Predictors that Specialize

Yoav Freund

March 3, 2025

Outline

Outline

The specialists setup

bounding cumulative loss using relative entropy

Outline

The specialists setup

- Up till now we assumed that each expert makes a prediction at each iteration.

The specialists setup

- Up till now we assumed that each expert makes a prediction at each iteration.
- Imagine that experts are specialists, they predict only some of the time.

The specialists setup

- Up till now we assumed that each expert makes a prediction at each iteration.
- Imagine that experts are specialists, they predict only some of the time.
- Gives the designer a lot of flexibility.

The specialists setup

- Up till now we assumed that each expert makes a prediction at each iteration.
- Imagine that experts are specialists, they predict only some of the time.
- Gives the designer a lot of flexibility.
- Generalizes the switching experts setup.

The specialists game

On each iteration $t = 1, 2, 3, \ldots$

- Adversary chooses a set $E^t \subseteq \{1, \ldots, N\}$ of awake specialists.

The specialists game

On each iteration $t = 1, 2, 3, \ldots$

- Adversary chooses a set $E^t \subseteq \{1, \ldots, N\}$ of awake specialists.
- Adversary chooses predictions for specialists in E^t

The specialists game

On each iteration $t = 1, 2, 3, \ldots$

- Adversary chooses a set $E^t \subseteq \{1, \ldots, N\}$ of awake specialists.
- Adversary chooses predictions for specialists in E^t
- Algorithm chooses it's prediction.

The specialists game

On each iteration $t = 1, 2, 3, \ldots$

- Adversary chooses a set $E^t \subseteq \{1, \ldots, N\}$ of awake specialists.
- Adversary chooses predictions for specialists in E^t
- Algorithm chooses it's prediction.
- Adversary chooses outcome.

The specialists game

On each iteration $t = 1, 2, 3, \ldots$

- Adversary chooses a set $E^t \subseteq \{1, \ldots, N\}$ of awake specialists.
- Adversary chooses predictions for specialists in E^t
- Algorithm chooses it's prediction.
- Adversary chooses outcome.
- Algorithm suffers loss. Specialists in E^t suffer loss. Sleeping specialists suffer no loss.

Desired bound

- Algorithm has to predict on each iteration

Desired bound

- Algorithm has to predict on each iteration
- Each specialist might sleep some of the time.

Desired bound

- Algorithm has to predict on each iteration
- Each specialist might sleep some of the time.
- $\Rightarrow$ makes no sense to compare to total loss of best specialist.

Desired bound

- Algorithm has to predict on each iteration
- Each specialist might sleep some of the time.
- $\Rightarrow$ makes no sense to compare to total loss of best specialist.
- **u**: comparator distribution, $u_i \geq 0, \ \sum_i u_i = 1$.

Desired bound

- Algorithm has to predict on each iteration
- Each specialist might sleep some of the time.
- $\Rightarrow$ makes no sense to compare to total loss of best specialist.
- $\mathbf{u}$: comparator distribution, $u_i \geq 0, \ \sum_i u_i = 1$.
- Average loss w.r.t. $\mathbf{u}$: $\ell_{\mathbf{u}}^t \doteq \frac{\sum_{i \in E^t} u_i \ell_i^t}{\sum_{i \in E^t} u_i}$

Desired bound

- Algorithm has to predict on each iteration
- Each specialist might sleep some of the time.
- $\Rightarrow$ makes no sense to compare to total loss of best specialist.
- $\mathbf{u}$: comparator distribution, $u_i \geq 0, \ \sum_i u_i = 1$.
- Average loss w.r.t. $\mathbf{u}$: $\ell^t_{\mathbf{u}} \doteq \frac{\sum_{i \in E^t} u_i \ell^t_i}{\sum_{i \in E^t} u_i}$
- Goal: $L_A \leq \min_{\mathbf{u}} \sum_{t=1}^{T} \ell^t_{\mathbf{u}} + \text{something small}$

Ideas

- We focus on normalized weights:

$$v_i^t = \frac{w_i^t}{\sum_{j=1}^{N} w_i^t}, \quad \mathbf{v}^t = \frac{\mathbf{w}^t}{W^t}$$

Ideas

- We focus on normalized weights:

$$v_i^t = \frac{w_i^t}{\sum_{j=1}^N w_i^t}, \quad \mathbf{v}^t = \frac{\mathbf{w}^t}{W^t}$$

- Algorithm: treat the set E_t as the set of experts.

Ideas

- We focus on normalized weights:

$$v_i^t = \frac{w_i^t}{\sum_{j=1}^N w_i^t}, \quad \mathbf{v}^t = \frac{\mathbf{w}^t}{W^t}$$

- Algorithm: treat the set E_t as the set of experts.
- Normalize the weights of specialists in E_t so that

$$\sum_{i \in E^t} v_i^t = \sum_{i \in E^t} v_i^{t+1}$$

Ideas

- We focus on normalized weights:

$$v_i^t = \frac{w_i^t}{\sum_{j=1}^N w_i^t}, \quad \mathbf{v}^t = \frac{\mathbf{w}^t}{W^t}$$

- Algorithm: treat the set E_t as the set of experts.
- Normalize the weights of specialists in E_t so that

$$\sum_{i \in E^t} v_i^t = \sum_{i \in E^t} v_i^{t+1}$$

- In particular: total weight is always 1.

The log-loss case

- $x_{t,i}$ prediction of expert i on iteration t

The log-loss case

- $x_{t,i}$ prediction of expert i on iteration t
- $\hat{y}_t$ prediction of algorithm.

The log-loss case

- $x_{t,i}$ prediction of expert i on iteration t
- $\hat{y}_t$ prediction of algorithm.
- y_t outcome at iteration t (0 or 1)

The log-loss case

- $x_{t,i}$ prediction of expert i on iteration t
- $\hat{y}_t$ prediction of algorithm.
- y_t outcome at iteration t (0 or 1)
- $$\ell_A^t = L(\hat{y}_t, y_t) = \begin{cases} -\ln \hat{y}_t & \text{if } y_t = 1 \\ -\ln(1 - \hat{y}_t) & \text{if } y_t = 0 \end{cases}$$

The log-loss case

- $x_{t,i}$ prediction of expert i on iteration t
- $\hat{y}_t$ prediction of algorithm.
- y_t outcome at iteration t (0 or 1)
- $$\ell_A^t = L(\hat{y}_t, y_t) = \begin{cases} -\ln \hat{y}_t & \text{if } y_t = 1 \\ -\ln(1 - \hat{y}_t) & \text{if } y_t = 0 \end{cases}$$
- ℓ_i^t defined similarly for expert i

Algorithm Bayes

Iterate for $t = 1, 2, \ldots, T$

Algorithm Bayes

Iterate for $t = 1, 2, \ldots, T$

1. Predict with the weighted average of the experts' predictions:

$$\hat{y}_t = \sum_{i=1}^{N} p_{t,i} x_{t,i}$$

Algorithm Bayes

Iterate for $t = 1, 2, \dots, T$

1. Predict with the weighted average of the experts' predictions:

$$\hat{y}_t = \sum_{i=1}^{N} p_{t,i} x_{t,i}$$

2. Observe outcome y_t.

Algorithm Bayes

Iterate for $t = 1, 2, \ldots, T$

1. Predict with the weighted average of the experts' predictions:

$$\hat{y}_t = \sum_{i=1}^{N} p_{t,i} x_{t,i}$$

2. Observe outcome y_t.
3. Update the posterior distribution:

$$p_{t+1,i} = \begin{cases} \frac{p_{t,i} x_{t,i}}{\hat{y}_t} & \text{if } y_t = 1 \\ \frac{p_{t,i}(1 - x_{t,i})}{1 - \hat{y}_t} & \text{if } y_t = 0 \end{cases}$$

Algorithm SBayes

Iterate for $t = 1, 2, \ldots, T$

Algorithm SBayes

Iterate for $t = 1, 2, \ldots, T$

1. Predict with the weighted average of the predictions of the awake specialists:

$$\hat{y}_t = \frac{\sum_{i \in E_t} p_{t,i} x_{t,i}}{\sum_{i \in E_t} p_{t,i}}$$

where E_t is the set of awake specialists.

Algorithm SBayes

Iterate for $t = 1, 2, \ldots, T$

1. Predict with the weighted average of the predictions of the awake specialists:
$$\hat{y}_t = \frac{\sum_{i \in E_t} p_{t,i} x_{t,i}}{\sum_{i \in E_t} p_{t,i}}$$
where E_t is the set of awake specialists.
2. Observe outcome y_t.

Algorithm SBayes

Iterate for $t = 1, 2, \ldots, T$

1. Predict with the weighted average of the predictions of the awake specialists:
$$\hat{y}_t = \frac{\sum_{i \in E_t} p_{t,i} x_{t,i}}{\sum_{i \in E_t} p_{t,i}}$$
where E_t is the set of awake specialists.
2. Observe outcome y_t.
3. Update the posterior distribution:
$$\text{if } i \in E_t : \quad p_{t+1,i} = \begin{cases} \frac{p_{t,i} x_{t,i}}{\hat{y}_t} & \text{if } y_t = 1 \\ \frac{p_{t,i}(1 - x_{t,i})}{1 - \hat{y}_t} & \text{if } y_t = 0 \end{cases}$$
$$\text{if } i \notin E_t : \quad p_{t+1,i} = p_{t,i}$$

Bound for SBayes

- For any sequence of awake specialists $E_1, \ldots, E_T$, specialist predictions and outcomes, and for any comparator $\mathbf{u}$:

$$\sum_{t=1} u(E^t)\ell_A^t \leq \sum_{t=1}^{T} \sum_{i \in E^t} u_i \ell_i^t + \mathrm{RE}\left(\mathbf{u} \parallel \mathbf{v}^1\right)$$

Bound for SBayes

- For any sequence of awake specialists $E_1, \ldots, E_T$, specialist predictions and outcomes, and for any comparator $\mathbf{u}$:

$$\sum_{t=1} u(E^t)\ell_A^t \leq \sum_{t=1}^{T} \sum_{i \in E^t} u_i \ell_i^t + \mathrm{RE}\left(\mathbf{u} \parallel \mathbf{v}^1\right)$$

- $\mathrm{RE}\left(\mathbf{u} \parallel \mathbf{v}\right) \doteq \sum_i u_i \log \frac{u_i}{v_i}$

Bound for SBayes

- For any sequence of awake specialists $E_1, \ldots, E_T$, specialist predictions and outcomes, and for any comparator $\mathbf{u}$:

$$\sum_{t=1} u(E^t)\ell_A^t \leq \sum_{t=1}^{T} \sum_{i \in E^t} u_i \ell_i^t + \mathrm{RE}\left(\mathbf{u} \parallel \mathbf{v}^1\right)$$

- $\mathrm{RE}\left(\mathbf{u} \parallel \mathbf{v}\right) \doteq \sum_i u_i \log \frac{u_i}{v_i}$
- $u(E^t) \doteq \sum_{i \in E^t} u_i$

Bound for SBayes

- For any sequence of awake specialists $E_1, \ldots, E_T$, specialist predictions and outcomes, and for any comparator $\mathbf{u}$:

$$\sum_{t=1} u(E^t)\ell_A^t \leq \sum_{t=1}^{T} \sum_{i \in E^t} u_i \ell_i^t + \mathrm{RE}\left(\mathbf{u} \parallel \mathbf{v}^1\right)$$

- $\mathrm{RE}\left(\mathbf{u} \parallel \mathbf{v}\right) \doteq \sum_i u_i \log \frac{u_i}{v_i}$
- $u(E^t) \doteq \sum_{i \in E^t} u_i$
- If we assume that $u(E^t) = U$ is constant, we get

$$L_A \leq \sum_{t=1}^{T} \ell_{\mathbf{u}}^t + \frac{\mathrm{RE}\left(\mathbf{u} \parallel \mathbf{v}^1\right)}{U}$$

Proof of Bound (1)

Lemma:

$$\mathrm{RE}\left(\mathbf{u} \parallel p_t\right) - \mathrm{RE}\left(\mathbf{u} \parallel p_{t+1}\right) = u(E_t)L(\hat{y}_t, y_t) - \sum_{i \in E_t} u_i L(x_{t,i}, y_t)$$

Proof of Bound (1)

Lemma:

$$\mathrm{RE}\left(\mathbf{u} \;\|\; p_t\right) - \mathrm{RE}\left(\mathbf{u} \;\|\; p_{t+1}\right) = u(E_t)L(\hat{y}_t, y_t) - \sum_{i \in E_t} u_i L(x_{t,i}, y_t)$$

From definition of $\mathrm{RE}\left(\;\|\; \right)$:

$$\mathrm{RE}\left(\mathbf{u} \;\|\; p_t\right) - \mathrm{RE}\left(\mathbf{u} \;\|\; p_{t+1}\right) = \sum_{i \in E_t} u_i \ln \frac{p_{t+1,i}}{p_{t,i}}$$

Proof of Bound (1)

Lemma:

$$\mathrm{RE}\left(\mathbf{u} \parallel p_t\right) - \mathrm{RE}\left(\mathbf{u} \parallel p_{t+1}\right) = u(E_t)L(\hat{y}_t, y_t) - \sum_{i \in E_t} u_i L(x_{t,i}, y_t)$$

From definition of $\mathrm{RE}\left(\parallel \right)$:

$$\mathrm{RE}\left(\mathbf{u} \parallel p_t\right) - \mathrm{RE}\left(\mathbf{u} \parallel p_{t+1}\right) = \sum_{i \in E_t} u_i \ln \frac{p_{t+1,i}}{p_{t,i}}$$

If $y_t = 1$ the RHS is equal to

$$\begin{aligned} \sum_{i \in E_t} u_i \ln \frac{x_{t,i}}{\hat{y}_t} &= \sum_{i \in E_t} u_i \ln x_{t,i} - u(E_t) \ln \hat{y}_t \\ &= -\sum_{i \in E_t} u_i L(X_{t,i}, y_t) + u(E_t) L(\hat{y}_t, y_t) \end{aligned}$$

Proof of Bound (1)

Lemma:

$$\mathrm{RE}\left(\mathbf{u} \parallel p_t\right) - \mathrm{RE}\left(\mathbf{u} \parallel p_{t+1}\right) = u(E_t)L(\hat{y}_t, y_t) - \sum_{i \in E_t} u_i L(x_{t,i}, y_t)$$

From definition of $\mathrm{RE}\left(\parallel \right)$:

$$\mathrm{RE}\left(\mathbf{u} \parallel p_t\right) - \mathrm{RE}\left(\mathbf{u} \parallel p_{t+1}\right) = \sum_{i \in E_t} u_i \ln \frac{p_{t+1,i}}{p_{t,i}}$$

If $y_t = 1$ the RHS is equal to

$$\begin{aligned} \sum_{i \in E_t} u_i \ln \frac{x_{t,i}}{\hat{y}_t} &= \sum_{i \in E_t} u_i \ln x_{t,i} - u(E_t) \ln \hat{y}_t \\ &= -\sum_{i \in E_t} u_i L(X_{t,i}, y_t) + u(E_t)L(\hat{y}_t, y_t) \end{aligned}$$

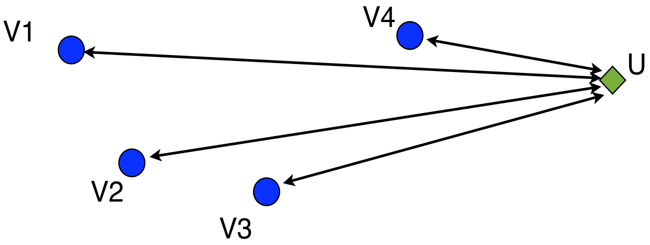
Similarly for $y_t = 0$

Visual intuition

$$\mathrm{RE}\left(\mathbf{u} \;\|\; \mathbf{v}^t\right) - \mathrm{RE}\left(\mathbf{u} \;\|\; \mathbf{v}^{t+1}\right) = \ell_A^t - \mathbf{u} \cdot \boldsymbol{\ell}^t$$

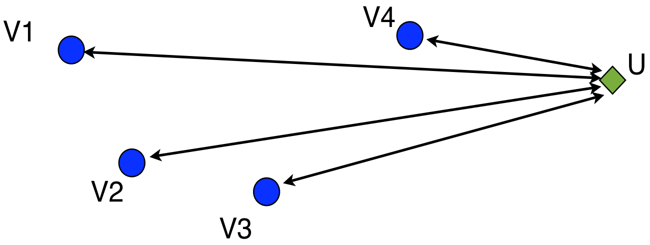
Visual intuition

$$\mathrm{RE}\left(\mathbf{u} \;\|\; \mathbf{v}^{t}\right) - \mathrm{RE}\left(\mathbf{u} \;\|\; \mathbf{v}^{t+1}\right) = \ell_A^t - \mathbf{u} \cdot \boldsymbol{\ell}^t$$



Visual intuition

$$\mathrm{RE}\left(\mathbf{u} \;\|\; \mathbf{v}^t\right) - \mathrm{RE}\left(\mathbf{u} \;\|\; \mathbf{v}^{t+1}\right) = \ell_A^t - \mathbf{u} \cdot \boldsymbol{\ell}^t$$



$\mathbf{v}^{t+1}$ is chosen to minimize $\mathrm{RE}\left(\mathbf{v}^{t+1} \;\|\; \mathbf{v}^t\right) + \mathbf{v}^{t+1} \cdot \boldsymbol{\ell}^t$

Proof of Bound (2)

Summing over $t = 1, \ldots, T$:

$$\mathrm{RE}\left(\mathbf{u} \parallel p_t\right) - \mathrm{RE}\left(\mathbf{u} \parallel p_{t+1}\right) = u(E_t)L(\hat{y}_t, y_t) - \sum_{i \in E_t} u_i L(x_{t,i}, y_t)$$

Proof of Bound (2)

Summing over $t = 1, \ldots, T$:

$$\mathrm{RE}\left(\mathbf{u} \parallel p_t\right) - \mathrm{RE}\left(\mathbf{u} \parallel p_{t+1}\right) = u(E_t)L(\hat{y}_t, y_t) - \sum_{i \in E_t} u_i L(x_{t,i}, y_t)$$

We get

$$\begin{aligned}
\mathrm{RE}\left(\mathbf{u} \parallel p_1\right) &\geq \mathrm{RE}\left(\mathbf{u} \parallel p_1\right) - \mathrm{RE}\left(\mathbf{u} \parallel p_{T+1}\right) \\
&= \sum_{t=1}^{T} u(E_t)L(\hat{y}_t, y_t) - \sum_{t=1}^{T} \sum_{i \in E_t} u_i L(x_{t,i}, y_t)
\end{aligned}$$

bounding general loss using relative entropy

- Suppose that loss is (a, c)-achievable.

bounding general loss using relative entropy

- Suppose that loss is (a, c)-achievable.
- Achievable with Vovk algorithm, learning rate $\eta = \frac{a}{c}$

bounding general loss using relative entropy

- Suppose that loss is (a, c)-achievable.
- Achievable with Vovk algorithm, learning rate $\eta = \frac{a}{c}$
- Let **u** be an arbitrary distribution vector over experts.

bounding general loss using relative entropy

- Suppose that loss is (a, c)-achievable.
- Achievable with Vovk algorithm, learning rate $\eta = \frac{a}{c}$
- Let $\mathbf{u}$ be an arbitrary distribution vector over experts.
- Lemma: $\mathrm{RE}\left(\mathbf{u} \parallel \mathbf{v}^t\right) - \mathrm{RE}\left(\mathbf{u} \parallel \mathbf{v}^{t+1}\right) \geq \frac{1}{c}\ell_A^t - \frac{a}{c}\mathbf{u} \cdot \ell^t$

bounding general loss using relative entropy

- Suppose that loss is (a, c)-achievable.
- Achievable with Vovk algorithm, learning rate $\eta = \frac{a}{c}$
- Let $\mathbf{u}$ be an arbitrary distribution vector over experts.
- Lemma: $\mathrm{RE}\left(\mathbf{u} \parallel \mathbf{v}^t\right) - \mathrm{RE}\left(\mathbf{u} \parallel \mathbf{v}^{t+1}\right) \geq \frac{1}{c}\ell_A^t - \frac{a}{c}\mathbf{u} \cdot \boldsymbol{\ell}^t$
- Summing over $t = 1, \ldots, T$ we get:
  $\mathrm{RE}\left(\mathbf{u} \parallel \mathbf{v}^1\right) - \mathrm{RE}\left(\mathbf{u} \parallel \mathbf{v}^{T+1}\right) = \frac{1}{c}L_A - \frac{a}{c}\mathbf{u} \cdot \sum_{t=1}^T \boldsymbol{\ell}^t$

bounding general loss using relative entropy

- Suppose that loss is (a, c)-achievable.
- Achievable with Vovk algorithm, learning rate $\eta = \frac{a}{c}$
- Let $\mathbf{u}$ be an arbitrary distribution vector over experts.
- Lemma: $\mathrm{RE}\left(\mathbf{u} \parallel \mathbf{v}^t\right) - \mathrm{RE}\left(\mathbf{u} \parallel \mathbf{v}^{t+1}\right) \geq \frac{1}{c}\ell_A^t - \frac{a}{c}\mathbf{u} \cdot \boldsymbol{\ell}^t$
- Summing over $t = 1, \ldots, T$ we get:
  $\mathrm{RE}\left(\mathbf{u} \parallel \mathbf{v}^1\right) - \mathrm{RE}\left(\mathbf{u} \parallel \mathbf{v}^{T+1}\right) = \frac{1}{c}L_A - \frac{a}{c}\mathbf{u} \cdot \sum_{t=1}^T \boldsymbol{\ell}^t$
- $L_A \leq \min_{\mathbf{u}} \left(a\mathbf{u} \cdot \sum_{t=1}^T \boldsymbol{\ell}^t + c\mathrm{RE}\left(\mathbf{u} \parallel \mathbf{v}^1\right)\right)$

bounding general loss using relative entropy

- Suppose that loss is (a, c)-achievable.
- Achievable with Vovk algorithm, learning rate $\eta = \frac{a}{c}$
- Let $\mathbf{u}$ be an arbitrary distribution vector over experts.
- Lemma: $\mathrm{RE}\left(\mathbf{u} \parallel \mathbf{v}^t\right) - \mathrm{RE}\left(\mathbf{u} \parallel \mathbf{v}^{t+1}\right) \geq \frac{1}{c}\ell_A^t - \frac{a}{c}\mathbf{u} \cdot \boldsymbol{\ell}^t$
- Summing over $t = 1, \ldots, T$ we get:
  $\mathrm{RE}\left(\mathbf{u} \parallel \mathbf{v}^1\right) - \mathrm{RE}\left(\mathbf{u} \parallel \mathbf{v}^{T+1}\right) = \frac{1}{c}L_A - \frac{a}{c}\mathbf{u} \cdot \sum_{t=1}^{T} \boldsymbol{\ell}^t$
- $L_A \leq \min_{\mathbf{u}} \left(a\mathbf{u} \cdot \sum_{t=1}^{T} \boldsymbol{\ell}^t + c\mathrm{RE}\left(\mathbf{u} \parallel \mathbf{v}^1\right)\right)$
- For any mixable loss, $a = 1$, using $\mathbf{u} = \langle 0, \ldots, 0, 1, 0, \ldots, 0 \rangle$ and $\mathbf{v}^1 = \langle 1/N, \ldots, 1/N \rangle$ we get the old bound: $L_A \leq \min_i L_i + c \log N$

Example 1 Pruning trees

- Consider the context algorithm.

Example 1 Pruning trees

- Consider the context algorithm.
- Each pruning is a generalist.

Example 1 Pruning trees

- Consider the context algorithm.
- Each pruning is a generalist.
- Each node is a specialist.

Example 1 Pruning trees

- Consider the context algorithm.
- Each pruning is a generalist.
- Each node is a specialist.
- Gives an inferior algorithm (regret bound is twice as large as Context alg

Example 1 Pruning trees

- Consider the context algorithm.
- Each pruning is a generalist.
- Each node is a specialist.
- Gives an inferior algorithm (regret bound is twice as large as Context alg
- But much easier to generalize.

Example 2: Switching Experts

- Consider the fixed share switching algorithm

Example 2: Switching Experts

- Consider the fixed share switching algorithm
- Each sequence of d switches between base expert = generalist.

Example 2: Switching Experts

- Consider the fixed share switching algorithm
- Each sequence of d switches between base expert = generalist.
- Specialist for each base expert - sleeps unless active.

Example 2: Switching Experts

- Consider the fixed share switching algorithm
- Each sequence of d switches between base expert = generalist.
- Specialist for each base expert - sleeps unless active.
- gives the same algorithm.

Example 3: Routing

- Consider a communication network defined by a DAG.

Example 3: Routing

- Consider a communication network defined by a DAG.
- goal: send packets from source to sink with minimal delay.

Example 3: Routing

- Consider a communication network defined by a DAG.
- goal: send packets from source to sink with minimal delay.
- protocol: after route is selected, delay is known.

Example 3: Routing

- Consider a communication network defined by a DAG.
- goal: send packets from source to sink with minimal delay.
- protocol: after route is selected, delay is known.
- This is a multiple arm problem.

Example 3: Routing

- Consider a communication network defined by a DAG.
- goal: send packets from source to sink with minimal delay.
- protocol: after route is selected, delay is known.
- This is a multiple arm problem.
- Generalist: a possible route.

Example 3: Routing

- Consider a communication network defined by a DAG.
- goal: send packets from source to sink with minimal delay.
- protocol: after route is selected, delay is known.
- This is a multiple arm problem.
- Generalist: a possible route.
- Specialist: The choice of next hop for an individual router.

Example 3: Routing

- Consider a communication network defined by a DAG.
- goal: send packets from source to sink with minimal delay.
- protocol: after route is selected, delay is known.
- This is a multiple arm problem.
- Generalist: a possible route.
- Specialist: The choice of next hop for an individual router.
- Number of generalist is exponential in the number of specialists.

Example 3: Routing

- Consider a communication network defined by a DAG.
- goal: send packets from source to sink with minimal delay.
- protocol: after route is selected, delay is known.
- This is a multiple arm problem.
- Generalist: a possible route.
- Specialist: The choice of next hop for an individual router.
- Number of generalist is exponential in the number of specialists.
- Is it possible to achieve, using specialists?

Example 3: Routing

- Consider a communication network defined by a DAG.
- goal: send packets from source to sink with minimal delay.
- protocol: after route is selected, delay is known.
- This is a multiple arm problem.
- Generalist: a possible route.
- Specialist: The choice of next hop for an individual router.
- Number of generalist is exponential in the number of specialists.
- Is it possible to achieve, using specialists?
- I don't know, could not find in the literature.