Wisdom of Crowds versus Groupthink: learning in groups and isolation

Presented by: Joel Maupin

Mayo-Wilson, Conor, Kevin Zollman, and David Danks. "Wisdom Of Crowds Versus Groupthink: Learning In Groups And In Isolation." International Journal Of Game Theory 42.3 (2013): 695-723.

Overview

- Compare learning ability of networked agents to isolated agents
- Multi-armed bandit problem
- 6 different approaches
  - Simulated Annealing
  - Epsilon Greedy
  - Delta-Epsilon
  - Reinforcement learning
  - Weighted Reinforcement learning
  - Upper Confidence Bound
- **Hypothesis: No significant difference between networked and isolated agents in a general case.**

Agenda

1. Introduction to Multi-armed bandit problem
2. Overview of the 6 methods
3. Summarize the experiment
4. Evaluate the results
5. Discuss future work
6. Review and Conclude

Multi-armed Bandits

- Colloquial slot machine name: “one-armed bandit”
- Agent pulls one “arm” and receives reward
- Reward is independent of other arms
  - Typically also random
- Regret defined as (possible reward – actual reward)
- Agent’s goal is to maximize reward and minimize regret
  - Achieve by alternating exploration and exploitation
- Agents in paper seek to find optimal reward
- Let’s look at an example

Rosin, Christopher. "Multi-Armed Bandits With Episode Context." Annals Of Mathematics & Artificial Intelligence 61.3 (2011): 203-230.

Multi-armed Bandit example

	Arm 1	Arm 2	Arm 3
First Pass	Reward = 1	Reward = 1	Reward = 0
Second pass	Reward = 1	Reward = 2	Reward = 0
Best choice?	✗	✓	✗
Actual reward	P(1) = 1.0	P(1) = 0.5 P(2) = 0.5	P(0) = 0.9 P(1000) = 0.1
Actual best	✗	✗	✓

Bounded rationality

- Limitation on decision making ability
- Make the best decision based on information available in the time allotted [Jones]
- Large subset of Computing problems
  - Example, Multi-armed bandit problem
  - All information => trivial solution

Jones, Bryan. “Bounded Rationality” Northwestern University Press. (1999) Accessed via web on November 5, 2014.

Agenda

1. Introduction to Multi-armed bandit problems
2. **Overview of each of 6 methods**
   1. Epsilon Greedy
   2. Sigma Epsilon
   3. Reinforcement Learning
   4. Weighted Reinforcement Learning
   5. Simulated Annealing
   6. Upper Confidence Bound
3. Summarize the experiment
4. Evaluate the results
5. Discuss future work
6. Review and Conclude

Epsilon Greedy

- Pick a best action most of the time and explore remainder
- P(a = best) : $1 - \varepsilon$
- P(a = random) : ε
- "Best" is action with highest expected reward
- Pro:
  - Focus on exploitation
  - Also guaranteed to search
- Con:
  - Very slow exploration compared to other methods
  - Continues to explore after finding true optimal

Delta Epsilon

- Uses a set of “favorite arms”
- Choose best with $1 - (\partial + \varepsilon)$ probability
  - $P(a = \text{best}) : 1 - (\partial + \varepsilon)$
  - $P(a = \text{random}) : \varepsilon$
  - $P(a = \text{favorite}) : \partial$
- Pro:
  - Same as Epsilon Greedy
  - Better approximates human decision making
- Con:
  - Set of favorites may not be optimal
    - Could perform very bad

Reinforcement Learning

- Agent seeks to maximize cumulative reward
- Choose based on ratio of reward
  - P(a) = R_a / R_t
  - R_a is the expected reward for action a.
  - R_t is the total reward of all actions
- Pro:
  - Great exploration of search space
- Con:
  - Very slow when all rewards are similar
  - Will still explore even when the “best” is found
    - Optimal arm just has highest selection probability

Weighted Reinforcement Learning

- Same as Reinforcement learning with a weighted function
- P(a) = $f\,(\,R_a\,)\,/\,f\,(\,R_{\mathrm{t}}\,)$
  - Where f () is a function that increases with time
  - Weights recent actions more strongly than distant actions
- Pro:
  - Over time, probabilities greatly favor optimal actions
- Con:
  - Depending on weights, might exploit too fast
    - Agent would choose non optimal action
  - Difficult to determine the exact weighting factor

Simulated Annealing

- Similar to hill climbing with multiple restarts
  - Over time “cool” the problem so we widen our search space
  - $P(r_{cur}, r_{next}, T)$
    - r_{cur} is the reward of the current arm.
    - r_{next} is the expected reward of the next arm.
    - T is the temperature of the system
- Pro:
  - Settles on a single answer
- Con:
  - Temperature is the dampening factor
    - Too high and system will never exploit
    - Too low and system will find local (not global) maximum

Hasani, Keramat, Svetlana A. Kravchenko, and Frank Werner.
"Simulated Annealing And Genetic Algorithms For The Two-Machine Scheduling Problem With A Single Server."
International Journal Of Production Research 52.13 (2014): 3778-3792.

Upper Confidence Bound

- Also known as Confidence Interval
- Statistically measure how well the agent is doing.
- Over time the confidence increases
  - Statistical certainty of level of performance.
- Pro:
  - Minimize regret after each “play”
    - Each “play” involves a sample from all arms
  - Self describing level of fitness
    - Useful for mathematical comparisons
- Con:
  - Requires a huge number of samples for each round
    - Difficult to break down search space

Mitchell, Tom. *Machine Learning*. McGraw-Hill Publications. March 1 (1997)

Agenda

1. Introduction to Multi-armed bandit problems
2. Overview of each of 6 methods
3. **Summarize the experiment**
4. Evaluate the results
5. Discuss future work
6. Review and Conclude

Math Notations

- Our learning problem is defined as { Ω, *A* , *O*, *p*}
  - Ω is the finite set of states of the world
  - A is a finite set of actions
  - O is a finite set of non-negative outcomes (rewards)
  - p is the probability of obtaining a particular award based on an action

$$p = \{p(* \mid a, \omega)\}_{a \in A, \omega \in \Omega}$$

Math Notations

- A Strategic Network is defined as $S = \{G, M\}$
  - G is the network (group) of agents
  - M is the method used by the agents
- Focus on networks S applied to learning problems that are:
  - Non-trivial
    - Same history of actions could lead to different states
  - Difficult
    - No action guaranteed to succeed
    - Little to no certainty of optimality

Experiment

- Mathematically state each learning method
- Evaluate networked and isolated agents
  - Comparing each method to themselves
  - Show any performance difference
- Hypothesis: No significant difference between networked and isolated agents in a general case.

RL vs RL

UCB vs UCB

wRL vs wRL

εG vs εG

∂ε vs ∂ε

SA vs SA

Experiment

- How to define convergence in bounded rational problem?
  - Define individual metrics
    - Isolation Consistency (*IC*)
      - Agent converges to optimal while in isolation
    - Universal Consistency (*UC*)
      - Agent converges to optimal while in an arbitrary network
        - Could be similar or foreign agents
  - Define the same but for groups!
    - Group Isolation Consistency (*GIC*)
      - One agent uses the strategy *M* and all agents converge
    - Group Universal Consistency (*GUC*)
      - All agents use strategy *M* and all converge

Individual Consistency

- Some implementations satisfy for all 6 methods
  - Does not need to be all!
  - We are looking for the existence of consistency in the methods
    - Not necessarily the robustness
- Consider the case of ε Greedy.
  - Small values of ε will converge
  - Large values will explore too often and shift the average away from the optimal return

Universal Consistency

- UC methods will always be IC
  - Just separate the connections in the grouping
- Not all IC methods are UC
- Consider ε Greedy strategies again
  - Experiment with rate $\frac{1}{n^{x/y}}$
    - x = number of actions observed
    - y = number of actions performed
  - In isolation that is $\frac{1}{n}$
  - In networks that becomes $\frac{1}{n^2}$
    - Too small to ensure optimality
- ε Greedy is IC but not UC

Group Consistencies

- Similar idea to Individual and Universal Consistency
- If Network is GUC then any single agent is GIC
- Not always true for the reverse
  - Consider ∂ε with sub-optimal favorite actions
    - Group may do very well but individual agents will not
- In general, GUC is to GIC as UC is to IC

Isolated vs Networked

- "Rational network of irrational individuals"
  - Paradox state
  - At least one IC individual in network
    - Broadcasts optimal choice to all others
    - Network settles on optimal action
- "Irrational network of rational individuals"
  - Conflicting information between individuals
  - Same problem we saw with ε Greedy and Universal Consistency

Agenda

1. Introduction to Multi-armed bandit problems
2. Overview of each of 6 methods
3. Summarize the experiment
4. **Evaluate the results**
5. Discuss future work
6. Review and Conclude

Results

- Several of the methods lack Universal Consistency
  - ε and RL methods
  - This means they also lack Group Universal Consistency
- There are exceptions to the general case
  - Contextual information is hugely important
- Independent Consistency and Group consistency need not coincide
- Learning methods all have strengths and weaknesses like anything else
  - Here some are better for isolation while others favor the performance of groups
- Group Learning is not necessarily more accurate!

The winner is: UCB

- None of the methods hold up against all criteria
  - UCB meets both IC and UC
    - Less promising in GIC and GUC
- Seems to hold up to math models the best
- The devil is in the details
  - Implementations vary and there may be more factors
  - Would need to study other applications on a case by case basis

Agenda

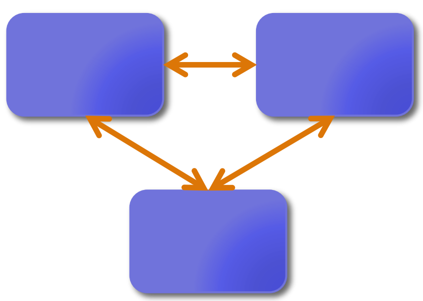
1. Introduction to Multi-armed bandit problems
2. Overview of each of 6 methods
3. Summarize the experiment
4. Evaluate the results
5. **Discuss future work**
6. Review and Conclude

Future Work and Opportunities

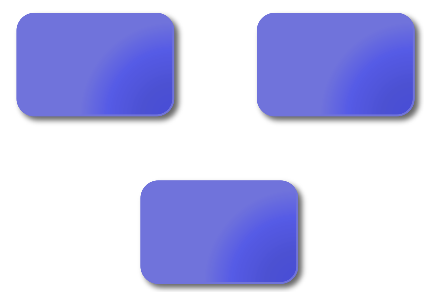
- Implementation
  - Can we be successful even without proof
    - Heuristics could be something we use for performance
- Incorporate real world data and testing
  - Theory is helpful but real data has new set of challenges
    - Quality of data and ability of methods to filter bad data
- Confidence Intervals look very promising
  - Implementations vary
- Expand analysis to other problem domains
  - Plenty of other Bounded Rational problems to study
  - Including the adversarial Bandit problem

Challenges

- Upper Confidence Bound misleading performance
  - Each "round" requires a sample of all arms
  - A UCB "round" is much more computationally complex than εG
    - Intuitively we would expect a greater knowledge gain
- Downplay of unconnected groups
  - Focus on connected only



Test consistency of Groups



Test individual variance?

Agenda

1. Introduction to Multi-armed bandit problems
2. Overview of each of 6 methods
3. Summarize the experiment
4. Evaluate the results
5. Discuss future work
6. **Review and Conclude**

Review

- 6 Different Learning methods
- Compared across 4 criteria
- None stand out as a general solution
    - Authors choose Upper Confidence Bound as a top performer in the general case
- Need to test with real world data to verify model
- Expand analysis to more domains

Conclusion

- Takeaways
  - Information transfer is difficult to accomplish in general form
  - Careful implementation details would bypass many difficulties
    - Dynamic programming
    - Use of heuristics
- Real world disconnect
  - There exists optimal policy
  - Finite number of actions to that policy
  - Can’t write that policy without exploring all actions and states
    - Still limited by physical hardware for complex problems.

References

- Mayo-Wilson, Conor, Kevin Zollman, and David Danks. "Wisdom Of Crowds Versus Groupthink: Learning In Groups And In Isolation." International Journal Of Game Theory 42.3 (2013): 695-723.
- Rosin, Christopher. "Multi-Armed Bandits With Episode Context." Annals Of Mathematics & Artificial Intelligence 61.3 (2011): 203-230.
- Jones, Bryan. “Bounded Rationality” Northwestern University Press. (1999) Accessed via web on November 5, 2014.
- Hasani, Keramat, Svetlana A. Kravchenko, and Frank Werner. "Simulated Annealing And Genetic Algorithms For The Two-Machine Scheduling Problem With A Single Server.” International Journal Of Production Research 52.13 (2014): 3778-3792.
- Mitchell, Tom. *Machine Learning*. McGraw-Hill Publications. March 1 (1997)

Thank you

Questions?