Prediction, Learning and Games - Chapter 5

Jérôme Thai
UC Berkeley

October 21, 2013

EXPERTS FRAMEWORK

Recall

- On iteration t: experts reveal their advice "i"

EXPERTS FRAMEWORK

Recall

- On iteration t: experts reveal their advice " i "
- forecaster makes a decision $\hat{p}_t = \sum_{i=1}^{N} w_{i,t} i$

EXPERTS FRAMEWORK

Recall

- On iteration t: experts reveal their advice " i "
- forecaster makes a decision $\hat{p}_t = \sum_{i=1}^{N} w_{i,t} i$
- the losses are revealed $\ell(i, y_t)$ and $\ell(\hat{p}_t, y_t)$

EXPERTS FRAMEWORK

Recall

- On iteration t: experts reveal their advice " i "
- forecaster makes a decision $\hat{p}_t = \sum_{i=1}^{N} w_{i,t} i$
- the losses are revealed $\ell(i, y_t)$ and $\ell(\hat{p}_t, y_t)$
- forecaster updates weights $w_{i,t+1}$

EXPERTS FRAMEWORK

Recall

- On iteration t: experts reveal their advice " i "
- forecaster makes a decision $\hat{p}_t = \sum_{i=1}^{N} w_{i,t} i$
- the losses are revealed $\ell(i, y_t)$ and $\ell(\hat{p}_t, y_t)$
- forecaster updates weights $w_{i,t+1}$
- $\ell(\cdot, y_t)$ *is convex*

EXPERTS FRAMEWORK

Recall

- On iteration t: experts reveal their advice " i "
- forecaster makes a decision $\hat{p}_t = \sum_{i=1}^{N} w_{i,t} i$
- the losses are revealed $\ell(i, y_t)$ and $\ell(\hat{p}_t, y_t)$
- forecaster updates weights $w_{i,t+1}$
- $\ell(\cdot, y_t)$ *is convex*

Regret: $R_{i,T} = \hat{L}_T - L_{i,T} = \sum_{t=1}^{T} \ell(\hat{p}_t, y_t) - \ell(i, y_t)$

Goal: $\frac{R_T}{T} = o(T)$

MULTIPLICATIVE WEIGHT ALGORITHMS

Hedge algorithm: $w_{i,t+1} \propto w_{i,t} \exp(-\gamma \ell(i, y_t))$

MULTIPLICATIVE WEIGHT ALGORITHMS

Hedge algorithm: $w_{i,t+1} \propto w_{i,t} \exp(-\gamma \ell(i, y_t))$

- Regret $R_T \leq \frac{\ln N}{\gamma} + \frac{\gamma T}{8}$

MULTIPLICATIVE WEIGHT ALGORITHMS

Hedge algorithm: $w_{i,t+1} \propto w_{i,t} \exp(-\gamma \ell(i, y_t))$

- Regret $R_T \leq \frac{\ln N}{\gamma} + \frac{\gamma T}{8}$
- taking $\gamma_t = \sqrt{\frac{8 \ln N}{T}}$ yields $R_T \leq \sqrt{\frac{T}{2} \ln N}$

MULTIPLICATIVE WEIGHT ALGORITHMS

Hedge algorithm: $w_{i,t+1} \propto w_{i,t} \exp(-\gamma \ell(i, y_t))$

- Regret $R_T \leq \frac{\ln N}{\gamma} + \frac{\gamma T}{8}$
- taking $\gamma_t = \sqrt{\frac{8 \ln N}{T}}$ yields $R_T \leq \sqrt{\frac{T}{2} \ln N}$
- small losses: $\frac{R_T}{T} \leq \frac{1}{T} \left(\frac{\gamma}{1-e^{-\gamma}} - 1 \right) L_i^* + \frac{1}{T} \frac{\ln N}{1-e^{-\gamma}}$

Problem

Problem

- N experts

PROBLEM

- N experts
- Hedge algorithm: $w_{i,t+1} \propto w_{i,t} \exp(-\gamma \ell(i, y_t)) \quad i \in [N]$

PROBLEM

- N experts
- Hedge algorithm: $w_{i,t+1} \propto w_{i,t} \exp(-\gamma \ell(i, y_t)) \quad i \in [N]$
- complexity $\propto N$

PROBLEM

- N experts
- Hedge algorithm: $w_{i,t+1} \propto w_{i,t} \exp(-\gamma \ell(i, y_t)) \quad i \in [N]$
- complexity $\propto N$
- N very large $\implies$ Hedge algorithm infeasible

PROBLEM

- N experts
- Hedge algorithm: $w_{i,t+1} \propto w_{i,t} \exp(-\gamma \ell(i, y_t)) \quad i \in [N]$
- complexity $\propto N$
- N very large $\implies$ Hedge algorithm infeasible
- Experts assume a certain structure

PROBLEM

- N experts
- Hedge algorithm: $w_{i,t+1} \propto w_{i,t} \exp(-\gamma \ell(i, y_t)) \quad i \in [N]$
- complexity $\propto N$
- N very large $\implies$ Hedge algorithm infeasible
- Experts assume a certain structure
- possible construction of efficient prediction algorithms

PROBLEM

- N experts
- Hedge algorithm: $w_{i,t+1} \propto w_{i,t} \exp(-\gamma \ell(i, y_t)) \quad i \in [N]$
- complexity $\propto N$
- N very large $\implies$ Hedge algorithm infeasible
- Experts assume a certain structure
- possible construction of efficient prediction algorithms
- two cases: *Tracking the best expert* and *tree experts*

PROBLEM

- N experts
- Hedge algorithm: $w_{i,t+1} \propto w_{i,t} \exp(-\gamma \ell(i, y_t)) \quad i \in [N]$
- complexity $\propto N$
- N very large $\implies$ Hedge algorithm infeasible
- Experts assume a certain structure
- possible construction of efficient prediction algorithms
- two cases: *Tracking the best expert* and *tree experts*
- Idea: track N "base" experts (actions) for $M = f(N)$ experts

PROBLEM

- N experts
- Hedge algorithm: $w_{i,t+1} \propto w_{i,t} \exp(-\gamma \ell(i, y_t)) \quad i \in [N]$
- complexity $\propto N$
- N very large $\implies$ Hedge algorithm infeasible
- Experts assume a certain structure
- possible construction of efficient prediction algorithms
- two cases: *Tracking the best expert* and *tree experts*
- Idea: track N "base" experts (actions) for $M = f(N)$ experts
- Same weights, same bound $R_T \leq \sqrt{\frac{T}{2} \ln M}$

TRAKING BEST EXPERT: SETTING

Regret against the best performing single action:

$$R_T = \hat{L}_T - \min_{i=1,\cdots,N} L_{i,T} = \sum_{t=1}^{T} \ell(\hat{p}_t, y_t) - \min_{i=1,\cdots,N} \sum_{t=1}^{T} \ell(i, y_t)$$

TRAKING BEST EXPERT: SETTING

Regret against the best performing single action:

$$R_T = \hat{L}_T - \min_{i=1,\cdots,N} L_{i,T} = \sum_{t=1}^{T} \ell(\hat{p}_t, y_t) - \min_{i=1,\cdots,N} \sum_{t=1}^{T} \ell(i, y_t)$$

When allowed to switch actions, Tracking regret:

$$\tilde{R}_T = \sum_{t=1}^{T} \ell(\hat{p}_t, y_t) - \min_{(i_1,\cdots,i_T)} \sum_{t=1}^{T} \ell(i_t, y_t)$$

where $(i_1, \cdots, i_T) \in \{1, \cdots, N\}^T$

TRAKING BEST EXPERT: SETTING

Regret against the best performing single action:

$$R_T = \hat{L}_T - \min_{i=1,\cdots,N} L_{i,T} = \sum_{t=1}^{T} \ell(\hat{p}_t, y_t) - \min_{i=1,\cdots,N} \sum_{t=1}^{T} \ell(i, y_t)$$

When allowed to switch actions, Tracking regret:

$$\tilde{R}_T = \sum_{t=1}^{T} \ell(\hat{p}_t, y_t) - \min_{(i_1,\cdots,i_T)} \sum_{t=1}^{T} \ell(i_t, y_t)$$

where $(i_1, \cdots, i_T) \in \{1, \cdots, N\}^T$

- N "base" experts $i \in \{1, \cdots, N\} = [N]$
- N^T "compound" experts $(i_1, \cdots, i_T) \in [N]^T$

BOUNDED NUMBER OF SWITCHES

We impose $\leq m$ switches:

$$\text{actions: } \{(i_1, \cdots, i_1, \quad i_2, \cdots, i_2, \quad \cdots, i_{m+1}, \cdots, i_{m+1})\}$$

BOUNDED NUMBER OF SWITCHES

We impose $\leq m$ switches:

$$\text{actions: } \{(i_1, \cdots, i_1, \quad i_2, \cdots, i_2, \quad \cdots, i_{m+1}, \cdots, i_{m+1})\}$$

- less "compound experts"
- more tractable, scalable algorithms

BOUNDED NUMBER OF SWITCHES

We impose $\leq m$ switches:

$$\text{actions: } \{(i_1, \cdots, i_1, \quad i_2, \cdots, i_2, \quad \cdots, i_{m+1}, \cdots, i_{m+1})\}$$

- less "compound experts"
- more tractable, scalable algorithms

$$\begin{aligned} M &:= \#\text{ "compound" experts} \\ &= \sum_{k=0}^{m} \binom{T-1}{k} N(N-1)^k \\ &\leq N^{m+1} \exp\left((T-1)\,\mathrm{H}(m/(T-1))\right) \end{aligned}$$

with $\mathrm{H}(x) = -x \ln x - (1-x)\ln(1-x)$

BOUNDED NUMBER OF SWITCHES

We impose $\leq m$ switches:

$$\text{actions: } \{(i_1, \cdots, i_1, \quad i_2, \cdots, i_2, \quad \cdots, i_{m+1}, \cdots, i_{m+1})\}$$

- less "compound experts"
- more tractable, scalable algorithms

$$\begin{aligned} M &:= \#\text{ "compound" experts} \\ &= \textstyle\sum_{k=0}^{m} \binom{T-1}{k} N(N-1)^k \\ &\leq N^{m+1} \exp\left((T-1)\,\mathrm{H}(m/(T-1))\right) \end{aligned}$$

with $\quad \mathrm{H}(x) = -x \ln x - (1-x)\ln(1-x)$

$$\tilde{R}_T \leq \sqrt{\frac{T}{2}\ln N} = \sqrt{\frac{T}{2}\left((m+1)\ln M + (T-1)\,\mathrm{H}(m/(T-1))\right)}$$

THE FIXED SHARE FORECASTER

Parameters: Real numbers $\eta > 0$ and $0 \leq \alpha \leq 1$.

Initialization: $\mathbf{w}_0 = (1/N, \ldots, 1/N)$.

For each round $t = 1, 2, \ldots$

(1) draw an action I_t from $\{1, \ldots, N\}$ according to the distribution

$$p_{i,t} = \frac{w_{i,t-1}}{\sum_{j=1}^{N} w_{j,t-1}}, \qquad i = 1, \ldots, N.$$

(2) obtain Y_t and compute

$$v_{i,t} = w_{i,t-1}\, e^{-\eta\, \ell(i, Y_t)} \qquad \text{for each } i = 1, \ldots, N.$$

(3) let

$$w_{i,t} = \alpha \frac{W_t}{N} + (1-\alpha) v_{i,t} \qquad \text{for each } i = 1, \ldots, N,$$

where $W_t = v_{1,t} + \cdots + v_{N,t}$.

THE FIXED SHARE FORECASTER

Theorem 5.1. Distribution of the action I_t by the fixed share forecaster = distribution of action I'_t by the Hedge algorithm (with specific initialization).

THE FIXED SHARE FORECASTER

Theorem 5.1. Distribution of the action I_t by the fixed share forecaster = distribution of action I'_t by the Hedge algorithm (with specific initialization).

Theorem 5.2. For all compound actions $(i_1, \cdots, i_T)$ with $\leq m$ switches, the tracking regret of the fixed share forecaster satisfies:

$$\tilde{R}_T \leq \frac{m+1}{\gamma} \ln N + \frac{1}{\gamma} \ln \frac{1}{(\alpha/N)^m (1-\alpha)^{T-m-1}} + \frac{\gamma}{8} T$$

THE FIXED SHARE FORECASTER

Theorem 5.1. Distribution of the action I_t by the fixed share forecaster = distribution of action I'_t by the Hedge algorithm (with specific initialization).

Theorem 5.2. For all compound actions $(i_1, \cdots, i_T)$ with $\leq m$ switches, the tracking regret of the fixed share forecaster satisfies:

$$\tilde{R}_T \leq \frac{m+1}{\gamma} \ln N + \frac{1}{\gamma} \ln \frac{1}{(\alpha/N)^m (1-\alpha)^{T-m-1}} + \frac{\gamma}{8} T$$

Corollary 5.1. For $\alpha = m/(T-1)$, and

$$\sqrt{\frac{8}{T}\left((m+1)\ln M + (T-1)\,\mathrm{H}(m/(T-1))\right)}$$

we have same performance bound as Hedge algorithm.

THE FIXED SHARE FORECASTER

Proof of Th 5.1. Let $w^t(i_1, \cdots, i_T)$ = weight of $(i_1, \cdots, i_T)$ at t for Hedge algorithm. Initialize:

$$w_0(i_1, \cdots, i_T) = \frac{1}{N}\left(\frac{\alpha}{N}\right)^{\text{size}(i_1,\cdots,i_T)}\left(1-\alpha+\frac{\alpha}{N}\right)^{T-\text{size}(i_1,\cdots,i_T)} \quad (1)$$

THE FIXED SHARE FORECASTER

Proof of Th 5.1. Let $w^t(i_1, \cdots, i_T)$ = weight of $(i_1, \cdots, i_T)$ at t for Hedge algorithm. Initialize:

$$w_0(i_1, \cdots, i_T) = \frac{1}{N}\left(\frac{\alpha}{N}\right)^{\text{size}(i_1,\cdots,i_T)}\left(1-\alpha+\frac{\alpha}{N}\right)^{T-\text{size}(i_1,\cdots,i_T)} \tag{1}$$

Update: $w'_t(i_1, \cdots, i_T) = w'_{t-1}(i_1, \cdots, i_T)\exp\left(-\gamma\ell(i_t, y_t)\right)$

THE FIXED SHARE FORECASTER

Proof of Th 5.1. Let $w^t(i_1, \cdots, i_T)$ = weight of $(i_1, \cdots, i_T)$ at t for Hedge algorithm. Initialize:

$$w_0(i_1, \cdots, i_T) = \frac{1}{N}\left(\frac{\alpha}{N}\right)^{\text{size}(i_1,\cdots,i_T)}\left(1-\alpha+\frac{\alpha}{N}\right)^{T-\text{size}(i_1,\cdots,i_T)} \tag{1}$$

Update: $w'_t(i_1, \cdots, i_T) = w'_{t-1}(i_1, \cdots, i_T)\exp\left(-\gamma\ell(i_t, y_t)\right)$

Choose with distribution $p_{i,t} \propto w'_{i,t} = \sum_{i_1,\cdots,i_T \mid i_{t+1}=i} w'_t(i_1, \cdots, i_T)$

THE FIXED SHARE FORECASTER

Proof of Th 5.1. Let $w^t(i_1, \cdots, i_T)$ = weight of $(i_1, \cdots, i_T)$ at t for Hedge algorithm. Initialize:

$$w_0(i_1, \cdots, i_T) = \frac{1}{N}\left(\frac{\alpha}{N}\right)^{\text{size}(i_1,\cdots,i_T)}\left(1-\alpha+\frac{\alpha}{N}\right)^{T-\text{size}(i_1,\cdots,i_T)} \tag{1}$$

Update: $w'_t(i_1, \cdots, i_T) = w'_{t-1}(i_1, \cdots, i_T)\exp\left(-\gamma\ell(i_t, y_t)\right)$

Choose with distribution $p_{i,t} \propto w'_{i,t} = \sum_{i_1,\cdots,i_T \mid i_{t+1}=i} w'_t(i_1, \cdots, i_T)$

Then we have $w_{i,t} = w'_{i,t}$ by induction.

THE FIXED SHARE FORECASTER

Lemma 5.1. Hedge algorithm with initial weights $w_{1,0} + \cdots + w_{N,0} \leq 1$, then

$$\sum_{t=1}^{T} \ell(\hat{p}_t, y_t) \leq \frac{1}{\gamma} \ln \frac{1}{W_T} + \frac{\gamma T}{8}$$

with $\quad W_T = \sum_{i=1}^{N} w_{i,T} = \sum_{i=1}^{N} w_{i,0} \exp(-\gamma \sum_{t=1}^{T} \ell(i, y_t))$

THE FIXED SHARE FORECASTER

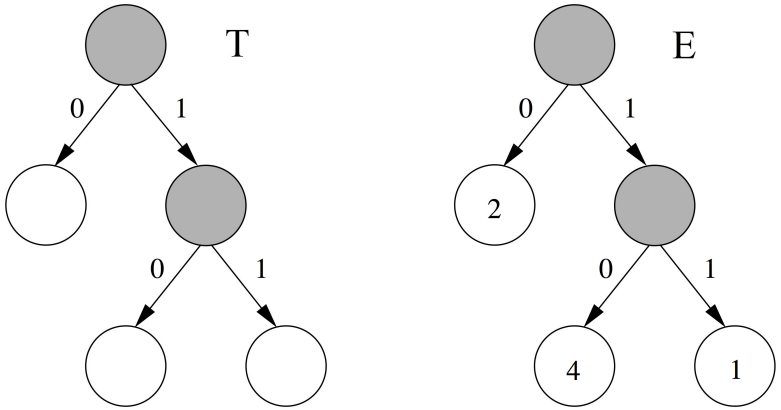
Lemma 5.1. Hedge algorithm with initial weights $w_{1,0} + \cdots + w_{N,0} \leq 1$, then

$$\sum_{t=1}^{T} \ell(\hat{p}_t, y_t) \leq \frac{1}{\gamma} \ln \frac{1}{W_T} + \frac{\gamma T}{8}$$

with $\quad W_T = \sum_{i=1}^{N} w_{i,T} = \sum_{i=1}^{N} w_{i,0} \exp(-\gamma \sum_{t=1}^{T} \ell(i, y_t))$

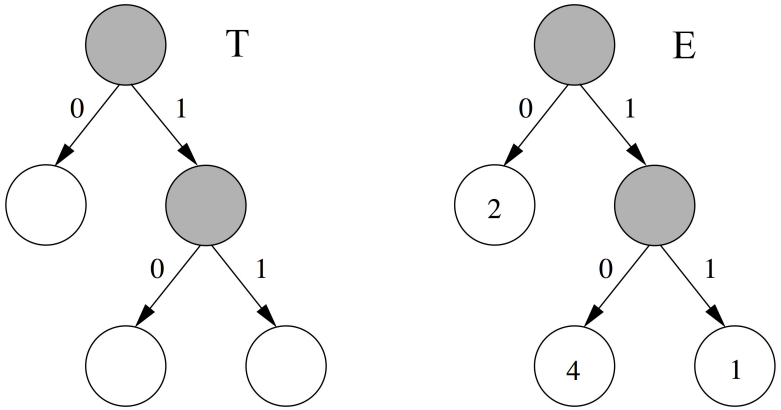
Proof of Th 5.2. Apply lemma 5.1 with weights (1)

Tree Experts: setting



$N = 4$ actions ("base" experts).

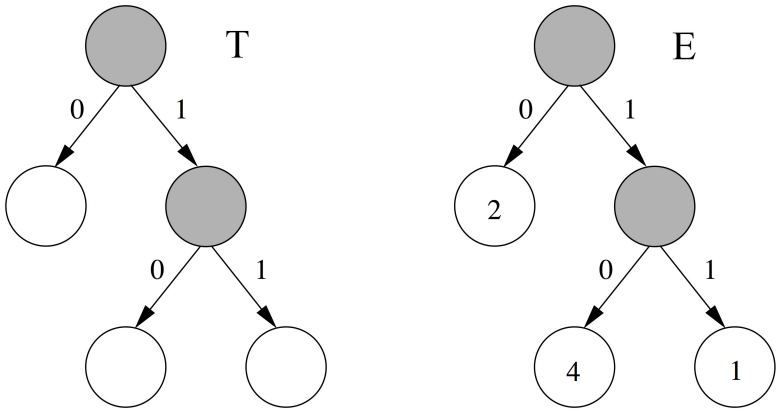
TREE EXPERTS: SETTING



$N = 4$ actions ("base" experts).

Tree expert E
= binary tree T with leaves labeled with actions $\in \{1, 2, 3, 4\}$.

TREE EXPERTS: SETTING

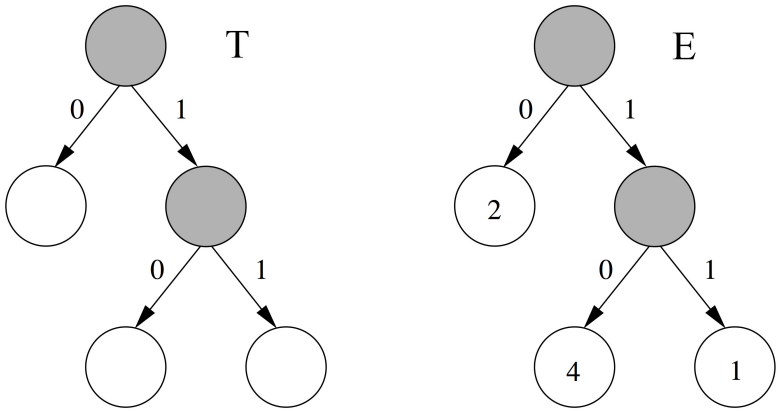


$N = 4$ actions ("base" experts).

Tree expert E
= binary tree T with leaves labeled with actions $\in \{1, 2, 3, 4\}$.

side information: $x = (x_1, x_2, \cdots) \in \{0, 1\}^{\mathbb{N}}$

TREE EXPERTS: SETTING



$N = 4$ actions ("base" experts).

Tree expert E
= binary tree T with leaves labeled with actions $\in \{1, 2, 3, 4\}$.

side information: $x = (x_1, x_2, \cdots) \in \{0, 1\}^{\mathbb{N}}$

- $x = (0, \cdots) \implies i_E(x) = 2$ "E chooses action 2 given x"
- $x = (1, 0, \cdots) \implies i_E(x) = 4$ "E chooses action 4 given x"
- $x = (1, 1, \cdots) \implies i_E(x) = 1$ "E chooses action 1 given x"

Tree Experts: setting

Regret against tree expert E:

$$\bar{R}_{E,T} = \sum_{t=1}^{T} \ell(p_t, y_t) - \sum_{t=1}^{T} \ell(i_E(x_t), y_t)$$

Tree Experts: setting

Regret against tree expert E:

$$\bar{R}_{E,T} = \sum_{t=1}^{T} \ell(p_t, y_t) - \sum_{t=1}^{T} \ell(i_E(x_t), y_t)$$

Some definitions:

- $\text{depth}(E) \leq D \implies$ finite # of trees

Tree Experts: setting

Regret against tree expert E:

$$\bar{R}_{E,T} = \sum_{t=1}^{T} \ell(p_t, y_t) - \sum_{t=1}^{T} \ell(i_E(x_t), y_t)$$

Some definitions:

- $\text{depth}(E) \leq D \implies$ finite # of trees
- $\text{length}(x_t) = D$

Tree Experts: setting

Regret against tree expert E:

$$\bar{R}_{E,T} = \sum_{t=1}^{T} \ell(p_t, y_t) - \sum_{t=1}^{T} \ell(i_E(x_t), y_t)$$

Some definitions:

- $\text{depth}(E) \leq D \implies$ finite # of trees
- $\text{length}(x_t) = D$
- $\|E\| := |\{v \in \text{nodes}(E)\}|$

Tree Experts: setting

Regret against tree expert E:

$$\bar{R}_{E,T} = \sum_{t=1}^{T} \ell(p_t, y_t) - \sum_{t=1}^{T} \ell(i_E(x_t), y_t)$$

Some definitions:

- $\text{depth}(E) \leq D \implies$ finite # of trees
- $\text{length}(x_t) = D$
- $\|E\| := |\{v \in \text{nodes}(E)\}|$
- $\|E\|_D := \|E\| - |\{v \in \text{leaves}(E) \,:\, \text{depth}(v) = D\}|$

TREE EXPERTS: SETTING

Regret against tree expert E:

$$\bar{R}_{E,T} = \sum_{t=1}^{T} \ell(p_t, y_t) - \sum_{t=1}^{T} \ell(i_E(x_t), y_t)$$

Some definitions:

- $\text{depth}(E) \leq D \implies$ finite # of trees
- $\text{length}(x_t) = D$
- $\|E\| := |\{v \in \text{nodes}(E)\}|$
- $\|E\|_D := \|E\| - |\{v \in \text{leaves}(E) \,:\, \text{depth}(v) = D\}|$
- $N^{|\text{leaves}(E)|}$ = # of tree experts E for binary tree T

TREE EXPERTS: SETTING

Regret against tree expert E:

$$\bar{R}_{E,T} = \sum_{t=1}^{T} \ell(p_t, y_t) - \sum_{t=1}^{T} \ell(i_E(x_t), y_t)$$

Some definitions:

- $\text{depth}(E) \leq D \implies$ finite # of trees
- $\text{length}(x_t) = D$
- $\|E\| := |\{v \in \text{nodes}(E)\}|$
- $\|E\|_D := \|E\| - |\{v \in \text{leaves}(E) \,:\, \text{depth}(v) = D\}|$
- $N^{|\text{leaves}(E)|}$ = # of tree experts E for binary tree T
- $u \sqsubseteq v : \quad \text{length}(u) \leq \text{length}(v),\ u_1 = v_1, \cdots, u_d = v_d$

TREE EXPERTS: SETTING

Regret against tree expert E:

$$\bar{R}_{E,T} = \sum_{t=1}^{T} \ell(p_t, y_t) - \sum_{t=1}^{T} \ell(i_E(x_t), y_t)$$

Some definitions:

- $\text{depth}(E) \leq D \implies$ finite # of trees
- $\text{length}(x_t) = D$
- $\|E\| := |\{v \in \text{nodes}(E)\}|$
- $\|E\|_D := \|E\| - |\{v \in \text{leaves}(E) \,:\, \text{depth}(v) = D\}|$
- $N^{|\text{leaves}(E)|}$ = # of tree experts E for binary tree T
- $u \sqsubseteq v : \quad \text{length}(u) \leq \text{length}(v),\ u_1 = v_1, \cdots, u_d = v_d$
- $u \sqsubset v : \quad \text{length}(u) < \text{length}(v),\ u_1 = v_1, \cdots, u_d = v_d$

TREE EXPERTS: HEDGE ALGORITHM

infeasible: $M = N^{2^D}$ tree experts $\implies$ complexity $\propto N^{2^D}$

TREE EXPERTS: HEDGE ALGORITHM

infeasible: $M = N^{2^D}$ tree experts $\implies$ complexity $\propto N^{2^D}$

In theory:

TREE EXPERTS: HEDGE ALGORITHM

infeasible: $M = N^{2^D}$ tree experts $\implies$ complexity $\propto N^{2^D}$

In theory:

- initialize: $w_{E,0} = 2^{-\|E\|_D} N^{-|\text{leaves}(E)|}$

TREE EXPERTS: HEDGE ALGORITHM

infeasible: $M = N^{2^D}$ tree experts $\implies$ complexity $\propto N^{2^D}$

In theory:

- initialize: $w_{E,0} = 2^{-\|E\|_D} N^{-|\text{leaves}(E)|}$
- define: $w_{E,t-1} = w_{E,0} \prod_{v \in \text{leaves}(E)} w_{E,v,t-1}$

TREE EXPERTS: HEDGE ALGORITHM

infeasible: $M = N^{2^D}$ tree experts $\implies$ complexity $\propto N^{2^D}$

In theory:

- initialize: $w_{E,0} = 2^{-\|E\|_D} N^{-|\text{leaves}(E)|}$
- define: $w_{E,t-1} = w_{E,0} \prod_{v \in \text{leaves}(E)} w_{E,v,t-1}$
- update weight of leaf v in E:

$$w_{E,v,t} = \begin{cases} w_{E,v,t-1} \exp(-\gamma \ell(i_E(v), y_t)) & \text{if } v \sqsubseteq x_t \\ w_{E,v,t-1} & \text{otherwise} \end{cases}$$

TREE EXPERTS: HEDGE ALGORITHM

infeasible: $M = N^{2^D}$ tree experts $\implies$ complexity $\propto N^{2^D}$

In theory:

- initialize: $w_{E,0} = 2^{-\|E\|_D} N^{-|\text{leaves}(E)|}$
- define: $w_{E,t-1} = w_{E,0} \prod_{v \in \text{leaves}(E)} w_{E,v,t-1}$
- update weight of leaf v in E:

$$w_{E,v,t} = \begin{cases} w_{E,v,t-1} \exp(-\gamma \ell(i_E(v), y_t)) & \text{if } v \sqsubseteq x_t \\ w_{E,v,t-1} & \text{otherwise} \end{cases}$$

- v unique $\Rightarrow$ one leaf updated: $w_{E,t} = w_{E,t-1} e^{-\gamma \ell(i_E(x_t), y_t)}$

TREE EXPERTS: HEDGE ALGORITHM

infeasible: $M = N^{2^D}$ tree experts $\implies$ complexity $\propto N^{2^D}$

In theory:

- initialize: $w_{E,0} = 2^{-\|E\|_D} N^{-|\text{leaves}(E)|}$
- define: $w_{E,t-1} = w_{E,0} \prod_{v \in \text{leaves}(E)} w_{E,v,t-1}$
- update weight of leaf v in E:

$$w_{E,v,t} = \begin{cases} w_{E,v,t-1} \exp(-\gamma \ell(i_E(v), y_t)) & \text{if } v \sqsubseteq x_t \\ w_{E,v,t-1} & \text{otherwise} \end{cases}$$

- v unique $\Rightarrow$ one leaf updated: $w_{E,t} = w_{E,t-1} e^{-\gamma \ell(i_E(x_t), y_t)}$
- Conditional distribution: $w_{k,t-1} = \sum_{E \mid i_E(x_t) = k} w_{E,t-1}$

TREE EXPERTS: HEDGE ALGORITHM

infeasible: $M = N^{2^D}$ tree experts $\implies$ complexity $\propto N^{2^D}$

In theory:

- initialize: $w_{E,0} = 2^{-\|E\|_D} N^{-|\text{leaves}(E)|}$
- define: $w_{E,t-1} = w_{E,0} \prod_{v \in \text{leaves}(E)} w_{E,v,t-1}$
- update weight of leaf v in E:

$$w_{E,v,t} = \begin{cases} w_{E,v,t-1} \exp(-\gamma \ell(i_E(v), y_t)) & \text{if } v \sqsubseteq x_t \\ w_{E,v,t-1} & \text{otherwise} \end{cases}$$

- v unique $\Rightarrow$ one leaf updated: $w_{E,t} = w_{E,t-1} e^{-\gamma \ell(i_E(x_t), y_t)}$
- Conditional distribution: $w_{k,t-1} = \sum_{E \mid i_E(x_t) = k} w_{E,t-1}$
- choose with probability: $p_{k,t} = \sum_{E \mid i_E(x_t) = k} w_{E,t-1} / \sum'_E w_{E',t-1}$

THE TREE EXPERT FORECASTER

Parameters: Real number $\eta > 0$, integer $D \geq 0$.

Initialization: $\overline{w}_{i,\mathbf{v},0} = 1$, $w_{i,\mathbf{v},0} = 1$ for each $i = 1, \ldots, N$ and for each node $\mathbf{v} = (v_1, \ldots, v_d)$ with $d \leq D$.

For each round $t = 1, 2, \ldots$

(1) draw an action I_t from $\{1, \ldots, N\}$ according to the distribution

$$p_{i,t} = \frac{\overline{w}_{i,\lambda,t-1}}{\sum_{j=1}^{N} \overline{w}_{j,\lambda,t-1}}, \qquad i = 1, \ldots, N;$$

(2) obtain Y_t and compute, for each $\mathbf{v}$ and for each $i = 1, \ldots, N$,

$$w_{i,\mathbf{v},t} = \begin{cases} w_{i,\mathbf{v},t-1}\, e^{-\eta\, \ell(i,Y_t)} & \text{if } \mathbf{v} \sqsubseteq \mathbf{x}_t \\ w_{i,\mathbf{v},t-1} & \text{otherwise;} \end{cases}$$

(3) recursively update each node $\mathbf{v} = (v_1, \ldots, v_d)$ with $d = D, D-1, \ldots, 0$

$$\overline{w}_{i,\mathbf{v},t} = \begin{cases} \frac{1}{2N} w_{i,\mathbf{v},t} & \text{if } \mathbf{v} = \mathbf{x}_t \\ \frac{1}{2N} \sum_{j=1}^{N} w_{j,\mathbf{v},t} & \text{if depth}(\mathbf{v}) = D \\ & \quad \text{and } \mathbf{v} \neq \mathbf{x}_t \\ \frac{1}{2N} w_{i,\mathbf{v},t} + \frac{1}{2N}\left(\overline{w}_{i,\mathbf{v}_0,t} + \overline{w}_{i,\mathbf{v}_1,t}\right) & \text{if } \mathbf{v} \sqsubset \mathbf{x}_t \\ \overline{w}_{i,\mathbf{v},t-1} & \text{if depth}(\mathbf{v}) < D \\ & \quad \text{and } \mathbf{v} \not\sqsubseteq \mathbf{x}_t \end{cases}$$

where $\mathbf{v}_0 = (v_1, \ldots, v_d, 0)$ and $\mathbf{v}_1 = (v_1, \ldots, v_d, 1)$.

TREE EXPERTS

Theorem 5.4. Regret bound of Hedge algorithm over tree experts of depth at most D:

$$\max_{E\,:\,\mathrm{depth}(E)\le D} \bar{R}_{E,T} \le \frac{2^D}{\gamma}\ln(2N) + \frac{\gamma T}{8}$$

$$\max_{E\,:\,\mathrm{depth}(E)\le D} \bar{R}_{E,T} \le \sqrt{\frac{T}{2}2^D\ln(2N)}$$

TREE EXPERTS

Theorem 5.4. Regret bound of Hedge algorithm over tree experts of depth at most D:

$$\max_{E\,:\,\mathrm{depth}(E)\le D} \bar{R}_{E,T} \le \frac{2^D}{\gamma}\ln(2N) + \frac{\gamma T}{8}$$
$$\max_{E\,:\,\mathrm{depth}(E)\le D} \bar{R}_{E,T} \le \sqrt{\frac{T}{2}2^D\ln(2N)}$$

- Recall: $M = N^{2^D} \implies \text{bound} = \sqrt{\frac{T}{2}\ln M} = \sqrt{\frac{T}{2}2^D\ln N}$

TREE EXPERTS

Theorem 5.4. Regret bound of Hedge algorithm over tree experts of depth at most D:

$$\max_{E\,:\,\mathrm{depth}(E)\le D} \bar{R}_{E,T} \le \frac{2^D}{\gamma}\ln(2N) + \frac{\gamma T}{8}$$
$$\max_{E\,:\,\mathrm{depth}(E)\le D} \bar{R}_{E,T} \le \sqrt{\frac{T}{2}2^D\ln(2N)}$$

- Recall: $M = N^{2^D} \implies \text{bound} = \sqrt{\frac{T}{2}\ln M} = \sqrt{\frac{T}{2}2^D\ln N}$
- Three expert algorithm has $N(2^{D+1}-1)$ weights

TREE EXPERTS

Theorem 5.4. Regret bound of Hedge algorithm over tree experts of depth at most D:

$$\max_{E\,:\,\mathrm{depth}(E)\le D} \bar{R}_{E,T} \le \frac{2^D}{\gamma}\ln(2N) + \frac{\gamma T}{8}$$
$$\max_{E\,:\,\mathrm{depth}(E)\le D} \bar{R}_{E,T} \le \sqrt{\frac{T}{2}2^D\ln(2N)}$$

- Recall: $M = N^{2^D} \quad \Longrightarrow \quad \text{bound} = \sqrt{\frac{T}{2}\ln M} = \sqrt{\frac{T}{2}2^D\ln N}$
- Three expert algorithm has $N(2^{D+1}-1)$ weights

Theorem 5.5. Distribution of the action I_t by the tree expert forecaster = distribution of action I'_t by the Hedge algorithm (with initialization $w_{E,0} = 2^{-\|E\|_D}N^{-|\mathrm{leaves}(E)|}$).

Shortest path problem

Shortest path problem

- directed acyclic graph with vertices V, edges $E = \{e_i\}_i$

Shortest path problem

- directed acyclic graph with vertices V, edges $E = \{e_i\}_i$
- paths from u to v: $e^{(1)} = (u, v_1), \cdots, e^{(k)} = (v_{k-1}, v)$

SHORTEST PATH PROBLEM

- directed acyclic graph with vertices V, edges $E = \{e_i\}_i$
- paths from u to v: $e^{(1)} = (u, v_1), \cdots, e^{(k)} = (v_{k-1}, v)$
- exponential number of path experts, but structured on a graph

SHORTEST PATH PROBLEM

- directed acyclic graph with vertices V, edges $E = \{e_i\}_i$
- paths from u to v: $e^{(1)} = (u, v_1), \cdots, e^{(k)} = (v_{k-1}, v)$
- exponential number of path experts, but structured on a graph
- representation: $\mathbf{i} \in \text{paths}(u, v) \subset \{0, 1\}^{|E|}$

SHORTEST PATH PROBLEM

- directed acyclic graph with vertices V, edges $E = \{e_i\}_i$
- paths from u to v: $e^{(1)} = (u, v_1), \cdots, e^{(k)} = (v_{k-1}, v)$
- exponential number of path experts, but structured on a graph
- representation: $\mathbf{i} \in \text{paths}(u, v) \subset \{0, 1\}^{|E|}$
- assume $\forall e \in E$, $\exists\, \mathbf{i} \in \text{paths}(u, v) \mid$ "$e \in \mathbf{i}$"

SHORTEST PATH PROBLEM

- directed acyclic graph with vertices V, edges $E = \{e_i\}_i$
- paths from u to v: $e^{(1)} = (u, v_1), \cdots, e^{(k)} = (v_{k-1}, v)$
- exponential number of path experts, but structured on a graph
- representation: $\mathbf{i} \in \text{paths}(u, v) \subset \{0, 1\}^{|E|}$
- assume $\forall e \in E, \exists\, \mathbf{i} \in \text{paths}(u, v) \mid$ "$e \in \mathbf{i}$"
- outcome y_t = vector of losses $\boldsymbol{\ell}_t \in [0, 1]^{|E|}$ (j-th entry = $\ell_{e_j,t}$)

SHORTEST PATH PROBLEM

- directed acyclic graph with vertices V, edges $E = \{e_i\}_i$
- paths from u to v: $e^{(1)} = (u, v_1), \cdots, e^{(k)} = (v_{k-1}, v)$
- exponential number of path experts, but structured on a graph
- representation: $\mathbf{i} \in \text{paths}(u, v) \subset \{0, 1\}^{|E|}$
- assume $\forall e \in E, \exists\, \mathbf{i} \in \text{paths}(u, v) \mid$ "$e \in \mathbf{i}$"
- outcome y_t = vector of losses $\boldsymbol{\ell}_t \in [0, 1]^{|E|}$ (j-th entry = $\ell_{e_j,t}$)
- loss of path $\mathbf{i}$: $\ell(\mathbf{i}, y_t) = \mathbf{i} \cdot \boldsymbol{\ell}_t$

SHORTEST PATH PROBLEM

- directed acyclic graph with vertices V, edges $E = \{e_i\}_i$
- paths from u to v: $e^{(1)} = (u, v_1), \cdots, e^{(k)} = (v_{k-1}, v)$
- exponential number of path experts, but structured on a graph
- representation: $\mathbf{i} \in \text{paths}(u, v) \subset \{0,1\}^{|E|}$
- assume $\forall e \in E, \exists\, \mathbf{i} \in \text{paths}(u, v) \mid$ "$e \in \mathbf{i}$"
- outcome y_t = vector of losses $\boldsymbol{\ell}_t \in [0,1]^{|E|}$ (j-th entry = $\ell_{e_j,t}$)
- loss of path $\mathbf{i}$: $\ell(\mathbf{i}, y_t) = \mathbf{i} \cdot \boldsymbol{\ell}_t$
- expected regret against shortest path:

$$\sum_{t=1}^{T} \ell(\hat{p}_t, y_t) - \min_{\mathbf{i} \in \text{paths}(u,v)} \sum_{t=1}^{T} \ell(\mathbf{i}, y_t)$$

$$\sum_{t=1}^{T} \ell(\hat{p}_t, y_t) - \min_{\mathbf{i} \in \text{paths}(u,v)} \mathbf{i} \cdot \sum_{t=1}^{T} \boldsymbol{\ell}_t$$

Follow the Perturbed Leader

Follow the Perturbed Leader

- $\mathbf{Z}_1, \cdots, \mathbf{Z}_T$ i.i.d. random vectors $\in \mathbb{R}^{|E|}$

Follow the Perturbed Leader

- $\mathbf{Z}_1, \cdots, \mathbf{Z}_T$ i.i.d. random vectors $\in \mathbb{R}^{|E|}$
- Forecaster chooses:

$$\mathbf{I}_t = \underset{\mathbf{i} \in \text{paths}(u,v)}{\text{argmin}} \ \mathbf{i} \cdot \left(\sum_{s=1}^{t-1} \boldsymbol{\ell}_s + \mathbf{Z_t} \right)$$

Follow the Perturbed Leader

- $\mathbf{Z}_1, \cdots, \mathbf{Z}_T$ i.i.d. random vectors $\in \mathbb{R}^{|E|}$
- Forecaster chooses:

$$\mathbf{I}_t = \underset{\mathbf{i} \in \text{paths}(u,v)}{\text{argmin}} \ \mathbf{i} \cdot \left(\sum_{s=1}^{t-1} \boldsymbol{\ell}_s + \mathbf{Z_t} \right)$$

- efficient linear-time algorithms for shortest path problems

FOLLOW THE PERTURBED LEADER

- $\mathbf{Z}_1, \cdots, \mathbf{Z}_T$ i.i.d. random vectors $\in \mathbb{R}^{|E|}$
- Forecaster chooses:

$$\mathbf{I}_t = \underset{\mathbf{i} \in \text{paths}(u,v)}{\text{argmin}} \ \mathbf{i} \cdot \left(\sum_{s=1}^{t-1} \boldsymbol{\ell}_s + \mathbf{Z_t} \right)$$

- efficient linear-time algorithms for shortest path problems
- apply them to forecaster

Follow the Perturbed Leader

- $\mathbf{Z}_1, \cdots, \mathbf{Z}_T$ i.i.d. random vectors $\in \mathbb{R}^{|E|}$
- Forecaster chooses:

$$\mathbf{I}_t = \underset{\mathbf{i} \in \text{paths}(u,v)}{\text{argmin}} \ \mathbf{i} \cdot \left(\sum_{s=1}^{t-1} \ell_s + \mathbf{Z_t} \right)$$

- efficient linear-time algorithms for shortest path problems
- apply them to forecaster
- good bounds for the forecaster for $\mathbf{Z_t} \sim U\left([0, \Delta]^{|E|}\right)$

Hedge algorithm for shortest path

Hedge algorithm for shortest path

- In theory, choose among exponential # of path experts:

$$p_{\mathbf{i},t} = \exp\left(-\gamma \sum_{s=1}^{t-1} \mathbf{i} \cdot \boldsymbol{\ell}_s\right) \Big/ \sum_{\mathbf{i}' \in \text{paths}(\mathbf{u},\mathbf{v})} \exp\left(-\gamma \sum_{s=1}^{t-1} \mathbf{i}' \cdot \boldsymbol{\ell}_s\right)$$

Hedge algorithm for shortest path

- In theory, choose among exponential # of path experts:

$$p_{\mathbf{i},t} = \exp\left(-\gamma \sum_{s=1}^{t-1} \mathbf{i} \cdot \boldsymbol{\ell}_s\right) / \sum_{\mathbf{i}' \in \text{paths}(\mathbf{u},\mathbf{v})} \exp\left(-\gamma \sum_{s=1}^{t-1} \mathbf{i}' \cdot \boldsymbol{\ell}_s\right)$$

- Cumulative loss of expert $\mathbf{i}$:

$$\sum_{s=1}^{t} \mathbf{i} \cdot \boldsymbol{\ell}_s = \sum_{s=1}^{t} \sum_{e \in \mathbf{i}} \ell_{e,s} = \sum_{e \in \mathbf{i}} L_{e,t}$$

with $L_{e,t} = \sum_{s=1}^{t} \ell_{e,s}$ = cumulative loss by edge e

Hedge algorithm for shortest path

- In theory, choose among exponential # of path experts:

$$p_{\mathbf{i},t} = \exp\left(-\gamma \sum_{s=1}^{t-1} \mathbf{i} \cdot \boldsymbol{\ell}_s\right) / \sum_{\mathbf{i}' \in \text{paths}(\mathbf{u},\mathbf{v})} \exp\left(-\gamma \sum_{s=1}^{t-1} \mathbf{i}' \cdot \boldsymbol{\ell}_s\right)$$

- Cumulative loss of expert $\mathbf{i}$:

$$\sum_{s=1}^{t} \mathbf{i} \cdot \boldsymbol{\ell}_s = \sum_{s=1}^{t} \sum_{e \in \mathbf{i}} \ell_{e,s} = \sum_{e \in \mathbf{i}} L_{e,t}$$

  with $L_{e,t} = \sum_{s=1}^{t} \ell_{e,s}$ = cumulative loss by edge e
- idea: Hedge algorithm on $|V|$ edges and construct path by choosing edges one by one

Hedge algorithm for shortest path

Hedge algorithm for shortest path

- recall: $\sum_{s=1}^{t} \mathbf{i} \cdot \boldsymbol{\ell}_s = \sum_{e \in \mathbf{i}} L_{e,t}$

HEDGE ALGORITHM FOR SHORTEST PATH

- recall: $\sum_{s=1}^{t} \mathbf{i} \cdot \boldsymbol{\ell}_s = \sum_{e \in \mathbf{i}} L_{e,t}$
- weight of vertex w at time t:

$$G_t(w) = \sum_{\mathbf{i} \in \text{paths}(w,v)} \exp\left(-\gamma \sum_{e \in \mathbf{i}} L_{e,t}\right)$$

HEDGE ALGORITHM FOR SHORTEST PATH

- recall: $\sum_{s=1}^{t} \mathbf{i} \cdot \boldsymbol{\ell}_s = \sum_{e \in \mathbf{i}} L_{e,t}$
- weight of vertex w at time t:

$$G_t(w) = \sum_{\mathbf{i} \in \text{paths}(w,v)} \exp\left(-\gamma \sum_{e \in \mathbf{i}} L_{e,t}\right)$$

- weights can be computed efficiently in $O(|E|)$

HEDGE ALGORITHM FOR SHORTEST PATH

- recall: $\sum_{s=1}^{t} \mathbf{i} \cdot \boldsymbol{\ell}_s = \sum_{e \in \mathbf{i}} L_{e,t}$
- weight of vertex w at time t:

$$G_t(w) = \sum_{\mathbf{i} \in \text{paths}(w,v)} \exp\left(-\gamma \sum_{e \in \mathbf{i}} L_{e,t}\right)$$

- weights can be computed efficiently in $O(|E|)$
- for $\mathbf{i} \in \text{paths}(u, v)$, let $v_{\mathbf{i},k} = k$-th vertex along $\mathbf{i}$

HEDGE ALGORITHM FOR SHORTEST PATH

- recall: $\sum_{s=1}^{t} \mathbf{i} \cdot \boldsymbol{\ell}_s = \sum_{e \in \mathbf{i}} L_{e,t}$
- weight of vertex w at time t:

$$G_t(w) = \sum_{\mathbf{i} \in \text{paths}(w,v)} \exp\left(-\gamma \sum_{e \in \mathbf{i}} L_{e,t}\right)$$

- weights can be computed efficiently in $O(|E|)$
- for $\mathbf{i} \in \text{paths}(u, v)$, let $v_{\mathbf{i},k} = k$-th vertex along $\mathbf{i}$
- $\mathbf{I}_t$ has distribution $\Pi_k \mathbb{P}_t[v_{\mathbf{I}_t,k} \mid v_{\mathbf{I}_t,k-1}, \cdots, v_{\mathbf{I}_t,0}]$

HEDGE ALGORITHM FOR SHORTEST PATH

- recall: $\sum_{s=1}^{t} \mathbf{i} \cdot \boldsymbol{\ell}_s = \sum_{e \in \mathbf{i}} L_{e,t}$
- weight of vertex w at time t:

$$G_t(w) = \sum_{\mathbf{i} \in \text{paths}(w,v)} \exp\left(-\gamma \sum_{e \in \mathbf{i}} L_{e,t}\right)$$

- weights can be computed efficiently in $O(|E|)$
- for $\mathbf{i} \in \text{paths}(u, v)$, let $v_{\mathbf{i},k} = k$-th vertex along $\mathbf{i}$
- $\mathbf{I}_t$ has distribution $\Pi_k \mathbb{P}_t[v_{\mathbf{I}_t,k} \mid v_{\mathbf{I}_t,k-1}, \cdots, v_{\mathbf{I}_t,0}]$
- we have

$$\mathbb{P}_t[v_{\mathbf{I}_t,k} \mid v_{\mathbf{I}_t,k-1}, \cdots, v_{\mathbf{I}_t,0}] = \begin{cases} G_{t-1}(v_{\mathbf{I}_t,k})/G_{t-1}(v_{\mathbf{I}_t,k-1}) & \text{if } (v_{\mathbf{I}_t,k-1}, v_{\mathbf{I}_t,k}) \in E \\ 0 & \text{otherwise} \end{cases}$$