Online Learning: Expert Setting
CS6780 – Advanced Machine Learning
Spring 2019
Thorsten Joachims
Cornell University
Reading: Shalev-Shwartz/Ben-David, 287-297

Online Classification Model

- Setting
  - Classification
  - Hypothesis space $H$ with $h: X \rightarrow Y$
  - Measure misclassifications (i.e. zero/one loss)

- Interaction Model
  - Initialize hypothesis $h \in H$
  - FOR $t$ from 1 to $T$
    - Receive $x_t$
    - Make prediction $\hat{y}_t = h(x_t)$
    - Receive true label $y_t$
    - Record if prediction was correct (e.g., $\hat{y}_t = y_t$)
    - Update $h$

(Online) Perceptron Algorithm

- Input: $S = ((x_1, y_1), \ldots, (x_n, y_n))$, $x_i \in \mathbb{R}^N$, $y_i \in \{-1, 1\}$
- Algorithm:
  - $w_0 = 0$, $k = 0$
  - FOR $i = 1$ TO $n$
    - IF $y_i (w^T_k \cdot x_i) < 0$  #### makes mistake
      - $w_{k+1} = w_k + y_i x_i$
      - $k = k + 1$
    - ENDIF
  - ENDFOR
- Output: $w_k$

Perceptron Mistake Bound

Theorem: For any sequence of training examples $S = ((x_1, y_1), \ldots, (x_n, y_n))$ with $R = \max \|x_i\|$, if there exists a weight vector $w_{opt}$ with $\|w_{opt}\| = 1$ and $y_i (w_{opt} \cdot x_i) \geq \delta$ for all $1 \leq i \leq n$, then the Perceptron makes at most $\frac{R^2}{\delta^2}$ errors.

Expert Learning Model

- Setting
  - $N$ experts named $H = \{h_1, \ldots, h_N\}$
  - Each expert $h_i$ takes an action $y = h_i(x_t)$ in each round $t$ and incurs loss $\Delta_{i,t}$
  - Algorithm can select which expert’s action to follow in each round

- Interaction Model
  - FOR $t$ from 1 to $T$
    - Algorithm selects expert $h_{opt}$ according to strategy $A_{w_k}$ and follows its action $y$
    - Algorithm incurs losses $\Delta_{opt} - \Delta_{i,t}$
    - Algorithm incurs loss $\Delta_{i,t}$
    - Algorithm updates $w_k$ to $w_{k+1}$ based on $\Delta_{i,t} - \Delta_{opt}$

Halving Algorithm

- Setting
  - $N$ experts named $H = \{h_1, \ldots, h_N\}$
  - Binary actions $y = \{+1, -1\}$ given input $x$, zero/one loss
  - Perfect expert exists in $H$

- Algorithm
  - $V_{S_1} = H$
  - FOR $t$ from 1 to $T$
    - Algorithm selects the same $y$ as majority of $h_i \in V_{S_t}$
    - $V_{S_{t+1}} = V_{S_t}$ minus those $h_i \in V_{S_t}$ that were wrong

- Mistake Bound
  - How many mistakes can the Halving algorithm make before predicting perfectly?
### Idea
- $N$ experts named $H = \{h_1, ..., h_N\}$
- Compare performance of $A$ to best expert $i^*$ in hindsight.

### Regret
- Overall loss of best expert $i^*$ in hindsight is
  $$\Delta^* = \min_{i^*} \frac{1}{T} \sum_{t=1}^{T} \Delta_t^{i^*}$$
- Loss of algorithm $A$ at time $t$ is
  $$\Delta_t = \sum_{i \neq i^*} w_i \Delta_t$$

### Exponentiated Gradient Algorithm for Expert Setting (EG)
**Setting**
- $N$ experts named $H = \{h_1, ..., h_N\}$
- Any actions, any positive and bounded loss
- There may be no expert in $H$ that acts perfectly

**Algorithm**
- Initialize $\tilde{w}_t = (1, ..., 1)$
- FOR $t$ from 1 to $T$
  - Algorithm randomly picks $i_t$ from $P(i_t = i) = w_i$
  - Experts incur losses $\Delta_t = \sum_i \tilde{w}_t \Delta_t$
  - Algorithm updates $w$ for all experts $i$ as
    $$\tilde{w}_t \Delta_t = \tilde{w}_t \exp(-\Delta_t)$$

### Weighted Majority Algorithm (Deterministic)
**Setting**
- $N$ experts named $H = \{h_1, ..., h_N\}$
- Binary actions $y = \{+1, -1\}$ given input $x$, zero/one loss
- There may be no expert in $H$ that acts perfectly

**Algorithm**
- Initialize $w_1 = (1, 1, ..., 1)$
- FOR $t = 1$ TO $T$
  - Predict the same $y$ as majority of $h_t \in H$, each weighted by $w_t$
  - FOR EACH $h_t \in H$
    - IF $h_t$ incorrect THEN $w_{t+1} = w_t + \beta$
    - ELSE $w_{t+1} = w_t$

**Regret Bound**
- How close is the number of mistakes the Weighted Majority Algorithm makes to the number of mistakes of the best expert in hindsight?

### Regret Bound for Exponentiated Gradient Algorithm
**Theorem**
The expected regret of the exponentiated gradient algorithm in the expert setting is bounded by

$$\text{ExpectedRegret}(T) \leq \sqrt{2T \log(|H|)}$$

where $\Delta \in [0, 1]$ and $\eta = \sqrt{2 \log(|H|)/T}$ and $T > 2 \log(|H|)$. 

### Expected Regret
- Overall loss of best expert $i^*$ in hindsight is
  $$\Delta^* = \min_{i^*} \frac{1}{T} \sum_{t=1}^{T} \Delta_t^{i^*}$$
- Expected loss of algorithm $A(w_t)$ at time $t$ is
  $$E_{A(w_t)}[\Delta_t] = w_t \Delta_t$$

for randomized algorithm that picks recommendation of expert $i$ at time $t$ with probability $w_t$.

- Regret is difference between expected loss of algorithm and best fixed expert in hindsight
  $$\text{ExpectedRegret}(T) = \sum_{t=1}^{T} w_t \Delta_t - \min_{i \neq i^*} \frac{1}{T} \sum_{t=1}^{T} \Delta_t$$

### LaTeX Equations
- $\Delta^* = \min_{i^*} \frac{1}{T} \sum_{t=1}^{T} \Delta_t^{i^*}$
- $\Delta_t = \sum_{i \neq i^*} w_i \Delta_t$
- $\tilde{w}_t \Delta_t = \tilde{w}_t \exp(-\Delta_t)$
- $E_{A(w_t)}[\Delta_t] = w_t \Delta_t$
- $\text{ExpectedRegret}(T) \leq \sqrt{2T \log(|H|)}$
- $\Delta \in [0, 1]$ and $\eta = \sqrt{2 \log(|H|)/T}$ and $T > 2 \log(|H|)$. 

### Diagrams
- Diagrams illustrating the Exponentiated Gradient Algorithm and Weighted Majority Algorithm with detailed steps and equations.