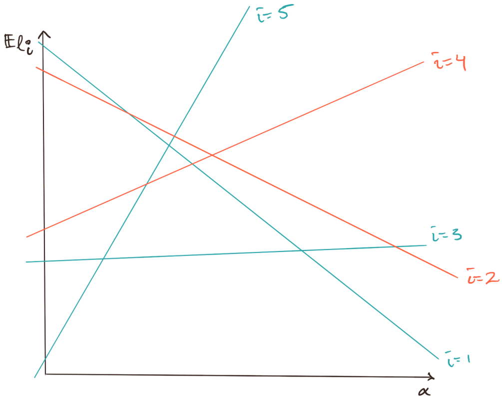
Setting: the means for each expert are jointly defined by a parameter α,
$N = 5$, $\mathcal{I}_0 = \{1, 3, 5\}$, $N_0 = 3$.

Constraint Example

$\mathcal{I}_0 = \{$experts that are optimal for some $\mu \in \mathcal{D}\}$ $\qquad N_0 = |\mathcal{I}_0|$

$$\Delta_0 = \inf_{\mu \in \mathcal{D}} \{\mu\text{-expected difference in loss of best expert and best expert not in } \mathcal{I}_0\}$$

Setting: the means for each expert are jointly defined by a parameter α,
$N = 5$, $\mathcal{I}_0 = \{1, 3, 5\}$, $N_0 = 3$.

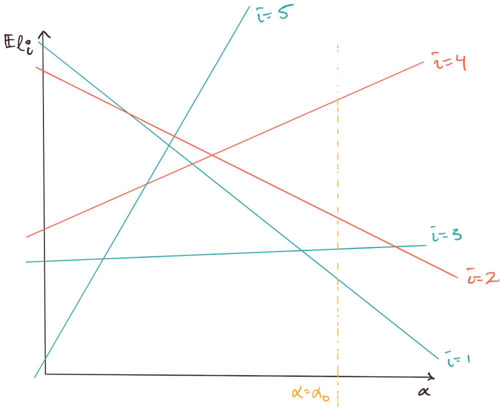


Constraint Example

$\mathcal{I}_0 = \{$experts that are optimal for some $\mu \in \mathcal{D}\}$ $\qquad N_0 = |\mathcal{I}_0|$

$\Delta_0 = \inf_{\mu \in \mathcal{D}} \{\mu$-expected difference in loss of best expert and best expert not in $\mathcal{I}_0\}$

Setting: the means for each expert are jointly defined by a parameter α,
$N = 5$, $\mathcal{I}_0 = \{1, 3, 5\}$, $N_0 = 3$.

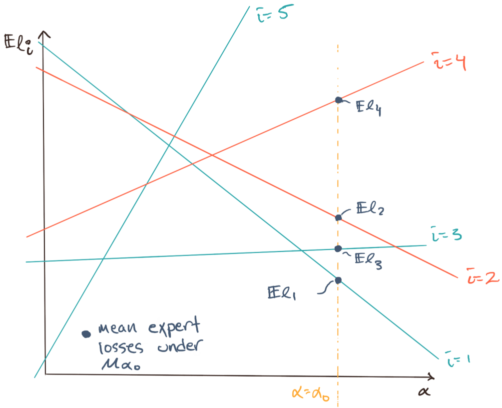


Constraint Example

$\mathcal{I}_0 = \{$experts that are optimal for some $\mu \in \mathcal{D}\}$ $\quad N_0 = |\mathcal{I}_0|$

$$\Delta_0 = \inf_{\mu \in \mathcal{D}} \{\mu\text{-expected difference in loss of best expert and best expert not in } \mathcal{I}_0\}$$

Setting: the means for each expert are jointly defined by a parameter α,
$N = 5$, $\mathcal{I}_0 = \{1, 3, 5\}$, $N_0 = 3$.

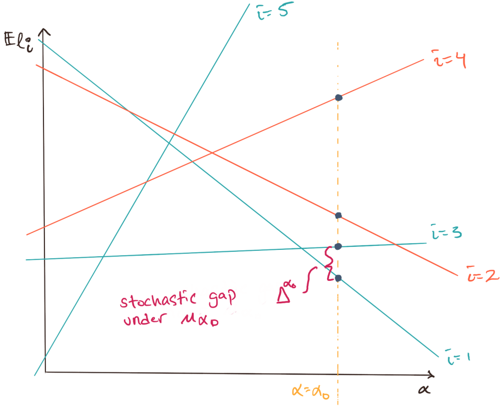


Constraint Example

$\mathcal{I}_0 = \{$experts that are optimal for some $\mu \in \mathcal{D}\}$ $\qquad N_0 = |\mathcal{I}_0|$

$\Delta_0 = \inf_{\mu \in \mathcal{D}} \{\mu$-expected difference in loss of best expert and best expert not in $\mathcal{I}_0\}$

Setting: the means for each expert are jointly defined by a parameter α,
$N = 5$, $\mathcal{I}_0 = \{1, 3, 5\}$, $N_0 = 3$.

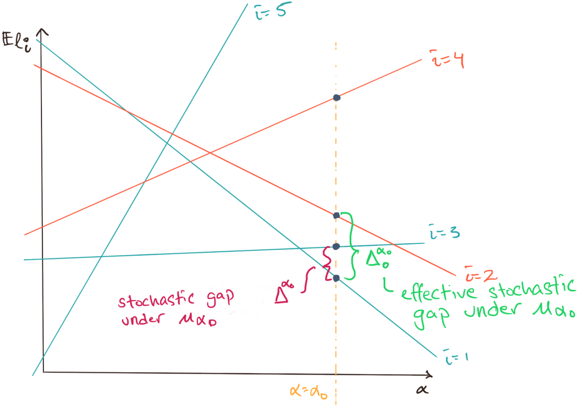


Constraint Example

$\mathcal{I}_0 = \{$experts that are optimal for some $\mu \in \mathcal{D}\}$ $\qquad N_0 = |\mathcal{I}_0|$

$\Delta_0 = \inf_{\mu \in \mathcal{D}} \{\mu$-expected difference in loss of best expert and best expert not in $\mathcal{I}_0\}$

Setting: the means for each expert are jointly defined by a parameter α,
$N = 5$, $\mathcal{I}_0 = \{1, 3, 5\}$, $N_0 = 3$.

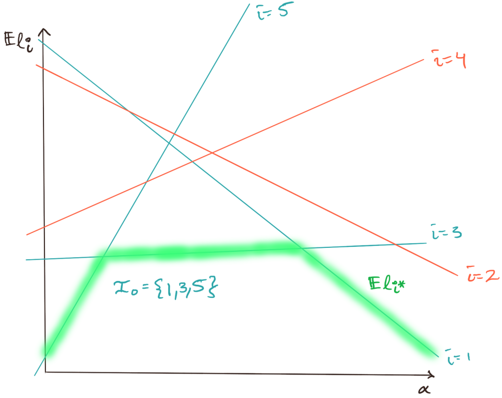


Constraint Example

$\mathcal{I}_0 = \{$experts that are optimal for some $\mu \in \mathcal{D}\}$ $\qquad N_0 = |\mathcal{I}_0|$

$\Delta_0 = \inf_{\mu \in \mathcal{D}} \{\mu$-expected difference in loss of best expert and best expert not in $\mathcal{I}_0\}$

Setting: the means for each expert are jointly defined by a parameter α,
$N = 5$, $\mathcal{I}_0 = \{1, 3, 5\}$, $N_0 = 3$.

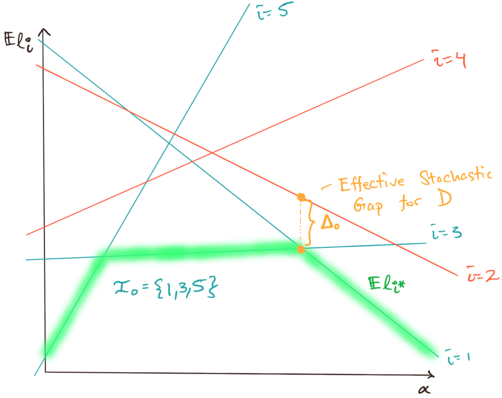


Constraint Example

$\mathcal{I}_0 = \{\text{experts that are optimal for some } \mu \in \mathcal{D}\}$ $\qquad N_0 = |\mathcal{I}_0|$

$$\Delta_0 = \inf_{\mu \in \mathcal{D}} \{\mu\text{-expected difference in loss of best expert and best expert not in } \mathcal{I}_0\}$$

Setting: the means for each expert are jointly defined by a parameter α,
$N = 5$, $\mathcal{I}_0 = \{1, 3, 5\}$, $N_0 = 3$.



Examples

Examples

Stochastic-with-a-gap: $\mathcal{D} = \{\mu_0\}$,

Examples

Stochastic-with-a-gap: $\mathcal{D} = \{\mu_0\}$,

- $N_0 = 1$,

Examples

Stochastic-with-a-gap: $\mathcal{D} = \{\mu_0\}$,

- $N_0 = 1$, $\mathcal{I}_0 = \left\{ i^* = \arg\min_{i \in [N]} \mathbb{E}_{\mu_0}[\ell_i] \right\}$,

Examples

Stochastic-with-a-gap: $\mathcal{D} = \{\mu_0\}$,

- $N_0 = 1$, $\mathcal{I}_0 = \left\{ i^* = \arg\min_{i \in [N]} \mathbb{E}_{\mu_0}[\ell_i] \right\}$, $\Delta_0 = \min_{i \neq i^*} \mathbb{E}_{\mu_0}[\ell_i - \ell_{i^*}]$

Examples

Stochastic-with-a-gap: $\mathcal{D} = \{\mu_0\}$,

- $N_0 = 1$, $\mathcal{I}_0 = \left\{ i^* = \arg\min_{i \in [N]} \mathbb{E}_{\mu_0}[\ell_i] \right\}$, $\Delta_0 = \min_{i \neq i^*} \mathbb{E}_{\mu_0}[\ell_i - \ell_{i^*}]$

Adversarial: $\mathcal{D} = \mathcal{M}(\hat{\mathcal{Y}}^N \times \mathcal{Y})$

Examples

Stochastic-with-a-gap: $\mathcal{D} = \{\mu_0\}$,

- $N_0 = 1$, $\mathcal{I}_0 = \left\{ i^* = \arg\min_{i \in [N]} \mathbb{E}_{\mu_0}[\ell_i] \right\}$, $\Delta_0 = \min_{i \neq i^*} \mathbb{E}_{\mu_0}[\ell_i - \ell_{i^*}]$

Adversarial: $\mathcal{D} = \mathcal{M}(\hat{\mathcal{Y}}^N \times \mathcal{Y}) \leftarrow$ contains point masses!

Examples

Stochastic-with-a-gap: $\mathcal{D} = \{\mu_0\}$,

- $N_0 = 1$, $\mathcal{I}_0 = \left\{ i^* = \arg\min_{i \in [N]} \mathbb{E}_{\mu_0}[\ell_i] \right\}$, $\Delta_0 = \min_{i \neq i^*} \mathbb{E}_{\mu_0}[\ell_i - \ell_{i^*}]$

Adversarial: $\mathcal{D} = \mathcal{M}(\hat{\mathcal{Y}}^N \times \mathcal{Y}) \leftarrow$ contains point masses!

- $N_0 = N$,

Examples

Stochastic-with-a-gap: $\mathcal{D} = \{\mu_0\}$,

- $N_0 = 1$, $\mathcal{I}_0 = \left\{ i^* = \arg\min_{i \in [N]} \mathbb{E}_{\mu_0}[\ell_i] \right\}$, $\Delta_0 = \min_{i \neq i^*} \mathbb{E}_{\mu_0}[\ell_i - \ell_{i^*}]$

Adversarial: $\mathcal{D} = \mathcal{M}(\hat{\mathcal{Y}}^N \times \mathcal{Y}) \leftarrow$ contains point masses!

- $N_0 = N$, $\Delta_0 = +\infty$

Examples

Stochastic-with-a-gap: $\mathcal{D} = \{\mu_0\}$,

- $N_0 = 1$, $\mathcal{I}_0 = \left\{ i^* = \arg\min_{i \in [N]} \mathbb{E}_{\mu_0}[\ell_i] \right\}$, $\Delta_0 = \min_{i \neq i^*} \mathbb{E}_{\mu_0}[\ell_i - \ell_{i^*}]$

Adversarial: $\mathcal{D} = \mathcal{M}(\hat{\mathcal{Y}}^N \times \mathcal{Y}) \leftarrow$ contains point masses!

- $N_0 = N$, $\Delta_0 = +\infty$

Adversarial-with-an-$\mathbb{E}$-gap (Mourtada and Gaïffas 2019)

Examples

Stochastic-with-a-gap: $\mathcal{D} = \{\mu_0\}$,

- $N_0 = 1$, $\mathcal{I}_0 = \left\{ i^* = \arg\min_{i \in [N]} \mathbb{E}_{\mu_0}[\ell_i] \right\}$, $\Delta_0 = \min_{i \neq i^*} \mathbb{E}_{\mu_0}[\ell_i - \ell_{i^*}]$

Adversarial: $\mathcal{D} = \mathcal{M}(\hat{\mathcal{Y}}^N \times \mathcal{Y}) \leftarrow$ contains point masses!

- $N_0 = N$, $\Delta_0 = +\infty$

Adversarial-with-an-$\mathbb{E}$-gap (Mourtada and Gaïffas 2019)

- All measures where a common expert is better than others in $\mathbb{E}$ by $\Delta > 0$.

Examples

Stochastic-with-a-gap: $\mathcal{D} = \{\mu_0\}$,

- $N_0 = 1$, $\mathcal{I}_0 = \left\{ i^* = \arg\min_{i \in [N]} \mathbb{E}_{\mu_0}[\ell_i] \right\}$, $\Delta_0 = \min_{i \neq i^*} \mathbb{E}_{\mu_0}[\ell_i - \ell_{i^*}]$

Adversarial: $\mathcal{D} = \mathcal{M}(\hat{\mathcal{Y}}^N \times \mathcal{Y}) \leftarrow$ contains point masses!

- $N_0 = N$, $\Delta_0 = +\infty$

Adversarial-with-an-$\mathbb{E}$-gap (Mourtada and Gaïffas 2019)

- All measures where a common expert is better than others in $\mathbb{E}$ by $\Delta > 0$.
- By design, $N_0 = 1$ and $\Delta_0 = \Delta$.

Examples

Stochastic-with-a-gap: $\mathcal{D} = \{\mu_0\}$,

- $N_0 = 1$, $\mathcal{I}_0 = \left\{ i^* = \arg\min_{i \in [N]} \mathbb{E}_{\mu_0}[\ell_i] \right\}$, $\Delta_0 = \min_{i \neq i^*} \mathbb{E}_{\mu_0}[\ell_i - \ell_{i^*}]$

Adversarial: $\mathcal{D} = \mathcal{M}(\hat{\mathcal{Y}}^N \times \mathcal{Y}) \leftarrow$ contains point masses!

- $N_0 = N$, $\Delta_0 = +\infty$

Adversarial-with-an-$\mathbb{E}$-gap (Mourtada and Gaïffas 2019)

- All measures where a common expert is better than others in $\mathbb{E}$ by $\Delta > 0$.
- By design, $N_0 = 1$ and $\Delta_0 = \Delta$.

Neighborhood-of-I.I.D.

Examples

Stochastic-with-a-gap: $\mathcal{D} = \{\mu_0\}$,

- $N_0 = 1$, $\mathcal{I}_0 = \left\{ i^* = \arg\min_{i \in [N]} \mathbb{E}_{\mu_0}[\ell_i] \right\}$, $\Delta_0 = \min_{i \neq i^*} \mathbb{E}_{\mu_0}[\ell_i - \ell_{i^*}]$

Adversarial: $\mathcal{D} = \mathcal{M}(\hat{\mathcal{Y}}^N \times \mathcal{Y}) \leftarrow$ contains point masses!

- $N_0 = N$, $\Delta_0 = +\infty$

Adversarial-with-an-$\mathbb{E}$-gap (Mourtada and Gaïffas 2019)

- All measures where a common expert is better than others in $\mathbb{E}$ by $\Delta > 0$.
- By design, $N_0 = 1$ and $\Delta_0 = \Delta$.

Neighborhood-of-I.I.D.

- Pick any distribution μ_0, and any radius, $r \geq 0$. $\mathcal{D} = \mathrm{Ball}(\mu_0, r)$

Examples

Stochastic-with-a-gap: $\mathcal{D} = \{\mu_0\}$,

- $N_0 = 1$, $\mathcal{I}_0 = \left\{ i^* = \arg\min_{i \in [N]} \mathbb{E}_{\mu_0}[\ell_i] \right\}$, $\Delta_0 = \min_{i \neq i^*} \mathbb{E}_{\mu_0}[\ell_i - \ell_{i^*}]$

Adversarial: $\mathcal{D} = \mathcal{M}(\hat{\mathcal{Y}}^N \times \mathcal{Y}) \leftarrow$ contains point masses!

- $N_0 = N$, $\Delta_0 = +\infty$

Adversarial-with-an-$\mathbb{E}$-gap (Mourtada and Gaïffas 2019)

- All measures where a common expert is better than others in $\mathbb{E}$ by $\Delta > 0$.
- By design, $N_0 = 1$ and $\Delta_0 = \Delta$.

Neighborhood-of-I.I.D.

- Pick any distribution μ_0, and any radius, $r \geq 0$. $\mathcal{D} = \text{Ball}(\mu_0, r)$
- Suppose that μ_0 has a gap between each of the mean losses.

Examples

Stochastic-with-a-gap: $\mathcal{D} = \{\mu_0\}$,

- $N_0 = 1$, $\mathcal{I}_0 = \left\{ i^* = \arg\min_{i \in [N]} \mathbb{E}_{\mu_0}[\ell_i] \right\}$, $\Delta_0 = \min_{i \neq i^*} \mathbb{E}_{\mu_0}[\ell_i - \ell_{i^*}]$

Adversarial: $\mathcal{D} = \mathcal{M}(\hat{\mathcal{Y}}^N \times \mathcal{Y}) \leftarrow$ contains point masses!

- $N_0 = N$, $\Delta_0 = +\infty$

Adversarial-with-an-$\mathbb{E}$-gap (Mourtada and Gaïffas 2019)

- All measures where a common expert is better than others in $\mathbb{E}$ by $\Delta > 0$.
- By design, $N_0 = 1$ and $\Delta_0 = \Delta$.

Neighborhood-of-I.I.D.

- Pick any distribution μ_0, and any radius, $r \geq 0$. $\mathcal{D} = \mathrm{Ball}(\mu_0, r)$
- Suppose that μ_0 has a gap between each of the mean losses.
- N_0, Δ_0 depend on the radius of the ball...

Examples

Stochastic-with-a-gap: $\mathcal{D} = \{\mu_0\}$,

- $N_0 = 1$, $\mathcal{I}_0 = \left\{ i^* = \arg\min_{i \in [N]} \mathbb{E}_{\mu_0}[\ell_i] \right\}$, $\Delta_0 = \min_{i \neq i^*} \mathbb{E}_{\mu_0}[\ell_i - \ell_{i^*}]$

Adversarial: $\mathcal{D} = \mathcal{M}(\hat{\mathcal{Y}}^N \times \mathcal{Y}) \leftarrow$ contains point masses!

- $N_0 = N$, $\Delta_0 = +\infty$

Adversarial-with-an-$\mathbb{E}$-gap (Mourtada and Gaïffas 2019)

- All measures where a common expert is better than others in $\mathbb{E}$ by $\Delta > 0$.
- By design, $N_0 = 1$ and $\Delta_0 = \Delta$.

Neighborhood-of-I.I.D.

- Pick any distribution μ_0, and any radius, $r \geq 0$. $\mathcal{D} = \text{Ball}(\mu_0, r)$
- Suppose that μ_0 has a gap between each of the mean losses.
- N_0, Δ_0 depend on the radius of the ball...

Interpreting (N_0, Δ_0^{-1})

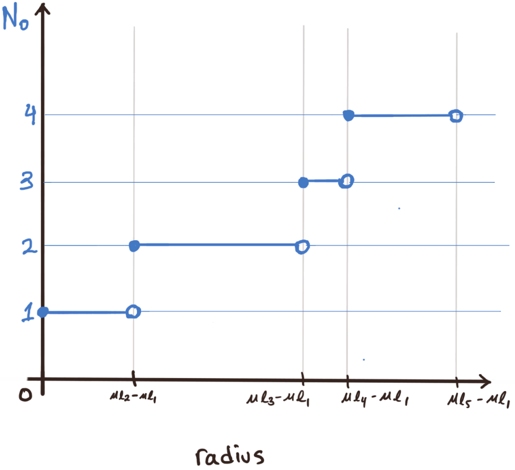
$\mathcal{D} = \mathsf{Ball}(\mu, \texttt{radius})$ w/ $\mathbb{E}_\mu \ell_1 < \mathbb{E}_\mu \ell_2 < \dots$

Interpreting (N_0, Δ_0^{-1})

$\mathcal{D} = \mathsf{Ball}(\mu, \texttt{radius})$ w/ $\mathbb{E}_\mu \ell_1 < \mathbb{E}_\mu \ell_2 < \dots$

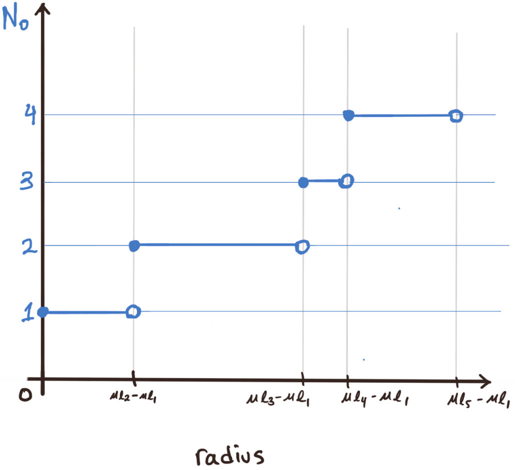
Interpreting (N_0, Δ_0^{-1})

$\mathcal{D} = \text{Ball}(\mu, \texttt{radius})$ w/ $\mathbb{E}_\mu \ell_1 < \mathbb{E}_\mu \ell_2 < \dots$



Interpreting (N_0, Δ_0^{-1})

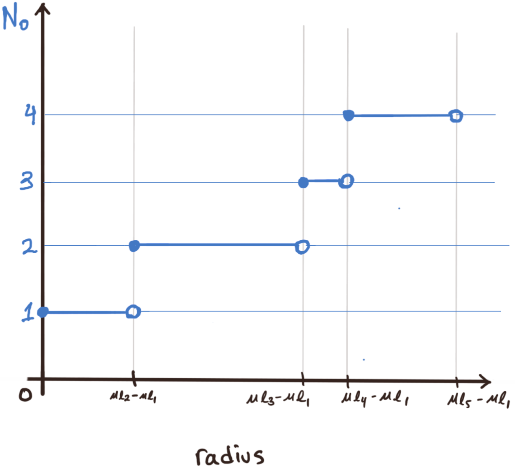
$\mathcal{D} = \text{Ball}(\mu, \texttt{radius})$ w/ $\mathbb{E}_\mu \ell_1 < \mathbb{E}_\mu \ell_2 < \dots$



- N_0 non-decreasing with radius

Interpreting (N_0, Δ_0^{-1})

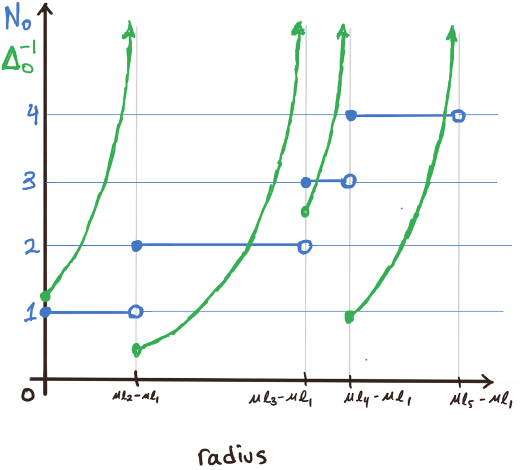
$\mathcal{D} = \text{Ball}(\mu, \texttt{radius})$ w/ $\mathbb{E}_\mu \ell_1 < \mathbb{E}_\mu \ell_2 < \dots$



- N_0 non-decreasing with radius
- N_0 increases discretely

Interpreting (N_0, Δ_0^{-1})

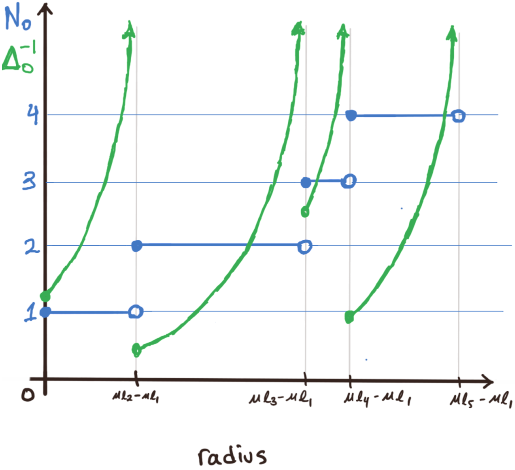
$\mathcal{D} = \text{Ball}(\mu, \texttt{radius})$ w/ $\mathbb{E}_\mu \ell_1 < \mathbb{E}_\mu \ell_2 < \ldots$



- N_0 non-decreasing with radius
- N_0 increases discretely

Interpreting (N_0, Δ_0^{-1})

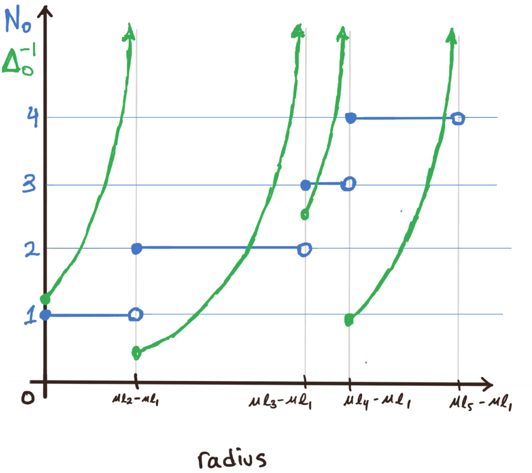
$\mathcal{D} = \mathrm{Ball}(\mu, \mathtt{radius})$ w/ $\mathbb{E}_\mu \ell_1 < \mathbb{E}_\mu \ell_2 < \dots$



- N_0 non-decreasing with radius
- N_0 increases discretely
- Δ_0^{-1} increases between N_0 jumps

Interpreting (N_0, Δ_0^{-1})

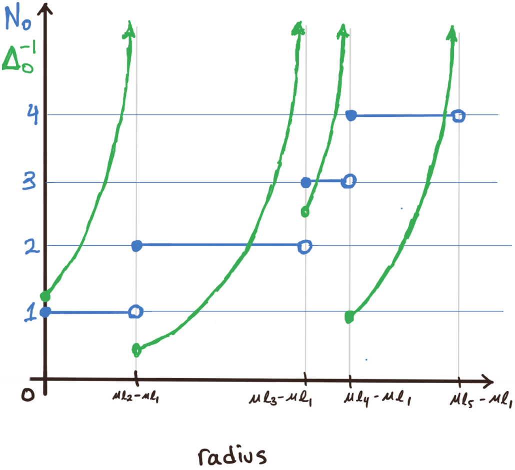
$\mathcal{D} = \text{Ball}(\mu, \texttt{radius})$ w/ $\mathbb{E}_\mu \ell_1 < \mathbb{E}_\mu \ell_2 < \dots$



- N_0 non-decreasing with radius
- N_0 increases discretely
- Δ_0^{-1} increases between N_0 jumps
- Δ_0^{-1} resets at each jump

Interpreting (N_0, Δ_0^{-1})

$\mathcal{D} = \text{Ball}(\mu, \texttt{radius})$ w/ $\mathbb{E}_\mu \ell_1 < \mathbb{E}_\mu \ell_2 < \dots$

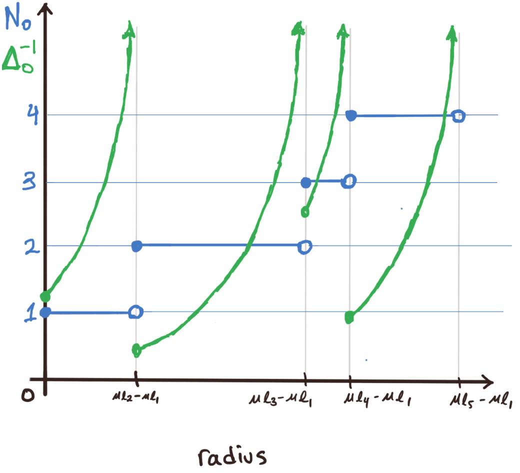


- N_0 non-decreasing with radius
- N_0 increases discretely
- Δ_0^{-1} increases between N_0 jumps
- Δ_0^{-1} resets at each jump

Lexicographical order on (N_0, Δ_0^{-1})

Interpreting (N_0, Δ_0^{-1})

$\mathcal{D} = \text{Ball}(\mu, \texttt{radius})$ w/ $\mathbb{E}_\mu \ell_1 < \mathbb{E}_\mu \ell_2 < \dots$



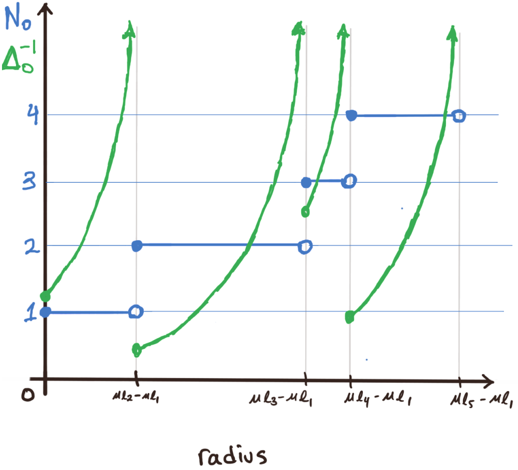
- N_0 non-decreasing with radius
- N_0 increases discretely
- Δ_0^{-1} increases between N_0 jumps
- Δ_0^{-1} resets at each jump

Lexicographical order on (N_0, Δ_0^{-1})

- For nested $\mathcal{D}$s, larger one is "harder" to learn.

Interpreting (N_0, Δ_0^{-1})

$\mathcal{D} = \text{Ball}(\mu, \texttt{radius})$ w/ $\mathbb{E}_\mu \ell_1 < \mathbb{E}_\mu \ell_2 < \ldots$



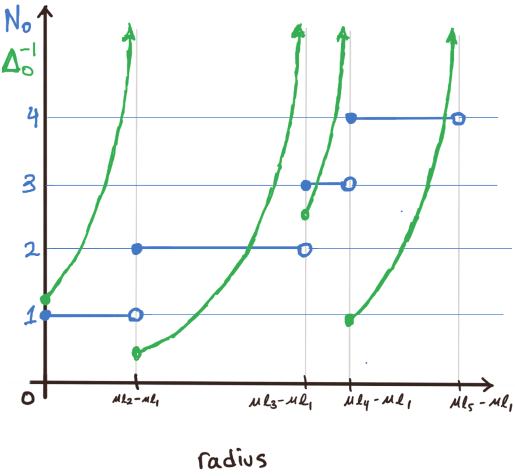
- N_0 non-decreasing with radius
- N_0 increases discretely
- Δ_0^{-1} increases between N_0 jumps
- Δ_0^{-1} resets at each jump

Lexicographical order on (N_0, Δ_0^{-1})

- For nested $\mathcal{D}$s, larger one is "harder" to learn.
- (N_0, Δ_0^{-1}) quantifies the difficulty.

Interpreting (N_0, Δ_0^{-1})

$\mathcal{D} = \text{Ball}(\mu, \texttt{radius})$ w/ $\mathbb{E}_\mu \ell_1 < \mathbb{E}_\mu \ell_2 < \dots$

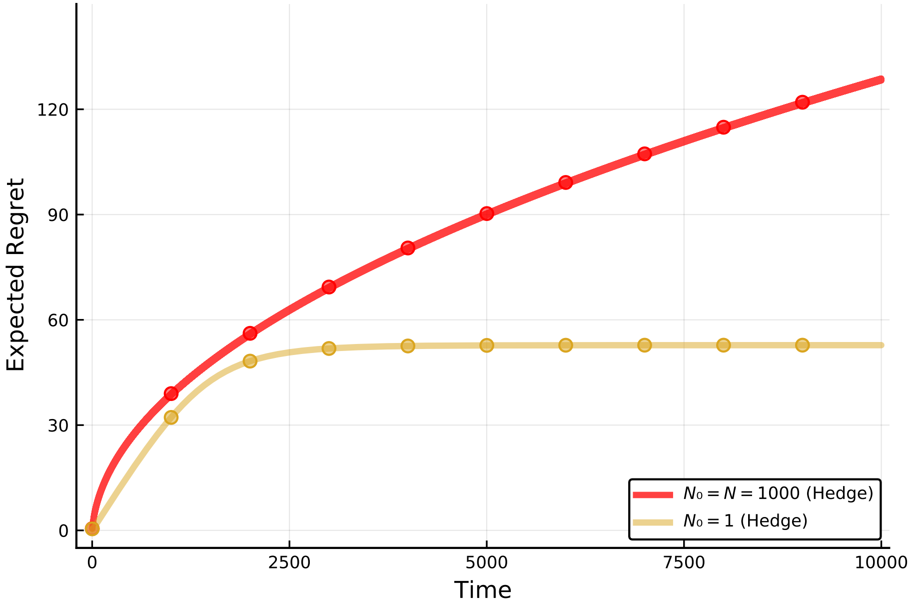


- N_0 non-decreasing with radius
- N_0 increases discretely
- Δ_0^{-1} increases between N_0 jumps
- Δ_0^{-1} resets at each jump

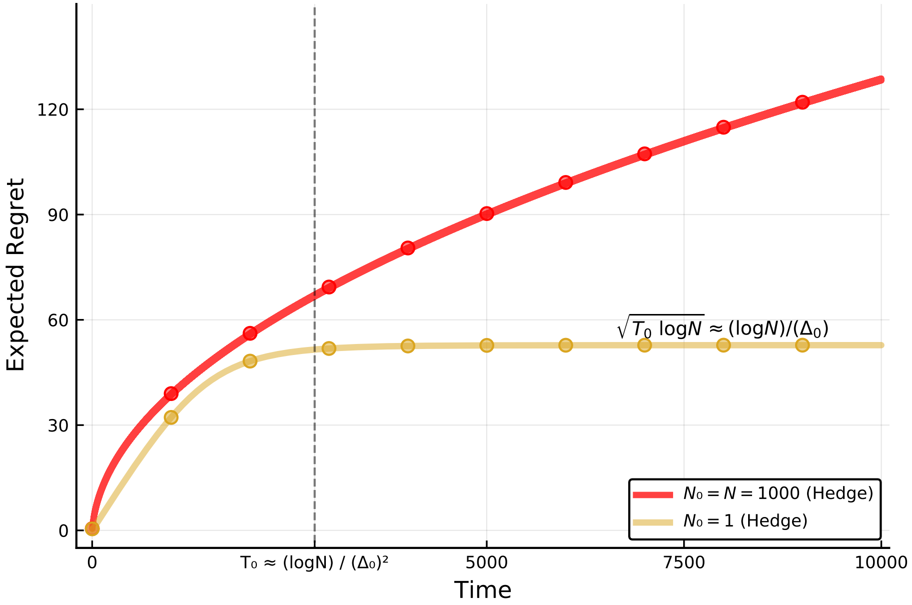
Lexicographical order on (N_0, Δ_0^{-1})

- For nested $\mathcal{D}$s, larger one is "harder" to learn.
- (N_0, Δ_0^{-1}) quantifies the difficulty.
- How does regret change with Δ_0?

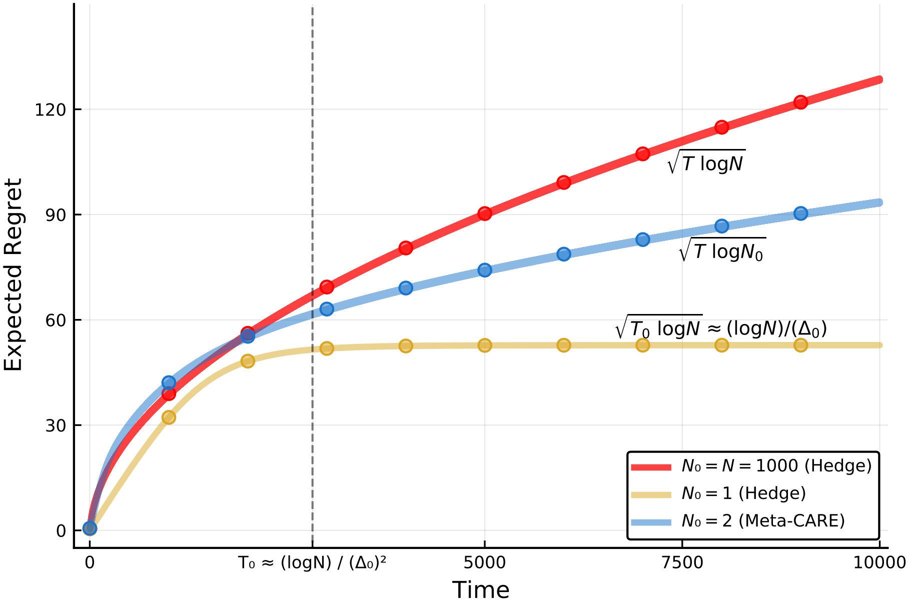
Impact of (N_0, Δ_0^{-1}) on Regret



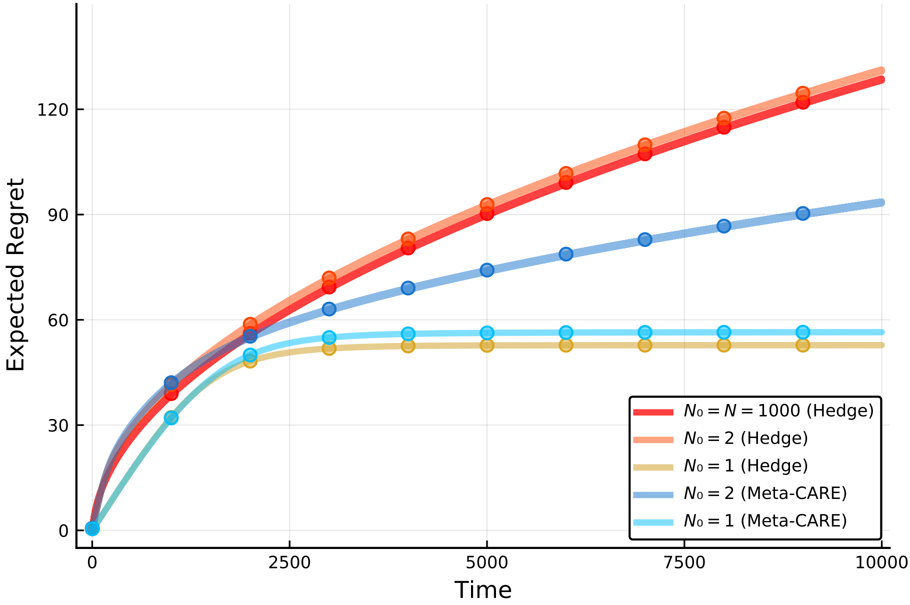
Impact of (N_0, Δ_0^{-1}) on Regret



Impact of (N_0, Δ_0^{-1}) on Regret



Impact of (N_0, Δ_0^{-1}) on Regret



Performance of Hedge

Hedge Algorithm

We will consider only finite expert classes and bounded losses $\ell : \hat{\mathcal{Y}} \times \mathcal{Y} \to [0, 1]$.

Hedge Algorithm

We will consider only finite expert classes and bounded losses $\ell : \hat{\mathcal{Y}} \times \mathcal{Y} \to [0, 1]$.

All explicit algorithms we will consider are *proper*:

Hedge Algorithm

We will consider only finite expert classes and bounded losses $\ell : \hat{\mathcal{Y}} \times \mathcal{Y} \to [0, 1]$.

All explicit algorithms we will consider are *proper*:

the player randomly selects an expert to emulate at each time.

Hedge Algorithm

We will consider only finite expert classes and bounded losses $\ell : \hat{\mathcal{Y}} \times \mathcal{Y} \to [0, 1]$.

All explicit algorithms we will consider are *proper*:

the player randomly selects an expert to emulate at each time.

A proper algorithm assigns probability $w_i(t)$ to expert i at time t.

Hedge Algorithm

We will consider only finite expert classes and bounded losses $\ell : \hat{\mathcal{Y}} \times \mathcal{Y} \to [0, 1]$.

All explicit algorithms we will consider are *proper*:

the player randomly selects an expert to emulate at each time.

A proper algorithm assigns probability $w_i(t)$ to expert i at time t.

Hedge Algorithm

Hedge Algorithm

We will consider only finite expert classes and bounded losses $\ell : \hat{\mathcal{Y}} \times \mathcal{Y} \to [0, 1]$.

All explicit algorithms we will consider are *proper*:

the player randomly selects an expert to emulate at each time.

A proper algorithm assigns probability $w_i(t)$ to expert i at time t.

Hedge Algorithm

- Fix learning rate schedule $\eta : \mathbb{N} \to \mathbb{R}$; initialize the weights as uniform; define

Hedge Algorithm

We will consider only finite expert classes and bounded losses $\ell : \hat{\mathcal{Y}} \times \mathcal{Y} \to [0, 1]$.

All explicit algorithms we will consider are *proper*:

the player randomly selects an expert to emulate at each time.

A proper algorithm assigns probability $w_i(t)$ to expert i at time t.

Hedge Algorithm

- Fix learning rate schedule $\eta : \mathbb{N} \to \mathbb{R}$; initialize the weights as uniform; define

$$\ell_i(t) = \ell(x_i(t), y(t)), \qquad L_i(t) = \sum_{s=1}^{t} \ell_i(s).$$

Hedge Algorithm

We will consider only finite expert classes and bounded losses $\ell : \hat{\mathcal{Y}} \times \mathcal{Y} \to [0, 1]$.

All explicit algorithms we will consider are *proper*:

the player randomly selects an expert to emulate at each time.

A proper algorithm assigns probability $w_i(t)$ to expert i at time t.

Hedge Algorithm

- Fix learning rate schedule $\eta : \mathbb{N} \to \mathbb{R}$; initialize the weights as uniform; define

$$\ell_i(t) = \ell(x_i(t), y(t)), \qquad L_i(t) = \sum_{s=1}^{t} \ell_i(s).$$

- Update weights for each $i \in [N]$ using

$$w_i(t) \propto \exp\left\{-\eta(t) L_i(t-1)\right\}.$$

Hedge Algorithm

We will consider only finite expert classes and bounded losses $\ell : \hat{\mathcal{Y}} \times \mathcal{Y} \to [0, 1]$.

All explicit algorithms we will consider are *proper*:

the player randomly selects an expert to emulate at each time.

A proper algorithm assigns probability $w_i(t)$ to expert i at time t.

Hedge Algorithm

- Fix learning rate schedule $\eta : \mathbb{N} \to \mathbb{R}$; initialize the weights as uniform; define

$$\ell_i(t) = \ell(x_i(t), y(t)), \qquad L_i(t) = \sum_{s=1}^{t} \ell_i(s).$$

- Update weights for each $i \in [N]$ using

$$w_i(t) \propto \exp\left\{-\eta(t) L_i(t-1)\right\}.$$

Hedge Regret Bounds

Consider playing Hedge with $\eta(t) = c/\sqrt{t}$ for any convex $\mathcal{D}$.

Recall:

- N_0 is the number of of effective experts,
- Δ_0 is the effective stochastic gap.

Hedge Regret Bounds

Consider playing Hedge with $\eta(t) = c/\sqrt{t}$ for any convex $\mathcal{D}$.

Recall:

- N_0 is the number of of effective experts,
- Δ_0 is the effective stochastic gap.

Theorem BNR20

Taking $c \propto \sqrt{\log N}$ gives

$$\mathbb{E}R(T) \lesssim \begin{cases} \sqrt{T \log N} + \frac{\log N}{\Delta_0} & : N_0 \geq 2 \\ (\log N)/\Delta_0 & : N_0 = 1. \end{cases}$$

Taking $c \propto 1$ gives

$$\mathbb{E}R(T) \lesssim (\log N_0)\sqrt{T} + \frac{(\log N)^2}{\Delta_0}$$

We also prove matching lower bounds!

Hedge Regret Bounds

Consider playing Hedge with $\eta(t) = c/\sqrt{t}$ for any convex $\mathcal{D}$.

Recall:

- N_0 is the number of of effective experts,
- Δ_0 is the effective stochastic gap.

Theorem BNR20

If the player has oracle knowledge of $N_0 > 1$, taking $c \propto \sqrt{\log N_0}$ gives

$$\mathbb{E}R(T) \lesssim \sqrt{T \log N_0} + \frac{(\log N)^2}{(\log N_0)\Delta_0}.$$

Hedge Regret Bounds

Consider playing Hedge with $\eta(t) = c/\sqrt{t}$ for any convex $\mathcal{D}$.

Recall:

- N_0 is the number of of effective experts,
- Δ_0 is the effective stochastic gap.

Theorem BNR20

If the player has oracle knowledge of $N_0 > 1$, taking $c \propto \sqrt{\log N_0}$ gives

$$\mathbb{E}R(T) \lesssim \sqrt{T \log N_0} + \frac{(\log N)^2}{(\log N_0)\Delta_0}.$$

In all three cases, we interpret terms involving...

Hedge Regret Bounds

Consider playing Hedge with $\eta(t) = c/\sqrt{t}$ for any convex $\mathcal{D}$.

Recall:

- N_0 is the number of of effective experts,
- Δ_0 is the effective stochastic gap.

Theorem BNR20

If the player has oracle knowledge of $N_0 > 1$, taking $c \propto \sqrt{\log N_0}$ gives

$$\mathbb{E}R(T) \lesssim \sqrt{T \log N_0} + \frac{(\log N)^2}{(\log N_0)\Delta_0}.$$

In all three cases, we interpret terms involving...

- T: long run regret accumulation after adapting

Hedge Regret Bounds

Consider playing Hedge with $\eta(t) = c/\sqrt{t}$ for any convex $\mathcal{D}$.

Recall:

- N_0 is the number of of effective experts,
- Δ_0 is the effective stochastic gap.

Theorem BNR20

If the player has oracle knowledge of $N_0 > 1$, taking $c \propto \sqrt{\log N_0}$ gives

$$\mathbb{E}R(T) \lesssim \sqrt{T \log N_0} + \frac{(\log N)^2}{(\log N_0)\Delta_0}.$$

In all three cases, we interpret terms involving...

- T: long run regret accumulation after adapting
- $(\log N, \Delta_0)$: adversarial regret over adaptation period of duration $\mathcal{O}\left(\frac{(\log N)^2}{c^2 \Delta_0^2}\right)$

Can we do better than Hedge?

Question: If we don't know N_0, can we learn adaptively and minimax optimally?

Can we do better than Hedge?

Question: If we don't know N_0, can we learn adaptively and minimax optimally?

In particular, is there an algorithm for which...

Can we do better than Hedge?

Question: If we don't know N_0, can we learn adaptively and minimax optimally?

In particular, is there an algorithm for which...

1. (T, N_0)-dependence matches Hedge with oracle knowledge of N_0,

Can we do better than Hedge?

Question: If we don't know N_0, can we learn adaptively and minimax optimally?

In particular, is there an algorithm for which...

1. (T, N_0)-dependence matches Hedge with oracle knowledge of N_0,
2. $(\log N, \Delta_0)$-dependence optimal for the stochastic case when $N_0 = 1$, and

Can we do better than Hedge?

Question: If we don't know N_0, can we learn adaptively and minimax optimally?

In particular, is there an algorithm for which...

1. (T, N_0)-dependence matches Hedge with oracle knowledge of N_0,
2. $(\log N, \Delta_0)$-dependence optimal for the stochastic case when $N_0 = 1$, and
3. no information is needed about the true setting?

Can we do better than Hedge?

Question: If we don't know N_0, can we learn adaptively and minimax optimally?

In particular, is there an algorithm for which...

1. (T, N_0)-dependence matches Hedge with oracle knowledge of N_0,
2. $(\log N, \Delta_0)$-dependence optimal for the stochastic case when $N_0 = 1$, and
3. no information is needed about the true setting?

Answer: Yes!

Can we do better than Hedge?

Question: If we don't know N_0, can we learn adaptively and minimax optimally?

In particular, is there an algorithm for which...

1. (T, N_0)-dependence matches Hedge with oracle knowledge of N_0,
2. $(\log N, \Delta_0)$-dependence optimal for the stochastic case when $N_0 = 1$, and
3. no information is needed about the true setting?

Answer: Yes!

We introduce two new algorithms in order to do this...

Can we do better than Hedge?

Question: If we don't know N_0, can we learn adaptively and minimax optimally?

In particular, is there an algorithm for which...

1. (T, N_0)-dependence matches Hedge with oracle knowledge of N_0,
2. $(\log N, \Delta_0)$-dependence optimal for the stochastic case when $N_0 = 1$, and
3. no information is needed about the true setting?

Answer: Yes!

We introduce two new algorithms in order to do this...

- FTRL-CARE, accomplished 1 and 3, but not 2.

Can we do better than Hedge?

Question: If we don't know N_0, can we learn adaptively and minimax optimally?

In particular, is there an algorithm for which...

1. (T, N_0)-dependence matches Hedge with oracle knowledge of N_0,
2. $(\log N, \Delta_0)$-dependence optimal for the stochastic case when $N_0 = 1$, and
3. no information is needed about the true setting?

Answer: Yes!

We introduce two new algorithms in order to do this...

- FTRL-CARE, accomplished 1 and 3, but not 2.
  - slightly worse dependence on N.

Can we do better than Hedge?

Question: If we don't know N_0, can we learn adaptively and minimax optimally?

In particular, is there an algorithm for which...

1. (T, N_0)-dependence matches Hedge with oracle knowledge of N_0,
2. $(\log N, \Delta_0)$-dependence optimal for the stochastic case when $N_0 = 1$, and
3. no information is needed about the true setting?

Answer: Yes!

We introduce two new algorithms in order to do this...

- FTRL-CARE, accomplished 1 and 3, but not 2.
  - slightly worse dependence on N.
- Meta-CARE, accomplished all 3 by *boosting* FTRL-CARE with Hedge.

Can we do better than Hedge?

Question: If we don't know N_0, can we learn adaptively and minimax optimally?

In particular, is there an algorithm for which...

1. (T, N_0)-dependence matches Hedge with oracle knowledge of N_0,
2. $(\log N, \Delta_0)$-dependence optimal for the stochastic case when $N_0 = 1$, and
3. no information is needed about the true setting?

Answer: Yes!

We introduce two new algorithms in order to do this...

- FTRL-CARE, accomplished 1 and 3, but not 2.
  - slightly worse dependence on N.
- Meta-CARE, accomplished all 3 by *boosting* FTRL-CARE with Hedge.

Improved Algorithms and Bounds

Intuition for Improving on Hedge

Three Key Insights:

Intuition for Improving on Hedge

Three Key Insights:

1. From our oracle Hedge bound, we want a learning rate $\propto \sqrt{(\log N_0)/t}$.

Intuition for Improving on Hedge

Three Key Insights:

1. From our oracle Hedge bound, we want a learning rate $\propto \sqrt{(\log N_0)/t}$.
2. The regret of Hedge closely depends on the *entropy* of the weights:

$$H(w) = -\sum_{i \in [N]} w_i \log(w_i).$$

Intuition for Improving on Hedge

Three Key Insights:

1. From our oracle Hedge bound, we want a learning rate $\propto \sqrt{(\log N_0)/t}$.
2. The regret of Hedge closely depends on the *entropy* of the weights:

$$H(w) = -\sum_{i \in [N]} w_i \log(w_i).$$

3. Worst-case adversary forces weights to concentrate to $\mathrm{Unif}(\mathcal{I}_0)$, so

$$H(w) \approx \log N_0.$$

Intuition for Improving on Hedge

Three Key Insights:

1. From our oracle Hedge bound, we want a learning rate $\propto \sqrt{(\log N_0)/t}$.
2. The regret of Hedge closely depends on the *entropy* of the weights:

$$H(w) = -\sum_{i \in [N]} w_i \log(w_i).$$

3. Worst-case adversary forces weights to concentrate to $\mathrm{Unif}(\mathcal{I}_0)$, so

$$H(w) \approx \log N_0.$$

What if we could have our learning rate at time t, $\eta(t)$, look like

$$\eta(t) = \sqrt{\frac{H(w(t))}{t}} \quad ?$$

Follow the Regularized Leader

FTRL is a fundamental online linear optimization algorithm.

Follow the Regularized Leader

FTRL is a fundamental online linear optimization algorithm.

Parametrized by a sequence of regularizers $(\psi_t)_{t\in\mathbb{N}} \subseteq \mathtt{simp}([N]) \to \mathbb{R}$,

Follow the Regularized Leader

FTRL is a fundamental online linear optimization algorithm.

Parametrized by a sequence of regularizers $(\psi_t)_{t\in\mathbb{N}} \subseteq \mathtt{simp}([N]) \to \mathbb{R}$,

$$w(t+1) = \operatorname*{arg\,min}_{w\in\mathtt{simp}([N])} \left(\langle w,\ L(t)\rangle + \psi_{t+1}(w)\right).$$

Follow the Regularized Leader

FTRL is a fundamental online linear optimization algorithm.

Parametrized by a sequence of regularizers $(\psi_t)_{t\in\mathbb{N}} \subseteq \mathtt{simp}([N]) \to \mathbb{R}$,

$$w(t+1) = \operatorname*{arg\,min}_{w\in\mathtt{simp}([N])} \left(\langle w,\ L(t)\rangle + \psi_{t+1}(w)\right).$$

Hedge corresponds to $\psi_{t+1}(w) = -\frac{H(w)}{\eta(t+1)}$.

Follow the Regularized Leader

FTRL is a fundamental online linear optimization algorithm.

Parametrized by a sequence of regularizers $(\psi_t)_{t\in\mathbb{N}} \subseteq \mathtt{simp}([N]) \to \mathbb{R}$,

$$w(t+1) = \operatorname*{arg\,min}_{w\in\mathtt{simp}([N])} \left(\langle w,\ L(t)\rangle + \psi_{t+1}(w)\right).$$

Hedge corresponds to $\psi_{t+1}(w) = -\frac{H(w)}{\eta(t+1)}$. That is,

$$\frac{\exp\{-\eta(t+1)L(t)\}}{\sum_{i\in[N]}\exp\{-\eta(t+1)L_i(t)\}} = \operatorname*{arg\,min}_{w\in\mathtt{simp}([N])} \left(\langle w,\ L(t)\rangle - \frac{H(w)}{\eta(t+1)}\right).$$

Follow the Regularized Leader

FTRL is a fundamental online linear optimization algorithm.

Parametrized by a sequence of regularizers $(\psi_t)_{t\in\mathbb{N}} \subseteq \mathtt{simp}([N]) \to \mathbb{R}$,

$$w(t+1) = \underset{w\in\mathtt{simp}([N])}{\arg\min} \left(\langle w,\ L(t)\rangle + \psi_{t+1}(w)\right).$$

Hedge corresponds to $\psi_{t+1}(w) = -\frac{H(w)}{\eta(t+1)}$. That is,

$$\frac{\exp\{-\eta(t+1)L(t)\}}{\sum_{i\in[N]}\exp\{-\eta(t+1)L_i(t)\}} = \underset{w\in\mathtt{simp}([N])}{\arg\min}\left(\langle w,\ L(t)\rangle - \frac{H(w)}{\eta(t+1)}\right).$$

Introducing **FTRL-CARE**:

Follow the Regularized Leader

FTRL is a fundamental online linear optimization algorithm.

Parametrized by a sequence of regularizers $(\psi_t)_{t\in\mathbb{N}} \subseteq \texttt{simp}([N]) \to \mathbb{R}$,

$$w(t+1) = \operatorname*{arg\,min}_{w\in\texttt{simp}([N])} \left(\langle w,\ L(t)\rangle + \psi_{t+1}(w)\right).$$

Hedge corresponds to $\psi_{t+1}(w) = -\frac{H(w)}{\eta(t+1)}$. That is,

$$\frac{\exp\{-\eta(t+1)L(t)\}}{\sum_{i\in[N]}\exp\{-\eta(t+1)L_i(t)\}} = \operatorname*{arg\,min}_{w\in\texttt{simp}([N])} \left(\langle w,\ L(t)\rangle - \frac{H(w)}{\eta(t+1)}\right).$$

Introducing **FTRL-CARE**:

Follow the Regularized Leader with Constraint-Adaptive Root-Entropic regularization

$$w(t+1) \in \operatorname*{arg\,min}_{w\in\texttt{simp}([N])} \left(\langle w,\ L(t)\rangle - \frac{\sqrt{t+1}}{c_1}\sqrt{H(w)+c_2}\right).$$

Follow the Regularized Leader

FTRL is a fundamental online linear optimization algorithm.

Parametrized by a sequence of regularizers $(\psi_t)_{t\in\mathbb{N}} \subseteq \mathtt{simp}([N]) \to \mathbb{R}$,

$$w(t+1) = \operatorname*{arg\,min}_{w\in\mathtt{simp}([N])} \left(\langle w,\ L(t)\rangle + \psi_{t+1}(w)\right).$$

Hedge corresponds to $\psi_{t+1}(w) = -\frac{H(w)}{\eta(t+1)}$. That is,

$$\frac{\exp\{-\eta(t+1)L(t)\}}{\sum_{i\in[N]}\exp\{-\eta(t+1)L_i(t)\}} = \operatorname*{arg\,min}_{w\in\mathtt{simp}([N])} \left(\langle w,\ L(t)\rangle - \frac{H(w)}{\eta(t+1)}\right).$$

Introducing **FTRL-CARE**:

Follow the Regularized Leader with Constraint-Adaptive Root-Entropic regularization

$$w(t+1) \in \operatorname*{arg\,min}_{w\in\mathtt{simp}([N])} \left(\langle w,\ L(t)\rangle - \frac{\sqrt{t+1}}{c_1}\sqrt{H(w)+c_2}\right).$$

CARE if you can, Hedge if you must; or, Meta-CARE for All

CARE if you can, Hedge if you must; or, Meta-CARE for All

FTRL-CARE is *almost* adaptively minimax optimal.

Theorem BNR20

For any convex $\mathcal{D}$, FTRL-CARE achieves

$$\mathbb{E}R(T) \lesssim \sqrt{T \log N_0} + \frac{(\log N)^{3/2}}{\Delta_0}.$$

When $N_0 = 1$, it incurs total regret of order $\frac{(\log N)^{3/2}}{\Delta_0}$ instead of $\frac{(\log N)}{\Delta_0}$.

CARE if you can, Hedge if you must; or, Meta-CARE for All

FTRL-CARE is *almost* adaptively minimax optimal.

Theorem BNR20

For any convex $\mathcal{D}$, FTRL-CARE achieves

$$\mathbb{E}R(T) \lesssim \sqrt{T \log N_0} + \frac{(\log N)^{3/2}}{\Delta_0}.$$

When $N_0 = 1$, it incurs total regret of order $\frac{(\log N)^{3/2}}{\Delta_0}$ instead of $\frac{(\log N)}{\Delta_0}$.
To be minimax optimal even when $N_0 = 1$, combine Hedge and FTRL-CARE.

CARE if you can, Hedge if you must; or, Meta-CARE for All

FTRL-CARE is *almost* adaptively minimax optimal.

Theorem BNR20

For any convex $\mathcal{D}$, FTRL-CARE achieves

$$\mathbb{E}R(T) \lesssim \sqrt{T \log N_0} + \frac{(\log N)^{3/2}}{\Delta_0}.$$

When $N_0 = 1$, it incurs total regret of order $\frac{(\log N)^{3/2}}{\Delta_0}$ instead of $\frac{(\log N)}{\Delta_0}$.
To be minimax optimal even when $N_0 = 1$, combine Hedge and FTRL-CARE.

Meta-CARE

- Treat the predictions of Hedge and FTRL-CARE as *meta-experts*.

CARE if you can, Hedge if you must; or, Meta-CARE for All

FTRL-CARE is *almost* adaptively minimax optimal.

Theorem BNR20

For any convex $\mathcal{D}$, FTRL-CARE achieves

$$\mathbb{E}R(T) \lesssim \sqrt{T \log N_0} + \frac{(\log N)^{3/2}}{\Delta_0}.$$

When $N_0 = 1$, it incurs total regret of order $\frac{(\log N)^{3/2}}{\Delta_0}$ instead of $\frac{(\log N)}{\Delta_0}$.
To be minimax optimal even when $N_0 = 1$, combine Hedge and FTRL-CARE.

Meta-CARE

- Treat the predictions of Hedge and FTRL-CARE as *meta-experts*.
- Use Hedge on these two meta-experts.

CARE if you can, Hedge if you must; or, Meta-CARE for All

FTRL-CARE is *almost* adaptively minimax optimal.

Theorem BNR20

For any convex $\mathcal{D}$, FTRL-CARE achieves

$$\mathbb{E}R(T) \lesssim \sqrt{T \log N_0} + \frac{(\log N)^{3/2}}{\Delta_0}.$$

When $N_0 = 1$, it incurs total regret of order $\frac{(\log N)^{3/2}}{\Delta_0}$ instead of $\frac{(\log N)}{\Delta_0}$.
To be minimax optimal even when $N_0 = 1$, combine Hedge and FTRL-CARE.

Meta-CARE

- Treat the predictions of Hedge and FTRL-CARE as *meta-experts*.
- Use Hedge on these two meta-experts.
- Incur best regret of the two, plus negligible excess from meta-learning.

CARE if you can, Hedge if you must; or, Meta-CARE for All

FTRL-CARE is *almost* adaptively minimax optimal.

Theorem BNR20

For any convex $\mathcal{D}$, FTRL-CARE achieves

$$\mathbb{E}R(T) \lesssim \sqrt{T \log N_0} + \frac{(\log N)^{3/2}}{\Delta_0}.$$

When $N_0 = 1$, it incurs total regret of order $\frac{(\log N)^{3/2}}{\Delta_0}$ instead of $\frac{(\log N)}{\Delta_0}$.
To be minimax optimal even when $N_0 = 1$, combine Hedge and FTRL-CARE.

Meta-CARE

- Treat the predictions of Hedge and FTRL-CARE as *meta-experts*.
- Use Hedge on these two meta-experts.
- Incur best regret of the two, plus negligible excess from meta-learning.

Theorem BNR20

For any convex $\mathcal{D}$, Meta-CARE achieves

$$\mathbb{E}R(T) \lesssim \sqrt{T \log N_0} + \mathbb{I}_{[N_0=1]} \frac{\log N}{\Delta_0} + \mathbb{I}_{[N_0 \geq 2]} \frac{(\log N)^{3/2}}{\Delta_0}.$$

CARE if you can, Hedge if you must; or, Meta-CARE for All

FTRL-CARE is *almost* adaptively minimax optimal.

Theorem BNR20

For any convex $\mathcal{D}$, FTRL-CARE achieves

$$\mathbb{E}R(T) \lesssim \sqrt{T \log N_0} + \frac{(\log N)^{3/2}}{\Delta_0}.$$

When $N_0 = 1$, it incurs total regret of order $\frac{(\log N)^{3/2}}{\Delta_0}$ instead of $\frac{(\log N)}{\Delta_0}$.
To be minimax optimal even when $N_0 = 1$, combine Hedge and FTRL-CARE.

Meta-CARE

- Treat the predictions of Hedge and FTRL-CARE as *meta-experts*.
- Use Hedge on these two meta-experts.
- Incur best regret of the two, plus negligible excess from meta-learning.

Theorem BNR20

For any convex $\mathcal{D}$, Meta-CARE achieves

$$\mathbb{E}R(T) \lesssim \sqrt{T \log N_0} + \mathbb{I}_{[N_0=1]} \frac{\log N}{\Delta_0} + \mathbb{I}_{[N_0 \geq 2]} \frac{(\log N)^{3/2}}{\Delta_0}.$$

Proof Details

Proof Details

Proof Technique 1: FTRL-CARE looks like Hedge with oracle knowledge.

Proof Details

Proof Technique 1: FTRL-CARE looks like Hedge with oracle knowledge.

Lemma BNR20

FTRL-CARE is equivalent to solving the following system of equations:

$$\eta(t+1) = c_1 \sqrt{\frac{H(w(t+1)) + c_2}{t+1}} \quad \text{and} \quad w(t+1) = \frac{\exp\{-\eta(t+1)L(t)\}}{\sum_{i\in[N]} \exp\{-\eta(t+1)L_i(t)\}}.$$

Proof Details

Proof Technique 1: FTRL-CARE looks like Hedge with oracle knowledge.

Lemma BNR20

FTRL-CARE is equivalent to solving the following system of equations:

$$\eta(t+1) = c_1 \sqrt{\frac{H(w(t+1)) + c_2}{t+1}} \quad \text{and} \quad w(t+1) = \frac{\exp\{-\eta(t+1)L(t)\}}{\sum_{i\in[N]} \exp\{-\eta(t+1)L_i(t)\}}.$$

Proof Technique 2: Concentration of measure holds under our relaxation of i.i.d.

Proof Details

Proof Technique 1: FTRL-CARE looks like Hedge with oracle knowledge.

Lemma BNR20

FTRL-CARE is equivalent to solving the following system of equations:

$$\eta(t+1) = c_1 \sqrt{\frac{H(w(t+1)) + c_2}{t+1}} \quad \text{and} \quad w(t+1) = \frac{\exp\{-\eta(t+1)L(t)\}}{\sum_{i\in[N]} \exp\{-\eta(t+1)L_i(t)\}}.$$

Proof Technique 2: Concentration of measure holds under our relaxation of i.i.d.

Lemma BNR20

For any prediction algorithm, constraint $\mathcal{D}$, and data-generating mechanism,

$$\sup_{i\in[N]\setminus\mathcal{I}_0} \mathbb{E} \min_{i_0\in\mathcal{I}_0} \exp\left\{\lambda \sum_{t=0}^{T} [\ell_{i_0}(t) - \ell_i(t)]\right\} \le \exp\left\{ T\left[\lambda^2/2 - \lambda\Delta_0\right]\right\}.$$

Proof Details

Proof Technique 1: FTRL-CARE looks like Hedge with oracle knowledge.

Lemma BNR20

FTRL-CARE is equivalent to solving the following system of equations:

$$\eta(t+1) = c_1\sqrt{\frac{H(w(t+1)) + c_2}{t+1}} \quad \text{and} \quad w(t+1) = \frac{\exp\{-\eta(t+1)L(t)\}}{\sum_{i\in[N]} \exp\{-\eta(t+1)L_i(t)\}}.$$

Proof Technique 2: Concentration of measure holds under our relaxation of i.i.d.

Lemma BNR20

For any prediction algorithm, constraint $\mathcal{D}$, and data-generating mechanism,

$$\sup_{i\in[N]\setminus\mathcal{I}_0} \mathbb{E} \min_{i_0\in\mathcal{I}_0} \exp\left\{\lambda \sum_{t=0}^{T} [\ell_{i_0}(t) - \ell_i(t)]\right\} \leq \exp\left\{ T\left[\lambda^2/2 - \lambda\Delta_0\right]\right\}.$$

Summary

Our Contributions

Our Contributions

1. Introduced a spectrum of relaxations of the I.I.D. assumption.

Our Contributions

1. Introduced a spectrum of relaxations of the I.I.D. assumption.
   - Data that we want to predict won't be purely adversarial or stochastic.

Our Contributions

1. Introduced a spectrum of relaxations of the I.I.D. assumption.
   - Data that we want to predict won't be purely adversarial or stochastic.
   - We want to know that we do well in intermediate scenarios as well.

Our Contributions

1. Introduced a spectrum of relaxations of the I.I.D. assumption.
   - Data that we want to predict won't be purely adversarial or stochastic.
   - We want to know that we do well in intermediate scenarios as well.

2. Characterized minimax regret under time-homogeneous convex constraints.

Our Contributions

1. Introduced a spectrum of relaxations of the I.I.D. assumption.
   - Data that we want to predict won't be purely adversarial or stochastic.
   - We want to know that we do well in intermediate scenarios as well.

2. Characterized minimax regret under time-homogeneous convex constraints.
   - Depends on the number of effective experts, N_0,

     and the effective stochastic gap, Δ_0.

Our Contributions

1. Introduced a spectrum of relaxations of the I.I.D. assumption.
   - Data that we want to predict won't be purely adversarial or stochastic.
   - We want to know that we do well in intermediate scenarios as well.

2. Characterized minimax regret under time-homogeneous convex constraints.
   - Depends on the number of effective experts, N_0,
     and the effective stochastic gap, Δ_0.

3. Formalized the notion of adaptive minimax optimality.

Our Contributions

1. Introduced a spectrum of relaxations of the I.I.D. assumption.
   - Data that we want to predict won't be purely adversarial or stochastic.
   - We want to know that we do well in intermediate scenarios as well.

2. Characterized minimax regret under time-homogeneous convex constraints.
   - Depends on the number of effective experts, N_0,
     and the effective stochastic gap, Δ_0.

3. Formalized the notion of adaptive minimax optimality.

4. Proved Hedge **is not** adaptively minimax optimal
   along spectrum from I.I.D. to adversarial.

Our Contributions

1. Introduced a spectrum of relaxations of the I.I.D. assumption.
   - Data that we want to predict won't be purely adversarial or stochastic.
   - We want to know that we do well in intermediate scenarios as well.

2. Characterized minimax regret under time-homogeneous convex constraints.
   - Depends on the number of effective experts, N_0,
     and the effective stochastic gap, Δ_0.

3. Formalized the notion of adaptive minimax optimality.

4. Proved Hedge **is not** adaptively minimax optimal
   along spectrum from I.I.D. to adversarial.
   - Requires oracle knowledge to get minimax optimal dependence on T and N_0.

Our Contributions

1. Introduced a spectrum of relaxations of the I.I.D. assumption.
   - Data that we want to predict won't be purely adversarial or stochastic.
   - We want to know that we do well in intermediate scenarios as well.

2. Characterized minimax regret under time-homogeneous convex constraints.
   - Depends on the number of effective experts, N_0,
     and the effective stochastic gap, Δ_0.

3. Formalized the notion of adaptive minimax optimality.

4. Proved Hedge **is not** adaptively minimax optimal
   along spectrum from I.I.D. to adversarial.
   - Requires oracle knowledge to get minimax optimal dependence on T and N_0.

5. Provided a new algorithm, Meta-CARE that **is** adaptively minimax optimal.

Our Contributions

1. Introduced a spectrum of relaxations of the I.I.D. assumption.
   - Data that we want to predict won't be purely adversarial or stochastic.
   - We want to know that we do well in intermediate scenarios as well.

2. Characterized minimax regret under time-homogeneous convex constraints.
   - Depends on the number of effective experts, N_0,
     and the effective stochastic gap, Δ_0.

3. Formalized the notion of adaptive minimax optimality.

4. Proved Hedge **is not** adaptively minimax optimal
   along spectrum from I.I.D. to adversarial.
   - Requires oracle knowledge to get minimax optimal dependence on T and N_0.

5. Provided a new algorithm, Meta-CARE that **is** adaptively minimax optimal.
   - Performs as well as possible relative to the constraint on the adversary,

Our Contributions

1. Introduced a spectrum of relaxations of the I.I.D. assumption.
   - Data that we want to predict won't be purely adversarial or stochastic.
   - We want to know that we do well in intermediate scenarios as well.

2. Characterized minimax regret under time-homogeneous convex constraints.
   - Depends on the number of effective experts, N_0,
     and the effective stochastic gap, Δ_0.

3. Formalized the notion of adaptive minimax optimality.

4. Proved Hedge **is not** adaptively minimax optimal
   along spectrum from I.I.D. to adversarial.
   - Requires oracle knowledge to get minimax optimal dependence on T and N_0.

5. Provided a new algorithm, Meta-CARE that **is** adaptively minimax optimal.
   - Performs as well as possible relative to the constraint on the adversary, without knowledge of the constraint.

Our Contributions

1. Introduced a spectrum of relaxations of the I.I.D. assumption.
   - Data that we want to predict won't be purely adversarial or stochastic.
   - We want to know that we do well in intermediate scenarios as well.

2. Characterized minimax regret under time-homogeneous convex constraints.
   - Depends on the number of effective experts, N_0,
     and the effective stochastic gap, Δ_0.

3. Formalized the notion of adaptive minimax optimality.

4. Proved Hedge **is not** adaptively minimax optimal
   along spectrum from I.I.D. to adversarial.
   - Requires oracle knowledge to get minimax optimal dependence on T and N_0.

5. Provided a new algorithm, Meta-CARE that **is** adaptively minimax optimal.
   - Performs as well as possible relative to the constraint on the adversary, without knowledge of the constraint.

References

- [CL06] N. Cesa-Bianchi and G. Lugosi. *Prediction, Learning, and Games*. Cambirdge University Press, 2006.
- [FS97] Y. Freund and R. Schapire. "A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting". *Journal of Computer and System Sciences* 55 (1 1997), pp. 119–139.
- [GSE14] P. Gaillard, G. Stoltz, and T. van Erven. "A second-order bound with excess losses". In: *Proceedings of the 27th Conference on Learning Theory*. 2014.
- [MG19] J. Mourtada and S. Gaïffas. "On the optimality of the Hedge algorithm in the stochastic regime.". *Journal of Machine Learning Research* 20.83 (2019), pp. 1–28.
- [RST11] A. Rakhlin, K. Sridharan, and A. Tewari. "Online learning: Stochastic, constrained, and smoothed adversaries". In: *Advances in Neural Information Processing Systems 25*. 2011.

References (cont.)

- [Vov98] V. Vovk. "A Game of Prediction with Expert Advice". *Journal of Computer and System Sciences* 56 (2 1998), pp. 153–173.