ActiveHedge: Hedge meets Active Learning

Bhuvesh Kumar
(Georgia Tech)

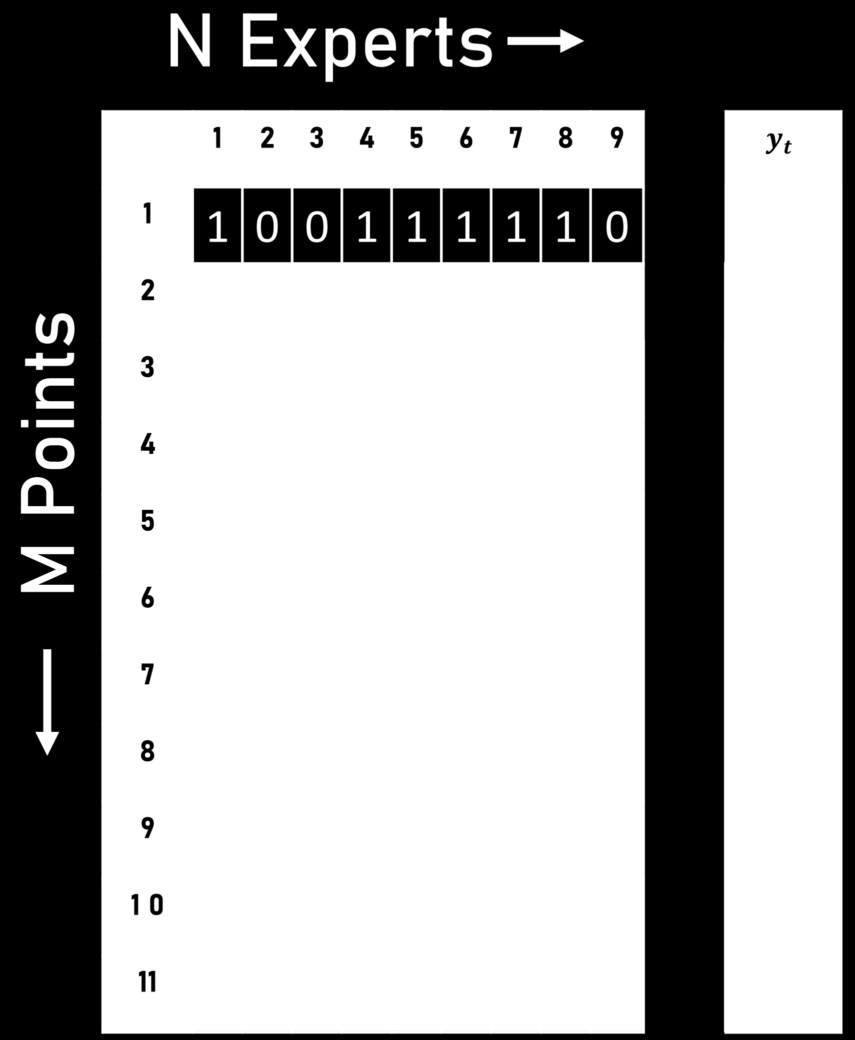
Jacob Abernethy
(Georgia Tech)

Venkatesh Saligrama
(Boston University)

Online Learning with Expert Advice

For $t = 1, \dots, M$ do:

- Receive advice from N experts, $X_t \in \{0,1\}^N$
- Predict $\hat{y}_t = (p_t, 1 - p_t), p_t \in [0,1]$
- Obtain label $y_t \in \{0,1\}$
- Suffer loss
  $\ell(\hat{y}_t, y_t) = (1 - \mathrm{p_t})\mathbb{I}_{\{y_t=1\}} + p_t\mathbb{I}_{\{y_t=0\}}$



Online Learning with Expert Advice

For $t = 1, \ldots, M$ do:

- Receive advice from N experts, $X_t \in \{0,1\}^N$
- Predict $\widehat{y_t} = (p_t, 1 - p_t), p_t \in [0,1]$
- Obtain label $y_t \in \{0,1\}$
- Suffer loss
  $\ell(\widehat{y_t}, y_t) = (1 - \mathrm{p_t})\mathbb{I}_{\{y_t=1\}} + p_t\mathbb{I}_{\{y_t=0\}}$

N Experts →

M Points ↓

	1	2	3	4	5	6	7	8	9	y_t
1	1	0	0	1	1	1	1	1	0	1
2										
3										
4										
5										
6										
7										
8										
9										
10										
11										

Online Learning with Expert Advice

For $t = 1, \dots, M$ do:

- Receive advice from N experts, $X_t \in \{0,1\}^N$
- Predict $\widehat{y_t} = (p_t, 1 - p_t), p_t \in [0,1]$
- Obtain label $y_t \in \{0,1\}$
- Suffer loss
  $\ell(\widehat{y_t}, y_t) = (1 - \mathrm{p_t})\mathbb{I}_{\{y_t=1\}} + p_t\mathbb{I}_{\{y_t=0\}}$

N Experts →

M Points ↓

	1	2	3	4	5	6	7	8	9	y_t
1	1	0	0	1	1	1	1	1	0	1
2	1	1	0	0	0	0	0	0	1	0
3										
4										
5										
6										
7										
8										
9										
10										
11										

Online Learning with Expert Advice

Adversarial expert advice and labels

N Experts →

M Points ↓

	1	2	3	4	5	6	7	8	9	y_t
1	1	0	0	1	1	1	1	1	0	1
2	1	1	0	0	0	0	0	0	1	0
3										
4										
g										
6										
7										
8										
9										
10										
11										

Online Learning with Expert Advice

Regret Minimization:

$$REG_{Alg} = \underbrace{\sum_{t=1}^{M} \ell(\widehat{y}_t, y_t)}_{\text{Loss of the Algorithm}} - \min_{j \in [n]} \underbrace{\sum_{t=1}^{M} \ell\left(X_{t,j}, y_t\right)}_{\text{Loss of the Best Expert}}$$

Hedge

In round t, Hedge weights each expert j's advice $\propto \exp\left(-\eta \sum_{t'=1}^{t} \ell\left(X_{t',j}, y_{t'}\right)\right)$

Hedge: Regret Guarantee

Theorem (Freund and Schapire ' 96)**: Given** L^* **s.t.** $\min_{j \in [n]} \sum_{t=1}^{M} \ell(X_{t,j}, y_t) \leq L^*$**, then by setting** η **appropriately,**

$$REG_{Hedge} \leq \sqrt{2L^* \ln N} + \ln N$$

Hedge: Regret Guarantee

Theorem (Freund and Schapire ' 96)**:** Given L^* s.t. $\min_{j\in[n]} \sum_{t=1}^{M} \ell(X_{t,j}, y_t) \leq L^*$, then by setting η appropriately,

$$REG_{Hedge} \leq \sqrt{2L^* \ln N} + \ln N$$

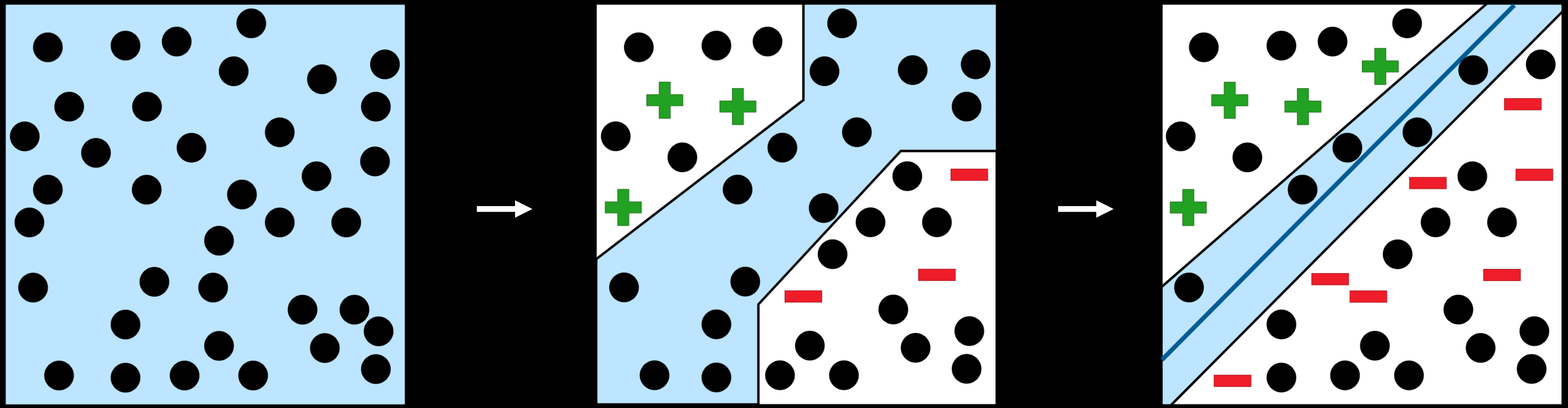
Requires true label at every round to make updates

Labels can be expensive to obtain

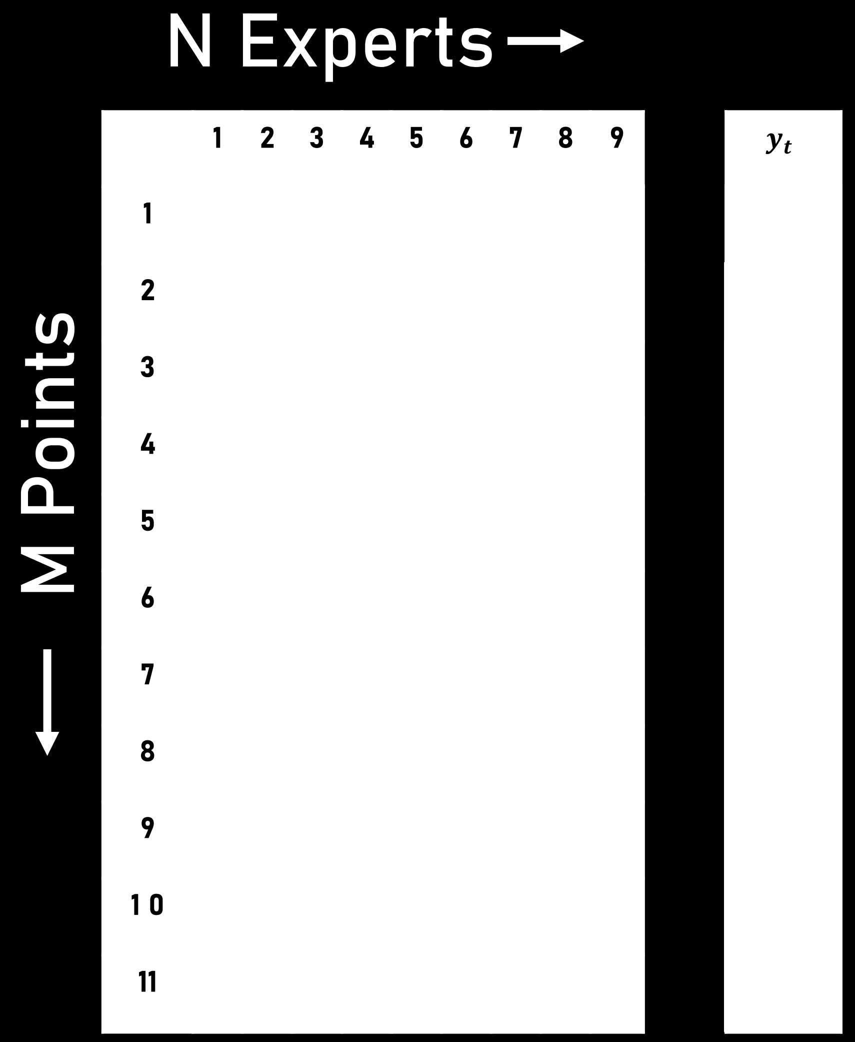
- Medical diagnoses that may require expensive tests
- Content moderation that requires humans in the loop
- Product testing that may require prolonged experiments

Active Learning

"Actively" decide the most informative points to request the label for



Active Online Learning



Active Online Learning

Expert advice matrix $X \in \{0,1\}^{M \times N}$ is available to the learner

1. Small burn-in phase: "Actively" move informative points ahead in the queue

N Experts →

M Points ↓

	1	2	3	4	5	6	7	8	9	y_t
1	1	0	0	1	1	1	1	1	0	
2	1	1	0	0	0	0	0	0	1	
3	0	0	0	1	1	0	1	0	1	
4	1	0	0	1	0	0	1	0	0	
5	1	0	1	1	1	1	1	1	0	
6	1	1	1	1	1	1	0	1	0	
7	0	0	1	0	1	1	1	0	1	
8	0	0	1	1	1	1	0	1	0	
9	0	0	1	0	1	1	0	0	1	
10	1	1	0	1	0	0	1	1	1	
11	0	1	1	0	1	1	0	0	1	

Active Online Learning

Expert advice matrix $X \in \{0,1\}^{M \times N}$ is available to the learner

1. Small burn-in phase: “Actively” move informative points ahead in the queue

N Experts →

M Points ↓

	1	2	3	4	5	6	7	8	9	y_t
1	1	0	0	1	1	1	1	1	0	
2	1	1	0	0	0	0	0	0	1	
3	0	0	0	1	1	0	1	0	1	
4	1	0	0	1	0	0	1	0	0	
5	1	0	1	1	1	1	1	1	0	
6	1	1	1	1	1	1	0	1	0	
7	0	0	1	0	1	1	1	0	1	
8	0	0	1	1	1	1	0	1	0	
9	0	0	1	0	1	1	0	0	1	
10	1	1	0	1	0	0	1	1	1	
11	0	1	1	0	1	1	0	0	1	

Active Online Learning

Expert advice matrix $X \in \{0,1\}^{M \times N}$ is available to the learner

1. Small burn-in phase: "Actively" move informative points ahead in the queue

N Experts →

M Points ↓

	1	2	3	4	5	6	7	8	9	y_t
5	1	0	1	1	1	1	1	1	0	
8	0	0	1	1	1	1	0	1	0	
11	0	1	1	0	1	1	0	0	1	
1	1	0	0	1	1	1	1	1	0	
2	1	1	0	0	0	0	0	0	1	
3	0	0	0	1	1	0	1	0	1	
4	1	0	0	1	0	0	1	0	0	
6	1	1	1	1	1	1	0	1	0	
7	0	0	1	0	1	1	1	0	1	
9	0	0	1	0	1	1	0	0	1	
10	1	1	0	1	0	0	1	1	1	

Active Online Learning

Expert advice matrix $X \in \{0,1\}^{M \times N}$ is available to the learner

1. Small burn-in phase: “Actively” move informative points ahead in the queue

N Experts →

M Points ↓

	1	2	3	4	5	6	7	8	9	y_t
5	1	0	1	1	1	1	1	1	0	1
8	0	0	1	1	1	1	0	1	0	1
11	0	1	1	0	1	1	0	0	1	0
1	1	0	0	1	1	1	1	1	0	
2	1	1	0	0	0	0	0	0	1	
3	0	0	0	1	1	0	1	0	1	
4	1	0	0	1	0	0	1	0	0	
6	1	1	1	1	1	1	0	1	0	
7	0	0	1	0	1	1	1	0	1	
9	0	0	1	0	1	1	0	0	1	
10	1	1	0	1	0	0	1	1	1	

Active Online Learning

Expert advice matrix $X \in \{0,1\}^{M \times N}$ is available to the learner

1. Small burn-in phase: "Actively" move informative points ahead in the queue
2. Sequentially go through the remaining points requesting labels for as few points as possible

N Experts →

M Points ↓

	1	2	3	4	5	6	7	8	9	y_t
5	1	0	1	1	1	1	1	1	0	1
8	0	0	1	1	1	1	0	1	0	1
11	0	1	1	0	1	1	0	0	1	0
1	1	0	0	1	1	1	1	1	0	1
2	1	1	0	0	0	0	0	0	1	
3	0	0	0	1	1	0	1	0	1	
4	1	0	0	1	0	0	1	0	0	0
6	1	1	1	1	1	1	0	1	0	
7	0	0	1	0	1	1	1	0	1	
9	0	0	1	0	1	1	0	0	1	1
10	1	1	0	1	0	0	1	1	1	

ζ Compactness of Prediction Matrix

Measures the "active learnability" of the prediction matrix X

	1	2	3	4	5
1	1	1	1	1	1
2	1	1	1	1	0
3	1	1	1	0	0
4	1	1	0	0	0
5	1	0	0	0	0

Actively learnable -> Small ζ

	1	2	3	4	5
1	0	0	0	0	1
2	0	0	0	1	0
3	0	0	1	0	0
4	0	1	0	0	0
5	1	0	0	0	0

Less actively learnable -> Large ζ

ζ Compactness of Prediction Matrix

Measures the "active learnability" of the prediction matrix X

$$\begin{array}{c|ccccc} & 1 & 2 & 3 & 4 & 5 \\ \hline 1 & 1 & 1 & 1 & 1 & 1 \\ 2 & 1 & 1 & 1 & 1 & 0 \\ 3 & 1 & 1 & 1 & 0 & 0 \\ 4 & 1 & 1 & 0 & 0 & 0 \\ 5 & 1 & 0 & 0 & 0 & 0 \end{array}$$

Actively learnable -> Small ζ

$$\begin{array}{c|ccccc} & 1 & 2 & 3 & 4 & 5 \\ \hline 1 & 0 & 0 & 0 & 0 & 1 \\ 2 & 0 & 0 & 0 & 1 & 0 \\ 3 & 0 & 0 & 1 & 0 & 0 \\ 4 & 0 & 1 & 0 & 0 & 0 \\ 5 & 1 & 0 & 0 & 0 & 0 \end{array}$$

Less actively learnable -> Large ζ

Usually, a small constant

ActiveHedge

1. Burn-in Phase:
   - Maintains a candidate set of experts V^{τ}

N Experts →

M Points ↓

	1	2	3	4	5	6	7	8	9
1	1	1	1	1	1	1	1	1	1
2	1	1	1	1	1	1	1	1	0
3	1	1	1	1	1	1	1	0	0
4	1	1	1	1	1	1	0	0	0
5	1	1	1	1	1	0	0	0	0
6	1	1	1	1	1	0	0	0	0
7	1	1	1	1	0	0	0	0	0
8	1	1	1	0	0	0	0	0	0
9	1	1	0	0	0	0	0	0	0
10	1	1	0	0	0	0	0	0	0
11	1	0	0	0	0	0	0	0	0

ActiveHedge

1. Burn-in Phase:
    - Maintains a candidate set of experts V^τ
    - Samples points where experts in V^τ disagree
    - Shrink V^τ

N Experts →

M Points ↓

	1	2	3	4	5	6	7	8	9	y_t
1	1	1	1	1	1	1	1	1	1	
2	1	1	1	1	1	1	1	1	0	
3	1	1	1	1	1	1	1	0	0	
4	1	1	1	1	1	1	0	0	0	
5	1	1	1	1	1	1	0	0	0	1
6	1	1	1	1	1	0	0	0	0	
7	1	1	1	1	0	0	0	0	0	0
8	1	1	1	0	0	0	0	0	0	0
9	1	1	0	0	0	0	0	0	0	
10	1	1	0	0	0	0	0	0	0	
11	1	0	0	0	0	0	0	0	0	

ActiveHedge

1. Burn-in Phase:
    - Maintains a candidate set of experts V^τ
    - Samples points where experts in V^τ disagree
    - Shrink V^τ
    - Repeat

N Experts →

M Points

	1	2	3	4	5	6	7	8	9	y_t
5	1	1	1	1	1	1	0	0	0	1
7	1	1	1	1	0	0	0	0	0	0
8	1	1	1	0	0	0	0	0	0	0
1	1	1	1	1	1	1	1	1	1	
2	1	1	1	1	1	1	1	1	0	
3	1	1	1	1	1	1	1	0	0	
4	1	1	1	1	1	1	0	0	0	
6	1	1	1	1	1	0	0	0	0	
9	1	1	0	0	0	0	0	0	0	
10	1	1	0	0	0	0	0	0	0	
11	1	0	0	0	0	0	0	0	0	

ActiveHedge

1. Burn-in Phase:
    - Maintains a candidate set of experts V^τ
    - Samples points where experts in V^τ disagree
    - Shrink V^τ
    - Repeat
2. Sequentially label the remaining points
    - Query label only if confused

N Experts →

M Points ↓

	1	2	3	4	5	6	7	8	9	y_t
5	1	1	1	1	1	1	0	0	0	1
7	1	1	1	1	0	0	0	0	0	0
8	1	1	1	0	0	0	0	0	0	0
1	1	1	1	1	1	1	1	1	1	
2	1	1	1	1	1	1	1	1	0	
3	1	1	1	1	1	1	1	0	0	
4	1	1	1	1	1	1	0	0	0	
6	1	1	1	1	1	0	0	0	0	0
9	1	1	0	0	0	0	0	0	0	
10	1	1	0	0	0	0	0	0	0	
11	1	0	0	0	0	0	0	0	0	

ActiveHedge: Guarantees

Theorem: Given a ζ compact advice matrix X, L^* s.t. $\min_{j\in[n]} \sum_{t=1}^{M} \ell(X_{t,j}, y_t) \leq L^*$, with probability at least $1 - \rho$,

1. ActiveHedge queries at most $\tilde{O}(\zeta L^*)$ lables
2. The burn in phase is only $\tilde{O}(\zeta)$ long
3. $REG_{ActiveHedge} \leq \sqrt{2L^* \ln N} + \ln N$

ActiveHedge: Guarantees

Theorem: Given a ζ compact advice matrix X, L^* s.t. $\min_{j \in [n]} \sum_{t=1}^{M} \ell(X_{t,j}, y_t) \leq L^*$, with probability at least $1 - \rho$,

1. ActiveHedge queries at most $\tilde{O}(\zeta L^*)$ lables
2. The burn in phase is only $\tilde{O}(\zeta)$ long
3. $REG_{ActiveHedge} \leq \sqrt{2L^* \ln N} + \ln N$

Same Regret as Hedge!

ActiveHedge: Guarantees

Theorem: Given a ζ compact advice matrix X, L^* s.t. $\min_{j\in[n]} \sum_{t=1}^{M} \ell(X_{t,j}, y_t) \leq L^*$, with probability at least $1 - \rho$,

1. ActiveHedge queries at most $\tilde{O}(\zeta L^*)$ lables
2. The burn in phase is only $\tilde{O}(\zeta)$ long
3. $REG_{ActiveHedge} \leq \sqrt{2L^* \ln N} + \ln N$

Same Regret as Hedge!

Using much fewer samples

$\tilde{O}(\zeta L^*)$ vs M

Conclusions

- We introduce the active online learning setting
- ActiveHedge can
  1. Obtain the same regret guarantee as Hedge
  2. Request much fewer labels

Thanks!