On-Line Learning of Non-Stationary Data

Manfred K. Warmuth
UC, Santa Cruz

Joint work with:
Olivier Bousquet
Mark Herbster
Outline

- Motivate on-line learning
- Motivate relative loss bounds
- Halving Algorithm as example
- Loss Update
- Flavor of proof techniques
- Comparator on-line as well
- How to adapt the algs
- Future work
On-Line Learning

Loop
  Get next instance
  Predict
  Get label
  Incur loss

• Choose comparison class of predictors
e.g. linear

• Goal:
  Do well compared to best off-line comparator

• No statistical assumptions on the data
### Experts as Comparators

<table>
<thead>
<tr>
<th>experts</th>
<th>$E_1$</th>
<th>$E_2$</th>
<th>$E_3$</th>
<th>$E_n$</th>
<th>prediction</th>
<th>true label</th>
<th>loss</th>
</tr>
</thead>
<tbody>
<tr>
<td>day 1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>day 2</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>day 3</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>day $t$</td>
<td>$x_{t,1}$</td>
<td>$x_{t,2}$</td>
<td>$x_{t,3}$</td>
<td>$x_{t,n}$</td>
<td>$\hat{y}_t$</td>
<td>$y_t$</td>
<td>$</td>
</tr>
</tbody>
</table>

### Protocol of the Master Algorithm

For $t = 1$ To $T$ Do

- Get instance    $\boldsymbol{x}_t \in \{0, 1\}^n$
- Predict        $\hat{y}_t \in \{0, 1\}$
- Get label      $y_t \in \{0, 1\}$
- Incur loss     $|y_t - \hat{y}_t|$
Minimax Algorithm for $T$ Trials

- **Learner** against adversary

\[
\begin{align*}
\sup_{x_1} \inf_{\hat{y}_1} \sup_{y_1} \sup_{x_2} \inf_{\hat{y}_2} \sup_{y_2} \ldots \sup_{x_T} \inf_{\hat{y}_T} \sup_{y_T} \\
\sum_{t=1}^{T} L(y_t, \hat{y}_t) - \inf_{f \in C} \left( \sum_{t=1}^{T} L(f(x_t), y_t) \right)
\end{align*}
\]

- **$C$** is comparison class
- Minimax algorithm usually intractable
What kind of performance can we expect?

- $L_{1..T,A}$ be the total loss of algorithm $A$
- $L_{1..T,i}$ be the total loss of $i$-th expert $E_i$

• Form of bounds

$$\forall S : \quad L_{1..T,A} \leq \min_i (L_{1..T,i} + c \log n)$$

where $c$ is constant

• Bounds the loss of the algorithm relative to the loss of best expert
General Expert Algorithm

• Master algorithm predicts with weighted average

\[ \hat{y}_t = v_t \cdot x_t \]

• The weights are updated according to the Loss Update

\[ v_{t+1,i} := v_{t,i} e^{-\eta L_{t,i}} \text{ normaliz.} \]

where \( L_{t,i} \) is loss of expert \( i \) in trial \( t \)

→ Weighted Majority Algorithm \[ [LW89] \]

→ Generalized by Vovk \[ [Vovk90] \]
Comparator Changes with Time

- Off-line alg. partitions sequence into sections and chooses best expert in each section
- Goal:
  Do well compared to best off-line partition
- Problem:
  Loss Update learns too well and does not recover fast enough
Modifications to the Expert Alg. [HW98]

- Predict $\hat{y}_t = \mathbf{v}_t \cdot \mathbf{x}_t$

- Loss Update

\[ v_{t,i}^m := \frac{v_{t,i}e^{-\eta L_{t,i}}}{\text{normaliz.}} \]

- Share Update
  - Static Expert

\[ \mathbf{v}_{t+1} = \mathbf{v}_t^m \]

  - Fixed Share to Start Vector ($\alpha \in [0, 1]$)

\[ \mathbf{v}_{t+1} = (1 - \alpha)\mathbf{v}_t^m + \alpha \mathbf{v}_0 \]

where $\mathbf{v}_0 = \left( \frac{1}{n}, \frac{1}{n}, \ldots, \frac{1}{n} \right)$
• Square loss, target outcome always 0, experts have predictions between 0 and 1/2 uniform for typical experts and restricted to [0, 0.12] for current best expert

• $T = 1400$ trials, $n = 20000$ experts, $k = 6$ shifts
Weights of Fixed Share Alg.

- Tracks the best expert
Weights of Fixed Share Alg.
Weights of Static Expert Alg.

M. Warmuth: On-line Learning of Non-Stationary Data

AT&T, May 01
• Variable Share to Start Vector

- Replace

\[ \mathbf{v}_{t+1,i} = (1 - \alpha) \mathbf{v}_{t,i}^m + \alpha \frac{1}{n} \]

- by

\[ \mathbf{v}_{t+1,i} = (1 - \alpha)^{L_{t,i}} \mathbf{v}_{t,i}^m + \left( 1 - \sum_{i=1}^{n} (1 - \alpha)^{L_{t,i}} \mathbf{v}_{t,i}^m \right) \frac{1}{n} \text{ where } L_{t,i} \in [0, 1] \]
Fixed Share vs. Variable Share
Weights of Variable Share Alg.

M. Warmuth: On-line Learning of Non-Stationary Data

AT&T, May 01
Shifting Bounds

• Recall Static Expert bound

\[ L_{\text{Alg}}(S') \leq \min_i (L_i(S') + O(\log n)) \]

  – Comparison class: set of experts

• Bounds for Share Algs.

\[ L_{\text{Alg}}(S) \leq \min_P (L_P(S) + O(\text{# of bits for } P)) \]

  – Comparison class: set of partitions

  – # of bits for partitions with \( k \) shifts:

\[ k \log n + \log \binom{T}{k} \]
Yoav’s Open Problem

- Number of possible experts $n$ is large
  \[ n \approx 10^6 \]
- Experts in partition from small subset of size $m$
  \[ m \approx 10 \]
- # of bits for partitions with $k$ shifts:
  \[
  \log \binom{n}{m} + k \log m + \log \binom{T}{k}
  \]
Mixing Update

- Predict $\hat{y}_t = v_t \cdot x_t$

- Loss Update $v_{t,i}^m := \frac{v_{t,i}e^{-\eta L_{t,i}}}{\text{normaliz.}}$

- Mixing Update: $v_{t+1} = \sum_{q=0}^{t} \beta_{t+1,q} v_q^m$

- Mixing scheme

FS to Start Vector

FS to Uniform Past

FS to Decaying Past
• Square loss, target outcome always 0, experts have predictions between 0 and 1/2 uniform for typical experts and restricted to [0, 0.12] for current best expert

• $T = 1400$ trials, $n = 20000$ experts, $k = 6$ shifts, $m = 3$ experts in the small subset
Static Experts

M. Warmuth: On-line Learning of Non-Stationary Data

AT&T, May 01
Fixed Share to Start Vector

\[ \log(\alpha/n) \]

M. Warmuth: On-line Learning of Non-Stationary Data
Fixed Share to Decaying Past - Log Weights
More Experts Remembered

![Graph showing Total Loss vs Best Expert]

Key:
- Red: Typical Expert
- Green: Static
- Blue: FS Start
- Pink: FS Decaying Past
- Cyan: Best Partition

M. Warmuth: On-line Learning of Non-Stationary Data  
AT&T, May 01
Fixed Share to Decaying Past - Log Weights

- Weights past good expert remain at higher level
• No memory
Long-term Versus Short-term Memory

![Graph showing Total Loss over Best Expert for different strategies: Typical Expert, Static, FS Start, FS Decaying Past, Best Partition.]

M. Warmuth: On-line Learning of Non-Stationary Data  AT&T, May 01
Fixed Share to Decaying Past

M. Warmuth: On-line Learning of Non-Stationary Data

AT&T, May 01
Variable Share Beats Fixed Share

![Graph showing Total Loss across Best Experts for different Expert Start and Decaying conditions.](image)

- **Typical Expert**
- **FS Start**
- **VS Start**
- **FS Decaying Past**
- **VS Decaying**
- **Best Partition**

M. Warmuth: On-line Learning of Non-Stationary Data
Bounds Again

- Bounds still have the form

\[ L_{1..T,A} \leq \min_P (L_{1..T,P} + O(\# \text{ of bits for } P)) \]

- Excess loss for naive alg.

\[ O(\log \binom{n}{m} + k \log m + \log \binom{T}{k}) \]

- Excess loss for Fixed Share to Decaying Past

\[ O \left( m \log n + k \log m + 2 \log \binom{T}{k} \right) \]

→ Boundaries are encoded twice

→ Off-line problem NP-complete
Loss Bounds Versus Storage Complexity

- Naive alg. has optimal bound - exponential storage

- Fixed Share to Uniform Past - $O(n)$ weights

- Fixed Share to Decaying Past - $O(nT)$ weights and better bound

$\rightarrow$ With tricks $O(n \ln T)$ weights and essentially same bound
Alternates to Mixing

- What we need for bounds

\[ \mathbf{v}_{t+1} = \beta_{t+1,q} \mathbf{v}_q^m, \text{ for } 0 \leq q \leq t \quad (\ast) \]

<table>
<thead>
<tr>
<th>Update Type</th>
<th>Update Formula</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mixing Update</td>
<td>[ \mathbf{v}<em>{t+1} = \sum</em>{q=0}^{t} \beta_{t+1,q} \mathbf{v}_q^m ]</td>
</tr>
<tr>
<td>Max Update</td>
<td>[ \mathbf{v}<em>{t+1} = \frac{1}{\text{normaliz.}} \max</em>{q=0,\ldots,t} \beta_{t+1,q} \mathbf{v}_q^m ]</td>
</tr>
<tr>
<td>Projection Update</td>
<td>[ \mathbf{v}<em>{t+1} = \arg \min</em>{\mathbf{v} \in (\ast)} \Delta(\mathbf{v}, \mathbf{v}_t^m) ]</td>
</tr>
</tbody>
</table>
• $T = 1400$ trials, $n = 20000$ experts

• $k = 1$ shift (at trial 400), $m = 2$ experts in the small subset
Larger alpha gives better long-term memory
Fixed Share to Start Vector - Log Weights
Many Short Sections of the Same Experts

- $T = 3200$ trials, $n = 20000$ experts
- $k = 30$ shifts (every 200 and 50 trials), $m = 2$ experts in the small subset
• The memory from many short sections accumulates
Future

- Bayesian interpretation
- Variable share
- Lower bounds
- Automatic tuning
- Mixing Update works for EG family
- Connections to Universal Coding

Applications
  - Load balancing
  - Switching between a few users
  - Segmentation