Machine Learning Theory 2022
Lecture 8

Wouter M. Koolen

Download these slides now from elo.mastermath.nl!

- Online Learning Intro
- Basic Protocol
- Basic Algorithms
- Basic Performance Guarantees

midterm in
TA session

Online Learning Intro

The Need to Go Beyond IID

When/why might IID be an **unreasonable** assumption?

- When humans (other learning systems) are in the loop
- When predictions are turned into actions
- ...

The Need to Go Beyond IID

When/why might IID be an **unreasonable** assumption?

- When humans (other learning systems) are in the loop
- When predictions are turned into actions
- ...

Example (Fashion)

What shirt will consumers buy in spring?

The Need to Go Beyond IID

When/why might IID be an **unreasonable** assumption?

- When humans (other learning systems) are in the loop
- When predictions are turned into actions
- ...

Example (Fashion)

What shirt will consumers buy in spring? Hypotheses:

- Blue
- White
- Tiger print

The Need to Go Beyond IID

When/why might IID be an **unreasonable** assumption?

- When humans (other learning systems) are in the loop
- When predictions are turned into actions
- ...

Example (Fashion)

What shirt will consumers buy in spring? Hypotheses:

- Blue
- White
- Tiger print

Collect data revealing preferences ⇒ Compute ERM ⇒ **Tiger print**

The Need to Go Beyond IID

When/why might IID be an **unreasonable** assumption?

- When humans (other learning systems) are in the loop
- When predictions are turned into actions
- ...

Example (Fashion)

What shirt will consumers buy in spring? Hypotheses:

- Blue
- White
- Tiger print

Collect data revealing preferences ⇒ Compute ERM ⇒ **Tiger print**
Mass produce Tiger print shirts ⇒ Huge profit!

The Need to Go Beyond IID

When/why might IID be an **unreasonable** assumption?

- When humans (other learning systems) are in the loop
- When predictions are turned into actions
- ...

Example (Fashion)

What shirt will consumers buy in spring? Hypotheses:

- Blue
- White
- Tiger print

Collect data revealing preferences $\Rightarrow$ Compute ERM $\Rightarrow$ **Tiger print**
Mass produce Tiger print shirts $\Rightarrow$ Huge profit!
Forward to next spring ...

The Need to Go Beyond IID

When/why might IID be an **unreasonable** assumption?

- When humans (other learning systems) are in the loop
- When predictions are turned into actions
- ...

Example (Fashion)

What shirt will consumers buy in spring? Hypotheses:

- Blue
- White
- Tiger print

Collect data revealing preferences ⇒ Compute ERM ⇒ **Tiger print**
Mass produce Tiger print shirts ⇒ Huge profit!
Forward to next spring ...
Hardly anyone buys Tiger print shirts.

The Need to Go Beyond IID

When/why might IID be an **unreasonable** assumption?

- When humans (other learning systems) are in the loop
- When predictions are turned into actions
- ...

Example (Fashion)

What shirt will consumers buy in spring? Hypotheses:

- Blue
- White
- Tiger print

Collect data revealing preferences ⇒ Compute ERM ⇒ **Tiger print**
Mass produce Tiger print shirts ⇒ Huge profit!
Forward to next spring ...
Hardly anyone buys Tiger print shirts.
Why not?

The Need to Go Beyond IID

When/why might IID be an **unreasonable** assumption?

- When humans (other learning systems) are in the loop
- When predictions are turned into actions
- ...

Example (Fashion)

What shirt will consumers buy in spring? Hypotheses:

- Blue
- White
- Tiger print

Collect data revealing preferences ⇒ Compute ERM ⇒ **Tiger print**
Mass produce Tiger print shirts ⇒ Huge profit!
Forward to next spring ...
Hardly anyone buys Tiger print shirts.
Why not?
Our mass production **changed** consumer preferences.

Online learning focus

Main idea

No assumptions about the data $\Leftrightarrow$ An evil opponent controls the data.

Online learning focus

Main idea

No assumptions about the data $\Leftrightarrow$ An evil opponent controls the data.

Is learning possible? When/how/what does it even mean?

Online learning focus

Main idea

No assumptions about the data $\Leftrightarrow$ An evil opponent controls the data.

Is learning possible? When/how/what does it even mean?

Change of setup/perspective/emphasis

- Tight feedback loop (recurring prediction task)
- Continuous learning (no training/learning separation)
- Adversarial analysis (Prequential principle, individual sequence. There is only the data. Also establishes robustness of statistical estimators.)
- Emphasis on both computational and statistical performance
- Regret: relative notion of performance

Application domains

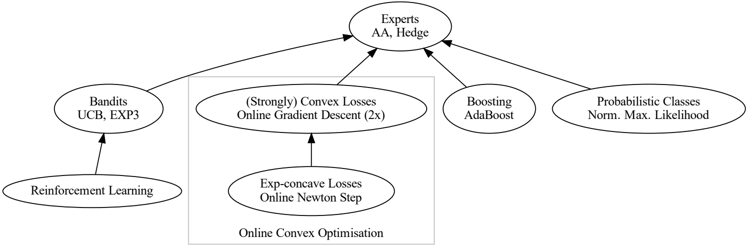
Truly sequential problems:

- electricity demand prediction (EDF, also Amazon)
- mobile device power management
- hybrid cars engine switching
- caching
- medical trials (bandits)
- online advertisement (bandits)
- weather forecasting
- data compression (CTW)
- statistical testing
- investment (Universal portfolios)
- input assistants (e.g. Dasher)
- prediction with expert advice (meld human and machine prediction)
- online convex optimisation

Wider application

- Big data sets (transport state of online algorithm instead of data, online to batch conversion)
- Convex optimisation
- Game theory (online learning methods for approximate equilibrium)
- General understanding
  - Uncertainty and ways to manipulate it
  - Makeup of and patterns in data
  - Complexity of classes of strategies

Overview of Second Half of Course



Material: course notes and selection of sources on MLT website.

The menu for today

Two fundamental and prototypical online learning problems

- The mix-loss game
  - Aggregating Algorithm
  - Performance analysis
- The dot-loss game
  - Hedge Algorithm
  - Performance analysis

The Mix Loss Game

Mix-loss game

Protocol

- For $t = 1, 2, \ldots$
  - Learner chooses a distribution $\boldsymbol{w}_t \in \triangle_K$ on K "experts".
  - Adversary reveals loss vector $\boldsymbol{\ell}_t \in (-\infty, \infty]^K$.
  - Learner's loss is the **mix loss** $-\ln\left(\sum_{k=1}^K w_t^k e^{-\ell_t^k}\right)$

Mix-loss game

Protocol

- For $t = 1, 2, \dots$
  - Learner chooses a distribution $\boldsymbol{w}_t \in \triangle_K$ on K "experts".
  - Adversary reveals loss vector $\boldsymbol{\ell}_t \in (-\infty, \infty]^K$.
  - Learner's loss is the **mix loss** $-\ln\left(\sum_{k=1}^K w_t^k e^{-\ell_t^k}\right)$

Instances:

- Investment (loss is *negative log-growth*)
- Data compression (loss is *code length*)
- Probability forecasting (loss is *logarithmic loss*)

Mix-loss game

Protocol

- For $t = 1, 2, \ldots$
  - Learner chooses a distribution $\boldsymbol{w}_t \in \triangle_K$ on K "experts".
  - Adversary reveals loss vector $\boldsymbol{\ell}_t \in (-\infty, \infty]^K$.
  - Learner's loss is the **mix loss** $-\ln\left(\sum_{k=1}^K w_t^k e^{-\ell_t^k}\right)$

Instances:

- Investment (loss is *negative log-growth*)
- Data compression (loss is *code length*)
- Probability forecasting (loss is *logarithmic loss*)

Connection to statistical learning:

- For any finite hypothesis class $\mathcal{H} = \{h_1, \ldots, h_K\}$ of binary classifiers, we may consider $\ell_t^k = \mathbf{1}\left[h_k(x_t) \neq y_t\right]$.

Two useful properties of the mix loss

Fact

Mix loss passes on additive constant $c \in \mathbb{R}$:

$$-\ln\left(\sum_{k=1}^{K} w_t^k e^{-(\ell_t^k + c)}\right) \;=\; c - \ln\left(\sum_{k=1}^{K} w_t^k e^{-\ell_t^k}\right)$$

Two useful properties of the mix loss

Fact

Mix loss passes on additive constant $c \in \mathbb{R}$*:*

$$-\ln\left(\sum_{k=1}^{K} w_t^k e^{-(\ell_t^k + c)}\right) \;=\; c - \ln\left(\sum_{k=1}^{K} w_t^k e^{-\ell_t^k}\right)$$

Fact

Mix loss of deterministic prediction $\boldsymbol{w}_t = \boldsymbol{e}_j \in \triangle_K$ *equals* ℓ_t^j*:*

$$-\ln\left(\sum_{k=1}^{K} w_t^k e^{-\ell_t^k}\right) \;=\; -\ln\left(e^{-\ell_t^j}\right) \;=\; \ell_t^j$$

Mix-loss objective

Obviously we cannot guarantee small loss.
Idea: relative evaluation, i.e. seek performance close to best expert.

Definition (Regret)

After T rounds of the mix-loss game, the *regret* is given by

$$R_T = \underbrace{\sum_{t=1}^{T} - \ln \left(\sum_{k=1}^{K} w_t^k e^{-\ell_t^k} \right)}_{\text{Learner's mix loss}} - \underbrace{\min_k \sum_{t=1}^{T} \ell_t^k}_{\text{loss of best expert}}$$

Goal: design a strategy for Learner that guarantees low regret.

Worst-case regret and Minimax regret

A *strategy* for the learner assigns to each **history** $(\boldsymbol{w}_1, \boldsymbol{\ell}_1), \ldots, (\boldsymbol{w}_{t-1}, \boldsymbol{\ell}_{t-1})$ a **next action** $\boldsymbol{w}_t$.

Definition (Worst-case regret)

The *worst-case regret* of a strategy S for the learner is

$$\max_{\boldsymbol{\ell}_1} \cdots \max_{\boldsymbol{\ell}_T} R_T$$

where the $\boldsymbol{w}_t$ are chosen according to S.

Definition (Minimax regret)

The *minimax regret* of the mix loss game is

$$\min_{\text{learner strategy}} \text{worst-case regret} \;=\; \min_{\boldsymbol{w}_1} \max_{\boldsymbol{\ell}_1} \cdots \min_{\boldsymbol{w}_T} \max_{\boldsymbol{\ell}_T} R_T$$

Mix-loss regret: lower bound (adversary construction)

Theorem

Any strategy for Learner has worst-case regret $\geq \ln K$, already in $T = 1$ round.

Proof.

Look at $k_{\text{low}} \in \arg\min_k w_1^k$ so that $w_1^{k_{\text{low}}} \leq \frac{1}{K}$.
Administer loss killing everyone but k_{low}

$$\ell_1^k \;=\; \begin{cases} \infty & k \neq k_{\text{low}} \\ 0 & k = k_{\text{low}} \end{cases}$$

Now Learner's mix loss equals

$$-\ln\left(\sum_{k=1}^K w_1^k e^{-\ell_1^k}\right) \;=\; -\ln\left(w_1^{k_{\text{low}}} e^{-\ell_t^{k_{\text{low}}}}\right) \;\geq\; \ln K + \ell_t^{k_{\text{low}}}$$

□

The Aggregating Algorithm for mix loss

Definition (Aggregating Algorithm)

The *Aggregating Algorithm* plays weights in round t:

$$w_t^k \;=\; \frac{e^{-\sum_{s=1}^{t-1} \ell_s^k}}{\sum_{j=1}^K e^{-\sum_{s=1}^{t-1} \ell_s^j}} \qquad \text{(AA)}$$

or, equivalently, $w_1^k = \frac{1}{K}$ and

$$w_{t+1}^k \;=\; \frac{w_t^k e^{-\ell_t^k}}{\sum_{j=1}^K w_t^j e^{-\ell_t^j}} \qquad \text{(AA, incremental)}$$

Many names

- (Generalisation of) Bayes' rule
- Exponentially weighted average

Mix-loss regret: upper bound (algorithm)

Theorem

The regret of the Aggregating Algorithm is at most $R_T \leq \ln K$ for all $T \geq 0$.

Proof.

Crucial observation is that mix loss *telescopes*

$$\begin{aligned}
\sum_{t=1}^{T} -\ln\left(\sum_{k=1}^{K} w_t^k e^{-\ell_t^k}\right) &= \sum_{t=1}^{T} -\ln\left(\sum_{k=1}^{K} \frac{e^{-\sum_{s=1}^{t-1} \ell_s^k}}{\sum_{j=1}^{K} e^{-\sum_{s=1}^{t-1} \ell_s^j}} e^{-\ell_t^k}\right) \\
&= \sum_{t=1}^{T} -\ln\left(\frac{\sum_{k=1}^{K} e^{-\sum_{s=1}^{t} \ell_s^k}}{\sum_{j=1}^{K} e^{-\sum_{s=1}^{t-1} \ell_s^j}}\right) \\
&= -\ln\left(\sum_{k=1}^{K} e^{-\sum_{t=1}^{T} \ell_t^k}\right) + \ln K.
\end{aligned}$$

Bounding the sum from below by a max results in

$$\leq \min_k \sum_{t=1}^{T} \ell_t^k + \ln K \tag{1}$$

The Dot Loss Game

Dot-loss game

Protocol

- For $t = 1, 2, \ldots$
  - Learner chooses a distribution $\boldsymbol{w}_t \in \triangle_K$ on K "experts".
  - Adversary reveals loss vector $\boldsymbol{\ell}_t \in [0,1]^K$.
  - Learner's loss is the **dot loss** $\boldsymbol{w}_t^\intercal \boldsymbol{\ell}_t = \sum_{k=1}^K w_t^k \ell_t^k$

Many names:

- Decision Theoretic Online Learning
- Prediction with Expert Advice
- The Hedge setting
- The Experts setting

Dot-loss objective

Definition (Regret)

Regret after T rounds:

$$R_T \;=\; \sum_{t=1}^{T} \boldsymbol{w}_t^{\intercal} \boldsymbol{\ell}_t - \min_k \sum_{t=1}^{T} \ell_t^k$$

Goal: design an algorithm for Learner that guarantees low regret.

Mix loss vs Dot-loss (Jensen)

By Jensen's Inequality for the convex function $x \mapsto -\ln(x)$

$$\underbrace{-\ln\left(\sum_{k=1}^{K} w_t^k e^{-\ell_t^k}\right)}_{\text{mix loss}} \leq \underbrace{\sum_{k=1}^{K} w_t^k \ell_t^k}_{\text{dot loss}} \tag{2}$$

So the dot loss game is harsher for the Learner ...
... but maybe we can find a converse inequality (with small overhead)

Mix loss vs Dot loss (Hoeffding)

Lemma (Hoeffding)

Fix zero-mean r.v. $X \in [a, b]$, and let $\eta \in \mathbb{R}$. Then

$$\mathbb{E}[e^{\eta X}] \;\leq\; e^{\eta^2 (b-a)^2/8}$$

(Note: Lemma is main ingredient of but not equal to Hoeffding's Bound)

Mix loss vs Dot loss (Hoeffding)

Lemma (Hoeffding)

Fix zero-mean r.v. $X \in [a, b]$, and let $\eta \in \mathbb{R}$. Then

$$\mathbb{E}[e^{\eta X}] \;\leq\; e^{\eta^2 (b-a)^2/8}$$

(Note: Lemma is main ingredient of but not equal to Hoeffding's Bound)
Application: Fix $\boldsymbol{w}_t \in \triangle_K$ and $\boldsymbol{\ell}_t \in [0,1]^K$. Define r.v. X to take value $\boldsymbol{w}_t^\intercal \boldsymbol{\ell}_t - \ell_t^k$ with probability w_t^k for all $k = 1, \ldots, K$. Then X has mean zero, and takes values in an interval of length 1. So

$$\sum_k w_t^k e^{\eta(\boldsymbol{w}_t^\intercal \boldsymbol{\ell}_t - \ell_t^k)} \;\leq\; e^{\eta^2/8}$$

and hence we obtain a converse to (2):

$$\underbrace{\boldsymbol{w}_t^\intercal \boldsymbol{\ell}_t}_{\text{dot loss}} \;\leq\; \underbrace{-\frac{1}{\eta} \ln \left(\sum_k w_t^k e^{-\eta \ell_t^k} \right)}_{\eta\text{-scaled mix loss}} + \frac{\eta}{8}$$

Hedge algorithm

Idea: re-use AA for mix loss, now with *learning rate* $\eta > 0$.

Definition (Hedge Algorithm)

The *Hedge algorithm* with *learning rate* η plays weights in round t:

$$w_t^k = \frac{e^{-\eta \sum_{s=1}^{t-1} \ell_s^k}}{\sum_{j=1}^K e^{-\eta \sum_{s=1}^{t-1} \ell_s^j}}. \tag{Hedge}$$

or, equivalently, $w_1^k = \frac{1}{K}$ and

$$w_{t+1}^k = \frac{w_t^k e^{-\eta \ell_t^k}}{\sum_{j=1}^K w_t^j e^{-\eta \ell_t^j}} \tag{Hedge, incremental}$$

Hedge analysis

Lemma

The regret of Hedge is bounded by

$$R_T \ \le \ \frac{\ln K}{\eta} + T\frac{\eta}{8}$$

Proof.

Applying Hoeffding's Lemma to the loss of each round gives

$$\sum_{t=1}^T \boldsymbol{w}_t^\intercal \boldsymbol{\ell}_t \ \le \ \sum_{t=1}^T \Bigg(\underbrace{\frac{-1}{\eta} \ln \left(\sum_{k=1}^K w_t^k e^{-\eta \ell_t^k} \right)}_{\text{mix loss}} + \underbrace{\ \eta/8\ }_{\text{overhead}} \Bigg)$$

The mix loss telescopes, and is bounded by (1) by

$$\sum_{t=1}^T \frac{-1}{\eta} \ln \left(\sum_{k=1}^K w_t^k e^{-\eta \ell_t^k} \right) \ \le \ \min_k \sum_{t=1}^T \ell_t^k + \frac{\ln K}{\eta}. \qquad (3)$$

Hedge tuning

Theorem

The Hedge regret bound is minimised at $\eta = \sqrt{\frac{8 \ln K}{T}}$ *where it states*

$$R_T \;\le\; \sqrt{T/2 \ln K}.$$

Proof.

Pick η to cancel the derivative. □

Note: tuning requires knowledge of the time horizon T. This can be solved by the "Doubling Trick". You will see it in the exercises.

Regret lower bound for the Dot loss game

Is the Hedge algorithm actually good?

Theorem

The minimax regret for the dot loss game is $\Omega\left(\sqrt{T \ln K}\right)$.

Regret lower bound for the Dot loss game

Is the Hedge algorithm actually good?

Theorem

The minimax regret for the dot loss game is $\Omega\left(\sqrt{T \ln K}\right)$.

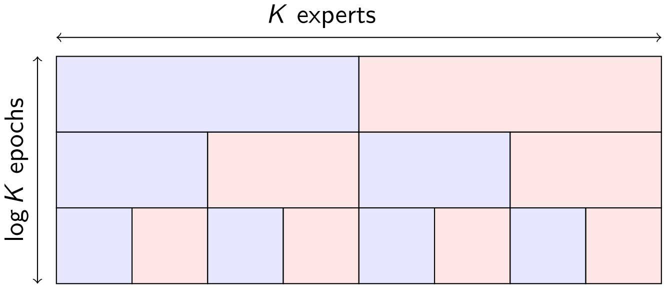
Proof. (Bonus Material).

We will see in the homework that there is an adversary for the 2-expert game with lower bound $c\sqrt{T}$. Here we boost it to K experts. The construction works by splitting the horizon T into $T/\log K$ epochs. Within each epoch, we will cluster the experts into 2 groups, and apply the 2-expert adversary to each group. This inflicts regret $c\sqrt{T/\log K}$ w.r.t. each expert in the winning group. With K experts, we can split them $\log K$ many times completely independently (see the figure below). The overall regret w.r.t. the expert that is in the wining group in every epoch is

$$R_T \;\geq\; \log(K) c \sqrt{T/\log K} \;=\; c\sqrt{T \log K}$$

□

Regret lower bound for the Dot loss game



Conclusion

Two simple settings.

- Adversary controls data
- Efficient learning algorithms
- With performance guarantees
- Matching lower bounds