TÉCNICO LISBOA

Planning, Learning and Decision Making

Lecture 21. Exploration vs. exploitation

Exploration vs exploitation

- You have 3 machines in the casino

Machine 1

Machine 2

Machine 3

TÉCNICO LISBOA

Exploration vs exploitation

- You play each machine once
  - Machine 1: You gain 10$
  - Machine 2: You gain 2$
  - Machine 3: You lose 1$

- Which machine should you play next?

Exploration vs exploitation

- You play each machine **twice**
  - Machine 1: You gain 10$, 2$
  - Machine 2: You gain 2$, 4$
  - Machine 3: You lose 1$, win 15$

- Which machine should you play next?

Exploration vs exploitation

- Pure exploration vs exploitation problem
  - When to explore (try new machines)?
  - When to stop exploring and start exploiting (play the apparently best machine)?

Multi-armed bandit

One-armed bandit

Multi-armed bandit

Multi-armed bandit

Multi-armed bandit

- Sequential decision problem
- Game between agent and “nature”
- At each time step t:
  - Agent selects an action
  - Nature selects cost function
  - Agent gets the cost for its action

Multi-armed bandit

- How can we play this game?

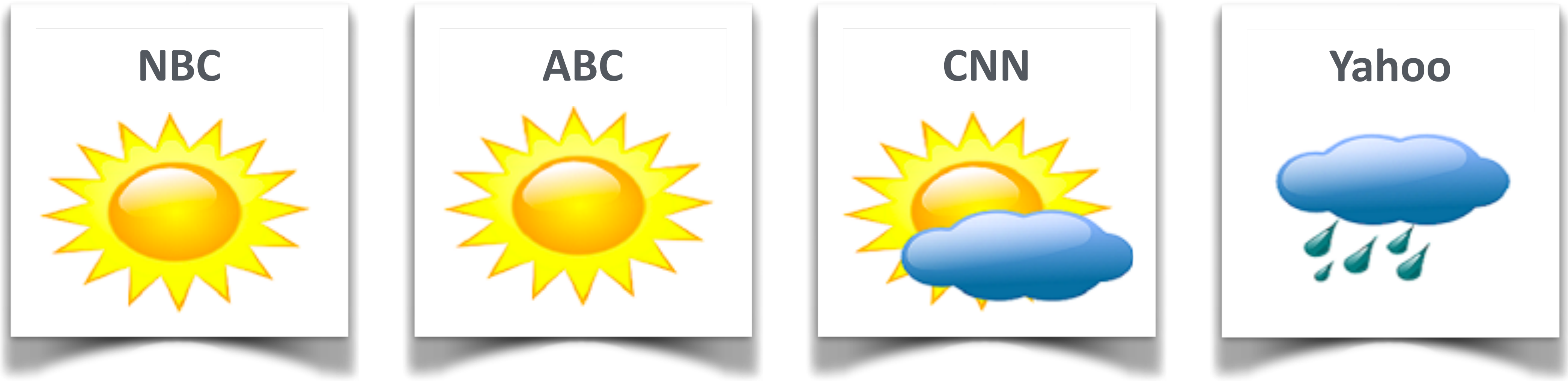
... hard!

Sequential prediction

- Let’s play a simpler game

Example

- You are a weather forecaster
- You have access to forecasts from different sources



Which source should you follow?

Example

- Suppose that you known that one source is always right
- How would you do this?

Follow the majority vote!

Example

- Possibility 1:
  - You get the prediction right
  - The cost is 0!


HELL YEAH!

Example

- Possibility 2:
  - You get the prediction wrong
  - You can eliminate half of your sources

Example

- What is the maximum number of mistakes?
  - Number of sources: N
  - Maximum number of (valid) sources after M mistakes:

$$\frac{N}{2^M}$$

  - There is always at least one valid source (always right)

$$\frac{N}{2^M} \geq 1$$

  - Maximum number of mistakes:

$$M \leq \log_2(N)$$

Nice!

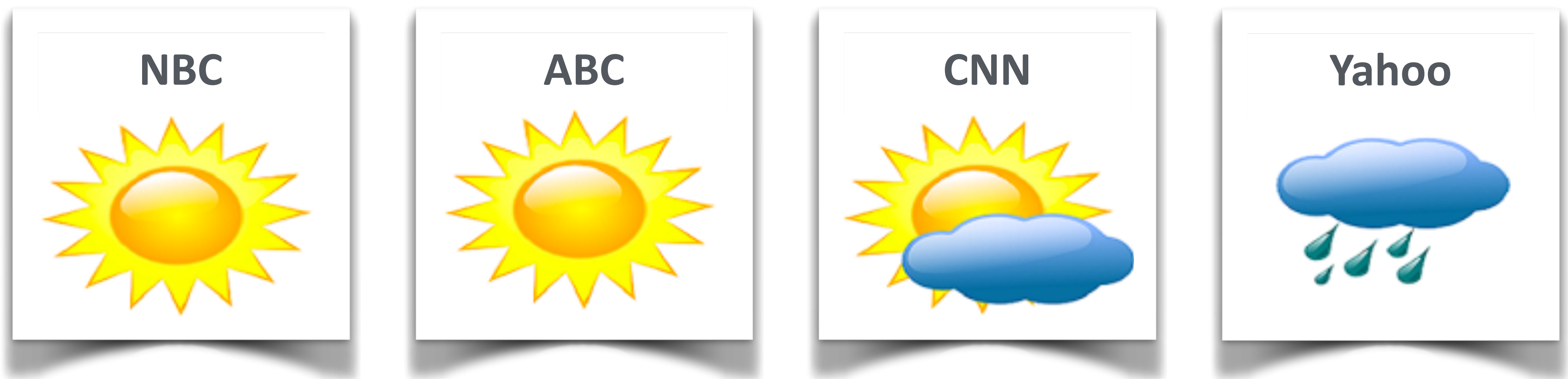
Example

- Even if you have exponentially many sources, you can still manage

But what if no source is always right?

Example

- Define a "confidence-level" for your sources
- Follow the majority vote



Confidence: 1 Confidence: 1 Confidence: 1 Confidence: 1

Example

- Possibility 1:
  - You get the prediction right
  - The cost is 0!

• • •

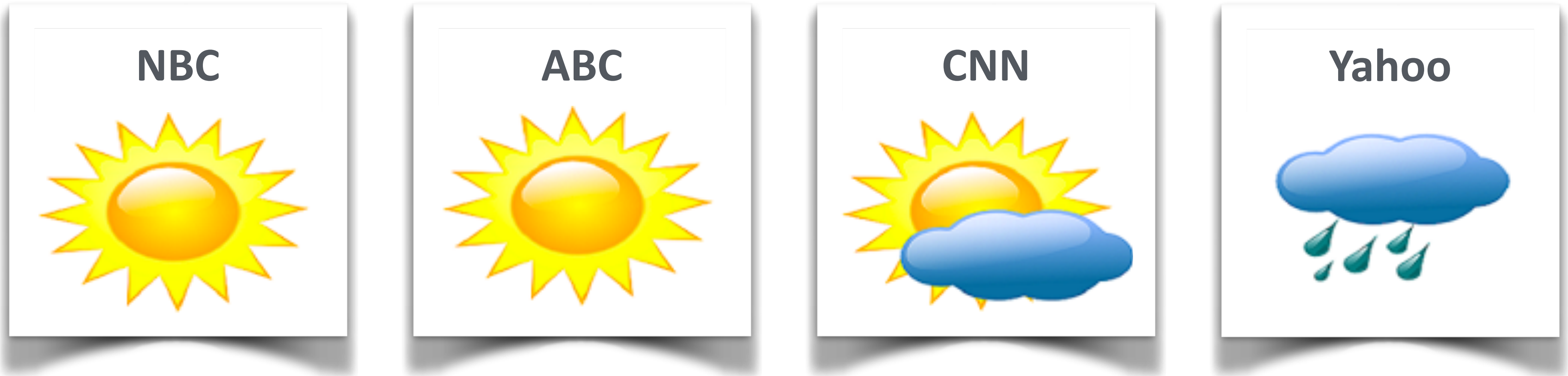

HELL YEAH!
memegenerator.net

Example

- Possibility 2:
  - You get the prediction wrong
  - You can no longer eliminate half of the sources

Example

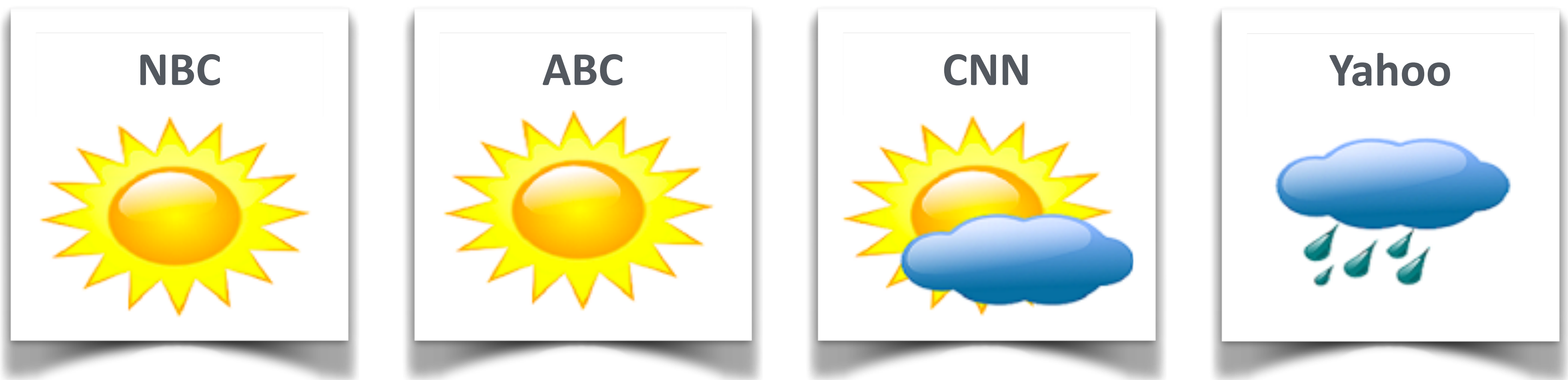
- … but you can decrease your confidence in those that failed



Confidence: 1 Confidence: 1 Confidence: 1 Confidence: 1

Example

- … but you can decrease your confidence in those that failed



Confidence: 1 Confidence: 1 Confidence: 0.8 Confidence: 0.8

Example

- Total confidence before mistake:

$$W_t = \sum_{n=1}^{N} w_t(n)$$

- After a mistake, at least 1/2 of the sources decrease by a factor of $(1 - \eta)$, with $\eta < 1/2$ so

$$W_{t+1} \leq \underbrace{\left(\frac{1}{2} + \frac{1}{2}(1 - \eta)\right)}_{<1} W_t = \left(1 - \frac{\eta}{2}\right) W_t$$

Example

- How many mistakes?
  - Number of sources: *N*
  - Total confidence after *M* mistakes:

$$N\left(1-\frac{\eta}{2}\right)^M$$

  - If the best source made *m* mistakes, then

$$N\left(1-\frac{\eta}{2}\right)^M \geq (1-\eta)^m$$

Logarithmic in number of sources

  - Maximum number of mistakes:

$$M \leq 2(1+\eta)m + \frac{2\log N}{\eta}$$

Important aspects

- We measure our performance compared against that of the best "guess"
- Usually, performance of the best guess can only be assessed *a posteriori*

Summarizing…

Weighted majority algorithm:

- Given a set of N "predictors" and $\eta < 1/2$
- Initialize predictor weights to $w_0(n) = 1$, $n = 1, \ldots, N$
- Make prediction based on the (weighted) majority vote
- Update weights of all wrong predictors as

$$w_{t+1}(n) = w_t(n)(1 - \eta)$$

Sequential prediction

- Let's consider a slightly more complex game
- At each time step *t*:
  - Agent selects an action
  - Nature selects cost function
  - Nature discloses cost function
- We make no assumptions on how nature selects cost function

Sequential prediction

- Use a similar principle:
  - Define a “confidence-level” for each action

Action 1 **Action 2** **Action 3** **Action 4**

Confidence: 1 Confidence: 1 Confidence: 1 Confidence: 1

Sequential prediction

- Select each action "proportionally" to its confidence:

$$p_t(a) = \frac{w_t(a)}{\sum_{a' \in \mathcal{A}} w_t(a')}$$

Sequential prediction

- When cost is revealed, we update each "confidence" according to the corresponding cost:

Cost: 0.1

Confidence: 1

Cost: 0.3

Confidence: 1

Cost: 1

Confidence: 1

Cost: 0.7

Confidence: 1

Sequential prediction

- When cost is revealed, we update each "confidence" according to the corresponding cost:

Cost: 0.1 **Cost: 0.3** **Cost: 1** **Cost: 0.7**

Confidence: 0.9 Confidence: 0.7 Confidence: 0.4 Confidence: 0.5

Sequential prediction

- When cost is revealed, we update each "confidence" according to the corresponding cost:

$$w_{t+1}(a) = w_t(a)e^{-\eta c_t(a)}$$

Cost

Sequential prediction

- When cost is revealed, we update each "confidence" according to the corresponding cost:

$$w_{t+1}(a) = w_t(a)\boxed{e^{-\eta c_t(a)}}$$

<1

- Then, at each step *t*,

$$w_t(a) = e^{-\eta \sum_{\tau=0}^{t-1} c_\tau(a)}$$

Total cost

Sequential prediction

- To measure the performance, we compare our total (expected) cost with that of the best action (in hindsight):

$$R_T = \mathbb{E}\left[\sum_{t=0}^{T-1} c_t(a_t)\right] - \min_{a \in \mathcal{A}} \sum_{t=0}^{T-1} c_t(a)$$

- The value R_T is called the **regret** at time T
  - It measures how much the agent regrets, in hindsight, not having following the minimizing action

Sequential prediction

- How much regret?
  - Initial weights: N
  - Total confidence after T steps smaller than:

$$Ne^{-\eta\mathbb{E}\left[\sum_{t=0}^{T-1} c_t(a_t)\right]_\rho}$$

Constant that depends on η and N

Sequential prediction

- How much regret?
  - Initial weights: *N*
  - Total confidence after *T* steps smaller than:

$$Ne^{-\eta\mathbb{E}\left[\sum_{t=0}^{T-1} c_t(a_t)\right]\rho}$$

  - Comparing with the best action (in hindsight):

$$Ne^{-\eta\mathbb{E}\left[\sum_{t=0}^{T-1} c_t(a_t)\right]\rho} \geq \min_{a\in\mathcal{A}} e^{-\eta\sum_{t=0}^{T-1} c_t(a)}$$

  - Finally,

$$R_t \leq \frac{\log N}{\eta} + \frac{\rho}{\eta}$$

Sequential prediction

- Selecting η properly, we get the final bound:

$$R_T \leq \sqrt{\frac{T}{2} \log N}$$

Exponentially Weighted Averager (EWA)

- This algorithm for sequential prediction is called **exponentially weighted averager**
  - It makes no assumptions on the process by which costs are selected (can be adversarial)
  - Depends logarithmically on the number of actions (works well even if there is an exponentially large number of actions to try)
  - Its regret is **sublinear in *T***

No-regret prediction

- What does it mean that the regret is sublinear in T?
- Recall that, for the EWA,

$$R_T \leq \sqrt{\frac{T}{2} \log N}$$

Grows with
$\sqrt{T}$
(slower than T)

No-regret prediction

- What does it mean that the regret is sublinear in *T*?
- Recall that, for the EWA,

$$R_T \leq \sqrt{\frac{T}{2} \log N}$$

- If we compute the average regret per step:

$$\frac{R_T}{T} \leq \sqrt{\frac{\log N}{2T}} \xrightarrow[T \to \infty]{} 0$$

No regret algorithm

Summarizing

Exponentially weighted averager:

- Given a set of N actions and $\eta > 0$
- Initialize weights to $w_0(a) = 1$, $a \in \mathscr{A}$
- Select an action according to the probabilities

$$p_t(a) = \frac{w_t(a)}{\sum_{a' \in \mathcal{A}} w_t(a')}$$

- Update weights of all actions as

$$w_{t+1}(a) = w_t(a) e^{-\eta c_t(a)}$$