Learning Methods for Online Prediction Problems

Peter Bartlett
Statistics and EECS
UC Berkeley

Online Learning

- Repeated game:

  Decision method plays a_t

  World reveals $\ell_t \in \mathcal{L}$

- Aim: minimize $\hat{L}_n = \sum_{t=1}^{n} \ell_t(a_t)$.
- For example, aim to minimize regret, that is, perform well compared to the best (in retrospect) from some class:

$$\begin{aligned}\text{regret} &= \sum_{t=1}^{n} \ell_t(a_t) - \min_{a \in \mathcal{A}} \sum_{t=1}^{n} \ell_t(a) \\ &= \hat{L}_n - L_n^*.\end{aligned}$$

- Data can be adversarially chosen.

Online Learning

Minimax regret is the value of the game:

$$\min_{a_1} \max_{\ell_1} \cdots \min_{a_n} \max_{\ell_n} \left(\sum_{t=1}^{n} \ell_t(a_t) - \min_{a \in \mathcal{A}} \sum_{t=1}^{n} \ell_t(a) \right).$$

Online Learning: Motivations

1. Adversarial model is appropriate for
   - Computer security.
   - Computational finance.
2. Adversarial model assumes little:
   It is often straightforward to convert a strategy for an adversarial environment to a method for a probabilistic environment.
3. Studying the adversarial model sometimes reveals the *deterministic core* of a statistical problem: there are strong similarities between the performance guarantees in the two cases, and in particular between their dependence on the complexity of the class of prediction rules.
4. There are significant overlaps in the design of methods for the two problems:
   - *Regularization* plays a central role.
   - Many online prediction strategies have a natural interpretation as a *Bayesian method*.

Computer Security: Spam Detection

Computer Security: Spam Email Detection

- Here, the action a_t might be a classification rule, and ℓ_t is the indicator for a particular email being incorrectly classified (e.g., spam allowed through).
- The sender can determine if an email is delivered (or detected as spam), and try to modify it.
- An adversarial model allows an arbitrary sequence.
- We cannot hope for good classification accuracy in an absolute sense; regret is relative to a comparison class.
- Minimizing regret ensures that the spam detection accuracy is close to the best performance in retrospect on the particular spam sequence.

Computer Security: Spam Email Detection

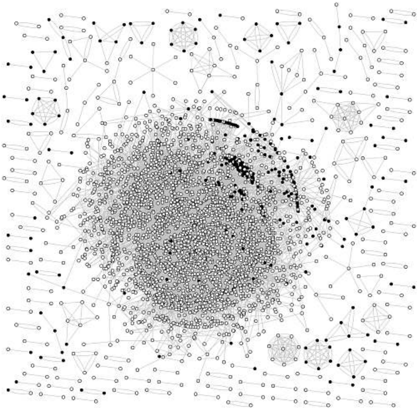
- Suppose we consider features of email messages from some set $\mathcal{X}$ (e.g., information about the header, about words in the message, about attachments).
- The decision method's action a_t is a mapping from $\mathcal{X}$ to $[0, 1]$ (think of the value as an estimated probability that the message is spam).
- At each round, the adversary chooses a feature vector $x_t \in \mathcal{X}$ and a label $y_t \in \{0, 1\}$, and the loss is defined as

$$\ell_t(a_t) = (y_t - a_t(x_t))^2 \, .$$

- The regret is then the excess squared error, over the best achievable on the data sequence:

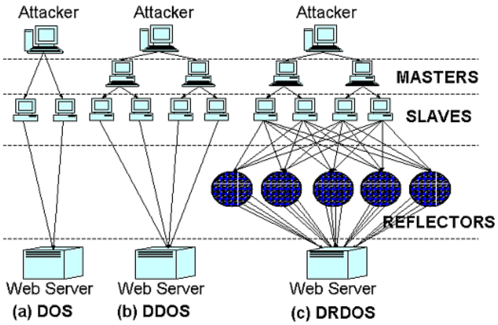
$$\sum_{t=1}^{n} \ell_t(a_t) - \min_{a \in \mathcal{A}} \sum_{t=1}^{n} \ell_t(a) = \sum_{t=1}^{n} (y_t - a_t(x_t))^2 - \min_{a \in \mathcal{A}} \sum_{t=1}^{n} (y_t - a(x_t))^2.$$

Computer Security: Web Spam Detection



Web Spam Challenge (www.iw3c2.org)

Computer Security: Detecting Denial of Service



ACM

Computational Finance: Portfolio Optimization

Computational Finance: Portfolio Optimization

- Aim to choose a portfolio (distribution over financial instruments) to maximize utility.
- Other market players can profit from making our decisions bad ones. For example, if our trades have a market impact, someone can *front-run* (trade ahead of us).
- Here, the action a_t is a distribution on instruments, and ℓ_t might be the negative logarithm of the portfolio's increase, $a_t \cdot r_t$, where r_t is the vector of relative price increases.
- We might compare our performance to the best stock (distribution is a delta function), or a set of indices (distribution corresponds to Dow Jones Industrial Average, etc), or the set of all distributions.

Computational Finance: Portfolio Optimization

- The decision method's action a_t is a distribution on the m instruments, $a_t \in \Delta^m = \{a \in [0,1]^m : \sum_i a_i = 1\}$.
- At each round, the adversary chooses a vector of returns $r_t \in \mathbb{R}_+^m$; the ith component is the ratio of the price of instrument i at time t to its price at the previous time, and the loss is defined as

$$\ell_t(a_t) = -\log\left(a_t \cdot r_t\right).$$

- The regret is then the log of the ratio of the maximum value the portfolio would have at the end (for the best mixture choice) to the final portfolio value:

$$\sum_{t=1}^{n} \ell_t(a_t) - \min_{a \in \mathcal{A}} \sum_{t=1}^{n} \ell_t(a) = \max_{a \in \mathcal{A}} \sum_{t=1}^{n} \log(a \cdot r_t) - \sum_{t=1}^{n} \log(a_t \cdot r_t).$$

Online Learning: Motivations

2. Online algorithms are also effective in probabilistic settings.

- Easy to convert an online algorithm to a batch algorithm.
- Easy to show that good online performance implies good i.i.d. performance, for example.

Online Learning: Motivations

3. Understanding statistical prediction methods.

- Many statistical methods, based on *probabilistic assumptions*, can be effective in an adversarial setting.
- Analyzing their performance in adversarial settings provides perspective on their robustness.
- We would like violations of the probabilistic assumptions to have a limited impact.

Key Points

- Online Learning:
  - repeated game.
  - aim to minimize *regret*.
  - Data can be *adversarially* chosen.
- Motivations:
  - Often appropriate (security, finance).
  - Algorithms also effective in probabilistic settings.
  - Can provide insight into statistical prediction methods.

Course Synopsis

- A finite comparison class: $\mathcal{A} = \{1, \ldots, m\}$.
- Converting online to batch.
- Online convex optimization.
- Log loss.
- Optimal regret.

Finite Comparison Class

1. "Prediction with expert advice."
2. With perfect predictions: $\log m$ regret.
3. Exponential weights strategy: $\sqrt{n \log m}$ regret.
4. Refinements and extensions:
   - Exponential weights and $L^* = 0$
   - n unknown
   - L^* unknown
   - Bayesian interpretation
   - *Convex* (versus linear) losses
5. Statistical prediction with a finite class.

Prediction with Expert Advice

Suppose we are predicting whether it will rain tomorrow. We have access to a set of m experts, who each make a forecast of 0 or 1. Can we ensure that we predict almost as well as the best expert?

Here, $\mathcal{A} = \{1, \ldots, m\}$. There are m experts, and each has a forecast sequence $f_1^i, f_2^i, \ldots$ from $\{0, 1\}$. At round t, the adversary chooses an outcome $y_t \in \{0, 1\}$, and sets

$$\ell_t(i) = 1[f_t^i \neq y_t] = \begin{cases} 1 & \text{if } f_t^i \neq y_t, \\ 0 & \text{otherwise.} \end{cases}$$

Online Learning

Minimax regret is the value of the game:

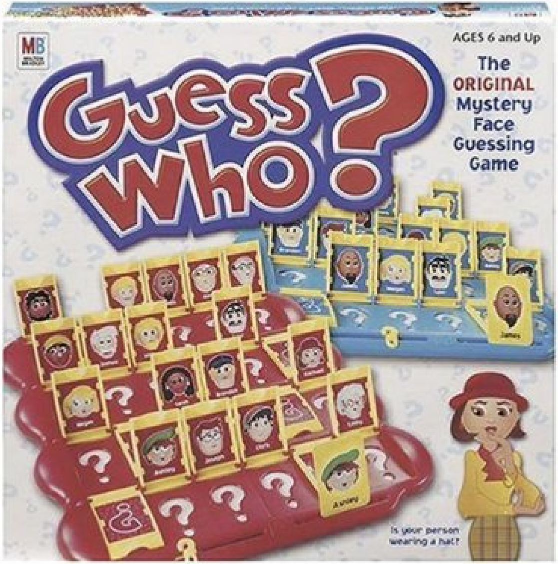
$$\min_{a_1} \max_{\ell_1} \cdots \min_{a_n} \max_{\ell_n} \left(\sum_{t=1}^{n} \ell_t(a_t) - \min_{a \in \mathcal{A}} \sum_{t=1}^{n} \ell_t(a) \right).$$

$$\hat{L}_n = \sum_{t=1}^{n} \ell_t(a_t), \qquad L_n^* = \min_{a \in \mathcal{A}} \sum_{t=1}^{n} \ell_t(a).$$

Prediction with Expert Advice

An easier game: suppose that the adversary is constrained to choose the sequence y_t so that some expert incurs no loss ($L_n^* = 0$), that is, there is an $i^* \in \{1, \ldots, m\}$ such that for all t, $y_t = f_t^{i^*}$.
How should we predict?

Prediction with Expert Advice: Guess Who?



Prediction with Expert Advice: Halving

- Define the set of experts who have been correct so far:

$$C_t = \{i : \ell_1(i) = \cdots = \ell_{t-1}(i) = 0\}.$$

- Choose a_t any element of

$$\left\{ i : f_t^i = \text{majority}\left(\{ f_t^j : j \in C_t \} \right) \right\}.$$

Theorem

This strategy has regret no more than $\log_2 m$.

Prediction with Expert Advice: Halving

Theorem

The halving strategy has regret no more than $\log_2 m$*.*

Proof.

If it makes a mistake (that is, $\ell_t(a_t) = 1$), then the minority of $\{f_t^j : j \in C_t\}$ is correct, so at least half of the experts are eliminated:

$$|C_{t+1}| \leq \frac{|C_t|}{2}.$$

And otherwise $|C_{t+1}| \leq |C_t|$ (because $|C_t|$ never increases). Thus,

$$\begin{aligned}
\hat{L}_n &= \sum_{t=1}^{n} \ell_t(a_t) \\
&\leq \log_2 \frac{|C_1|}{|C_{n+1}|} = \log_2 m - \log_2 |C_{n+1}| \leq \log_2 m.
\end{aligned}$$

Prediction with Expert Advice

The proof follows a pattern we shall see again:
find some measure of progress (here, $|C_t|$) that

- changes monotonically when excess loss is incurred (here, it halves),
- is somehow constrained (here, it cannot fall below 1, because there is an expert who predicts perfectly).

What if there is no perfect expert?
Maintaining C_t makes no sense.

Finite Comparison Class

1. "Prediction with expert advice."
2. With perfect predictions: $\log m$ regret.
3. Exponential weights strategy: $\sqrt{n \log m}$ regret.
4. Refinements and extensions:
   - Exponential weights and $L^* = 0$
   - n unknown
   - L^* unknown
   - Bayesian interpretation
   - *Convex* (versus linear) losses
5. Statistical prediction with a finite class.

Prediction with Expert Advice: Mixed Strategies

- We have m experts.
- Allow a mixed strategy, that is, a_t chosen from the simplex Δ^m—the set of distributions on $\{1, \ldots, m\}$,

$$\Delta^m = \left\{ a \in [0,1]^m : \sum_{i=1}^{m} a^i = 1 \right\}.$$

- We can think of the strategy as choosing an element of $\{1, \ldots, m\}$ randomly, according to a distribution a_t. Or we can think of it as playing an element a_t of Δ^m, and incurring the expected loss,

$$\ell_t(a_t) = \sum_{i=1}^{m} a_t^i \ell_t(e_i),$$

where $\ell_t(e_i) \in [0,1]$ is the *loss* incurred by expert i.
(e_i denotes the vector with a single 1 in the ith coordinate, and the rest zeros.)

Prediction with Expert Advice: Exponential Weights

- Maintain a set of (unnormalized) weights over experts:

$$\begin{aligned} w_0^i &= 1, \\ w_{t+1}^i &= w_t^i \exp\left(-\eta \ell_t(e_i)\right). \end{aligned}$$

- Here, $\eta > 0$ is a parameter of the algorithm.
- Choose a_t as the normalized vector,

$$a_t = \frac{1}{\sum_{i=1}^m w_t^i} w_t.$$

Prediction with Expert Advice: Exponential Weights

Theorem

The exponential weights strategy with parameter

$$\eta = \sqrt{\frac{8 \ln m}{n}}$$

has regret satisfying

$$\hat{L}_n - L_n^* \leq \sqrt{\frac{n \ln m}{2}}.$$

Exponential Weights: Proof Idea

We use a measure of progress:

$$W_t = \sum_{i=1}^{m} w_t^i.$$

1. W_n grows at least as

$$\exp\left(-\eta \min_i \sum_{t=1}^{n} \ell_t(e_i)\right).$$

2. W_n grows no faster than

$$\exp\left(-\eta \sum_{t=1}^{n} \ell_t(a_t)\right).$$

Exponential Weights: Proof 1

$$
\begin{aligned}
\ln \frac{W_{n+1}}{W_1} &= \ln\left(\sum_{i=1}^{m} w_{n+1}^i\right) - \ln m \\
&= \ln\left(\sum_{i=1}^{m} \exp\left(-\eta \sum_t \ell_t(e_i)\right)\right) - \ln m \\
&\geq \ln\left(\max_i \exp\left(-\eta \sum_t \ell_t(e_i)\right)\right) - \ln m \\
&= -\eta \min_i\left(\sum_t \ell_t(e_i)\right) - \ln m \\
&= -\eta L_n^* - \ln m.
\end{aligned}
$$

Exponential Weights: Proof 2

$$\begin{aligned}
\ln \frac{W_{t+1}}{W_t} &= \ln\left(\frac{\sum_{i=1}^m \exp(-\eta \ell_t(e_i)) w_t^i}{\sum_i w_t^i}\right) \\
&\leq -\eta \frac{\sum_i \ell_t(e_i) w_t^i}{\sum_i w_t^i} + \frac{\eta^2}{8} \\
&= -\eta \ell_t(a_t) + \frac{\eta^2}{8},
\end{aligned}$$

where we have used Hoeffding's inequality:
for a random variable $X \in [a, b]$ and $\lambda \in \mathbb{R}$,

$$\ln\left(\mathbf{E} e^{\lambda X}\right) \leq \lambda \mathbf{E} X + \frac{\lambda^2 (b-a)^2}{8}.$$

Aside: Proof of Hoeffding's inequality

Define

$$A(\lambda) = \log\left(\mathbf{E}e^{\lambda X}\right) = \log\left(\int e^{\lambda x}\, dP(x)\right),$$

where $X \sim P$. Then A is the log normalization of the exponential family random variable X_λ with reference measure P and sufficient statistic x. Since P has bounded support, $A(\lambda) < \infty$ for all λ, and we know that

$$A'(\lambda) = \mathbf{E}(X_\lambda),$$
$$A''(\lambda) = \mathrm{Var}(X_\lambda).$$

Since P has support in $[a, b]$, $\mathrm{Var}(X_\lambda) \leq (b - a)^2/4$. Then a Taylor expansion about $\lambda = 0$ (where X_λ has the same distribution as X) gives

$$A(\lambda) \leq \lambda \mathbf{E}X + \frac{\lambda^2}{8}(b - a)^2.$$

Exponential Weights: Proof

$$-\eta L_n^* - \ln m \leq \ln \frac{W_{n+1}}{W_1} \leq -\eta \hat{L}_n + \frac{n\eta^2}{8}.$$

Thus,

$$\hat{L}_n - L_n^* \leq \frac{\ln m}{\eta} + \frac{\eta n}{8}.$$

Choosing the optimal η gives the result:

Theorem

The exponential weights strategy with parameter $\eta = \sqrt{8 \ln m / n}$ has regret no more than $\sqrt{\frac{n \ln m}{2}}$.

Key Points

For a finite set of actions (experts):

- If one is perfect (zero loss), halving algorithm gives per round regret of
$$\frac{\ln m}{n}.$$
- Exponential weights gives per round regret of
$$O\left(\sqrt{\frac{\ln m}{n}}\right).$$

Prediction with Expert Advice: Refinements

1. Does exponential weights strategy give the faster rate if $L^* = 0$?
2. Do we need to know n to set η?

Prediction with Expert Advice: Refinements

1. Does exponential weights strategy give the faster rate if $L^* = 0$?
Replace Hoeffding:

$$\ln \mathbf{E} e^{\lambda X} \leq \lambda \mathbf{E} X + \frac{\lambda^2}{8},$$

with ‘Bernstein’:

$$\ln \mathbf{E} e^{\lambda X} \leq (e^{\lambda} - 1) \mathbf{E} X.$$

(for $X \in [0, 1]$).

Exponential Weights: Proof 2

$$\ln \frac{W_{t+1}}{W_t} = \ln \left(\frac{\sum_{i=1}^m \exp(-\eta \ell_t(e_i)) w_t^i}{\sum_i w_t^i} \right)$$
$$\leq \left(e^{-\eta} - 1\right) \ell_t(a_t).$$

Thus

$$\hat{L}_n \leq \frac{\eta}{1 - e^{-\eta}} L_n^* + \frac{\ln m}{1 - e^{-\eta}}.$$

For example, if $L_n^* = 0$ and η is large, we obtain a regret bound of roughly $\ln m/n$ again. And η large is like the halving algorithm (it puts roughly equal weight on all experts that have zero loss so far).

Prediction with Expert Advice: Refinements

2. Do we need to know n to set η?
   - We used the optimal setting $\eta = \sqrt{8 \ln m / n}$. But can this regret bound be achieved uniformly across time?
   - Yes; using a time-varying $\eta_t = \sqrt{8 \ln m / t}$ gives the same rate (worse constants).
   - It is also possible to set η as a function of L_t^*, the best cumulative loss so far, to give the improved bound for small losses uniformly across time (worse constants).

Prediction with Expert Advice: Refinements

3. We can interpret the exponential weights strategy as computing a Bayesian posterior.
Consider $f_t^i \in [0,1]$, $y_t \in \{0,1\}$, and $\ell_t^i = |f_t^i - y_t|$. Then consider a Bayesian prior that is uniform on m distributions.
Given the ith distribution, y_t is a Bernoulli random variable with parameter

$$\frac{e^{-\eta(1-f_t^i)}}{e^{-\eta(1-f_t^i)} + e^{-\eta f_t^i}}.$$

Then exponential weights is computing the posterior distribution over the m distributions.

Prediction with Expert Advice: Refinements

4. We could work with arbitrary convex losses on Δ^m:
We defined loss as linear in a:

$$\ell_t(a) = \sum_i a^i \ell_t(e^i).$$

We could replace this with any bounded convex function on Δ^m. The only change in the proof is an equality becomes an inequality:

$$-\eta \frac{\sum_i \ell_t(e_i) w_t^i}{\sum_i w_t^i} \leq -\eta \ell_t(a_t).$$

Prediction with Expert Advice: Refinements

But note that the exponential weights strategy only competes with the *corners* of the simplex:

Theorem

For convex functions $\ell_t : \Delta^m \to [0,1]$*, the exponential weights strategy, with* $\eta = \sqrt{8 \ln m/n}$*, satisfies*

$$\sum_{t=1}^{n} \ell_t(a_t) \leq \min_i \sum_{t=1}^{n} \ell_t(e^i) + \sqrt{\frac{n \ln m}{2}}.$$

Finite Comparison Class

1. "Prediction with expert advice."
2. With perfect predictions: $\log m$ regret.
3. Exponential weights strategy: $\sqrt{n \log m}$ regret.
4. Refinements and extensions:
   - Exponential weights and $L^* = 0$
   - n unknown
   - L^* unknown
   - Bayesian interpretation
   - *Convex* (versus linear) losses
5. Statistical prediction with a finite class.

Probabilistic Prediction Setting

Let's consider a probabilistic formulation of a prediction problem.

- There is a sample of size n drawn i.i.d. from an unknown probability distribution P on $\mathcal{X} \times \mathcal{Y}$: $(X_1, Y_1), \ldots, (X_n, Y_n)$.
- Some method chooses $\hat{f} : \mathcal{X} \to \mathcal{Y}$.
- It suffers regret

$$\mathbf{E}\ell(\hat{f}(X), Y) - \min_{f \in F} \mathbf{E}\ell(f(X), Y).$$

- Here, F is a class of functions from $\mathcal{X}$ to $\mathcal{Y}$.

Probabilistic Setting: Zero Loss

Theorem

If some $f^* \in F$ *has* $\mathbf{E}\ell(f^*(X), Y) = 0$*, then choosing*

$$\hat{f} \in C_n = \left\{ f \in F : \hat{\mathbf{E}}\ell(f(X), Y) = 0 \right\}$$

leads to regret that is

$$O\left(\frac{\log |F|}{n}\right).$$

Probabilistic Setting: Zero Loss

Proof.

$$
\begin{aligned}
\Pr(\mathbf{E}\ell(\hat{f}) \geq \epsilon) &\leq \Pr(\exists f \in F : \hat{\mathbf{E}}\ell(f) = 0,\ \mathbf{E}\ell(\hat{f}) \geq \epsilon) \\
&\leq |F|(1-\epsilon)^n \\
&\leq |F|e^{-n\epsilon}.
\end{aligned}
$$

Integrating the tail bound $\Pr(\mathbf{E}\ell(\hat{f})n/\ln|F| \geq x) \geq 1 - e^{-x}$ gives $\mathbf{E}\ell(\hat{f}) \leq c\ln|F|/n$. □

Probabilistic Setting

Theorem

Choosing $\hat{f}$ to minimize the empirical risk, $\hat{\mathbf{E}}\ell(f(X), Y)$, leads to regret that is

$$O\left(\sqrt{\frac{\log |F|}{n}}\right).$$

Probabilistic Setting

Proof.

By the triangle inequality and the definition of $\hat{f}$,

$\mathbf{E}\ell_{\hat{f}} - \min_{f\in F} \mathbf{E}\ell_f \le 2\mathbf{E}\sup_{f\in F}\left|\mathbf{E}\ell_f - \hat{\mathbf{E}}\ell_f\right|$.

$$\begin{aligned}
\mathbf{E}\sup_{f\in F}\left|\mathbf{E}\ell_f - \hat{\mathbf{E}}\ell_f\right| &= \mathbf{E}\sup_{f\in F}\left|\mathbf{E}\hat{\mathbf{E}}'\ell_f - \hat{\mathbf{E}}\ell_f\right| \\
&\le \mathbf{E}\sup_{f\in F}\left|\frac{1}{n}\sum_t \epsilon_t\left(\ell_f(X'_t, Y'_t) - \ell_f(X_t, Y_t)\right)\right| \\
&\le 2\mathbf{E}\sup_{f\in F}\left|\frac{1}{n}\sum_t \epsilon_t \ell_f(X_t, Y_t)\right| \\
&\le 2\max_{X_i, Y_i}\sqrt{\sum_t \ell(f(X_i, Y_i))^2}\,\frac{\sqrt{2\log|F|}}{n} \\
&\le 2\sqrt{\frac{2\log|F|}{n}}.
\end{aligned}$$

Key Points

For a finite function class

- If one is perfect (zero loss), minimizing empirical risk gives per round regret of
$$\frac{\ln |F|}{n}.$$
- In any case, it gives per round regret of
$$O\left(\sqrt{\frac{\ln |F|}{n}}\right).$$

just as in the adversarial setting.

Course Synopsis

- A finite comparison class: $\mathcal{A} = \{1, \ldots, m\}$.
  1. "Prediction with expert advice."
  2. With perfect predictions: $\log m$ regret.
  3. Exponential weights strategy: $\sqrt{n \log m}$ regret.
  4. Refinements and extensions.
  5. Statistical prediction with a finite class.
- Converting online to batch.
- Online convex optimization.
- Log loss.
- Optimal regret.