On-Line Learning

Nicolò Cesa-Bianchi

Università degli Studi di Milano

November 3, 2005

Background

- Theory of repeated games
  (Hannan, 1956; Blackwell, 1956)
- Compression of individual sequences
  (Lempel and Ziv, 1976)
- Gambling and portfolio selection
  (Cover, 1965 and 1991)
- Pattern classification
  (Novikov, 1962; Littlestone, 1989)

Unifying framework

Prediction with expert advice

Outline

1. Prediction with expert advice
2. Connections with game theory
3. Learning with linear experts
4. The Perceptron algorithm and its extensions
5. Mistake bounds
6. Online learning with kernels
7. From mistake bounds to risk bounds

- A forecaster predicts a binary sequence one bit at the time
- At each step $t = 1, 2, \ldots$ the forecaster predicts the t-th bit knowing the previous $t - 1$ bits

$$0100010110\,?\ldots$$

- After the prediction is made, the t-th bit is observed and the forecaster finds out whether a mistake was made

Goal

Bound the number of prediction mistakes without making any statistical assumptions on the way the data sequence is generated

The role of experts

- Want a nonstatistical framework where good forecasters can be distinguished from bad forecasters
- Any forecaster must use some map of the form

$$\text{past observations} \rightarrow \text{predictions}$$

- For each forecaster, there exists a bit sequence on which a mistake is made at each step

Competitive analysis

Compare the performance of the forecaster to that of a set of *reference forecasters* (experts)

A simple example

Forecaster competes against three experts on sequence 1101

	$t = 1$	$t = 2$	$t = 3$	$t = 4$	Mistakes
Expert 1	1	1	1	1	$M_1 = 1$
Expert 2	0	1	1	0	$M_2 = 3$
Expert 3	1	0	1	0	$M_3 = 3$
Forecaster	1	0	1	1	$M = 2$
Bit sequence	1	1	0	1	

Goal (refined)

Predict each sequence almost as well as the best expert for that sequence

A more general prediction model

- Predict an unknown sequence $y_1, y_2, \ldots \in \mathcal{Y}$ (outcome space)
- Predictions $\widehat{p}$ are chosen from $\mathcal{X}$ (decision space)
- Forecasters are scored with their cumulative loss

$$\ell(\widehat{p}_1, y_1) + \ell(\widehat{p}_2, y_2) + \ldots$$

where $\ell : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ is a loss function

Example

- Zero-one loss: $\mathcal{X} = \mathcal{Y} = \{0, 1\}$ and $\ell(\widehat{p}, y) = \mathbb{I}_{\{\widehat{p} \neq y\}}$
- Quadratic loss: $\mathcal{X} = \mathcal{Y} = [0, 1]$ and $\ell(\widehat{p}, y) = (\widehat{p} - y)^2$
- Absolute loss: $\mathcal{X} = [0, 1]$, $\mathcal{Y} = \{0, 1\}$ and $\ell(\widehat{p}, y) = |\widehat{p} - y|$

On-line prediction with expert advice

Measure performance relatively to a set of N experts

At each step $t = 1, 2, \ldots$

1. Get predictions (advice) $f_{1,t}, \ldots, f_{N,t} \in \mathcal{X}$ of the experts
2. Compute prediction $\widehat{p}_t \in \mathcal{X}$
3. Outcome $y_t \in \mathcal{Y}$ is revealed
4. Forecaster incurs loss $\ell(\widehat{p}_t, y_t)$ and each expert i incurs loss $\ell(f_{i,t}, y_t)$

Note

Experts are viewed as abstract entities, generating predictions in an unspecified way

Regret

$$
\begin{aligned}
r_{i,t} &= \ell(\widehat{p}_t, y_t) - \ell(f_{i,t}, y_t) \\
\widehat{L}_n &= \sum_{t=1}^{n} \ell(\widehat{p}_t, y_t) \qquad L_{i,n} = \sum_{t=1}^{n} \ell(f_{i,t}, y_t) \\
R_{i,n} &= \sum_{t=1}^{n} r_{i,t} = \widehat{L}_n - L_{i,n} \\
\max_{i=1,\dots,N} R_{i,n} &= \widehat{L}_n - \min_{i=1,\dots,N} L_{i,n}
\end{aligned}
$$

We want to design consistent forecasters, i.e. such that

$$
\lim_{n\to\infty} \frac{1}{n} \left(\max_{i=1,\dots,N} R_{i,n} \right) = 0
$$

for any sequence of outcomes and all choices of expert advice

Weighted average forecasters

- Assume decision space $\mathcal{X}$ is a convex subset of a linear space

$$x, x' \in \mathcal{X} \quad \Longrightarrow \quad \alpha x + (1-\alpha)x' \in \mathcal{X} \quad \text{for all } 0 \leqslant \alpha \leqslant 1$$

- If $R_{i,t-1}$ is big, then we should predict more like expert i

$$\widehat{p}_t = \frac{\sum_{i=1}^N \mu(R_{i,t-1}) f_{i,t}}{\sum_{j=1}^N \mu(R_{j,t-1})}$$

where μ is some positive monotone increasing function

- This is the weighted average forecaster

Convex loss functions

- Assume loss function $\ell(x, y)$ is convex in its first argument
- Then, the prediction $\widehat{p}_t$ satisfies

$$\ell(\widehat{p}_t, y_t) = \ell\left(\frac{\sum_{i=1}^N \mu(R_{i,t-1}) f_{i,t}}{\sum_{j=1}^N \mu(R_{j,t-1})}, y_t\right) \leqslant \frac{\sum_{i=1}^N \mu(R_{i,t-1}) \ell(f_{i,t}, y_t)}{\sum_{j=1}^N \mu(R_{j,t-1})}$$

- Using $r_{i,t} = \ell(\widehat{p}_t, y_t) - \ell(f_{i,t}, y_t)$ and rearranging we obtain that, irrespective to y_t,

$$\sum_{i=1}^N r_{i,t}\, \mu(R_{i,t-1}) \leqslant 0$$

Potential-based forecasters

- Choose $\mu = \phi'$
  where $\phi : \mathbb{R} \to \mathbb{R}$ is s.t. $\phi, \phi' \geqslant 0$ and ϕ'' exists
- Weighted average forecaster is then

$$\widehat{p}_t = \frac{\sum_{i=1}^N \phi'(R_{i,t-1}) f_{i,t}}{\sum_{j=1}^N \phi'(R_{j,t-1})}$$

Definition

Potential function $\Phi : \mathbb{R}^N \to \mathbb{R}$

$$\Phi(\mathbf{R}) = \psi\left(\sum_{i=1}^N \phi(R_i)\right)$$

where $\psi : \mathbb{R} \to \mathbb{R}$ is such that $\psi \geqslant 0, \psi' > 0, \psi'' \leqslant 0$

Blackwell condition

Then the prediction at time t becomes

$$\widehat{p}_t = \frac{\sum_{i=1}^{N} \nabla\Phi(R_{i,t-1})_i f_{i,t}}{\sum_{j=1}^{N} \nabla\Phi(R_{j,t-1})_j}$$

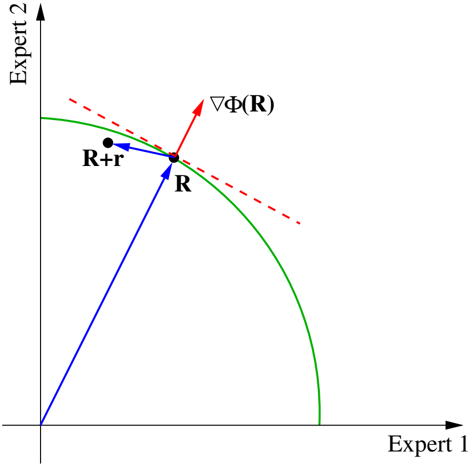
And the condition

$$\sum_{i=1}^{N} r_{i,t}\, \mu(R_{i,t-1}) \leqslant 0$$

gets rewritten as

$$\nabla\Phi(\mathbf{R}_{t-1})^\top \mathbf{r}_t \leqslant 0 \qquad \text{(Blackwell condition)}$$

Gradient descent interpretation



Proving regret bounds

Since ϕ is monotone increasing

$$\begin{aligned}\psi\left(\phi\left(\max_{i=1,\dots,N} R_{i,n}\right)\right) &= \psi\left(\max_{i=1,\dots,N} \phi(R_{i,n})\right) \\ &\leqslant \psi\left(\sum_{i=1}^{N} \phi(R_{i,n})\right) = \Phi(\mathbf{R}_n)\end{aligned}$$

Assuming ψ is also invertible,

$$\max_{i=1,\dots,N} R_{i,n} \leqslant \phi^{-1}\psi^{-1}\Big(\Phi(\mathbf{R}_n)\Big)$$

So, a bound on $\Phi(\mathbf{R}_n)$ implies a bound on $\max_{i=1,\dots,N} R_{i,n}$

Proof of the potential bound

We bound the increment $\Phi(\mathbf{R}_t) - \Phi(\mathbf{R}_{t-1})$ of the potential by taking a linear approximation of $\Phi(\mathbf{R}_t)$ around $\Phi(\mathbf{R}_{t-1})$

$$
\begin{aligned}
\Phi(\mathbf{R}_t) = \Phi(\mathbf{R}_{t-1} + \mathbf{r}_t) \\
= \quad & \Phi(\mathbf{R}_{t-1}) + \boxed{\nabla\Phi(\mathbf{R}_{t-1})^\top \mathbf{r}_t} \\
& + \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} \left.\frac{\partial^2\Phi(\mathbf{u})}{\partial u_i \partial u_j}\right|_{\mathbf{u}=\boldsymbol{\xi}} r_{i,t}\, r_{j,t} \quad (\text{for some } \boldsymbol{\xi} \in \mathbb{R}^N) \\
\leqslant \quad & \Phi(\mathbf{R}_{t-1}) + \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} \left.\frac{\partial^2\Phi(\mathbf{u})}{\partial u_i \partial u_j}\right|_{\mathbf{u}=\boldsymbol{\xi}} r_{i,t}\, r_{j,t}
\end{aligned}
$$

Proof (cont.)

As $\Phi(\mathbf{u}) = \psi\left(\sum_{k=1}^{N} \phi(u_k)\right) = \psi(\Sigma(\mathbf{u}))$, we have that

$$\frac{\partial \Phi(\mathbf{u})}{\partial u_i} = \psi'(\Sigma(\mathbf{u}))\,\phi'(u_i)$$

For $i \neq j$,

$$\frac{\partial^2 \Phi(\mathbf{u})}{\partial u_i \partial u_j} = \psi''(\Sigma(\mathbf{u}))\,\phi'(u_i)\,\phi'(u_j)$$

For $i = j$,

$$\frac{\partial^2 \Phi(\mathbf{u})}{\partial u_i^2} = \psi''(\Sigma(\mathbf{u}))\,\phi'(u_i)^2 + \psi'(\Sigma(\mathbf{u}))\,\phi''(u_i)$$

Hence,

$$\sum_{i=1}^{N}\sum_{j=1}^{N}\left.\frac{\partial^2\Phi(\mathbf{u})}{\partial u_i\,\partial u_j}\right|_{\mathbf{u}=\xi} r_{i,t}\,r_{j,t}$$

$$= \psi''\big(\Sigma(\xi)\big)\sum_{i=1}^{N}\sum_{j=1}^{N}\phi'(\xi_i)\phi'(\xi_j)r_{i,t}\,r_{j,t}$$

$$+\psi'\big(\Sigma(\xi)\big)\sum_{i=1}^{N}\phi''(\xi_i)r_{i,t}^2$$

$$= \psi''\big(\Sigma(\xi)\big)\left(\sum_{i=1}^{N}\phi'(\xi_i)r_{i,t}\right)^2+\psi'\big(\Sigma(\xi)\big)\sum_{i=1}^{N}\phi''(\xi_i)r_{i,t}^2$$

$$\leqslant \psi'\big(\Sigma(\xi)\big)\sum_{i=1}^{N}\phi''(\xi_i)r_{i,t}^2$$

Proof (cont.)

- We have proven that

$$\Phi(\mathbf{R}_t) \leqslant \Phi(\mathbf{R}_{t-1}) + \frac{1}{2} \underbrace{\max_{\mathbf{u}} \psi'\left(\sum_{k=1}^{N} \phi(u_k)\right) \sum_{i=1}^{N} \phi''(u_i) r_{i,t}^2}_{C(\mathbf{r}_t)}$$

- $C(\mathbf{r}_t)$ bounds Taylor error
- By iterating, we get $\Phi(\mathbf{R}_n) \leqslant \Phi(\mathbf{0}) + \frac{1}{2} \sum_{t=1}^{n} C(\mathbf{r}_t)$
- This holds for any forecaster satisfying $\nabla\Phi(\mathbf{R}_{t-1})^\top \mathbf{r}_t \leqslant 0$ (e.g., weighted average forecaster for convex losses)

Polynomial potential

Assume: Loss ℓ is convex and takes values in $[0,1]$

- Potential function

$$\Phi_p(\mathbf{R}) = \left(\sum_{i=1}^{N} (R_i)_+^p\right)^{2/p} = \|(\mathbf{R})_+\|_p^2 \qquad \text{for } p \geqslant 2$$

- Prediction

$$\widehat{p}_t = \frac{\sum_{i=1}^N \phi'(R_{i,t-1}) f_{i,t}}{\sum_{j=1}^N \phi'(R_{j,t-1})} = \frac{\sum_{i=1}^N (R_{i,t-1})_+^{p-1} f_{i,t}}{\sum_{j=1}^N \left(R_{j,t-1}\right)_+^{p-1}}$$

- Taylor error bound: $C(\mathbf{r}_t) = (p-1)\,\|\mathbf{r}_t\|_p^2 \leqslant (p-1)N^{2/p}$
- Bound on regret: $\max\limits_{i=1,\dots,N} R_{i,n} \leqslant \sqrt{n(p-1)N^{2/p}}$

Comments on polynomial potential

- The polynomial forecaster is consistent with rate $O(1/\sqrt{n})$
- There is a trade-off for the choice of p in the bound

$$\max_{i=1,\dots,N} R_{i,N} \leqslant \sqrt{n(p-1)N^{2/p}}$$

- Choosing $p = 2\ln N$ yields

$$\max_{i=1,\dots,N} R_{i,N} \leqslant \sqrt{(2e)n\ln N}$$

- Is this the best possible bound in term of n and N?

Exponential potential

Assume: Loss ℓ is convex and takes values in $[0,1]$

- Potential function

$$\Phi_\eta(\mathbf{R}) = \frac{1}{\eta} \ln\left(\sum_{i=1}^{N} e^{\eta R_i}\right) \qquad \text{for } \eta > 0$$

- Prediction:

$$\widehat{p}_t = \frac{\sum_{i=1}^{N} e^{\eta\left(\widehat{L}_{t-1} - L_{i,t-1}\right)} f_{i,t}}{\sum_{j=1}^{N} e^{\eta\left(\widehat{L}_{t-1} - L_{j,t-1}\right)}} = \frac{\sum_{i=1}^{N} e^{-\eta L_{i,t-1}} f_{i,t}}{\sum_{j=1}^{N} e^{-\eta L_{j,t-1}}}$$

- Taylor error bound: $C(\mathbf{r}_t) = \eta \max_{i=1,\dots,N} r_{i,t}^2 \leqslant \eta$
- Bound on regret: $\max_{i=1,\dots,N} R_{i,n} \leqslant \frac{\ln N}{\eta} + \frac{\eta}{2} n$

Tuned exponential forecaster

- Choosing $\eta = \sqrt{2(\ln N)/n}$ gets

$$\max_{i=1,\dots,N} R_{i,n} \leqslant \sqrt{2n \ln N}$$

- Better constant than polynomial potential but not consistent (horizon-dependent tuning)
- Disregarding consistency, is the constant $\sqrt{2}$ optimal?

Lower bound

- Best lower bound is for the absolute loss $\ell(p, y) = |p - y|$ ($p \in [0, 1]$ and $y \in \{0, 1\}$)

$$\max_{y_1,\dots,y_n} \frac{1}{n} \left(\max_{i=1,\dots,N} R_{i,n} \right) = (1 - o(1)) \sqrt{\frac{n}{2} \ln N}$$

for any forecasting strategy $\qquad o(1) \to 0$ for $N, n \to \infty$

- Refining the analysis of the exponential potential we get the optimal constant

$$\max_{i=1,\dots,N} R_{i,N} \leqslant \sqrt{\frac{n}{2} \ln N}$$

- But tuning of η is still horizon-dependent

More sophisticated forecasters

- Using a time-dependent tuning $\eta_t = \sqrt{8(\ln N)/t}$ we get

$$\max_{i=1,\dots,N} R_{i,N} \leqslant \sqrt{2n \ln N} + \sqrt{\frac{\ln N}{8}}$$

- Consistent, but suboptimal leading constant
- Loss-dependent tuning $\eta_t = \sqrt{c \frac{\ln N}{L^*_{t-1}}}$ gives

$$\max_{i=1,\dots,N} R_{i,N} \leqslant 2\sqrt{2L^*_n \ln N} + O(\ln N)$$

$$L^*_t = \min_{i=1,\dots,N} L_{i,t}$$

Summary

- Bound for forecasters satisfying Blackwell condition

$$\Phi(\mathbf{R}_n) \leqslant \Phi(\mathbf{0}) + \frac{1}{2}\sum_{t=1}^{n} C(\mathbf{r}_t)$$

- Polynomial potential with $p = 2\ln N$

$$\max_{i=1,\dots,N} R_{i,n} \leqslant \sqrt{(2e)n\ln N}$$

- Exponential potential with time-varying parameter

$$\max_{i=1,\dots,N} R_{i,n} \leqslant \sqrt{2n\ln N} + \sqrt{\frac{\ln N}{8}}$$

The greedy forecaster for the exponential potential

- Predicts by minimizing the increase of regret

$$\begin{aligned}\widehat{p}_t &= \operatorname*{argmin}_{\widehat{p}\in\mathcal{X}} \sup_{y_t\in\mathcal{Y}} \Phi\left(\mathbf{R}_{t-1}+\mathbf{r}_t\right)\\ &= \operatorname*{argmin}_{\widehat{p}\in\mathcal{X}} \sup_{y_t\in\mathcal{Y}} \left(\ell(\widehat{p},y_t)+\frac{1}{\eta}\ln\sum_{i=1}^{N} e^{-\eta L_{i,t}}\right)\end{aligned}$$

- Not necessarily a weighted average forecaster

Definition

A loss is mixable if for some η^* and for all $t = 1, 2, \ldots$ the greedy forecasters satisfies

$$\ell(\widehat{p}_t, y_t) \leqslant -\frac{1}{\eta}\ln\frac{\sum_{i=1}^{N} e^{-\eta^* L_{i,t}}}{\sum_{j=1}^{N} e^{-\eta^* L_{j,t-1}}}$$

Properties of mixable losses

- If a loss is mixable for η^*, then $\Phi_{\eta^*}(\mathbf{R}_n) \leqslant \Phi_{\eta^*}(\mathbf{0})$
- In fact, we have

$$\widehat{L}_n \leqslant -\frac{1}{\eta^*} \ln \frac{\sum_{j=1}^N e^{-\eta^* L_{j,n}}}{N} \leqslant L_{i,n} + \frac{\ln N}{\eta^*} \qquad \text{for any } i$$

- This immediately implies

$$\max_{i=1,\dots,N} R_{i,n} \leqslant \frac{1}{\eta^*} \ln N$$

- A constant regret bound!

Which losses are mixable?

- Assume $0 \leqslant \widehat{p}, y \leqslant 1$
- Square loss $\ell(\widehat{p}, y) = (\widehat{p} - y)^2$ is mixable for $\eta^* = 2$ (greedy $\neq$ weighted average)
- Relative entropy loss $\ell(\widehat{p}, y) = y \ln \frac{y}{\widehat{p}} + (1 - y) \ln \frac{1 - y}{1 - \widehat{p}}$ is mixable for $\eta^* = 1$ (greedy $=$ weighted average)
- Absolute loss $\ell(\widehat{p}, y) = |\widehat{p} - y|$ is not mixable for any $\eta^* > 0$

Zero-sum games

$N \times M$ known loss matrix with entries $0 \leqslant \ell(i,y) \leqslant 1$

$$\begin{array}{ccc} \ell(1,1) & \ell(1,2) & \dots \\ \ell(2,1) & \ell(2,2) & \dots \\ \vdots & \vdots & \ddots \end{array}$$

Row player has N actions
Column player has
M actions

- Players independently draw actions I (with law P) and Y (with law Q)
- Row player suffers loss $\ell(I,Y)$ (= gain of column player)
- Value of the game

$$\min_P \max_Q \sum_{i,y} \ell(i,y) P(i) Q(y) = \max_Q \min_P \sum_{i,y} \ell(i,y) P(i) Q(y)$$

Repeated zero-sum games

- Game is played repeatedly
- At round $t = 1, 2, \ldots$ players draw actions (I_t, Y_t) which may depend on past realizations of (I_s, Y_s), $s = 1, \ldots, t-1$
- Regret after n plays

$$R_{i,n} = \sum_{t=1}^{n} \ell(I_t, Y_t) - \sum_{t=1}^{n} \ell(i, Y_t)$$

- Regret is small when empirical distribution of row player's actions performs not much worse than any fixed action

Hannan consistency

Definition

A row player is Hannan consistent if

$$\limsup_{n\to\infty} \max_{i=1,\dots,N} \frac{R_{i,n}}{n} = 0 \qquad \text{with probability } 1$$

irrespective to what column player does

- H.C. implies $\limsup_{n\to\infty} \frac{1}{n}\sum_{t=1}^{n} \ell(I_t, Y_t)$ is at most the game value
- Note: Row player can beat the game value if column player is suboptimal

Randomized forecasting

At each round t:

1. **forecaster** announces distribution P_t over $\{1, \ldots, N\}$
2. **adversary** picks $y_t \in \mathcal{Y}$ and **forecaster** draws $I_t \sim P_t$
3. I_t and y_t are both revealed

- We may apply the exp. weighted average forecaster

$$P_t(i) \propto \exp\left(-\eta_t \sum_{s=1}^{t-1} \ell(i, y_s)\right) \qquad \eta_t = \sqrt{\frac{8 \ln N}{t}}$$

- This achieves Hannan consistency

$$\max_{i=1,\ldots,N} \frac{R_{i,n}}{n} \leqslant \left(\sqrt{2\frac{\ln N}{n}} + o(1)\right) \qquad \text{w.h.p.}$$

Label efficient prediction

- What if forecaster queries only m out of n adversary's actions?
- For any fixed n, we get

$$\max_{i=1,\dots,N} \frac{R_{i,n}}{n} \leqslant c\sqrt{\frac{\ln N}{m}} \qquad \text{w.h.p.}$$

- Optimal to within constants
- A query rate slightly faster than

$$\frac{(\ln n)(\ln \ln n)}{n}$$

is sufficient for Hannan consistency

Partial monitoring

- What if forecaster observes signal $h(I_t, y_t)$ instead of y_t?
- Feedback matrix H

Example

- Forecaster's action $I_t \in \{1, 2, \ldots, N\}$ is the price at which a product sold online is offered to t-th customer
- Adversary's action $y_t \in \{1, 2, \ldots, N\}$ is maximum price at which t-th customer is willing to buy the product
- Feedback matrix is

$$h(I_t, y_t) = \begin{cases} \textsc{sold} & \text{if } I_t \leqslant y_t \\ \textsc{not sold} & \text{otherwise} \end{cases}$$

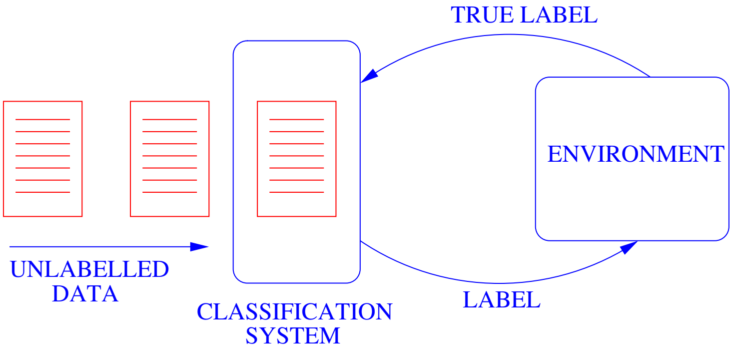
Partial monitoring and Hannan consistency

- Assume $\mathcal{Y}$ is finite
- L is the loss matrix, H is any (legal) encoding of the feedback matrix
- Sufficient (and almost necessary) condition:

$$L = K\,H \qquad \text{for some matrix } K$$

- Then per-round regret vanishes at rate $n^{-1/3}$
- Optimal, but worse than $n^{-1/2}$ of the full monitoring scenario
- Can still get $n^{-1/2}$ in special cases (e.g., nonstochastic bandits)

Binary pattern classification



Pattern classification model

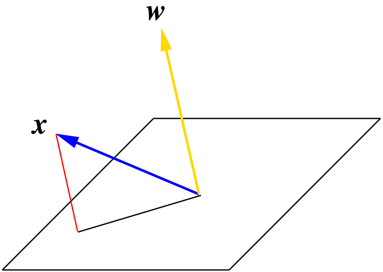
- Data instances encoded as vectors $\mathbf{x}_t \in [0,1]^d$
- A binary label $y_t \in \{-1, 1\}$ expresses some property of $\mathbf{x}_t$

Example

Given today's closing prices $\mathbf{x}_t$, predict whether tomorrow market index will increase ($y_t = 1$)

Linear classification

$$\text{predict with } \widehat{p}_t = \text{SGN}(\boldsymbol{w}_{t-1}^\top \boldsymbol{x}_t) \qquad \boldsymbol{w}_{t-1} \in \mathbb{R}^d$$



Linear classifiers (cont.)

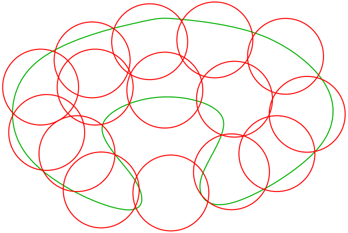
If $\widehat{p}_t \neq y_t$ then mistake at step t

Goal

On any arbitrary sequence $(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), \ldots$ perform not much worse than the best fixed linear classifier

Experts $=$ all linear classifiers

Direct application of experts' framework



- Consider the class $\mathcal{F}$ of all linear classifiers $\widehat{p}_t = \text{SGN}(\mathbf{u}^\top \mathbf{x}_t)$ for $\mathbf{u} \in \mathbb{R}^d$ with $\|\mathbf{u}\|$ bounded
- A covering of $\mathcal{F}$ has size exponential in d
- Running the weighted average forecaster on the covering requires managing an exponential number of weights

A reduction to prediction with expert advice

- Allocate d experts $F_1, \ldots, F_d$
- On instance $\boldsymbol{x}_t = (x_{t,1}, \ldots, x_{t,d})$ expert F_j predicts $x_{t,j}$

Regret

$$\boldsymbol{r}_t = y_t \, \boldsymbol{x}_t \, \mathbb{I}_{\{\widehat{p}_t \neq y_t\}}$$

We write m_t to denote $\mathbb{I}_{\{\widehat{p}_t \neq y_t\}}$

A reduction (cont.)

- Unnormalized weighted average forecaster for binary classification

$$\boldsymbol{w}_{t-1} = \nabla\Phi(\boldsymbol{R}_{t-1}) \qquad \widehat{p}_t = \operatorname{SGN}(\boldsymbol{w}_{t-1}^\top \boldsymbol{x}_t)$$

- To apply previous results we need Blackwell condition $\boldsymbol{w}_{t-1}^\top \boldsymbol{r}_t \leqslant 0$ to hold
- Indeed,

$$\boldsymbol{w}_{t-1}^\top \boldsymbol{r}_t = y_t\, \boldsymbol{w}_{t-1}^\top \boldsymbol{x}_t\, \mathbb{I}_{\{\widehat{p}_t \neq y_t\}} = \begin{cases} 0 & \text{if } \mathbb{I}_{\{\widehat{p}_t \neq y_t\}} = 0 \\ < 0 & \text{otherwise} \end{cases}$$

since $\mathbb{I}_{\{\widehat{p}_t \neq y_t\}} = 1$ iff $\operatorname{SGN}(\boldsymbol{w}_{t-1}^\top \boldsymbol{x}_t) \neq y_t$

Polynomial potential

- Recall

$$\Phi_p(\mathbf{R}) = \left(\sum_{i=1}^{N} R_i^p\right)^{2/p} = \|\mathbf{R}\|_p^2 \qquad \text{for } p \geqslant 2$$

Before we had $\Phi_p(\mathbf{R}) = \|(\mathbf{R})_+\|_p^2$
Now $(\cdot)_+$ dropped for technical reasons

- Applying the main result

$$\|\mathbf{R}_n\|_p^2 \leqslant \frac{p-1}{2}\sum_{t=1}^{n}\|\mathbf{r}_t\|_p^2 \leqslant \frac{p-1}{2}\left(\max_t \|\mathbf{x}_t\|_p\right)^2 \sum_{t=1}^{n} m_t$$

- Next we lower bound $\|\mathbf{R}_n\|_p = \sqrt{\Phi_p(\mathbf{R}_n)}$

Polynomial potential (cont.)

- For any linear classifier of parameter $\mathbf{u}$

$$\begin{aligned}
\|\mathbf{R}_n\|_p &\geqslant \mathbf{R}_n^\top \frac{\mathbf{u}}{\|\mathbf{u}\|_q} \quad \text{(by Hölder's inequality)} \\
&= \mathbf{R}_{n-1}^\top \frac{\mathbf{u}}{\|\mathbf{u}\|_q} + m_n \frac{y_n \mathbf{u}^\top \mathbf{x}_n}{\|\mathbf{u}\|_q} \\
&\geqslant \mathbf{R}_{n-1}^\top \frac{\mathbf{u}}{\|\mathbf{u}\|_q} + m_n \frac{1 - d_n(\mathbf{u})}{\|\mathbf{u}\|_q}
\end{aligned}$$

where $d_n(\mathbf{u}) = \left(1 - y_n \mathbf{u}^\top \mathbf{x}_n\right)_+$ is the hinge loss of $\mathbf{u}$

- Iterating we get

$$\|\mathbf{R}_n\|_p \geqslant \sum_{t=1}^{n} \frac{m_t - d_t(\mathbf{u})}{\|\mathbf{u}\|_q}$$

A general mistake bound

- Piecing together upper and lower bounds:

$$\sum_{t=1}^{n} \frac{m_t - d_t(\mathbf{u})}{\|\mathbf{u}\|_q} \leqslant \|\mathbf{R}_n\|_p \leqslant \sqrt{\frac{(p-1)X_p^2}{2} \sum_{t=1}^{n} m_t}$$

- Solving for $M_n = \sum_{t=1}^{n} m_t$ and overapproximating

$$M_n - D_n(\mathbf{u}) \leqslant \frac{p-1}{2} \left(X_p \|\mathbf{u}\|_q\right)^2 + \left(X_p \|\mathbf{u}\|_q\right) \sqrt{\frac{p-1}{2} D_n(\mathbf{u})}$$

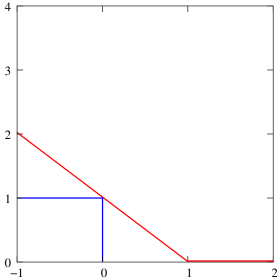
where $D_n(\mathbf{u}) = \sum_{t=1}^{n} d_t(\mathbf{u})$

This holds for all $\mathbf{u} \in \mathbb{R}^d$ and all sequences $(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \ldots \in \mathbb{R}^d \times \{-1, +1\}$

The hinge loss

The hinge loss upper bounds the mistake indicator function

$$d_t = \left(1 - y_t \mathbf{u}^\top \mathbf{x}_t\right)_+ \geqslant \textsc{sgn}(\mathbf{u}^\top \mathbf{x}_t)$$



This "regret" is a bit unfair

$$M_n - \inf_{\mathbf{u}} D_n(\mathbf{u}) \leqslant M_n - \inf_{\mathbf{u}} M_n(\mathbf{u})$$

Formulation as an incremental algorithm

We want to express $\boldsymbol{w}_t = \nabla\Phi(\mathbf{R}_t)$ recursively as $\boldsymbol{w}_t = F(\boldsymbol{w}_{t-1})$

Definition

A potential $\Phi : \mathbb{R}^d \to \mathbb{R}$ is Legendre if Φ is strictly convex, differentiable, and has a convex domain
(plus some additional technical requirements)

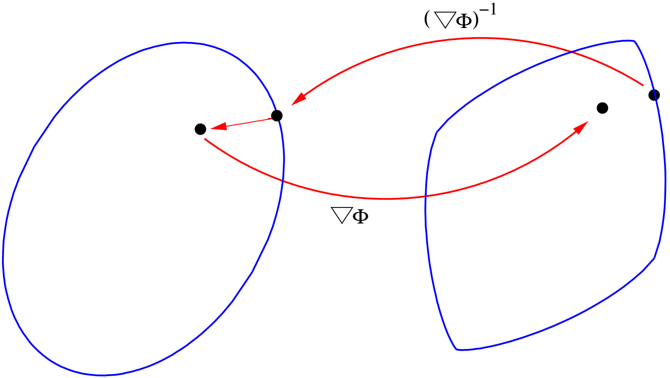
If a potential is Legendre, then $\nabla\Phi$ is invertible

$$\boldsymbol{w}_t = \nabla\Phi(\mathbf{R}_t) = \nabla\Phi(\mathbf{R}_{t-1} + \mathbf{r}_t) = \nabla\Phi\Big((\nabla\Phi)^{-1}(\boldsymbol{w}_{t-1}) + \mathbf{r}_t\Big)$$

Incremental formulation

$$\boldsymbol{w}_t = \nabla\Phi\Big((\nabla\Phi)^{-1}(\boldsymbol{w}_{t-1}) + y_t\,\boldsymbol{x}_t\,\mathbb{I}_{\{\widehat{p}_t \neq y_t\}}\Big) \qquad \text{update rule}$$

Incremental formulation (cont.)



$$\boldsymbol{w}_t = \nabla\Phi\Big((\nabla\Phi)^{-1}(\boldsymbol{w}_{t-1}) + y_t\,\boldsymbol{x}_t\,\mathbb{I}_{\{\widehat{p}_t \neq y_t\}} \Big)$$

Application to polynomial potential

- Polynomial potential $\Phi_p(\cdot) = \|\cdot\|_p^2$ is Legendre

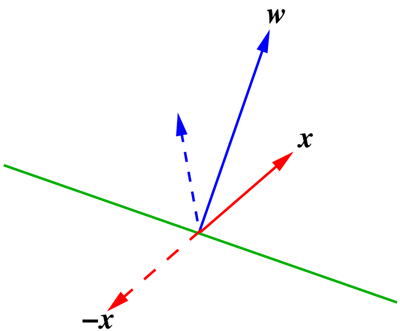
$$\left(\nabla \tfrac{1}{2}\|\mathbf{u}\|_p^2\right)_i = \frac{\mathrm{SGN}(u_i)\,|u_i|^{p-1}}{\|\mathbf{u}\|_p^{p-2}} \qquad \left(\nabla \tfrac{1}{2}\|\mathbf{u}\|_p^2\right)^{-1} = \nabla \tfrac{1}{2}\|\mathbf{u}\|_q^2$$

  where q is such that $1/p + 1/q = 1$
- When $p = 2$ we have $\nabla\Phi_2(\mathbf{R}) = \mathbf{R}$
- The incremental formulation then is simply

$$\boldsymbol{w}_t = \boldsymbol{w}_{t-1} + y_t\,\boldsymbol{x}_t\,\mathbb{I}_{\{\widehat{p}_t \neq y_t\}}$$

  the Perceptron algorithm (Rosenblatt, 1952)

The Perceptron algorithm



$$\boldsymbol{w}_t = \boldsymbol{w}_{t-1} + y_t\, \boldsymbol{x}_t\, \mathbb{I}_{\{\widehat{p}_t \neq y_t\}}$$

Mistake bounds

We can specialize our previous bound by setting $p = 2$

$$M_n - D_n(\mathbf{u}) \leqslant \left(X_2 \|\mathbf{u}\|_2\right)^2 + \left(X_2 \|\mathbf{u}\|_2\right) \sqrt{D_n(\mathbf{u})}$$

Definition

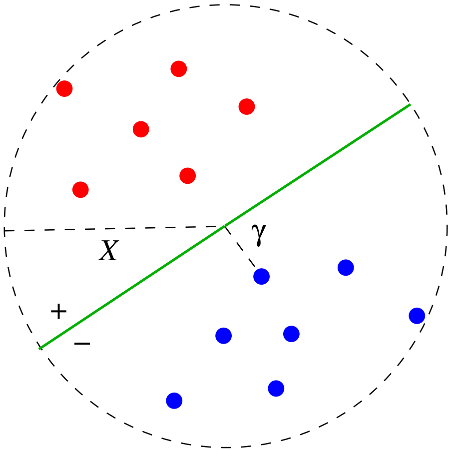
A sequence $(\boldsymbol{x}_1, y_1), \dots, (\boldsymbol{x}_n, y_n)$ is linearly separable with margin $\gamma > 0$ if there exists $\mathbf{u} \in \mathbb{R}^d$ with $\|\mathbf{u}\|_2 = 1$ such that

$$\min_{t=1,\dots,n} y_t \, \mathbf{u}^\top \boldsymbol{x}_t \geqslant \gamma$$

In this case we recover the Perceptron convergence theorem

$$M_n \leqslant \left(\frac{X_2}{\gamma}\right)^2$$

Linear separability



Exponential potential

- Recall

$$\Phi_\eta(\mathbf{R}) = \frac{1}{\eta} \ln \left(\sum_{i=1}^{d} e^{\eta \, R_i} \right)$$

- The weights have form

$$\boldsymbol{w}_t = \nabla \Phi_\eta(\mathbf{R}_t) = \frac{e^{\eta \, R_{i,n}}}{\displaystyle\sum_{k=1}^{d} e^{\eta \, R_{k,n}}}$$

- Note that $\boldsymbol{w}_t$ belongs to the simplex in $\mathbb{R}^d$

Incremental formulation

- Problem: Φ_η is not strictly convex because $\nabla\Phi_\eta(\mathbf{R})$ is constant along the line $R_1 = R_2 = \cdots = R_d$
- Thus, Φ is not Legendre and $(\nabla\Phi)^{-1}$ is not defined
- However, the potential $\Phi_{\exp}(\mathbf{R}) = e^{R_1} + \cdots + e^{R_d}$ is Legendre with

$$
\begin{aligned}
\nabla\Phi_{\exp}(\mathbf{R}) &= \left(e^{R_1}, \dots, e^{R_d}\right) \\
(\nabla\Phi_{\exp})^{-1}(\boldsymbol{w}') &= \left(\ln w'_1, \dots, \ln w'_d\right)
\end{aligned}
$$

- We get the following update rule

$$
\begin{aligned}
w'_{i,t} &= \left[\nabla\Phi_{\exp}\Big((\nabla\Phi_{\exp})^{-1}(\boldsymbol{w}'_{t-1}) + \eta\,\mathbf{r}_{t-1}\Big)\right]_i \\
&= \exp\left(\ln w'_{i,t-1} + \eta\, r_{i,t-1}\right) \\
&= w'_{i,t-1}\, e^{\eta\, r_{i,t-1}}
\end{aligned}
$$

Incremental formulation (cont.)

- Summarizing, we have

$$
\begin{aligned}
\boldsymbol{w}'_t &= \nabla\Phi_{\exp}\Big(\big(\nabla\Phi_{\exp}\big)^{-1}\big(\boldsymbol{w}'_{t-1}\big) + \eta\,\mathbf{r}_{t-1}\Big) \\
w_{i,t} &= \Big[\nabla\Phi_\eta\big(\mathbf{R}_t\big)\Big]_i = \frac{e^{\eta\,R_{i,t}}}{\displaystyle\sum_{k=1}^{d} e^{\eta\,R_{k,t}}} = \frac{w'_{i,t}}{\displaystyle\sum_{k=1}^{d} w'_{k,t}}
\end{aligned}
$$

- This is the Winnow algorithm (Littlestone, 1988)

Exponential potential (cont.)

- Applying the main result

$$
\begin{aligned}
\ln \Phi_\eta(\mathbf{R}_n) &\leqslant \frac{\ln d}{\eta} + \frac{\eta}{2} \sum_{t=1}^{n} \left(\max_{i=1,\dots,d} r_{i,t}^2 \right) \\
&\leqslant \frac{\ln d}{\eta} + \frac{\eta}{2} (X_\infty)^2 M_n
\end{aligned}
$$

- As before, we lower bound $\ln \Phi_\eta(\mathbf{R}_n)$

- Recalling that $\boldsymbol{w}$ belong to the simplex, we apply the log-sum inequality

$$\ln \sum_{i=1}^{d} v_i \geqslant \sum_{i=1}^{d} u_i \ln v_i + H(\mathbf{u}) \qquad u_i, v_i \geqslant 0 \quad \sum_{i=1}^{d} u_i = 1$$

for any linear classifier $\mathbf{u}$ in the simplex

- Applying this to the exponential potential we get

$$\ln \Phi_\eta(\mathbf{R}_n) = \frac{1}{\eta} \ln \sum_{i=1}^{d} e^{\eta R_{i,n}} \geqslant \mathbf{R}_n^\top \mathbf{u} + H(\mathbf{u})$$

Dropping $H(\mathbf{u})$ and proceeding as for the polynomial potential we get

$$\ln \Phi_\eta(\mathbf{R}_n) \geqslant \sum_{t=1}^{n} \big(m_t - d_t(\mathbf{u})\big) = M_n - D_n(\mathbf{u})$$

- Piecing together upper and lower bound

$$M_n - D_n(\mathbf{u}) \leqslant \ln \Phi_\eta(\mathbf{R}_n) \leqslant \frac{\ln d}{\eta} + \frac{\eta}{2}(X_\infty)^2 M_n$$

- Solving for M_n

$$M_n \leqslant \frac{1}{1-\eta\,(X_\infty)^2/2}\left(D_n(\mathbf{u}) + \frac{\ln d}{\eta}\right)$$

for any sequence $(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \ldots \in \mathbb{R}^d \times \{-1, +1\}$ and for any $\mathbf{u}$ in the simplex.

- Same tuning problem as in prediction with expert advice
- Proper choice of $\eta = \eta(D_n, X_\infty, d)$ gives

$$M_n - D_n(\mathbf{u}) \leqslant X_\infty \sqrt{2D_n(\mathbf{u}) \ln d} + \big(1 + o(1)\big)(X_\infty)^2 \ln d$$

where $o(1) \to 0$ for $D(\mathbf{u}) \to \infty$

Beyond the simplex

- Recall that, for the exponential potential, the difference $M_n - D_n(\mathbf{u})$ is bounded for all $\mathbf{u}$ in the simplex
- If we consider $\mathbf{u}$ in the scaled simplex $\{\mathbf{u} \in \mathbb{R}^d \,:\, u_i \geqslant 0,\, u_1 + \cdots + u_d = U\}$ we get

$$M_n - D_n(\mathbf{u}) \leqslant (X_\infty U)\sqrt{2D_n(\mathbf{u}) \ln d} + (1+o(1))(X_\infty U)^2 \ln d$$

- To remove the constraint $u_i \geqslant 0$ we can transform the instances

$$\mathbf{x} = (x_1, \ldots, x_d) \mapsto (x_1, -x_1, \ldots, x_d, -x_d)$$

at the price of replacing d by $2d$

Comparison between poly. and exp. potential

The two bounds on $M_n - D_n(\mathbf{u})$

$$\frac{p-1}{2}\left(X_p \|\mathbf{u}\|_q\right)^2 \quad + \quad \left(X_p \|\mathbf{u}\|_q\right)\sqrt{\frac{p-1}{2}D_n(\mathbf{u})}$$

$$(1+o(1))\ln(2d)\,(X_\infty \|\mathbf{u}\|_1)^2 \quad + \quad (X_\infty \|\mathbf{u}\|_1)\sqrt{2D_n(\mathbf{u})\ln(2d)}$$

- Bound for exp. pot. assumes tuning (previous knowledge of X_∞ and choice of $\|\mathbf{u}\|_1$)
- Both bounds depend on pairs of dual norms: $\|\mathbf{x}\|_p \, \|\mathbf{u}\|_q$ vs. $\|\mathbf{x}\|_\infty \, \|\mathbf{u}\|_1$
- For $p \approx 2\ln d$ the bounds are essentially equal

Comparison for spherical potential

- Consider a sequence $(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2) \dots$ such that $\boldsymbol{x}_t \in \{-1, 1\}^d$ and $y_t = \text{SGN}(x_{1,t})$
- Then $\boldsymbol{u} = (1, 0, \dots, 0)$ is an optimal classifier (no loss)
- Moreover,

$$\left(\|\boldsymbol{u}\|_2 X_2\right)^2 = d \qquad \text{and} \qquad \left(\|\boldsymbol{u}\|_1 X_\infty\right)^2 = 1$$

- Then

$$\begin{aligned} M_n &\leqslant d \qquad \text{(polynomial potential, } p = 2\text{)} \\ M_n &\leqslant 4 \ln(2d) \qquad \text{(exponential potential)} \end{aligned}$$

  an exponential advantage (verified by experiments)
- Opposite situation when instances $\boldsymbol{x}_t$ are sparse and best expert $\boldsymbol{u}$ is dense

On-line learning with kernels

- Feature map $\phi : \mathbb{R}^d \to \text{RKHS}$
- Kernel $K(\boldsymbol{x}, \boldsymbol{x}') = \langle \phi(\boldsymbol{x}), \phi(\boldsymbol{x}') \rangle$
- Assume a linear algorithm learns $\boldsymbol{w}$ such that

$$\boldsymbol{w} = \sum_i \alpha_i \boldsymbol{x}_{t_i}$$

- Then we can learn $w = \sum_i \alpha_i \phi(\boldsymbol{x}_{t_i})$ in the RKHS because

$$\begin{aligned}
\operatorname{SGN}\big(\langle w, \phi(\boldsymbol{x})\rangle\big) &= \operatorname{SGN}\left(\sum_i y_{t_i} \langle \phi(\boldsymbol{x}_{t_i}), \phi(\boldsymbol{x}) \rangle\right) \\
&= \operatorname{SGN}\left(\sum_i y_{t_i} K(\boldsymbol{x}_{t_i}, \boldsymbol{x})\right)
\end{aligned}$$

Checking applicability of kernels

Let $\mathbf{R}_t = \sum_t y_t \mathbf{x}_t \mathbb{I}_{\{\widehat{p}_t \neq y_t\}}$

- Winnow $w_{i,t} = \dfrac{e^{\eta\, R_{i,t}}}{\displaystyle\sum_{k=1}^{d} e^{\eta\, R_{k,t}}}$
- p-norm Perceptron $w_{i,t} = \dfrac{\mathrm{SGN}(R_{i,t})\, |R_{i,t}|^{p-1}}{\|\mathbf{R}_t\|_p^{p-2}}$
- Perceptron $\boldsymbol{w}_t = \mathbf{R}_t$

Perceptron's potential is spherical $\rightarrow$ rotational invariance

Kernel Perceptron

Initialize $\mathcal{L} = \emptyset$

For $t = 1, 2, \ldots$

1. Read next example $(\boldsymbol{x}_t, y_t)$
2. Compute prediction $\widehat{p} = \text{SGN}\left(\sum_{k \in \mathcal{L}} y_k K(\boldsymbol{x}_k, \boldsymbol{x}_t)\right)$
3. If $\widehat{p}_t \neq y_t$ then $\mathcal{L} \leftarrow \mathcal{L} \cup \{t\}$

Now mistake bounds are extended to the whole RKHS

$$M_n \leqslant D_n(f) + \left(\max_t K(\boldsymbol{x}_t, \boldsymbol{x}_t)\right) \|f\|^2 + \ldots$$

for any f in the RKHS

- Linear classifiers $H(\boldsymbol{x}) = \textsc{sgn}\big(\boldsymbol{w}^\top \boldsymbol{x}\big)$
- Examples $(\boldsymbol{x}_t, y_t)$ are i.i.d. according to a fixed and unknown probability distribution on $\mathbb{R}^d \times \{-1, +1\}$
- $\texttt{risk}(H) = \mathbb{P}\big(H(\boldsymbol{x}) \neq y\big)$
- Learning algorithm

$$(\boldsymbol{x}_1, y_1), \dots, (\boldsymbol{x}_n, y_n) \quad \longrightarrow \boxed{\quad A \quad} \longrightarrow \quad \widehat{H} : \mathbb{R}^d \to \{-1, +1\}$$

$\widehat{H}$ is (random) hypothesis output by learner

Data-dependent VC theory

- $\mathcal{H}$ is set of classifiers from which $\widehat{H}$ is selected
- For all $H \in \mathcal{H}$

$$
\begin{aligned}
& \mathtt{risk}(H) - \mathtt{risk}_{\mathtt{emp}}(H) \\
& \quad \leqslant \quad c_1 \sqrt{\mathtt{risk}_{\mathtt{emp}}(H) \frac{V_{\mathcal{H}} \ln n}{n}} + c_2 \frac{V_{\mathcal{H}} \ln n}{n} \quad \text{w.h.p.}
\end{aligned}
$$

- VC theory of generalization studies properties of $\mathcal{H}$
- We use on-line learning to study a small subclass of $\mathcal{H}$ generated by the interaction with the training data

The ensemble of hypotheses

- Run an incremental learner on the training set
- Everytime $H(x_t) \neq y_t$, H is changed by the update rule
- This process generates an ensemble of classifiers

$$H_0, H_1, \ldots, H_n$$

Goals

1. Bound the average risk of the ensemble in terms of the size of the ensemble
2. Find an element of the ensemble whose risk is close to the ensemble average

Step 1: bound the average risk

The difference

$$\mathtt{risk}(H_{t-1}) - \mathbb{I}_{\{H_{t-1}(x_t) \neq y_t\}}$$

is a martingale difference sequence because

$$\mathbb{E}\left[\mathtt{risk}(H_{t-1}) - \mathbb{I}_{\{H_{t-1}(x_t) \neq y_t\}} \,\middle|\, (x_1, y_1), \dots, (x_{t-1}, y_{t-1})\right] = 0$$

The associated martingale is

$$\sum_{t=1}^{n} \left(\mathtt{risk}(H_{t-1}) - \mathbb{I}_{\{H_{t-1}(x_t) \neq y_t\}}\right)$$

$$\iff \underbrace{\frac{1}{n}\sum_{t=1}^{n} \mathtt{risk}(H_{t-1})}_{\text{average risk}} - \underbrace{\frac{1}{n}\sum_{t=1}^{n} \mathbb{I}_{\{H_{t-1}(x_t) \neq y_t\}}}_{\text{fraction of mistakes}}$$

Bernstein's bound

If $Z_1, Z_2, \ldots$ is a martingale difference sequence with increments bounded by 1 and

$$V_n = \sum_{t=1}^{n} \mathbb{E}\left[Z_t^2 \mid Z_1, \ldots, Z_{t-1}\right]$$

then for all $S, K > 0$

$$\mathbb{P}\left(\sum_{t=1}^{n} Z_n \geqslant S, \quad V_n \leqslant K\right) \leqslant \exp\left(-\frac{S^2}{2(S/3 + K)}\right)$$

Application of Bernstein's bound

Since $0 \leqslant \mathbb{I}_{\{H(x) \neq y\}} \leqslant 1$,

$$\begin{aligned} &\mathrm{VAR}\left[\mathbb{I}_{\{H_{t-1}(x_t), y_t\}} \,\middle|\, (x_1, y_1), \dots, (x_{t-1}, y_{t-1}) \right] \\ &\leqslant \quad \mathbb{E}\left[\mathtt{risk}(H_{t-1}) \,\middle|\, (x_1, y_1), \dots, (x_{t-1}, y_{t-1}) \right] \end{aligned}$$

Applying Bernstein's gives

$$\frac{1}{n} \sum_{t=1}^{n} \mathtt{risk}(H_{t-1}) \quad \leqslant \quad \frac{M_n}{n} + \frac{c}{n} \left(\ln M_n + \sqrt{M_n \ln M_n} \right) \quad \text{w.h.p.}$$

Recall

$$\frac{M_n}{n} = \frac{1}{n} \sum_{t=1}^{n} \mathbb{I}_{\{H_{t-1}(x_t) \neq Y_t\}} \qquad \text{is the fraction of mistakes}$$

Step 2: pick a good classifier in the ensemble

- Start from the ensemble $H_0, H_1, \ldots, H_n$
- Do the following:
  1. test each H_t on $(x_{t+1}, y_{t+1}), \ldots, (x_n, y_n)$
  2. pick $\widehat{H} = H_{t^*}$ minimizing a penalized risk estimate

Guaranteed bound

$$\mathtt{risk}(\widehat{H}) \quad \leqslant \quad \frac{M_n}{n} + \frac{c}{n}\left((\ln n)^2 + \sqrt{M_n \, \ln n}\right) \qquad \text{w.h.p.}$$

Conclusions!

- We made a reduction from pattern classification to prection with experts
- Potential-based forecasters have on-line classifiers as natural counterparts
- We get a different viewpoint on kernel-based learning
- A simple large deviation inequality is enough to get data-dependent tail risk bounds