ML – Reinforcement Learning

CPSC 383: Explorations in Artificial Intelligence and Machine Learning
Fall 2025

Jonathan Hudson, Ph.D
Associate Professor (Teaching)
Department of Computer Science
University of Calgary

August 27, 2025

UNIVERSITY OF CALGARY

Categories

In **reinforcement learning**, the goal is to learn a mapping of inputs to outputs that maximizes the earned “reward”.

1. Observe state
2. Select action
3. Receive a reward based on state and action
4. Transition probabilistically to a new state based on previous state and action

Example

As a student, you are an example of reinforcement learning!

So are:

- Generative AI models like GPT and DALL-E
- Many AI models that play games (Starcraft, GO, etc.)
- Robotics
- … and many more

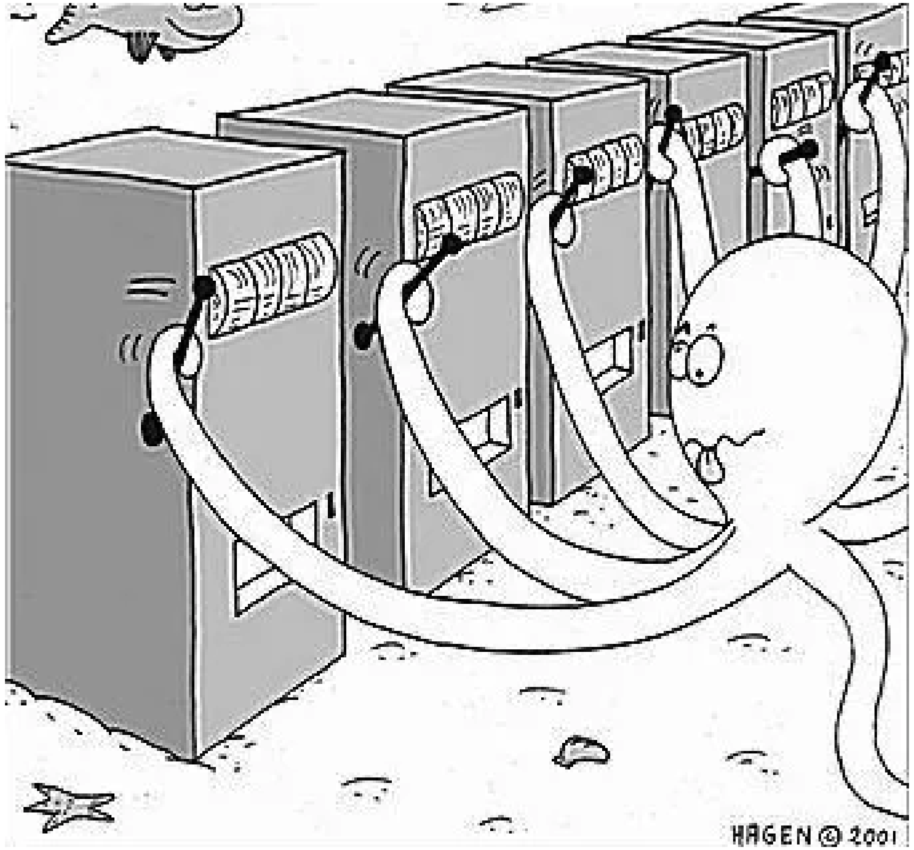
Multi-armed Bandit

Armed Bandits

- Slot machines -> one-armed bandits
- The arm you pull to spin the wheel is the one-arm
- The bandit being that they are ‘rigged’ so that the house always wins more than the user population in the long-tern
- Multi-armed bandit means multiple slot machines
- Multi-armed Bandit **Problem** is trying to figure out (while just playing) the slot machines which has the best odds

Multi-armed Bandit Problem

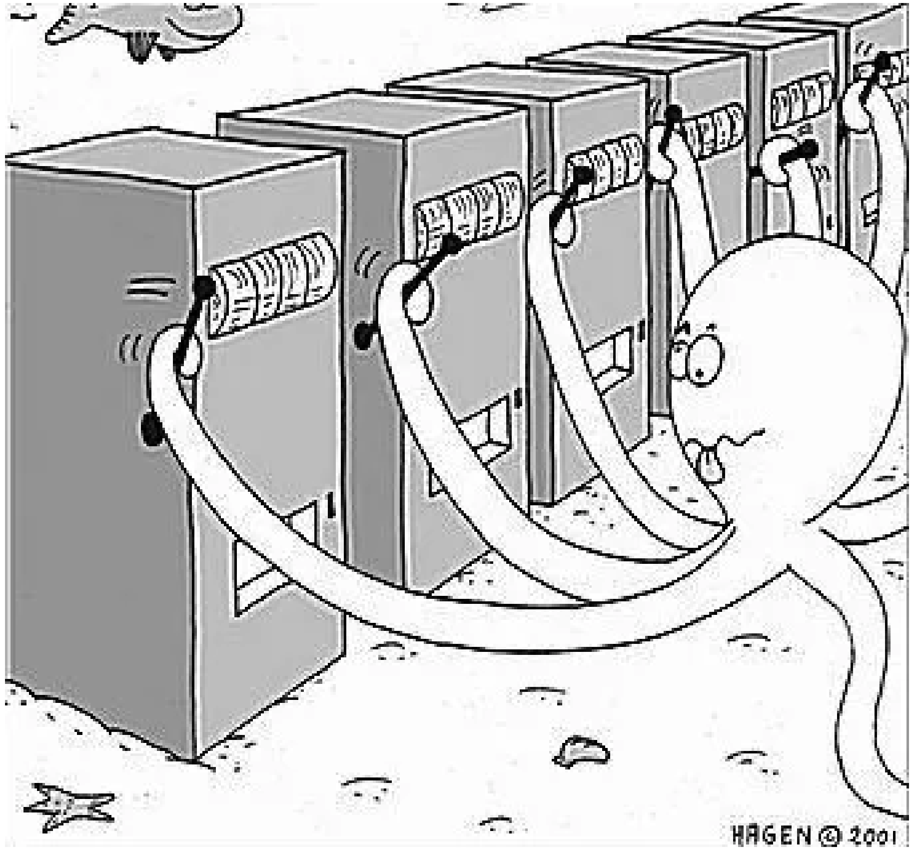
- This natural a problem with a structure made for re-inforcement learning
- You have an original plan that you use to choose which slot machine to play first
- The data you gain from that is used to adjust your plan for the next time you pick a slot machine to play



UNIVERSITY OF CALGARY

Important Assumptions

- Fundamental part of problem is requirement that a choice of an arm does not affect any slot machines payoff odds
- This is not true to the real-world where many slot machines like to show users a building payoff amount and promise better odds over time to overcome risk aversion
- Restless Bandit Problem allows for the odds of a played machine to change (using Markov state evolution of probabilities)



UNIVERSITY OF CALGARY

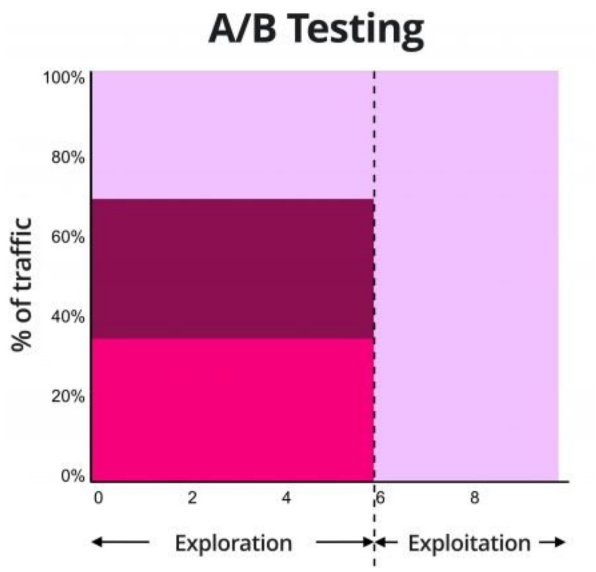
Exploitation vs Exploration

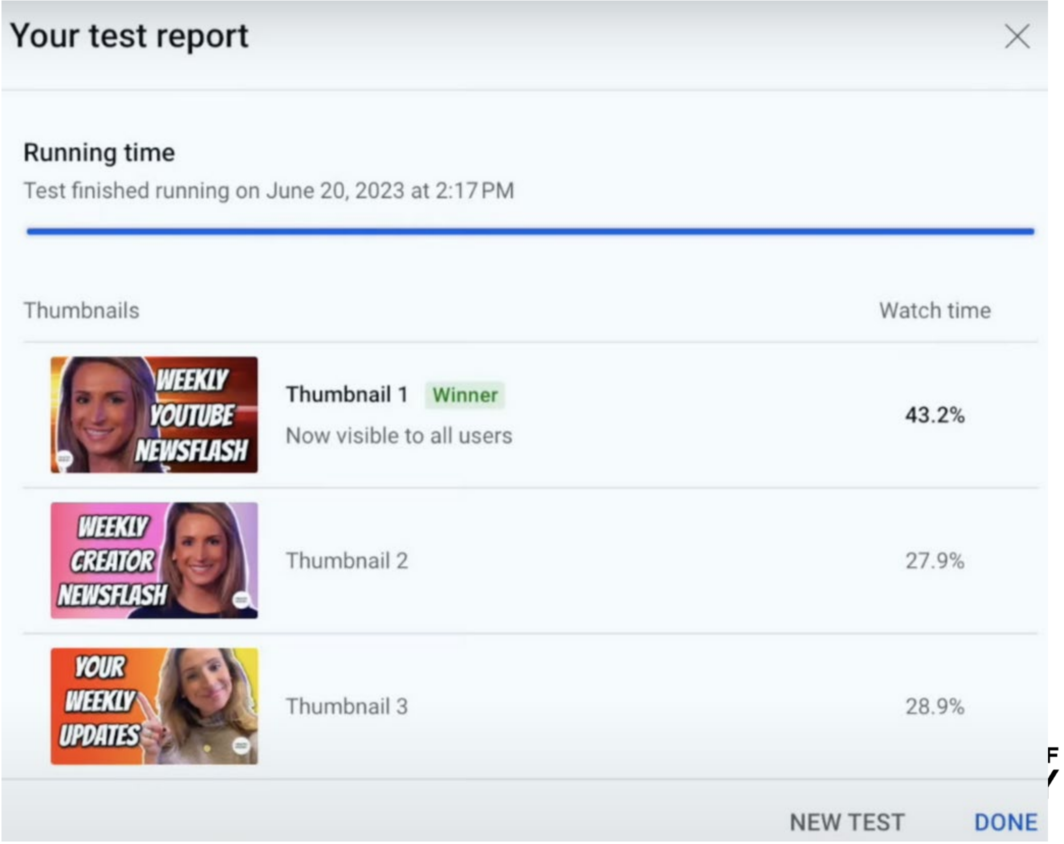
- Popular intro to reinforcement learning as it’s a great example of
- **Exploration vs exploitation trade-off**
  - Should a play a new machine that I want more knowledge about
  - Or should I play the one I currently think has the best odds
- Most version you begin with no knowledge of machine odds
- Goal is to pick one machine at some point
- All that is known is each machine has its own payoff probability distribution

Usages

Non gambling scenarios

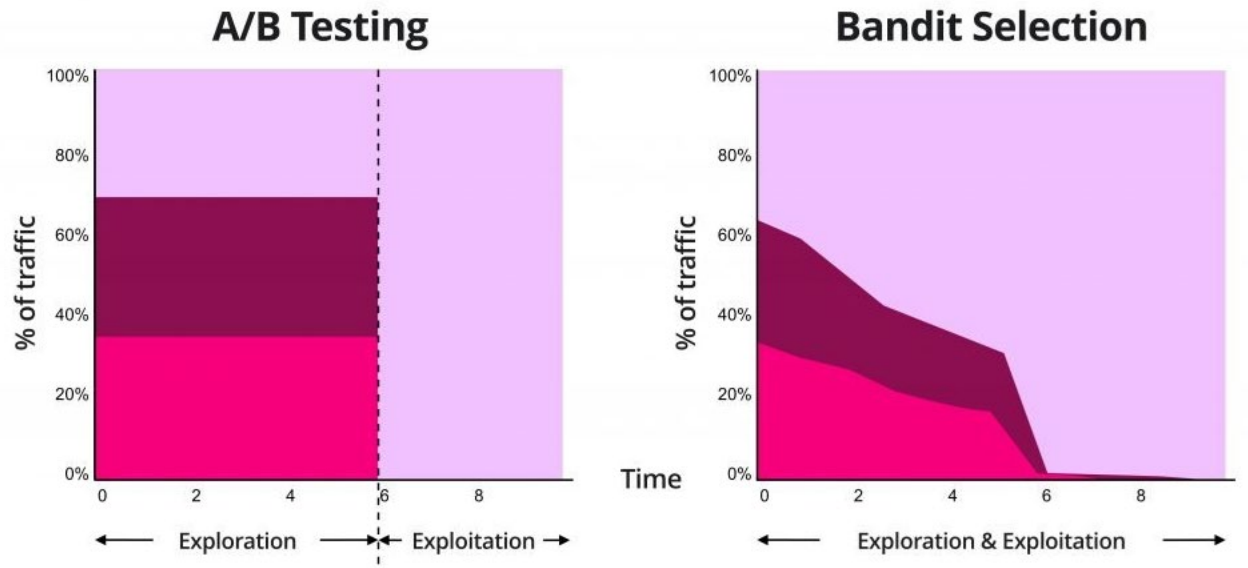
- Often when making design choices in ads, websites, content, companies use A/B testing where users are served variants of content, then response/engagement/etc. rates are reviewed and one of the variants chosen
- Youtube lets you do this with thumbnails for videos
- Bandit selection is a service offered by advertising companies, and they have us code up back-ends so that non-tech users can use it





Non gambling scenarios

- Bandit selection allows you to deploy many choices as well, but over-time as users engage the most successful design will be served most often
- The benefit of this, is when you are running an ad campaign you have limited time to make money
- You make most of it right away some times as you can only buy ad space in prominent places for short time
- Bandit selection allows you to start profiting earlier on your best design

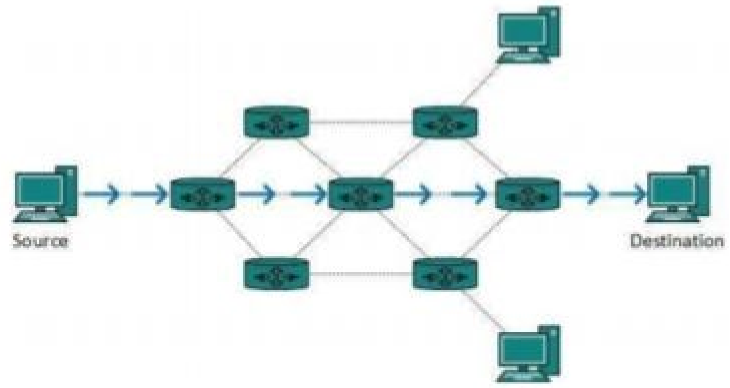


UNIVERSITY OF CALGARY

Non gambling scenarios

- You run a research or pharma. company with many projects
  - Each project has odds of being successful that you don't fully know
  - Should you continue to fund currently running projects
  - Or start other projects that are proposed with less known about them
- Often these variants include how much money should be invested in each, instead of just one usage like the slot machine scenario

Non gambling scenarios



- Network routing
  - Pick which next hop to propagate data through without perfect knowledge of which edges are fastest, or robust to data loss

- Game designing
  - Positively: Game testing different design choices
  - Negatively: Making exploitative game that tries to get you to pay more based on your personality type
    - Addictive behaviour exploitation
      - Live!
    - Like hosts at casinos or online gambling apps that identify you and exploit you to gamble more because you are a ‘big loser’

Solutions

Best Arm Identification

- Pick one arm of the k you start with
- Two main solution types
- Optimal solutions
  - You can ‘do the math’ and essentially define how long it would take to use a slot machine to determine its distribution (its mean payoff rate) with certain confidence you require
  - You can then ‘do the math’ with two machines to decide to choose the one with the most uncertain mean each time, and then at a certain point you can decide you are confident enough your sample means for each are different and pick the higher one
  - This can be extrapolated for 3, 4, etc. machines
- Approximate solutions
  - Optimal solutions are both limiting in requiring determinate time which may be excessive
  - That time may be unnecessary as you are interested in something that is ‘good enough’
    - Sometimes its easier to collect many data points before changing a strategy (serving webpages)
    - Sometimes you don’t need optimal answer, just actionable one that isn’t bad

Best Arm Identification

- Approximate solutions
  - Semi-uniform Strategies
    - Use a numerical controls, often ϵ epsilon
    - Epsilon-greedy
      - Select best lever for $1-\epsilon$ % of trials, and random otherwise, like 90% exploit, 10% explore
    - Epsilon-first
      - Phases, for N trials, then explore $N\epsilon$ in a row, then exploit $(1-N)\epsilon$ pulling best lever after
    - Epsilon-decreasing
      - Start with high exploration (high epsilon) and decay it overtime to exploit
    - Other adaptive methods
  - Semi-uniform requires picking one or more ‘magic numbers’
    - Epsilon, N, decay rates
    - Heuristics an average user may not understand

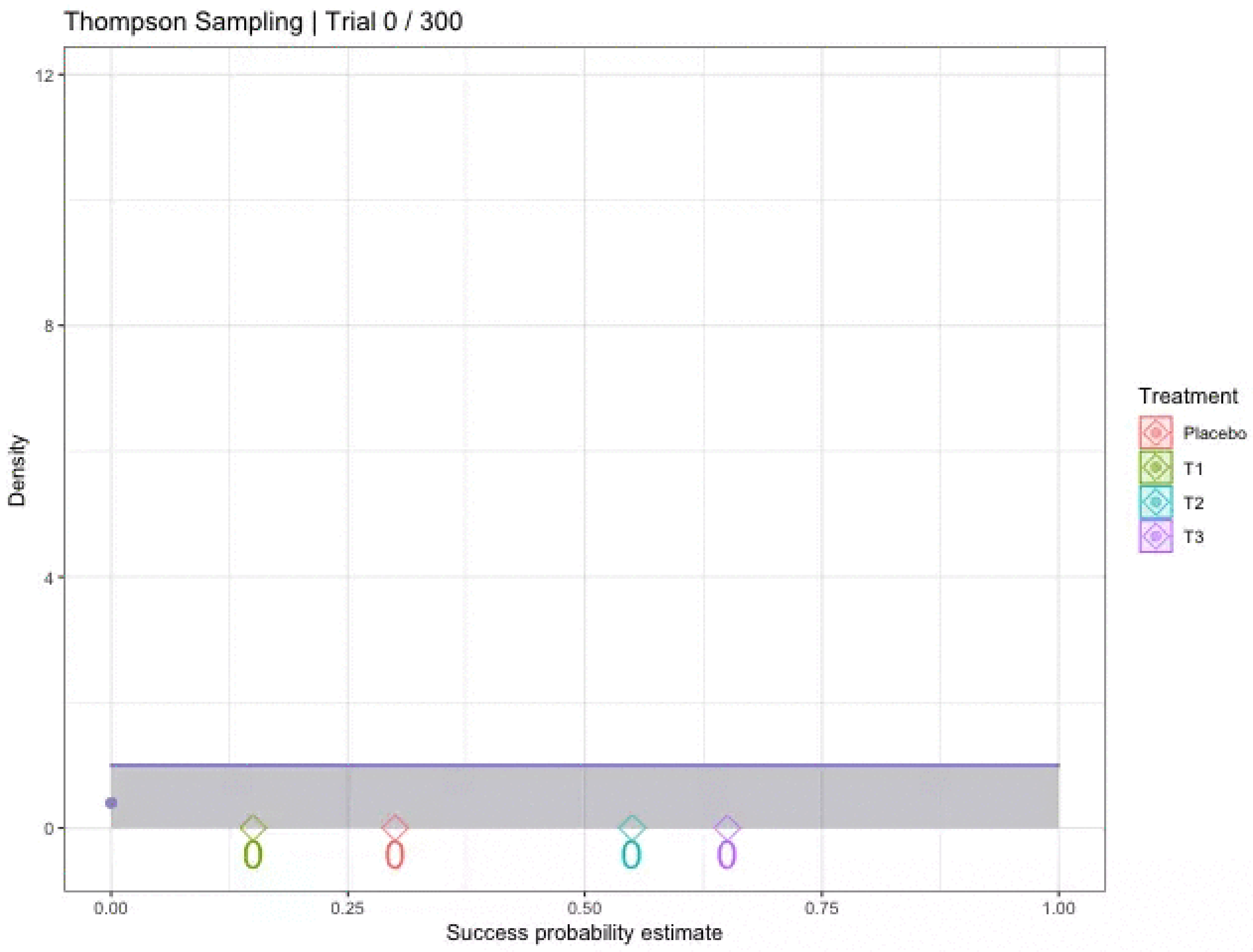
UNIVERSITY OF CALGARY

Best Arm Identification

- Other approximate solutions
  - Pricing Strategies
    - Turn choice into market problem by establishing cost for lever which is sum of expected award to pull it plus estimation of ‘reward of knowledge gained’
    - Pulling a never used lever could have a high value just from knowledge even if winning payoff is unknown
    - Requires some good math to make sure the balance of exploit and explore are merged in balanced way

Best Arm Identification

- Other approximate solutions
  - Probability Matching
    - Rather popular partially because they often can be done with user heuristics
    - Thompson Sampling best known
    - Engage with a lever in accordance with its currently known distribution of payoff
    - Popular in A/B replacement online as it scales well
      - Can collect many samples before you update the statistics that change the choices
      - Works well with web-caching and delivery limitations of real sized websites



Solutions

Sci Kit?

- No directly
- One option is bayesianbandits
- Built to integrate with scikit-learn and scipy
- https://github.com/bayesianbandits/bayesianbandits
- Example in code notebook

UNIVERSITY OF CALGARY

Next…neural networks

Jonathan Hudson, Ph.D.
jwhudson@ucalgary.ca
https://cspages.ucalgary.ca/~jwhudson/

UNIVERSITY OF CALGARY