2006-Cesa-Bianchi

Prediction, Learning, and Games

- **Date**: 2006
- **Link:** Textbook
- **Authors:**
  - Cesa-Bianchi, Nicolo
  - Lugosi, Gabor
- **Cites:**
- **Cited by:**
- **Keywords:** #exponential-weighted-forecaster #online-gradient-descent #online-learning
- **Collections:**
- **Status:** #in-progress

---

1 - Introduction

1.1. Prediction

- *Prediction* - guessing the short-term evolution of certain phenomena.

- **Objective** - Predict the next element of an unknown sequence given some knowledge about the past elements and possibly other available information.
  - Given the sequence of observations $y_1,\ y_2,\ ...$ for the $t = 1,\ ,2, ..., t-1$ time steps, the predictor must guess the next value, y_t.

- Entities involved in the forecasting task:
  - Elements forming the sequence
  - Criterion used to measure the quality of a forecast.
  - Protocol for how the predictor receives feedback about the sequence.
  - Any addition side information provided to the predictor.

- Classical statistics differs vastly from prediction as defined in this book:

  - *Classical statistical theory* - assumes the elements in the sequence are drawn from a stationary stochastic process that can be modeled by a statistical distribution.
    - Risk is the expected value of a loss function when measuring the discrepancy between predicted and observed values.

  - *Prediction of individual sequences* - assumes the elements are generated by an unknown mechanism (could be stochastic and stationary, or not).

    - Risk formulation is not clear under this setup, with many possibilities:

      - Basic model - loss accumulated during many rounds of prediction.

      - Baseline - use *reference forecasters* (aka *experts*) as a baseline to compare against the predictor.
        - Difference between predictor and expert loss is the *regret*.

- What is an expert?

  - Black box interpretation - an expert is a black box with unknown power and information.

  - Statistical model interpretation - each expert in a class is an optimal forecaster for some given 'state of nature.'

- Pioneers: Blackwell, Hannan, Robbins coined the *sequential compound decision problem* in the 1950s.

- *Online* - class of algorithms which receive inputs sequentially.

1.2. Learning

- *Online learning* - machine learning applied to the prediction of individual sequences.

- Pioneers:
  - De Santis, Markowski, and Wegman.
  - Littlestone and Warmuth.
  - Vovk

- *Side information* - additional information beyond the past elements of the sequence.

- Classical models of online pattern recognition take the side information as vectors and experts as linear functions of the side information.
  - Examples: Rosenblatt's Perceptron, Widrow-Hoff rule, ridge regression.

- Computational complexity of online learning algorithms are studied because of their ability to handle large, real-world datasets.

1.3. Games

- Online learning can be modeled as a sequential game between the predictor (agent) and the mechanism (environment) that generates the sequence.
- This formulation yields some interesting results:
    - Derivation of the minmax theorem by applying performance bounds to a sequential predictor.
    - Blackwell's approachability theorem can be used to define strong performing forecasters.
- Biggest result - set of regret minimizing player's in a normal form game reach an equilibrium.

1.4. A Gentle Start

- Simple example.
- Problem setup:
    - Forecaster must predict the next element $\hat{p}_t \in \{0, 1\}$, given a sequence of past bits: $y_1, , y_2, \ldots$ where $y_t \in \{0, 1\}$.
    - Using N experts, whose guesses for the next bit form the binary vector, $(f_{1,t}, \ldots, f_{N,t})$.
    - **Goal:** bound the number of mistakes made by the forecaster.
- Problem simplification:
    - Assume one expert, i, is never wrong.
    - Solution:
        - Start by assigning a weight of $w_j = 1$ for each expert, $j = 1, \ldots, N$.
        - For each time step t:
            - Predict $\hat{p}_t$ by taking a poll of the recommenders with $w_j = 1$.
            - After y_t is revealed, set $w_j = 0$ for any recommenders who guessed wrong.
            - Repeat until we have found the perfect expert.
    - We expect to lose half the experts per timestep, therefore we expect to converge to a solution in $\log_2 N$ time. (See book for a more rigorous derivation).
- For the general problem:
    - Modify the update rule to shrink an expert's weight for each incorrect guess, $w_k \leftarrow \beta w_k$ where $0 < \beta < 1$.
    - Predict $\hat{p}_t$ by taking a weighted poll of the recommenders.
- See text for convergence bounds.

---

2 - Prediction with Expert Advice

- Notation
    - *Outcome space* $\mathcal{Y}$ - space from which the sequence of outcomes are drawn, $y_1, y_2, \ldots$
    - *Decision space* $\mathcal{D}$ - convex subset of a vector space from which predictions are drawn $\hat{p}_1, \hat{p}_2, \ldots$
        - *Note* - in some cases, $\mathcal{Y} = \mathcal{D}$ but not generally.
    - *Experts* - set of reference forecasters from which the forecaster's performance is measured.
        - Set of forecaster predictions for a timestep: $\{f_{E,t} : E \in \mathcal{E}\}$ where $f_{E,t} \in \mathcal{D}$ and $\mathcal{E}$ is a fixed set of indices for the experts. Let N be the number of experts.
    - *Prediction* $\hat{p}_t$ - forecaster uses the expert advice to compute a guess for the next element.
    - *True outcome* y_t - the next element of the sequence.
    - *Loss function* ℓ - nonnegative function for scoring the forecaster's performance relative to the experts, $\ell : \mathcal{D} \times \mathcal{Y} \rightarrow \mathbb{R}$
- Prediction as a repeated game:
    - Parameters: $\mathcal{D}$, $\mathcal{Y}$, ℓ, $\mathcal{E}$.
    - For each round $t = 1, 2, \ldots$

- Environment chooses y_t and the expert advice.
- Expert advice is revealed to the forecaster.
- Forecaster chooses its prediction.
- Environment reveals the outcome.
- Forecaster and experts incur a loss.

- *Cumulative regret* (a.k.a "regret") $R_{E,n}$- difference between the forecaster's total loss and that of expert E after n rounds of prediction.

  How much the forecaster wishes it had listened to the expert for every round of prediction.

$$R_{E,n} = \sum_{t=1}^{n} \big(\ell(\hat{p}_t,\, y_t) - \ell(f_{E,t},\, y_t)\big) = \hat{L}_n - L_{E,n}$$

- *Instantaneous regret* $r_{E,t}$ - regret for one round of prediction, $r_{E,t} = \ell(\hat{p}_t,\, y_t) - \ell(f_{E,t},\, y_t)$.

- **Forecaster's Goal** - minimize its cumulative regret.

2.1. Weighted Average Prediction

- *Weighted average prediction* - strategy based on computing a weighted average of the experts' predictions.

  At timestep t, the forecaster predicts:

$$\hat{p}_t = \frac{\sum_{i=1}^{N} w_{i,t-1} f_{i,t}}{\sum_{j=1}^{N} w_{j,t-1}}$$

  Note - $\hat{p}_t$ is in $\mathcal{D}$ because it is a convex combination of $f_{i,t-1}$ which we know are in $\mathcal{D}$.

- Selecting the weight values:

  - Recall the forecaster's goal is to minimize its regret.

  - It would then be logical to assign the weights as a function of the regret associated with each expert, denoted $\phi'(R_{i,t-1})$.

  - Assign higher weights to experts with higher regrets (we should've listened to them more in the past), and vice versa.
    - Higher regret for an expert means that expert has a lower cumulative loss.

- Rewriting the weighted average prediction with updated regret based weight formulation:

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

- *Lemma 2.1.* - If the loss function is convex in its first argument, then

$$\sup_{y_t \in \mathcal{Y}} \sum_{i=1}^{N} r_{i,t}\, \phi'(R_{i,t-1}) \leq 0$$

- Added notation:

  - *Instantaneous regret vector* $\mathbf{r}_t$ - vector of instantaneous regrets for each expert.
    $\mathbf{r}_t = (r_{1,t},\, \ldots,\, r_{N,t}) \in \mathbb{R}^N$

  - *Regret vector* $\mathbf{R}_n$ - vector of cumulative regrets for each expert.
    $\mathbf{R}_n = \sum_{t=1}^{n} \mathbf{r}_t$

  - *Potential function* $\Phi(\mathbf{u})$ - maps $\mathbb{R}^N$ onto $\mathbb{R}$.
    $\Phi(\mathbf{u}) = \psi\left(\sum_{i=1}^{N} \phi(u_i)\right)$
    - $\psi \,:\, \mathbb{R} \to \mathbb{R}$ is nonnegative, strictly increasing, concave and twice differentiable.

- Using the new notation to update the weighted average prediction formula:

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

  where

$$\nabla\Phi(\mathbb{R}_{t-1})_i = \frac{\partial\Phi(\mathbf{R}_{t-1})}{\partial\mathbf{R}_{i,t-1}}$$

- *Blackwell condition* - key property used in the proof of Blackwell's approachability theorem, not discussed until Chapter 7.

  Rewriting Lemma 2.1. in terms of the potential function:

$$\sup_{y_t \in \mathcal{Y}} \mathbf{r}_t \cdot \nabla\Phi(\mathbf{R}_{t-1}) \leq 0$$

- Φ plays a similar role to that of a potential function in a dynamical system.

- The regret vector always points away from the gradient of Φ which forces $\mathbf{R}_t$ to point towards the minimum of Φ.

- *Theorem 2.1* - Assume that a forecaster satisfies the Blackwell condition for a potential, $\Phi(\mathbf{u}) = \phi\left(\sum_{i=1}^{N} \phi(u_i)\right)$.

  Then, for all $n = 1,\ 2,\ \ldots,$

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

  where

$$C(\mathbf{r}_t) = \sup_{\mathbf{u}\in\mathbb{R}^N} \psi'\left(\sum_{i=1}^{N}\phi(u_i)\right)\sum_{i=1}^{N}\phi''(u_i)\, r_{i,t}^2$$

- **NOTE TO SELF** - comeback to this section, I'm a bit lost at the moment as to the utility of this.
  - As shown in the next two subsections, we can use this theorem to generate bounds for arbitrary potential functions.

Polynomially Weighted Average Forecaster

- *Polynomial weighted average forecaster* - weighted average forecaster with the potential:

$$\Phi_p(\mathbf{u}) = \left(\sum_{i=1}^{N}(u_i)_+^p\right)^{2/p} = \|\mathbf{u}_+\|_p^2$$

  where $p \geq 2$.

  Note - $\mathbf{u}_+$ denotes the positive components of $\mathbf{u}$.

- Weights:

$$w_{i,\,t-1} = \nabla\Phi(\mathbf{R}_{t-1})_i$$

$$w_{i,\,t-1} = \frac{2(R_{i,\,t-1})_+^{p-1}}{\|(\mathbf{R}_{t-1})_+\|_p^{p-2}}$$

- Prediction:

$$\hat{p}_t = \frac{\sum_{i=1}^{N}\left(\sum_{s=1}^{t-1}(\ell(\hat{p}_s,\ y_s) - \ell(f_{i,s},\ y_s))\right)_+^{p-1} f_{i_t}}{\sum_{j=1}^{N}\left(\sum_{s=1}^{t-1}(\ell(\hat{p}_s, y_s) - \ell(f_{j,s},\ y_s))\right)_+^{p-1}}$$

- *Corollary 2.1* - Assuming ℓ is convex in its first argument and takes values [0,1].

  Then, for any sequence of outcomes:

$$\hat{L}_n - \min_{i=1,\ldots,N} L_{i,n} \leq \sqrt{n(p-1)N^{2/p}}$$

  $p = \sqrt{2\ln N}$ approximately minimizes the upper bound.

Exponentially Weighted Average Forecaster

- *Exponentially weighted average forecaster* - weighted average forecaster with the potential:

$$\Phi_\eta(\mathbf{u}) = \frac{1}{\eta}\ln\left(\sum_{i=1}^{N} e^{\eta\, u_i}\right)$$

- Weights:

$$w_{i,\,t-1} = \nabla\Phi(\mathbf{R}_{t-1})_i = \frac{e^{\eta\, R_{i,t-1}}}{\sum_{j=1}^{N} e^{\eta\, R_{j,t-1}}}$$

- Prediction:

$$\hat{p}_t = \frac{\sum_{i=1}^{N} e^{-\eta L_{i,t-1}} f_{i,t}}{\sum_{j=1}^{N} e^{-\eta L_{j,t-1}}}$$

- Properties of exponentially weighted avg forecasters:

  - Weights depend only on the past performance of the experts.
    - Opposed to also relying on the predictions as well, $\hat{p}_s$, for $s < t$.

  - Weights can be incremented as follows:

$$w_{i,t} = \frac{w_{i,t-1}e^{-\eta\,\ell(f_{i,t},y_t)}}{\sum_{j=1}^{N} w_{j,t-1}e^{-\eta\,\ell(f_{j,t-1},y_t)}}$$

- *Corollary 2.2* - Assuming ℓ is convex in its first argument and takes values [0,1].

  Then, for any sequence of outcomes:

$$\hat{L}_n - \min_{i=1,\ldots,N} L_{i,n} \leq \frac{\ln N}{\eta} + \frac{n\,\eta}{2}$$

  $\eta = \sqrt{2 \ln N/n}$ optimizes this upper bound to be $\sqrt{2n \ln N}$.

  (*Note*: this is a slightly better bound than the polynomial forecaster.)

- Drawback of the exponential forecaster - the optimal value of η requires knowledge of the horizon n.
  - Next two sections offer remedies to this problem.

2.2. An Optimal Bound

- **Goal** - show that Corollary 2.2 (for the exponential forecaster) can be improved by a *constant* factor.

- *Theorem 2.2.* - Assuming ℓ is convex in its first argument and takes values [0,1].

  Then, the regret of the exponential forecaster satisfies:

$$\hat{L}_n - \min_{i=1,\ldots,N} \leq \frac{\ln N}{\eta} + \frac{n\,\eta}{8}$$

  $\eta = \sqrt{8 \ln N/n}$ optimizes the upper bound to be $\sqrt{(n/2)\ln N}$.

- Similar to Corollary 2.2, but instead of bounding the evolution of $(1/\eta)\ln(\sum_i e^{\eta R_{i,t}})$ we bound the related quantity, $(1/\eta)\ln(W_t/W_{t-1})$.
  - See the proof for a better understanding of where these values are coming from.

2.3. Bounds That Hold Uniformly over Time

- *Problem* - Corollary 2.2. only holds for sequences of a given length n.

- *Solution 1* - "Doubling Trick" - partition the time into periods of exponentially increasing lengths.

  - At the start of each interval, an optimal η is selected for that interval length.

  - At the end of each interval, the forecaster is reset.

- The time doubling trick yields a bound that is worse than Theorem 2.2. by a constant factor. But, it does suggest that a solution that varies the potential as a function of time could work.

- *Solution 2* - η_t - let η (and thereby Φ) vary as a function of time. Given we know the form of the optimal η for fixed time, a natural choice is $\eta = \sqrt{8(\ln N)/t}$.

- *Theorem 2.3.* - Assuming ℓ is convex in its first argument and takes values $[0,1]$.

  Then, the regret of the exponential forecaster with $\eta_t = \sqrt{8(\ln N)/t}$ satisfies:

$$\hat{L}_n - \min_{i=1,\ldots,N} L_{i,n} \leq 2\sqrt{\frac{n}{2}\ln N} + \sqrt{\frac{\ln N}{8}}$$

  This yields a regret bound with a main term of $2\sqrt{(n/2)\ln N}$.

  See book for proof.

2.4. An Improvement for Small Losses

- Regret bounds can be improved when it is assumed the best expert suffers a small loss.

- *Theorem 2.4* - Assuming ℓ is convex in the first argument and takes values $[0,1]$.

  For any $\eta > 0$ the forecaster satisfies:

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

  where $L_n^* = \min_{i=1,\ldots,N} L_i, n$ is the loss of the best expert.

- *Corollary 2.4* - By tuning η to be $\ln(1 + \sqrt{(2\ln N)/L_n^*})$ where $L_n^* > 0$, then using Theorem 2.4:

$$\hat{L}_n - L_n^* \leq \sqrt{2 L_n^* \ln N} + \ln N$$

2.5. Forecasters Using the Gradient of the Loss

- *Gradient-based exponentially weighted average forecaster* - exponentially weighted average forecaster where the cumulative loss, $L_{i,t-1}$ is replaced by the gradient loss summed over the time $t-1$.

Assuming the loss is differentiable, $\nabla \ell(\hat{p}, y)$:

$$\hat{p}_t = \frac{\sum_{i=1}^{N} \exp\left(-\eta \sum_{s=1}^{t-1} \nabla \ell(\hat{p}_s, y_s) \cdot f_{i,s}\right) f_{i,t}}{\sum_{j=1}^{N} \exp\left(-\eta \sum_{s=1}^{t-1} \nabla \ell(\hat{p}_s, y_s) \cdot f_{j,s}\right)}$$

- Intuition:
  - The weight assigned to an expert is inversely proportional to $\nabla \ell(\hat{p}_s, y_s) \cdot f_{i,s}$
  - $\nabla \ell(\hat{p}_s, y_s) \cdot f_{i,s}$ is large if the expert's guess $f_{i,s}$ points in the direction of a large increasing loss.
  - This intuitively makes sense, we want to assign higher weights to experts whose predictions lead in the direction of decreasing loss.
- Comparing gradient-based forecaster to the standard forecaster:
  - Generally, the two forecasters yield different predictions.
  - The two forecasters converge in the special case of binary prediction with absolute loss.
- Regret bound:
  - *Corollary 2.5* -
    - IF:
      - $\mathcal{D}$ is a convex subset of the unit ball $\{q \in \mathbb{R}^d : \|q\| \leq 1\}$.
      - ℓ is convex in its first argument.
      - $\nabla \ell$ exists and ≤ 1.
    - THEN, for any n and $\eta > 0$, regret satisfies:

$$\hat{L}_n - \min_{i=1,\dots,N} L_{i,n} \leq \frac{\ln N}{\eta} + \frac{n\,\eta}{2}$$

2.6. Scaled Losses and Signed Games

- We have assumed the loss function, ℓ , is bound by the interval $[0,1]$.
- *Goal* - In this section we consider scaling and translating the loss range and interpret the new regret bounds.
- *Case 1* - **Scaled loss** - $\ell \in [0, M]$.
  - Use the scaled loss ℓ / M without any modification to the weighted forecaster.
  - For example, applying the scaled loss to the regret bound for the exponentially weighted average forecaster yields the regret bound:

$$\hat{L}_n - L_n^* \leq \sqrt{2L_n^* M \ln N} + M \ln N$$

- *Case 2* - **Translated loss** - $\ell \in [-M, 0]$.
  - Let $\hat{G}_n = -\hat{L}_n$ be the *gain* and $G_n^* = -L_n^*$ be the gain of the best expert.
  - Use the scaled gain $(-\ell)/M$ without modification to the weighted forecaster.
  - For example, applying the scaled gain to the regret bound for the exponentially weighted average forecaster yields the regret bound:

$$G_n^* - \hat{G}_n \leq \sqrt{2G_n^* M \ln N} + M \ln N$$

- *Case 3* - **General loss** - *payoff function* $h : \mathcal{D} \times \mathcal{Y} \to \mathbb{R}$
  - Goal of the forecaster is to maximize its payoff, the forecaster's regret is then:

$$\max_{i=1,\dots,N} \sum_{t=1}^{n} h(f_{i,t}, y_t) - \sum_{t=1}^{n} h(\hat{p}_t, y_t) = H_n^* - \hat{H}_n$$

  - Game categorization based on payoff function:
    - *Gain game* - $h \in [0, M]$
      - Regret bound on the order of $\sqrt{|H_n^*| M \ln N}$ (as seen in the above cases)
    - *Loss game* - $h \in [-M, 0]$
      - Regret bound on the order of $\sqrt{|H_n^*| M \ln N}$ (as seen in the above cases)
    - *Signed game* - $h \in [-M, M]$
      - Must use a linear transform to map h range onto $[0, 1]$:

- $h \mapsto (h+M)/(2M)$

- Applying to exponentially weighted average forecaster regret bound:

$$H_n^* - \hat{H}_n \leq \sqrt{4(H_n^* + Mn)(M \ln N)} + 2M \ln N$$

- Regret bound on the order of $M\sqrt{n \ln N}$.

- Shows converting a signed game to $[0, 1]$ might not be the best for regret.

2.7. The Multilinear Forecaster

- *Goal* - derive a good forecaster for signed games without using potential functions.

- *Multilinear forecaster* - weighted average forecaster where the weights are defined recursively:

$$w_{i,0} = 1$$
$$w_{i,t} = w_{i,t-1}\big(1 + \eta\, h(f_{i,t}, y_t)\big)$$

- Regret bound discussion:

  - The weights can not be expressed as functions of regret, $H_{i,n} - \hat{H}_n$, like we did with the potential function derivations.

  - However approximations can be made by observing $(1 + \eta h) \approx e^{\eta h}$

- *Theorem 2.5* - Assuming h is concave in its first argument and $h \in [-M, \infty]$.

  For any n and $0 < \eta < 1/(2M)$, the regret of the multilinear forecaster satisfies:

$$H_{i,n} - \hat{H}_n \leq \frac{\ln N}{\eta} + \eta \sum_{t=1}^{n} h(f_{i,t}, y_t)^2$$

  See the book for the proof

- If we know the payoff of the best performing expert then we can improve the regret bound by a constant factor compared to the exponential weighted forecaster bounds, under certain conditions.

2.8. The Exponential Forecaster for Signed Games

- Exponentially weighted average forecaster is capable of achieving small regret in signed games.

- Proof uses a modified version of Theorem 2.5's proof using the sum of quadratic terms.

  See book for full proof

2.9. Simulatable Experts

-