Tracking a small set of Experts

Yoav Freund

February 5, 2020

Based on “Tracking a Small Set of Experts by Mixing Past Posteriors” by Bousquet and Warmuth.

Vovk's meta-algorithm

- Fix an achievable pair (a, c) and set $\eta = a/c$

Vovk's meta-algorithm

- Fix an achievable pair (a, c) and set $\eta = a/c$

Vovk's meta-algorithm

- Fix an achievable pair (a, c) and set $\eta = a/c$
- 1.
$$W_i^t = \frac{1}{N}\, e^{-\eta L_i^t}$$

Vovk's meta-algorithm

- Fix an achievable pair (a, c) and set $\eta = a/c$
- 1. $$W_i^t = \frac{1}{N}\, e^{-\eta L_i^t}$$
  2. Choose γ_t so that, for all $\omega^t \in \Omega$:
  $$\lambda(\omega^t, \gamma^t) - c \ln \sum_i W_i^t \leq -c \ln \left(\sum_i W_i^t e^{-\eta \lambda(\omega^t, \gamma_i^t)} \right)$$

Vovk's meta-algorithm

- Fix an achievable pair (a, c) and set $\eta = a/c$
- 1.
$$W_i^t = \frac{1}{N}\, e^{-\eta L_i^t}$$
  2. Choose γ_t so that, for all $\omega^t \in \Omega$:
$$\lambda(\omega^t, \gamma^t) - c \ln \sum_i W_i^t \leq -c \ln \left(\sum_i W_i^t e^{-\eta\lambda(\omega^t, \gamma_i^t)} \right)$$
- If choice of γ_t always exists, then the total loss satisfies:
$$\sum_t \lambda(\omega^t, \gamma^t) \leq -c \ln \sum_i W_i^{T+1} \leq aL_{\min} + c \ln N$$

Vovk's meta-algorithm

- Fix an achievable pair (a, c) and set $\eta = a/c$
- 1.

$$W_i^t = \frac{1}{N} e^{-\eta L_i^t}$$

  2. Choose γ_t so that, for all $\omega^t \in \Omega$:

$$\lambda(\omega^t, \gamma^t) - c \ln \sum_i W_i^t \leq -c \ln \left(\sum_i W_i^t e^{-\eta \lambda(\omega^t, \gamma_i^t)} \right)$$

- If choice of γ_t always exists, then the total loss satisfies:

$$\sum_t \lambda(\omega^t, \gamma^t) \leq -c \ln \sum_i W_i^{T+1} \leq aL_{\min} + c \ln N$$

- Vovk's result: *yes!* a good choice for γ_t always exists!

The set of achievable bounds

- Fix loss function $\lambda : \Omega \times \Gamma \to [0, \infty)$

The set of achievable bounds

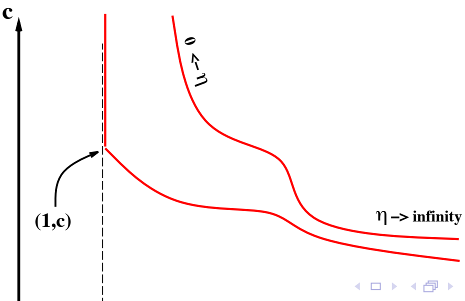
- Fix loss function $\lambda : \Omega \times \Gamma \rightarrow [0, \infty)$
- The pair (a, c) is *achievable* if there exists *some* prediction algorithm such that for *any* $N > 0$, *any* set of N prediction sequences and *any* sequence of outcomes

$$L_A \leq aL_{\min} + c \ln N$$

The set of achievable bounds

- Fix loss function $\lambda : \Omega \times \Gamma \rightarrow [0, \infty)$
- The pair (a, c) is *achievable* if there exists *some* prediction algorithm such that for *any* $N > 0$, *any* set of N prediction sequences and *any* sequence of outcomes

$$L_A \leq aL_{\min} + c \ln N$$



Definition of achievability for non-uniform prior

Definition 1 *(Haussler et al., 1998, Vovk, 1998) Let* $c, \eta > 0$. *A loss function* L *and prediction function* $\mathtt{pred}$ *are* (c, η)*-realizable if, for any weight vector* $\boldsymbol{v} \in \mathcal{P}_n$*, prediction vector* $\boldsymbol{x}$ *and outcome* y*,*

$$L(y, \mathtt{pred}(\boldsymbol{v}, \boldsymbol{x})) \leq -c \ln \sum_{i=1}^{n} v_i e^{-\eta L(y, x_i)} \quad . \tag{1}$$

- Specifies c, η instead of a, c and $\eta = a/c$

Definition of achievability for non-uniform prior

Definition 1 *(Haussler et al., 1998, Vovk, 1998) Let $c, \eta > 0$. A loss function L and prediction function* `pred` *are (c,η)-realizable if, for any weight vector $\boldsymbol{v} \in \mathcal{P}_n$, prediction vector $\boldsymbol{x}$ and outcome y,*

$$L(y, \mathbf{pred}(\boldsymbol{v}, \boldsymbol{x})) \leq -c \ln \sum_{i=1}^{n} v_i e^{-\eta L(y, x_i)} \quad . \tag{1}$$

- Specifies c, η instead of a, c and $\eta = a/c$
- Mixable: $a = 1$ or equivalently $\eta = 1/c$

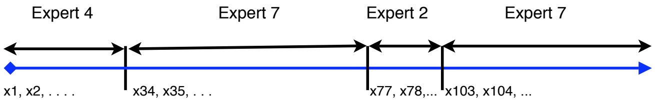
Switching in a small set

- Switching experts: n experts. Algorithm's total loss compared to total loss of best expert sequence with k switches.

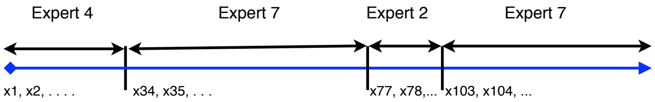
Switching in a small set

- Switching experts: n experts. Algorithm's total loss compared to total loss of best expert sequence with k switches.
- Regret bound **Regret** $\leq ck \log n + c \log \frac{T-1}{k}$

Switching in a small set

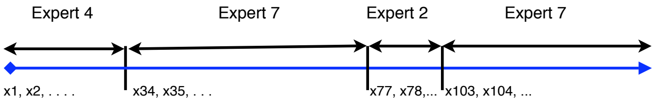
- Switching experts: n experts. Algorithm's total loss compared to total loss of best expert sequence with k switches.
- Regret bound **Regret** $\leq ck\log n + c\log\frac{T-1}{k}$
- 

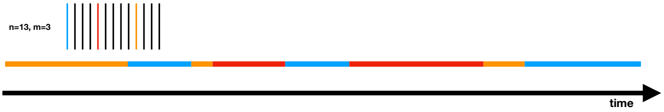
Switching in a small set

- Switching experts: n experts. Algorithm's total loss compared to total loss of best expert sequence with k switches.
- Regret bound **Regret** $\leq ck \log n + c \log \frac{T-1}{k}$
- 

- Switching in a small set: Same, but comparator restricted to a subset of $m \ll n$ experts.

Switching in a small set

- Switching experts: n experts. Algorithm's total loss compared to total loss of best expert sequence with k switches.
- Regret bound **Regret** $\leq ck \log n + c \log \frac{T-1}{k}$
- 

- Switching in a small set: Same, but comparator restricted to a subset of $m \ll n$ experts.
- 

Short term vs. long term

- In the short term - track the sequence of switching experts. Regret of $k \log n$ per switch.

Short term vs. long term

- In the short term - track the sequence of switching experts. Regret of $k \log n$ per switch.
- In the long term - Identify the set of $m \ll n$ experts and switch only among them. Regret of $c \log m$ per switch.

Short term vs. long term

- In the short term - track the sequence of switching experts. Regret of $k \log n$ per switch.
- In the long term - Identify the set of $m \ll n$ experts and switch only among them. Regret of $c \log m$ per switch.
- Practical implication: when the set of models is small, transitions are tracked more quickly.

An inefficient algorithm for switching experts

- Fix:

An inefficient algorithm for switching experts

- Fix:
  - l - sequence length

An inefficient algorithm for switching experts

- Fix:
  - l - sequence length
  - k - number of switches

An inefficient algorithm for switching experts

- Fix:
  - l - sequence length
  - k - number of switches
  - n - number of experts

An inefficient algorithm for switching experts

- Fix:
  - l - sequence length
  - k - number of switches
  - n - number of experts
- Consider one partition-expert per sequence of switching experts.

An inefficient algorithm for switching experts

- Fix:
  - l - sequence length
  - k - number of switches
  - n - number of experts
- Consider one partition-expert per sequence of switching experts.
- No. of partition-experts : $\binom{l}{k-1} n(n-1)^k = O\left(n^{k+1}\left(\frac{el}{k}\right)^k\right)$

An inefficient algorithm for switching experts

- Fix:
  - l - sequence length
  - k - number of switches
  - n - number of experts
- Consider one partition-expert per sequence of switching experts.
- No. of partition-experts : $\binom{l}{k-1} n(n-1)^k = O\left(n^{k+1}\left(\frac{el}{k}\right)^k\right)$
- The regret for a mixable loss with constant c is $c\left((k+1)\log n + k\log\frac{l}{k} + k\right)$

An inefficient algorithm for switching in a small set

- Define a meta-expert for each subset S of $\{1, \dots, n\}$ of size m.

An inefficient algorithm for switching in a small set

- Define a meta-expert for each subset S of $\{1, \ldots, n\}$ of size m.
- There are $\binom{n}{m}$ meta-experts.

An inefficient algorithm for switching in a small set

- Define a meta-expert for each subset S of $\{1, \ldots, n\}$ of size m.
- There are $\binom{n}{m}$ meta-experts.
- Each meta-expert considers switches among the experts in S

An inefficient algorithm for switching in a small set

- Define a meta-expert for each subset S of $\{1, \ldots, n\}$ of size m.
- There are $\binom{n}{m}$ meta-experts.
- Each meta-expert considers switches among the experts in S
- A standard exponential weights algorithm is used to combine all meta-experts.

Regret bound for inefficient algorithm

- For inefficient switching experts:
  $R \approx ck \log n + ck \log \frac{l}{k}$

Regret bound for inefficient algorithm

- For inefficient switching experts:
  $R \approx ck \log n + ck \log \frac{l}{k}$
- For inefficient switching in a small set:
  $R \approx ck \log m + ck \log \frac{l}{k} + cn \log \frac{n}{m}$

Regret bound for inefficient algorithm

- For inefficient switching experts:
  $R \approx ck \log n + ck \log \frac{l}{k}$
- For inefficient switching in a small set:
  $R \approx ck \log m + ck \log \frac{l}{k} + cn \log \frac{n}{m}$
- Remember two part coding for log loss: encode the model and then the data given the model. The length of the description of the model is the regret.

Efficient algorithm

Parameters: $0 < \eta, c$ and $0 \leq \alpha \leq 1$
Initialization: Initialize the weight vector to $\boldsymbol{v}_1 = \frac{1}{n}\mathbf{1}$ and denote $\boldsymbol{v}_0^m = \frac{1}{n}\mathbf{1}$
FOR $t = 1$ **TO** T **DO**

- **Prediction:** After receiving the vector of experts' predictions $\boldsymbol{x}_t$, predict with

$$\hat{y}_t = \mathtt{pred}(\boldsymbol{v}_t, \boldsymbol{x}_t) \ .$$

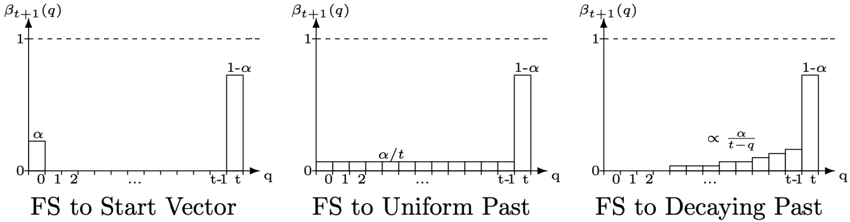
- **Loss Update:** After receiving the outcome y_t, compute for $1 \leq i \leq n$,

$$v_{t,i}^m = \frac{v_{t,i} e^{-\eta L_{t,i}}}{\sum_{j=1}^n v_{t,j} e^{-\eta L_{t,j}}}, \quad \text{where } L_{t,i} = L(y_t, x_{t,i}) \ .$$

- **Mixing Update:** Choose non-negative mixture coefficients $\beta_{t+1}(q)$ $(q = 0, \ldots, t)$ such that $\sum_{q=0}^t \beta_{t+1}(q) = 1$ and compute

$$\boldsymbol{v}_{t+1} = \sum_{q=0}^{t} \beta_{t+1}(q) \boldsymbol{v}_q^m \ .$$

Mixing in past weights



Bound For Uniform Past

Bound for the Fixed Share to Uniform Past Mixing Scheme. Consider a mixing scheme that equally penalizes all vectors in the past: $\beta_{t+1}(q) = \alpha \, \frac{1}{t} \; (q = 0..t-1)$.

Corollary 8 *For the Mixing Algorithm A with the Fixed Share to Uniform Past mixing scheme and for any sequence of T comparison vectors $\boldsymbol{u}_t$ with k shifts from a pool of m convex combinations, we have*

$$L_{1..T,A} \leq \sum_{t=1}^{T} \boldsymbol{L}_t \cdot \boldsymbol{u}_t + cm \ln n + ck \ln \frac{1}{\alpha} + c(T-k-1) \ln \frac{1}{1-\alpha} + ck \ln(T-1) \quad .$$

Bound For Decaying Past

Bound for the Fixed Share to Decaying Past Mixing Scheme. We now show that an improvement of the above corollary is possible by choosing $\beta_{t+1}(q) = \alpha \frac{1}{(t-q)^\gamma Z_t}$ for $0 \leq q \leq t-1$, with $Z_t = \sum_{q=0}^{t-1} \frac{1}{(t-q)^\gamma}$.

Corollary 9 *For the Mixing Algorithm A with the Fixed Share to Decaying Past mixing scheme with $\gamma = 1$ and for any sequence of T comparison vectors $\boldsymbol{u}_t$ with k shifts from a pool of m convex combinations, we have*

$$\begin{aligned} L_{1..T,A} \quad \leq \quad & \sum_{t=1}^{T} \boldsymbol{L}_t \cdot \boldsymbol{u}_t + cm \ln n + ck \ln \frac{1}{\alpha} + c(T-k-1) \ln \frac{1}{1-\alpha} \\ & + ck \ln \frac{(T-1)(m-1)}{k} + ck \ln \ln(eT) \ . \end{aligned}$$

Proof The proof follows the proof of Corollary 8 and is given in Appendix B. ∎

Experiments

Switch to paper (BousquetW02.pdf)