Teaching Teammates in Ad Hoc Teams

Peter Stone
Director, Learning Agents Research Group
Department of Computer Sciences
The University of Texas at Austin

Joint work with
Gal A. Kaminka, **Sarit Kraus**, Bar Ilan University
Jeffrey S. Rosenschein, Hebrew University

Research Question

To what degree can autonomous intelligent agents learn in the presence of teammates and/or adversaries in real-time, dynamic domains?

Research Question

To what degree can autonomous intelligent agents learn in the presence of teammates and/or adversaries in real-time, dynamic domains?

- Autonomous agents
- Robotics
- Machine learning (RL)
- Multiagent systems

Research Question

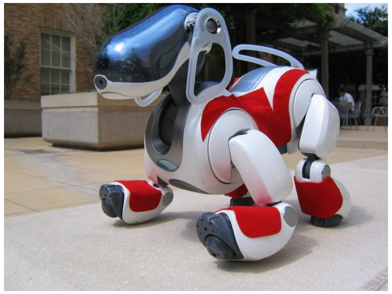
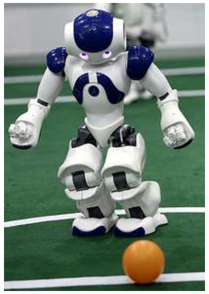
To what degree can autonomous intelligent agents learn in the presence of teammates and/or adversaries in real-time, dynamic domains?

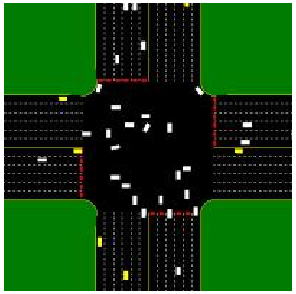
- Autonomous agents
- Robotics
- Machine learning (RL)
- Multiagent systems
  - e-commerce
  - mechanism design

Research Question

To what degree can autonomous intelligent agents learn in the presence of teammates and/or adversaries in real-time, dynamic domains?

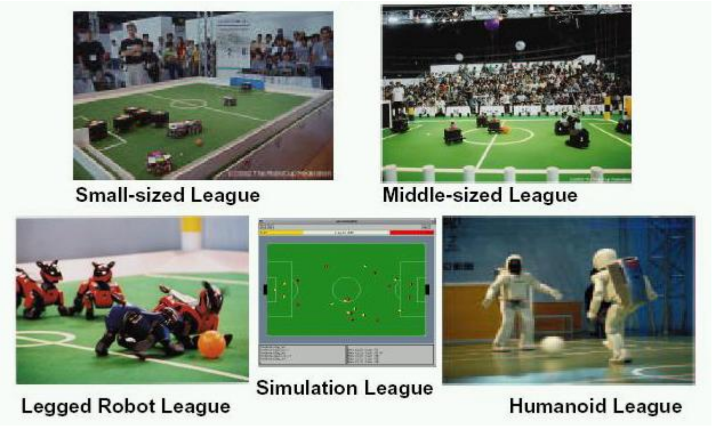
- Autonomous agents
- Robotics
- Machine learning (RL)
- Multiagent systems
  - e-commerce
  - mechanism design

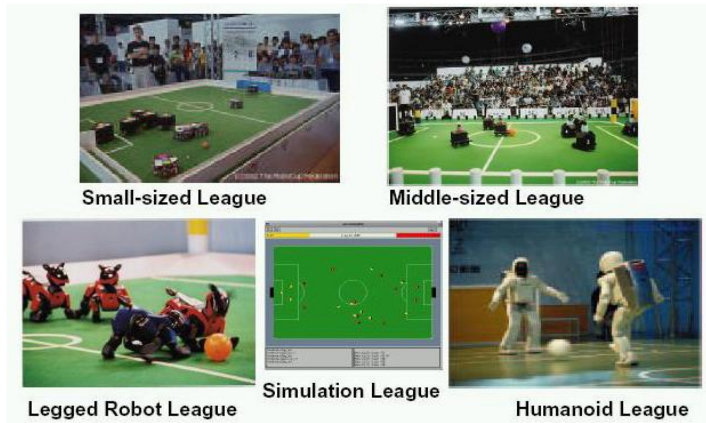


Teamwork

Teamwork


Small-sized League
Middle-sized League
Legged Robot League
Simulation League
Humanoid League

Teamwork



- Typical scenario: pre-coordination
  - People practice together
  - Robots given coordination languages, protocols
  - "Locker room agreement" (Stone & Veloso, '99)

Ad Hoc Teams

- Ad hoc team player is an individual
  - Unknown teammates (programmed by others)

Ad Hoc Teams

- Ad hoc team player is an individual
  – Unknown teammates (programmed by others)
- May or may not be able to communicate

Ad Hoc Teams

- Ad hoc team player is an individual
  - Unknown teammates (programmed by others)
- May or may not be able to communicate
- Teammates likely sub-optimal: no control

Ad Hoc Teams

- Ad hoc team player is an individual
  - Unknown teammates (programmed by others)
- May or may not be able to communicate
- Teammates likely sub-optimal: no control

Ad Hoc Teams

- Ad hoc team player is an individual
  - Unknown teammates (programmed by others)
- May or may not be able to communicate
- Teammates likely sub-optimal: no control

Goal: Create a good team player

Ad Hoc Teams

- Ad hoc team player is an individual
  - Unknown teammates (programmed by others)
- May or may not be able to communicate
- Teammates likely sub-optimal: no control

Goal: Create a good team player

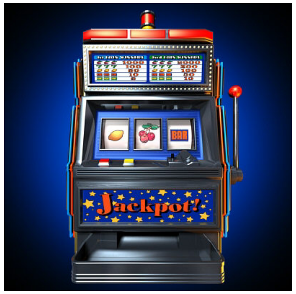
- Minimal representative scenarios
  - One teammate, no communication
  - Fixed and known behavior

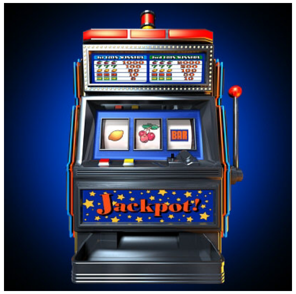
Scenarios

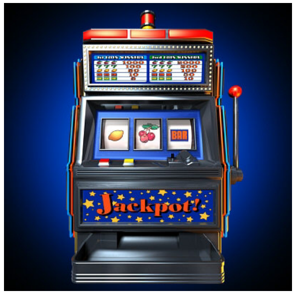
- Cooperative normal form game (w/ Kaminka & Rosenschein)

$M1$	b_0	b_1	b_2
a_0	25	1	0
a_1	10	30	10
a_2	0	33	40

- Cooperative k-armed bandit (w/ Kraus)







Scenarios

- Cooperative normal form game (w/ Kaminka & Rosenschein)

$M1$	b_0	b_1	b_2
a_0	25	1	0
a_1	10	30	10
a_2	0	33	40

- Cooperative k-armed bandit (w/ Kraus)

Formalism

- *Agent A* in our control: actions $a_0, a_1, \ldots a_{x-1}$
- *Agent B* reacts in a fixed way: $b_0, b_1, \ldots, b_{y-1}$

Formalism

- *Agent A* in our control: actions $a_0, a_1, \ldots a_{x-1}$
- *Agent B* reacts in a fixed way: $b_0, b_1, \ldots, b_{y-1}$
- **Game theory**: normal form, fully cooperative

$M1$	b_0	b_1	b_2
a_0	25	1	0
a_1	10	30	10
a_2	0	33	**40**

- Payoff from joint action (a_i, b_j): $m_{i,j}$

Formalism

- *Agent A* in our control: actions $a_0, a_1, \ldots a_{x-1}$
- *Agent B* reacts in a fixed way: $b_0, b_1, \ldots, b_{y-1}$
- **Game theory**: normal form, fully cooperative

$M1$	b_0	b_1	b_2
a_0	25	1	0
a_1	10	30	10
a_2	0	33	**40**

- Payoff from joint action (a_i, b_j): $m_{i,j}$
- Highest payoff m^* always at (a_{x-1}, b_{y-1})
- *Agent B*'s default action: b_0

Objective

- *Agent A*'s goal: action sequence with highest reward
  - Undiscounted, medium-term (finite)
  - Depends on *Agent B*'s strategy

Objective

- *Agent A*'s goal: action *sequence* with highest reward
  - Undiscounted, medium-term (finite)
  - Depends on *Agent B*'s strategy

***Agent B* not adaptive** $\Longrightarrow a_0$ **always**

$M1$	b_0	b_1	b_2
a_0	**25**	1	0
a_1	10	30	10
a_2	0	33	40

Objective

- *Agent A*'s goal: action sequence with highest reward
  - Undiscounted, medium-term (finite)
  - Depends on *Agent B*'s strategy

Agent B **not adaptive** $\Longrightarrow a_0$ **always**

$M1$	b_0	b_1	b_2
a_0	**25**	1	0
a_1	10	30	10
a_2	0	33	40

- Reward sequence: 25, 25, 25, . . .

Objective

- *Agent A*'s goal: action sequence with highest reward
  - Undiscounted, medium-term (finite)
  - Depends on *Agent B*'s strategy

***Agent B* best response $\Longrightarrow$ can do better**

$M1$	b_0	b_1	b_2
a_0	**25**	1	0
a_1	10	30	10
a_2	0	33	40

- Reward sequence: 25, 25, 25, . . .
- Reward sequence: 25

Objective

- *Agent A*'s goal: action sequence with highest reward
  - Undiscounted, medium-term (finite)
  - Depends on *Agent B*'s strategy

***Agent B* best response $\Longrightarrow$ can do better**

$M1$	b_0	b_1	b_2
a_0	25	1	0
a_1	10	30	10
a_2	**0**	33	40

- Reward sequence: 25, 25, 25, . . .
- Reward sequence: 25, 0

Objective

- *Agent A*'s goal: action sequence with highest reward
  - Undiscounted, medium-term (finite)
  - Depends on *Agent B*'s strategy

***Agent B* best response $\Longrightarrow$ can do better**

$M1$	b_0	b_1	b_2
a_0	25	1	0
a_1	10	30	10
a_2	0	33	**40**

- Reward sequence: 25, 25, 25, . . .
- Reward sequence: 25, 0, 40

Objective

- *Agent A*'s goal: action sequence with highest reward
  - Undiscounted, medium-term (finite)
  - Depends on *Agent B*'s strategy

***Agent B* best response $\Longrightarrow$ can do better**

$M1$	b_0	b_1	b_2
a_0	25	1	0
a_1	10	30	10
a_2	0	33	**40**

- Reward sequence: 25, 25, 25, . . .
- Reward sequence: 25, 0, 40, 40, 40, . . .

Objective

- *Agent A*'s goal: action sequence with highest reward
  - Undiscounted, medium-term (finite)
  - Depends on *Agent B*'s strategy

Agent B best response $\Longrightarrow$ or even better

$M1$	b_0	b_1	b_2
a_0	**25**	1	0
a_1	10	30	10
a_2	0	33	40

- Reward sequence: 25, 25, 25, . . .
- Reward sequence: 25, 0, 40, 40, 40, . . .
- How?

Objective

- *Agent A*'s goal: action sequence with highest reward
  - Undiscounted, medium-term (finite)
  - Depends on *Agent B*'s strategy

***Agent B* best response $\Longrightarrow$ or even better**

$M1$	b_0	b_1	b_2
a_0	25	1	0
a_1	**10**	30	10
a_2	0	33	40

- Reward sequence: 25, 25, 25, . . .
- Reward sequence: 25, 0, 40, 40, 40, . . .
- Reward sequence: 25, 10

Objective

- *Agent A*'s goal: action sequence with highest reward
  - Undiscounted, medium-term (finite)
  - Depends on *Agent B*'s strategy

***Agent B* best response $\Longrightarrow$ or even better**

$M1$	b_0	b_1	b_2
a_0	25	1	0
a_1	10	30	10
a_2	0	**33**	40

- Reward sequence: 25, 25, 25, . . .
- Reward sequence: 25, 0, 40, 40, 40, . . .
- Reward sequence: 25, 10, 33

Objective

- *Agent A*'s goal: action sequence with highest reward
  - Undiscounted, medium-term (finite)
  - Depends on *Agent B*'s strategy

Agent B best response $\Longrightarrow$ or even better

$M1$	b_0	b_1	b_2
a_0	25	1	0
a_1	10	30	10
a_2	0	33	**40**

- Reward sequence: 25, 25, 25, . . .
- Reward sequence: 25, 0, 40, 40, 40, . . .
- Reward sequence: 25, 10, 33, 40

Objective

- *Agent A*'s goal: action sequence with highest reward
  - Undiscounted, medium-term (finite)
  - Depends on *Agent B*'s strategy

***Agent B* best response $\Longrightarrow$ or even better**

$M1$	b_0	b_1	b_2
a_0	25	1	0
a_1	10	30	10
a_2	0	33	**40**

- Reward sequence: 25, 25, 25, . . .
- Reward sequence: 25, 0, 40, 40, 40, . . .
- Reward sequence: 25, 10, 33, 40, 40, 40, . . .

Objective

- *Agent A*'s goal: action sequence with highest reward
  - Undiscounted, medium-term (finite)
  - Depends on *Agent B*'s strategy

***Agent B* best response $\Longrightarrow$ or even better**

$M1$	b_0	b_1	b_2
a_0	25	1	0
a_1	10	30	10
a_2	0	33	**40**

- Reward sequence: 25, 25, 25, . . .
- Reward sequence: 25, 0, 40, 40,. . . (65 from 1st 3)
- Reward sequence: 25, 10, 33, 40, . . . (68 from 1st 3)

Objective

- *Agent A*'s goal: action sequence with highest reward
  - Undiscounted, medium-term (finite)
  - Depends on *Agent B*'s strategy

***Agent B* best response $\Longrightarrow$ or even better**

$M1$	b_0	b_1	b_2
a_0	25	1	0
a_1	10	30	10
a_2	0	33	**40**

- Reward sequence: 25, 25, 25, . . .
- Reward sequence: 25, 0, 40, 40,. . . **Cost:** 15+40=**55**
- Reward sequence: 25, 10, 33, 40, . . . **Cost:** 15+30+7=**52**

Objective

- *Agent A*'s goal: action sequence with highest reward
  - Undiscounted, medium-term (finite)
  - Depends on *Agent B*'s strategy

***Agent B* best response $\Longrightarrow$ or even better**

$M1$	b_0	b_1	b_2
a_0	25	1	0
a_1	10	30	10
a_2	0	33	**40**

- Reward sequence: 25, 25, 25, . . .
- Reward sequence: 25, 0, 40, 40,. . . Cost: 55, **Length: 2**
- Reward sequence: 25, 10, 33, 40, . . . Cost: 52, **Length: 3**

Assumptions

1. *Agent B*: bounded-memory BR, ϵ-greedy action strategy

Assumptions

1. *Agent B*: bounded-memory BR, ϵ-greedy action strategy
   - *mem*: memory size
   - ϵ: degree of randomness

Assumptions

1. *Agent B*: bounded-memory BR, ϵ-greedy action strategy
   - *mem*: memory size
   - ϵ: degree of randomness
2. *Agent A* knows *Agent B*'s type

Assumptions

1. *Agent B*: bounded-memory BR, ϵ-greedy action strategy
   - *mem*: memory size
   - ϵ: degree of randomness

2. *Agent A* knows *Agent B*'s type

$M1$	b_0	b_1	b_2
a_0	25	1	0
a_1	10	30	10
a_2	0	33	40

- Example: *mem*$= 4, \epsilon = 0.1$
  - *Agent A* previous actions: a_1, a_0, a_1, a_1

Assumptions

1. *Agent B*: bounded-memory BR, ϵ-greedy action strategy.
   – *mem*: memory size
   – ϵ: degree of randomness

2. *Agent A* knows *Agent B*'s type

$M1$	b_0	b_1	b_2
a_0	25	1	0
$\mathbf{a_1}$	**10**	**30**	**10**
a_2	0	33	40

- Example: *mem*$= 4, \epsilon = 0.1$
  – *Agent A* previous actions: a_1, a_0, a_1, a_1
  – *Agent B*: *A* will select a_0 (prob. 0.25) or a_1 (0.75)

Assumptions

1. *Agent B*: bounded-memory BR, ϵ-greedy action strategy.
   - *mem*: memory size
   - ϵ: degree of randomness
2. *Agent A* knows *Agent B*'s type

$M1$	b_0	b_1	b_2
a_0	25	1	0
$\mathbf{a_1}$	**10**	**30**	**10**
a_2	0	33	40

- Example: *mem* $= 4$, $\epsilon = 0.1$
  - *Agent A* previous actions: a_1, a_0, a_1, a_1
  - *Agent B*: *A* will select a_0 (prob. 0.25) or a_1 (0.75)
  - *BR*$(a_1, a_0, a_1, a_1) = b_1$

Assumptions

1. *Agent B*: bounded-memory BR, ϵ-greedy action strategy.
   - *mem*: memory size
   - ϵ: degree of randomness

$M1$	b_0	b_1	b_2
a_0	25	1	0
$\mathbf{a_1}$	**10**	**30**	**10**
a_2	0	33	40

2. *Agent A* knows *Agent B*'s type

- Example: *mem*$= 4$, $\epsilon = 0.1$
  - *Agent A* previous actions: a_1, a_0, a_1, a_1
  - *Agent B*: *A* will select a_0 (prob. 0.25) or a_1 (0.75)
  - *BR*$(a_1, a_0, a_1, a_1) = b_1$
  - *Agent B*: selects b_1 (1-ϵ) or uniformly random (ϵ)

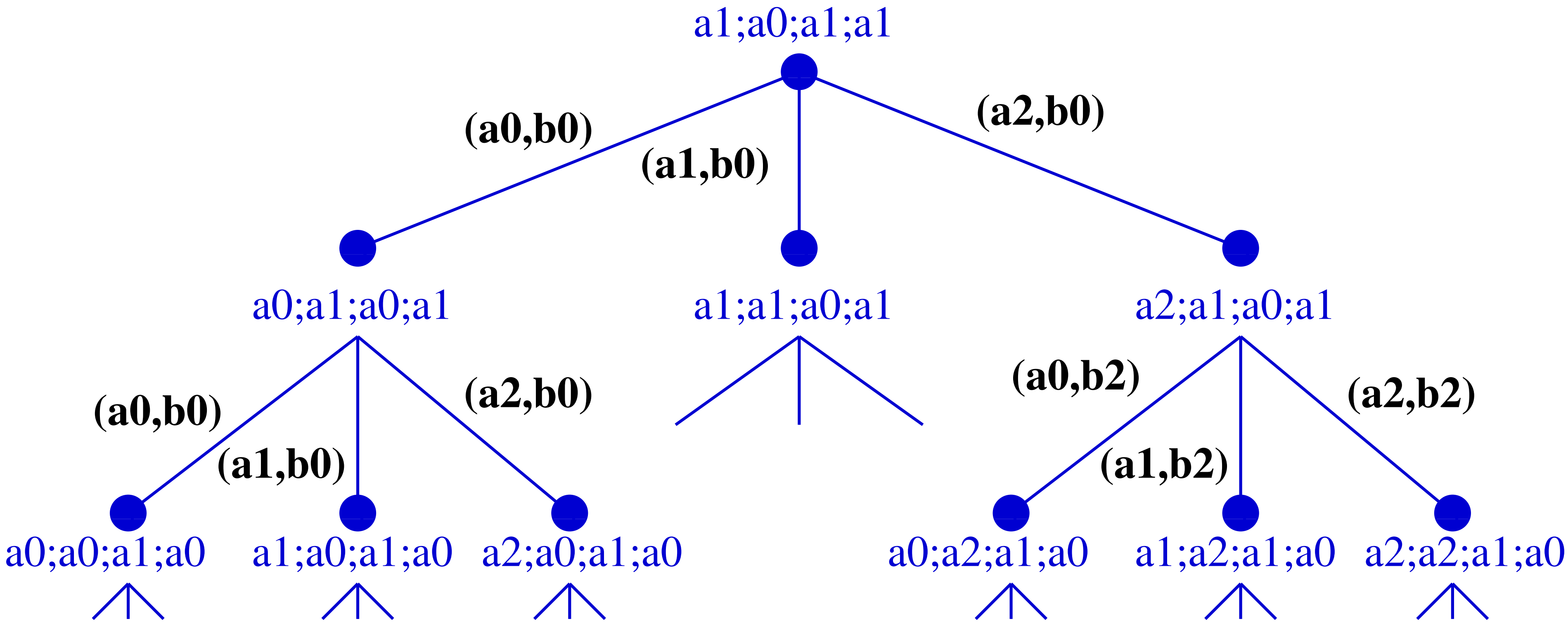
Assumptions

1. *Agent B*: bounded-memory BR, ϵ-greedy action strategy.
   - *mem*: memory size
   - ϵ: degree of randomness
2. *Agent A* knows *Agent B*'s type

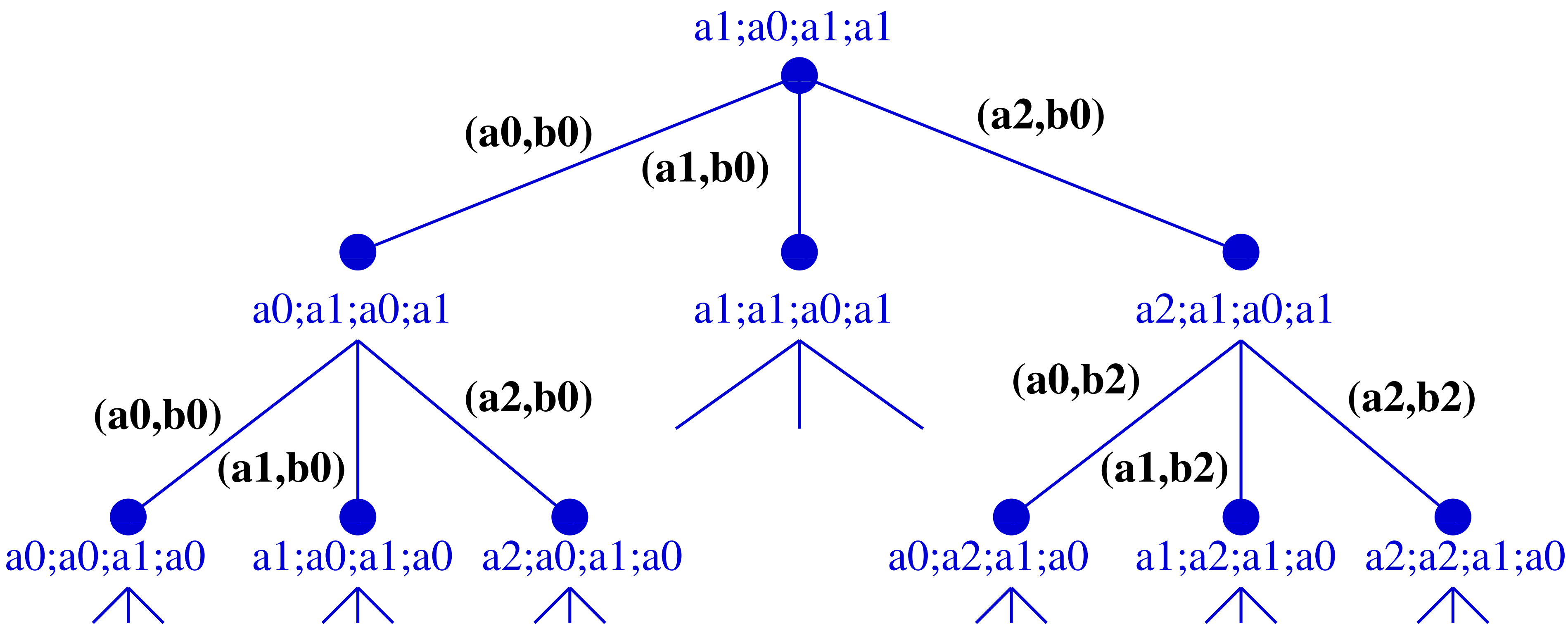
$M1$	b_0	b_1	b_2
a_0	25	1	0
$\mathbf{a_1}$	**10**	**30**	**10**
a_2	0	33	40

- Example: *mem*$= 4$, $\epsilon = 0.1$
  - *Agent A* previous actions: a_1, a_0, a_1, a_1
  - *Agent B*: *A* will select a_0 (prob. 0.25) or a_1 (0.75)
  - *BR*$(a_1, a_0, a_1, a_1) = b_1$
  - *Agent B*: selects b_1 (1-ϵ) or uniformly random (ϵ)
  - *Agent A*: action detemines payoff and next history

Extensive Form Version



Extensive Form Version



Stick with iterated normal form for presentation, algorithms

Questions

- Can we find the optimal action sequence efficiently?
- How long can the optimal action sequences be?

Cases

- Deterministic teammate, 1-step memory ($mem = 1, \epsilon = 0$)
- Longer teammate memory ($mem > 1, \epsilon = 0$)
- Teammate non-determinism ($mem > 1, \epsilon > 0$)

Dynamic Programming Algorithm

- Define $S_n^*(a_i, b_j)$ = optimal sequence of length n
- Define $S_0^*(a_i, b_j)$ to be cost 0 if $m_{i,j} = m_*$, else ∞

$S_0^*(a_2, b_2)$ Cost 0

$M1$	b_0	b_1	b_2
a_0	25	1	0
a_1	10	30	10
a_2	0	33	**40**

Dynamic Programming Algorithm

- Define $S_n^*(a_i, b_j)$ = optimal sequence of length n
- Define $S_0^*(a_i, b_j)$ to be cost 0 if $m_{i,j} = m_*$, else ∞

$S_0^*(a_2, b_2)$	Cost 0
$S_2^*(a_0, b_0)$	Cost 15+40
$S_2^*(a_1, b_0)$	Cost 30+7
$S_2^*(a_2, b_0)$	Cost 40

$M1$	b_0	b_1	b_2
a_0	25	1	0
a_1	10	30	10
a_2	0	33	40

Dynamic Programming Algorithm

- Define $S_n^*(a_i, b_j)$ = optimal sequence of length n
- Define $S_0^*(a_i, b_j)$ to be cost 0 if $m_{i,j} = m_*$, else ∞
- Find $S_n^*(a_i, b_j)$ using S_{n-1}^*'s (O(d), d = dim(M))
  - Either $S_{n-1}^*(a_i, b_j)$ or
  - Best sequence that prepends (a_i, b_j) to $S_{n-1}^*(a_{act}, b_{BR(a_i)})$

$S_3^*(a_0, b_0)$	**?**
$S_2^*(a_0, b_0)$	Cost 55
$S_2^*(a_1, b_0)$	Cost 37
$S_2^*(a_2, b_0)$	Cost 40

$M1$	b_0	b_1	b_2
a_0	25	1	0
a_1	10	30	10
a_2	0	33	40

Dynamic Programming Algorithm

- Define $S^*_n(a_i, b_j)$ = optimal sequence of length n
- Define $S^*_0(a_i, b_j)$ to be cost 0 if $m_{i,j} = m_*$, else ∞
- Find $S^*_n(a_i, b_j)$ using S^*_{n-1}'s ($O(d)$, d =dim(M))
  - Either $S^*_{n-1}(a_i, b_j)$ or
  - Best sequence that prepends (a_i, b_j) to $S^*_{n-1}(a_{act}, b_{BR(a_i)})$

$S^*_3(a_0, b_0)$	**Cost 52**
$S^*_2(a_0, b_0)$	Cost 55
$S^*_2(a_1, b_0)$	Cost 37+15
$S^*_2(a_2, b_0)$	Cost 40+15

$M1$	b_0	b_1	b_2
a_0	25	1	0
a_1	**10**	30	10
a_2	0	33	40

Dynamic Programming Algorithm

- Define $S^*_n(a_i, b_j)$ = optimal sequence of length n
- Define $S^*_0(a_i, b_j)$ to be cost 0 if $m_{i,j} = m_*$, else ∞
- Find $S^*_n(a_i, b_j)$ using S^*_{n-1}'s $\quad (O(d)$, $d =$dim$(M))$
  - Either $S^*_{n-1}(a_i, b_j)$ or
  - Best sequence that prepends (a_i, b_j) to $S^*_{n-1}(a_{act}, b_{BR(a_i)})$
- Sufficient to calculate $S^*_n(a_i, b_0), \forall i < x$ $\quad$ loop $O(d^2)$
  - How high do we need to let n get?

$S^*_3(a_0, b_0)$	Cost 52
$S^*_2(a_0, b_0)$	Cost 55
$S^*_2(a_1, b_0)$	Cost 37+15
$S^*_2(a_2, b_0)$	Cost 40+15

$M1$	b_0	b_1	b_2
a_0	25	1	0
a_1	10	30	10
a_2	0	33	40

Maximal Sequence Length

- Recall: *Agent A* has x actions, *Agent B* has y
- **Theorem:** No sequence is longer than $\min(x, y)$

Maximal Sequence Length

- Recall: *Agent A* has x actions, *Agent B* has y
- **Theorem:** No sequence is longer than $\min(x, y)$
  - Neither agent takes the same action twice
  - Otherwise, part of the sequence could be excised

Maximal Sequence Length

- Recall: *Agent A* has x actions, *Agent B* has y
- **Theorem:** No sequence is longer than $\min(x, y)$
  - Neither agent takes the same action twice
  - Otherwise, part of the sequence could be excised
- **Theorem:** $\exists M$ with optimal sequence $\min(x, y)$

Maximal Sequence Length

- Recall: *Agent A* has x actions, *Agent B* has y
- **Theorem:** No sequence is longer than $\min(x, y)$
  - Neither agent takes the same action twice
  - Otherwise, part of the sequence could be excised
- **Theorem:** $\exists M$ with optimal sequence $\min(x, y)$

$M2$	b_0	b_1	b_2	$\cdots$	b_{y-3}	b_{y-2}	b_{y-1}
a_0	$100-\delta$	0	0	$\cdots$	0	0	0
a_1	$100-2\delta$	$100-\delta$	0		$\vdots$	0	0
a_2	0	$100-2\delta$	$100-\delta$			$\vdots$	0
$\vdots$	$\vdots$		$\ddots$	$\ddots$			$\vdots$
a_{x-3}	0	$\vdots$		$\ddots$	$100-\delta$	0	0
a_{x-2}	0	0	$\vdots$		$100-2\delta$	$100-\delta$	0
a_{x-1}	0	0	0	$\cdots$	0	$100-2\delta$	100

Maximal Sequence Length

- Recall: *Agent A* has x actions, *Agent B* has y
- **Theorem:** No sequence is longer than $\min(x, y)$
  - Neither agent takes the same action twice
  - Otherwise, part of the sequence could be excised
- **Theorem:** $\exists M$ with optimal sequence $\min(x, y)$

$M2$	b_0	b_1	b_2	$\cdots$	b_{y-3}	b_{y-2}	b_{y-1}
a_0	$100-\delta$	0	0	$\cdots$	0	0	0
a_1	$100-2\delta$	$100-\delta$	0		$\vdots$	0	0
a_2	0	$100-2\delta$	$100-\delta$			$\vdots$	0
$\vdots$	$\vdots$		$\ddots$	$\ddots$			$\vdots$
a_{x-3}	0	$\vdots$		$\ddots$	$100-\delta$	0	0
a_{x-2}	0	0	$\vdots$		$100-2\delta$	$100-\delta$	0
a_{x-1}	0	0	0	$\cdots$	0	$100-2\delta$	100

Questions

- Find the optimal action sequence efficiently? $O(d^3)$
- Maximum length of optimal sequences? $\min(x, y)$

Cases

- Deterministic teammate, 1-step memory ($mem = 1, \epsilon = 0$)
- Longer teammate memory ($mem > 1, \epsilon = 0$)
- Teammate non-determinism ($mem > 1, \epsilon > 0$)

Questions

- Find the optimal action sequence efficiently?
- Maximum length of optimal sequences?

Cases

- Deterministic teammate, 1-step memory ($mem = 1, \epsilon = 0$)
- Longer teammate memory ($mem > 1, \epsilon = 0$)
- Teammate non-determinism ($mem > 1, \epsilon > 0$)

Longer Teammate Memory

- Algorithm extends naturally, but exponential in *mem*
  - Need S^*_{n-1} for every possible history of *Agent A* actions
  - Reaching m^* once not sufficient ("stability")

Longer Teammate Memory

- Algorithm extends naturally, but exponential in *mem*
  - Need S^*_{n-1} for every possible history of *Agent A* actions
  - Reaching m^* once not sufficient (“stability”)

History	$[a_2; a_1; a_0]$
Response	b_2

$M3$	b_0	b_1	b_2
a_0	0	30	50
a_1	41	20	0
a_2	99	20	**100**

Longer Teammate Memory

- Algorithm extends naturally, but exponential in *mem*
  - Need S^*_{n-1} for every possible history of *Agent A* actions
  - Reaching m^* once not sufficient ("stability")

History $[a_2; a_2; a_1]$
Response b_0

$M3$	b_0	b_1	b_2
a_0	0	30	50
a_1	41	20	0
a_2	99	20	**100**

Longer Teammate Memory

- Algorithm extends naturally, but exponential in *mem*
  - Need S^*_{n-1} for every possible history of *Agent A* actions
  - Reaching m^* once not sufficient ("stability")

History	$[a_2; a_2; a_2]$
Response	b_2

$M3$	b_0	b_1	b_2
a_0	0	30	50
a_1	41	20	0
a_2	**99**	20	100

Longer Teammate Memory

- Algorithm extends naturally, but exponential in *mem*
  - Need S^*_{n-1} for every possible history of *Agent A* actions
  - Reaching m^* once not sufficient ("stability")

History	$[a_2; a_2; a_2]$
Response	b_2

$M3$	b_0	b_1	b_2
a_0	0	30	50
a_1	41	20	0
a_2	**99**	20	100

  - NP-hard: reduction from Hamiltonian Path (Littman)

Longer Teammate Memory

- **Theorem:** $\exists M$ with optimal seq. $(\min(x, y) - 1) * mem + 1$
- **Conjecture:** No seq. longer than $(\min(x, y) - 1) * mem + 1$

Longer Teammate Memory

- **Theorem:** $\exists M$ with optimal seq. $(\min(x, y) - 1) * mem + 1$
- **Conjecture:** No seq. longer than $(\min(x, y) - 1) * mem + 1$
  - Can only prove no seq. longer than $\min(x, y) * x^{mem-1}$

Longer Teammate Memory

- **Theorem:** $\exists M$ with optimal seq. $(\min(x,y)-1)*mem+1$
- **Conjecture:** No seq. longer than $(\min(x,y)-1)*mem+1$
  - Can only prove no seq. longer than $\min(x,y)*x^{mem-1}$

$M2$	b_0	b_1	b_2	$\cdots$	b_{y-3}	b_{y-2}	b_{y-1}
a_0	$100-\delta$	0	0	$\cdots$	0	0	0
a_1	$100-2\delta$	$100-\delta$	0		$\vdots$	0	0
a_2	0	$100-2\delta$	$100-\delta$			$\vdots$	0
$\vdots$	$\vdots$		$\ddots$	$\ddots$			$\vdots$
a_{x-3}	0	$\vdots$		$\ddots$	$100-\delta$	0	0
a_{x-2}	0	0	$\vdots$		$100-2\delta$	$100-\delta$	0
a_{x-1}	0	0	0	$\cdots$	0	$100-2\delta$	100

Questions

- Find the optimal action sequence efficiently? no
- Maximum length of optimal sequences? ?

Cases

- Deterministic teammate, 1-step memory ($mem = 1, \epsilon = 0$)
- Longer teammate memory ($mem > 1, \epsilon = 0$)
- Teammate non-determinism ($mem > 1, \epsilon > 0$)

Questions

- Find the optimal action sequence efficiently?
- Maximum length of optimal sequences?

Cases

- Deterministic teammate, 1-step memory ($mem = 1, \epsilon = 0$)
- Longer teammate memory ($mem > 1, \epsilon = 0$)
- Teammate non-determinism ($mem > 1, \epsilon > 0$)

Teammate Non-Determinism

- $EV(a_i, b_j) = (1-\epsilon)m_{i,j} + \frac{\epsilon}{y}(\sum_{k=0}^{y-1} m_{i,k})$
  - Cost now sum of $m^* - EV(a_i, b_j)$ over sequence

Teammate Non-Determinism

- $EV(a_i, b_j) = (1 - \epsilon)m_{i,j} + \frac{\epsilon}{y}(\sum_{k=0}^{y-1} m_{i,k})$
  - Cost now sum of $m^* - EV(a_i, b_j)$ over sequence
  - m^* now maximum $EV(a_i, b_j)$ in M

Teammate Non-Determinism

- $EV(a_i, b_j) = (1-\epsilon)m_{i,j} + \frac{\epsilon}{y}(\sum_{k=0}^{y-1} m_{i,k})$
  - Cost now sum of $m^* - EV(a_i, b_j)$ over sequence
  - m^* now maximum $EV(a_i, b_j)$ in M

- “Target” (m^*) can change:

Teammate Non-Determinism

- $EV(a_i, b_j) = (1 - \epsilon)m_{i,j} + \frac{\epsilon}{y}(\sum_{k=0}^{y-1} m_{i,k})$
  - Cost now sum of $m^* - EV(a_i, b_j)$ over sequence
  - m^* now maximum $EV(a_i, b_j)$ in M

- "Target" (m^*) can change: $S^*(a_0, b_0)$ with *mem*=3

Teammate Non-Determinism

- $EV(a_i, b_j) = (1-\epsilon)m_{i,j} + \frac{\epsilon}{y}(\sum_{k=0}^{y-1} m_{i,k})$
  - Cost now sum of $m^* - EV(a_i, b_j)$ over sequence
  - m^* now maximum $EV(a_i, b_j)$ in M

- "Target" (m^*) can change: $S^*(a_0, b_0)$ with *mem*=3

$\epsilon = 0$: m_* at (a_3, b_3) L(S^*)=10

$\epsilon = 0.1$: m_* at (a_3, b_3) L(S^*)=8

$\epsilon = 0.3$: m_* at (a_3, b_3) L(S^*)=3

$\epsilon = 0.4$: m_* at (a_2, b_2) L(S^*)=3

$M4$	b_0	b_1	b_2	b_3
a_0	25	0	0	0
a_1	88	90	99	80
a_2	70	98	99	80
a_3	70	70	98	100

Teammate Non-Determinism

- $EV(a_i, b_j) = (1-\epsilon)m_{i,j} + \frac{\epsilon}{y}(\sum_{k=0}^{y-1} m_{i,k})$
  - Cost now sum of $m^* - EV(a_i, b_j)$ over sequence
  - m^* now maximum $EV(a_i, b_j)$ in M

- "Target" (m^*) can change: $S^*(a_0, b_0)$ with *mem*=3

$\epsilon = 0$:	m_* at (a_3, b_3)	L(S^*)=10
$\epsilon = 0.1$:	m_* at (a_3, b_3)	L(S^*)=8
$\epsilon = 0.3$:	m_* at (a_3, b_3)	L(S^*)=3
$\epsilon = 0.4$:	m_* at (a_2, b_2)	L(S^*)=3

$M4$	b_0	b_1	b_2	b_3
a_0	25	0	0	0
a_1	88	90	99	80
a_2	70	98	99	80
a_3	70	70	98	100

- Algorithm and theorems hold unchanged

Teammate Non-Determinism

- $EV(a_i, b_j) = (1-\epsilon)m_{i,j} + \frac{\epsilon}{y}(\sum_{k=0}^{y-1} m_{i,k})$
  - Cost now sum of $m^* - EV(a_i, b_j)$ over sequence
  - m^* now maximum $EV(a_i, b_j)$ in M

- "Target" (m^*) can change: $S^*(a_0, b_0)$ with *mem*=3

$\epsilon = 0$: m_* at (a_3, b_3) L(S^*)=10
$\epsilon = 0.1$: m_* at (a_3, b_3) L(S^*)=8
$\epsilon = 0.3$: m_* at (a_3, b_3) L(S^*)=3
$\epsilon = 0.4$: m_* at (a_2, b_2) L(S^*)=3

$M4$	b_0	b_1	b_2	b_3
a_0	25	0	0	0
a_1	88	90	99	80
a_2	70	98	99	80
a_3	70	70	98	100

- Algorithm and theorems hold unchanged
  - Except when $\epsilon = 1$

Questions

- Find the optimal action sequence efficiently? no
- Maximum length of optimal sequences? ?

Cases

- Deterministic teammate, 1-step memory ($mem = 1, \epsilon = 0$)
- Longer teammate memory ($mem > 1, \epsilon = 0$)
- Teammate non-determinism ($mem > 1, \epsilon > 0$)

Experiments

- All variations of the algorithm fully implemented

Experiments

- All variations of the algorithm fully implemented
- Test frequency of longest S^* of varying lengths
  - 3x3 matrix: how often $L(S^*(a_i, b_j)) = 3$?

Experiments

- All variations of the algorithm fully implemented
- Test frequency of longest S^* of varying lengths
  - 3x3 matrix: how often $L(S^*(a_i, b_j)) = 3$?
- $m_{i,j}$ uniformly random in $[0, 100]$; $m_{x-1,y-1} = 100$

Experiments

- All variations of the algorithm fully implemented
- Test frequency of longest S^* of varying lengths
  - 3x3 matrix: how often $L(S^*(a_i, b_j)) = 3$?
- $m_{i,j}$ uniformly random in $[0, 100]$; $m_{x-1,y-1} = 100$

mem=1	1	2	3	4	5	6	7	8	9	10
3×3	104	852	44							

Experiments

- All variations of the algorithm fully implemented
- Test frequency of longest S^* of varying lengths
  - 3x3 matrix: how often $L(S^*(a_i, b_j)) = 3$?
- $m_{i,j}$ uniformly random in $[0, 100]$; $m_{x-1,y-1} = 100$

mem=1	1	2	3	4	5	6	7	8	9	10
3×3	104	852	44							
4×4	12	825	158	5						
5×5	3	662	316	19	0					
6×6	0	465	489	45	1	0				
7×7	0	349	565	81	5	0	0			
8×8	0	236	596	159	8	1	0	0		
9×9	0	145	640	193	20	2	0	0	0	
10×10	0	72	636	263	29	0	0	0	0	0

Experiments

mem=1	1	2	3	4	5	6	7	8	9	10
3×3	104	852	44							
4×4	12	825	158	5						
5×5	3	662	316	19	0					
6×6	0	465	489	45	1	0				
7×7	0	349	565	81	5	0	0			
8×8	0	236	596	159	8	1	0	0		
9×9	0	145	640	193	20	2	0	0	0	
10×10	0	72	636	263	29	0	0	0	0	0

mem=3	1	2	3	4	5	6	7	8	9	10	11
3×3	98	178	344	340	28	8	4	0	0	0	0
4×4	15	76	266	428	134	60	21	0	0	0	0
5×5	1	19	115	408	234	145	71	7	0	0	0
6×6	0	0	22	282	272	222	164	27	11	0	0
7×7	0	0	5	116	293	282	220	55	17	10	1

Robot Experiments

- In progress. . .

Related Work

Game Theory

- Multiagent learning (Claus & Boutilier, '98),(Littman, '01),
  (Conitzer & Sandholm, '03),(Powers & Shoham, '05),(Chakraborty & Stone, '08)
- Economic repeated games (Hart & Mas-Colell, '00),(Neyman & Okada, '00)
- Fictitious play (Brown, '51)
- Adaptive play (Young, '93)

Opponent Modeling

- Intended plan recognition (Sidner, '85),(Lochbaum,'91),(Carberry, '01)
- SharedPlans (Grosz & Kraus, '96)
- Recursive Modeling (Vidal & Durfee, '95)

Ad Hoc Teams

- Ad hoc team player is an individual
  - Unknown teammates (programmed by others)
- May or may not be able to communicate
- Teammates likely sub-optimal: no control

Goal: Create a good team player

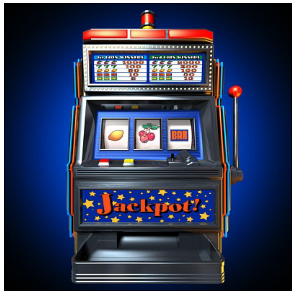
- Minimal representative scenarios
  - One teammate, no communication
  - Fixed and known behavior: best response

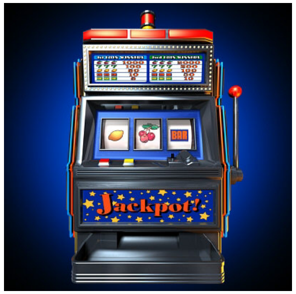
Scenarios

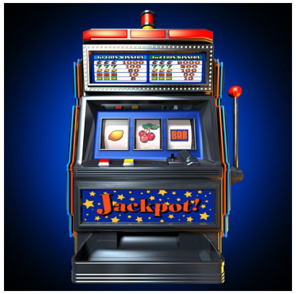
- Cooperative normal form game (w/ Kaminka & Rosenschein)

$M1$	b_0	b_1	b_2
a_0	25	1	0
a_1	10	30	10
a_2	0	33	40

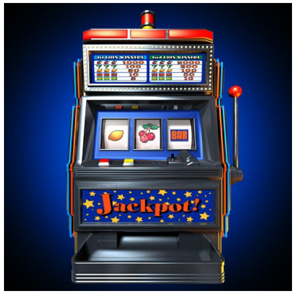
- Cooperative k-armed bandit (w/ Kraus)







3-armed bandit

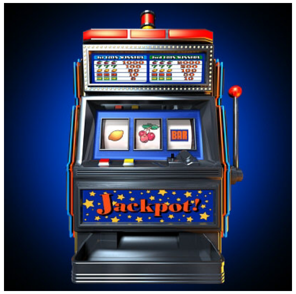


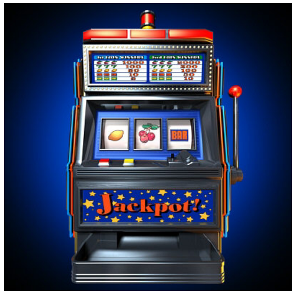
$\Longrightarrow$

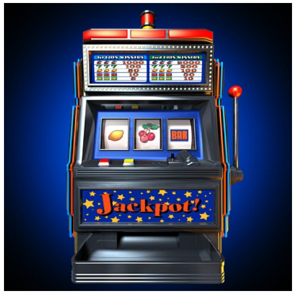
- Random value from a distribution
- Expected value μ

3-armed bandit

Arm_* Arm_1 Arm_2

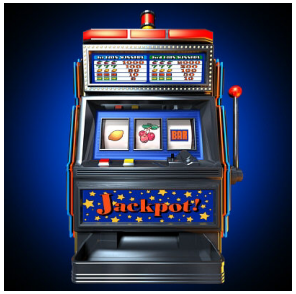


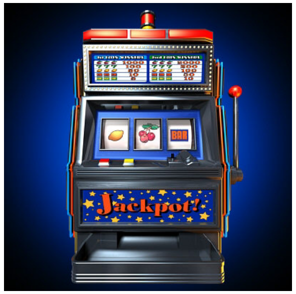


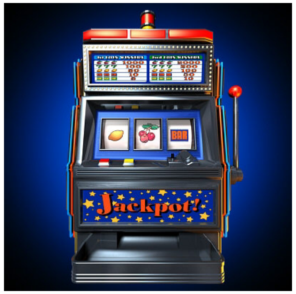


3-armed bandit

Arm$_*$ Arm$_1$ Arm$_2$



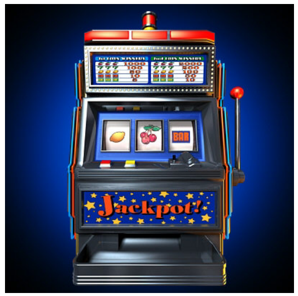
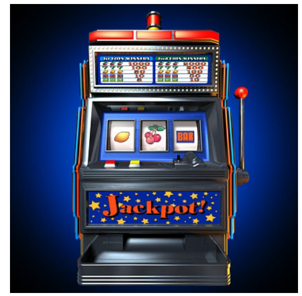
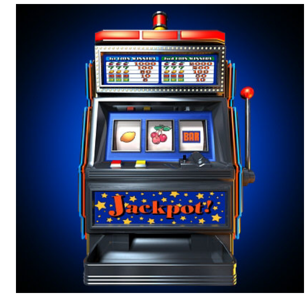




$$\mu_* > \mu_1 > \mu_2$$

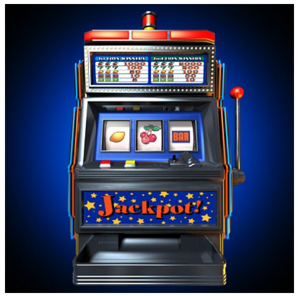
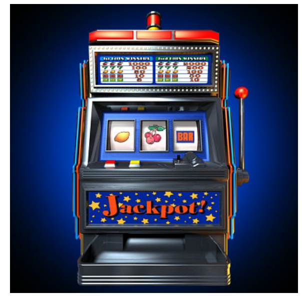
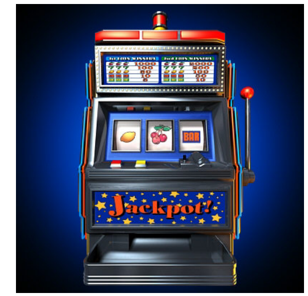
- Agent A: teacher
  - Knows payoff distributions
  - Objective: maximize expected sum of payoffs

3-armed bandit

Arm_*		Arm_1		Arm_2
				
μ_*	$>$	μ_1	$>$	μ_2

- Agent A: teacher
  - Knows payoff distributions
  - Objective: maximize expected sum of payoffs
  - If alone, always Arm_*

3-armed bandit

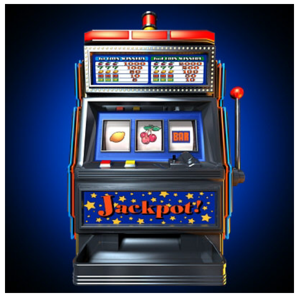
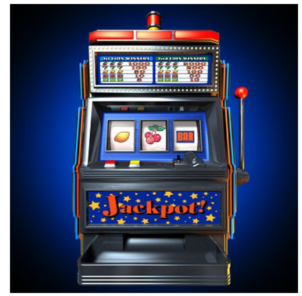
Arm$_*$	Arm$_1$	Arm$_2$
		

$$\mu_* \quad > \quad \mu_1 \quad > \quad \mu_2$$

- Agent A: teacher
  - Knows payoff distributions
  - Objective: maximize expected sum of payoffs
  - If alone, always Arm$_*$
- Agent B: learner
  - Can only pull Arm$_1$ or Arm$_2$

3-armed bandit

Arm_* | Arm_1 | Arm_2

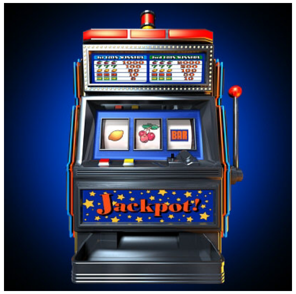




$\mu_* \quad > \quad \mu_1 \quad > \quad \mu_2$

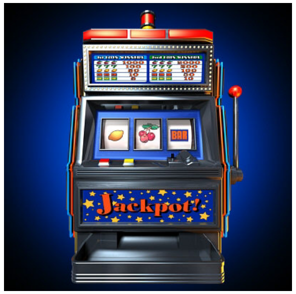
- Agent A: teacher
  - Knows payoff distributions
  - Objective: maximize expected sum of payoffs
  - If alone, always Arm_*
- Agent B: learner
  - Can only pull Arm_1 or Arm_2
  - Selects arm with highest observed sample average

Assumptions

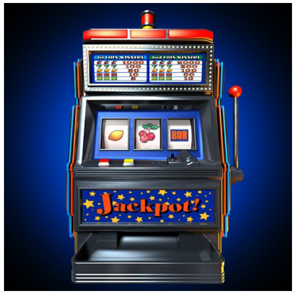
Arm_*



Arm_1

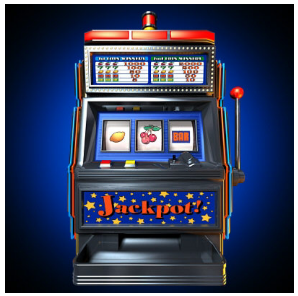


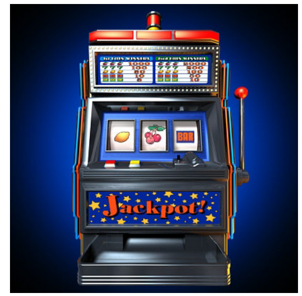
Arm_2



Assumptions

Arm$_*$ Arm$_1$ Arm$_2$

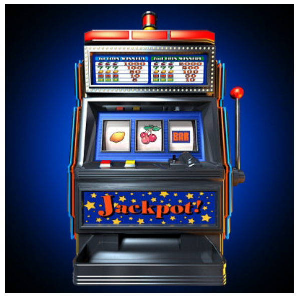
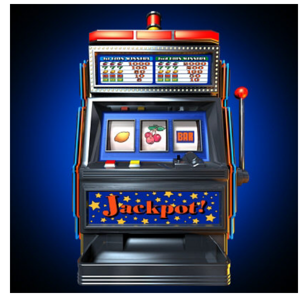




$$\mu_* > \mu_1 > \mu_2$$

- Alternate actions (teacher first)
- Results of all actions fully observable (to both)

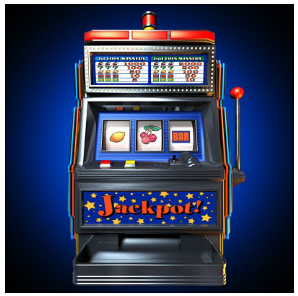
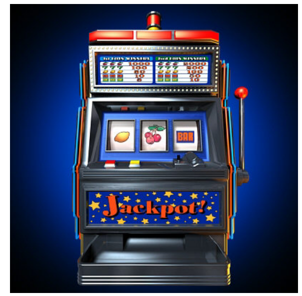
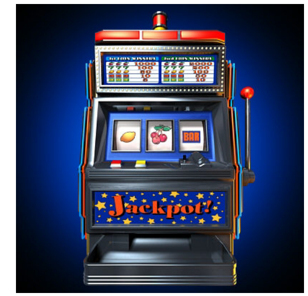
Assumptions

Arm_*		Arm_1		Arm_2
				
μ_*	$>$	μ_1	$>$	μ_2

- Alternate actions (teacher first)
- Results of all actions fully observable (to both)
- Number of rounds remaining finite, known to teacher

Assumptions

Arm$_*$ Arm$_1$ Arm$_2$

$\mu_* \quad > \quad \mu_1 \quad > \quad \mu_2$

- Alternate actions (teacher first)
- Results of all actions fully observable (to both)
- Number of rounds remaining finite, known to teacher

Objective: maximize expected sum of payoffs

Formalism

- μ_i: expected payoff of Arm$_i$ ($i \in \{1, 2, *\}$)
  - Assume $\mu_* > \mu_1 > \mu_2$: only interesting case

Formalism

- μ_i: expected payoff of Arm_i ($i \in \{1, 2, *\}$)
  - Assume $\mu_* > \mu_1 > \mu_2$: only interesting case
- n_i: number of times Arm_i has been pulled
- m_i: cumulative payoff from past pulls of Arm_i

Formalism

- μ_i: expected payoff of Arm$_i$ ($i \in \{1, 2, *\}$)
  - Assume $\mu_* > \mu_1 > \mu_2$: only interesting case
- n_i: number of times Arm$_i$ has been pulled
- m_i: cumulative payoff from past pulls of Arm$_i$
- $\bar{x}_i = \frac{m_i}{n_i}$: observed sample average so far

Formalism

- μ_i: expected payoff of Arm_i $(i \in \{1, 2, *\})$
  - Assume $\mu_* > \mu_1 > \mu_2$: only interesting case
- n_i: number of times Arm_i has been pulled
- m_i: cumulative payoff from past pulls of Arm_i
- $\bar{x}_i = \frac{m_i}{n_i}$: observed sample average so far
- r: number of rounds left

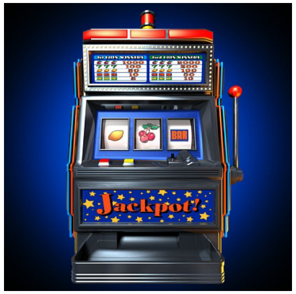
Formalism

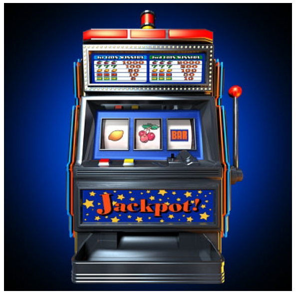
- μ_i: expected payoff of Arm$_i$ ($i \in \{1, 2, *\}$)
  - Assume $\mu_* > \mu_1 > \mu_2$: only interesting case
- n_i: number of times Arm$_i$ has been pulled
- m_i: cumulative payoff from past pulls of Arm$_i$
- $\bar{x}_i = \frac{m_i}{n_i}$: observed sample average so far
- r: number of rounds left

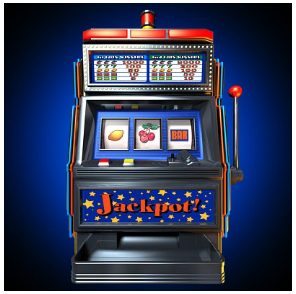
Which arm should the teacher pull, as a function of r and all the μ_i, n_i, and $\bar{x}_i$?

Teacher should consider Arm$_1$

$\mu_* = 10.0$ $\mu_1 = 9.0$ $\mu_2 = 5.0$ $r = 3$

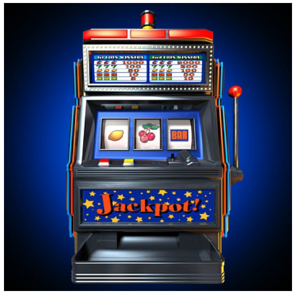




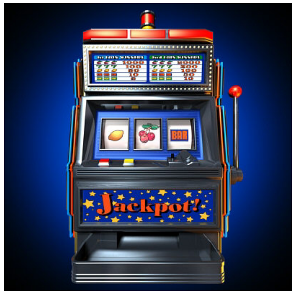


Teacher should consider Arm$_1$

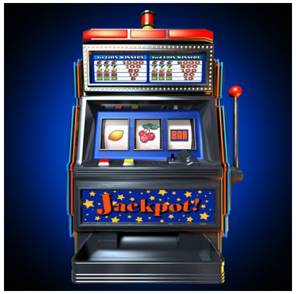
$\mu_* = 10.0$



$\mu_1 = 9.0$



$\mu_2 = 5.0$



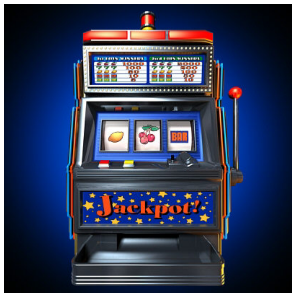
$r = 3$

i	m_i	n_i	$\bar{x}_i$
1	0.0	0	
2	0.0	0	

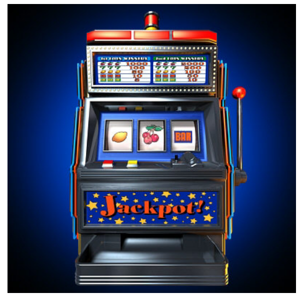
9.8

Teacher should consider Arm$_1$

$\mu_* = 10.0$



$\mu_1 = 9.0$



$\mu_2 = 5.0$

$r = 3$

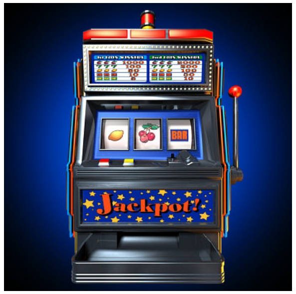
i	m_i	n_i	$\bar{x}_i$
1	0.0	0	
2	0.0	0	

9.8

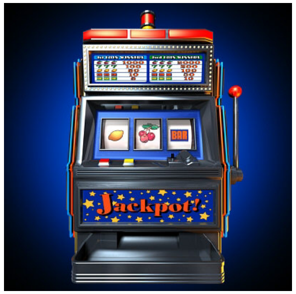
7.0

Teacher should consider Arm$_1$

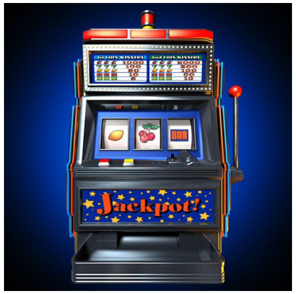
$\mu_* = 10.0$



$\mu_1 = 9.0$



$\mu_2 = 5.0$

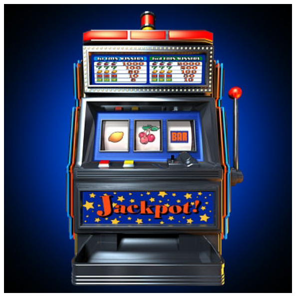


$r = 2$

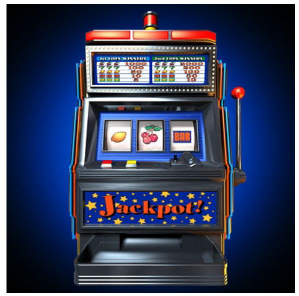
i	m_i	n_i	$\bar{x}_i$
1	0.0	0	
2	7.0	1	7.0

Teacher should consider Arm$_1$

$\mu_* = 10.0$



$\mu_1 = 9.0$



$\mu_2 = 5.0$

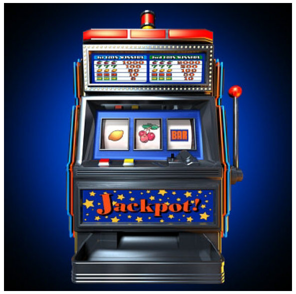
$r = 2$

i	m_i	n_i	$\bar{x}_i$
1	0.0	0	
2	7.0	1	7.0

10.3

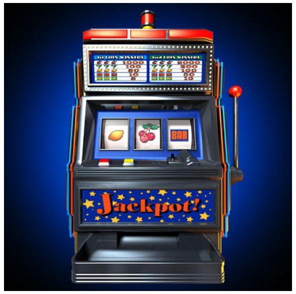
Teacher should consider Arm$_1$

$\mu_* = 10.0$



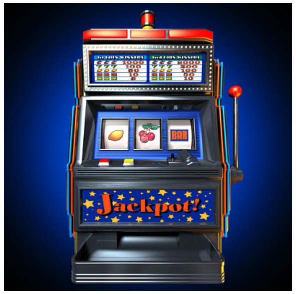
10.3

$\mu_1 = 9.0$



6.0

$\mu_2 = 5.0$

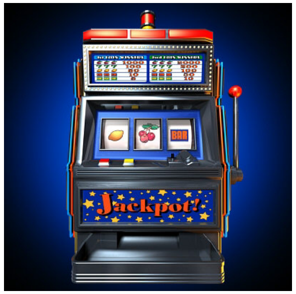


$r = 2$

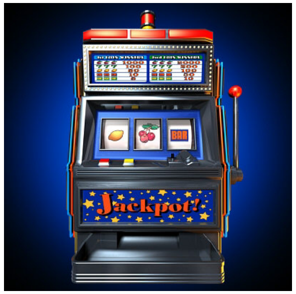
i	m_i	n_i	$\bar{x}_i$
1	0.0	0	
2	7.0	1	7.0

Teacher should consider Arm$_1$

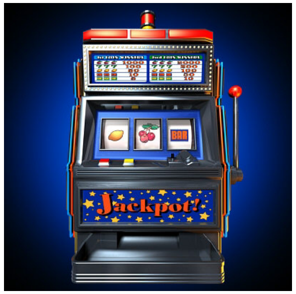
$\mu_* = 10.0$



$\mu_1 = 9.0$



$\mu_2 = 5.0$

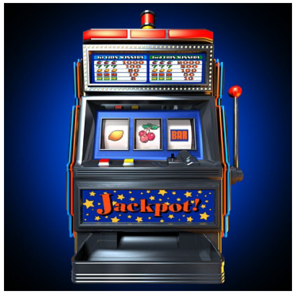


$r = 1$

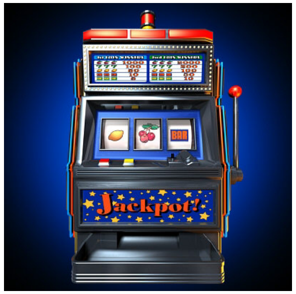
i	m_i	n_i	$\bar{x}_i$
1	6.0	1	6.0
2	7.0	1	7.0

Teacher should consider Arm$_1$

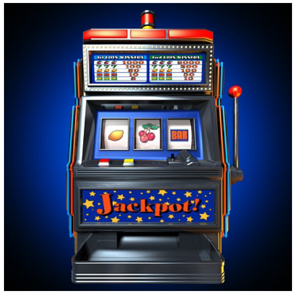
$\mu_* = 10.0$



$\mu_1 = 9.0$



$\mu_2 = 5.0$



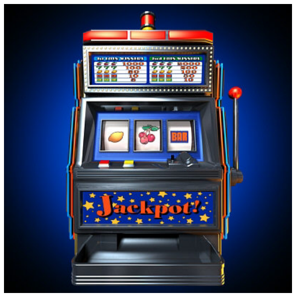
$r = 1$

i	m_i	n_i	$\bar{x}_i$
1	6.0	1	6.0
2	7.0	1	7.0

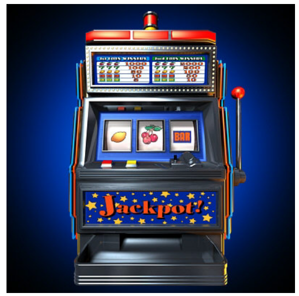
- Teacher Arm$_1$ expected value:

Teacher should consider Arm$_1$

$\mu_* = 10.0$



$\mu_1 = 9.0$



$\mu_2 = 5.0$

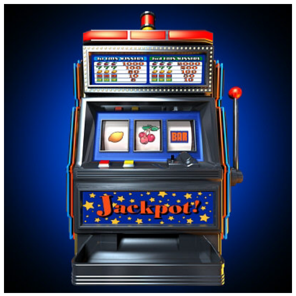
$r = 1$

i	m_i	n_i	$\bar{x}_i$
1	6.0	1	6.0
2	7.0	1	7.0

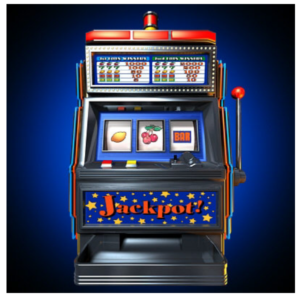
- Teacher Arm$_1$ expected value:
  - Define η: probability Arm$_1$ returns > 8
  - Assume: $\eta > \frac{1}{2}$

Teacher should consider Arm$_1$

$\mu_* = 10.0$



$\mu_1 = 9.0$



$\mu_2 = 5.0$

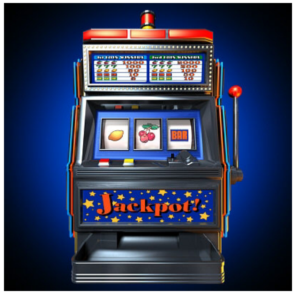
$r = 1$

i	m_i	n_i	$\bar{x}_i$
1	6.0	1	6.0
2	7.0	1	7.0

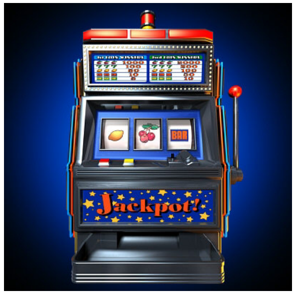
- Teacher Arm$_1$ expected value:
  - Define η: probability Arm$_1$ returns > 8
  - Assume: $\eta > \frac{1}{2}$
  - EV: $\mu_1 + \eta\mu_1 + (1 - \eta)\mu_2$

Teacher should consider Arm$_1$

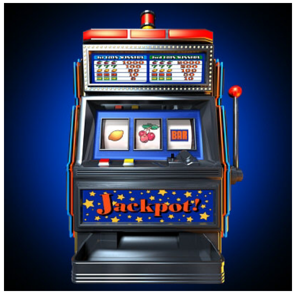
$\mu_* = 10.0$



$\mu_1 = 9.0$



$\mu_2 = 5.0$



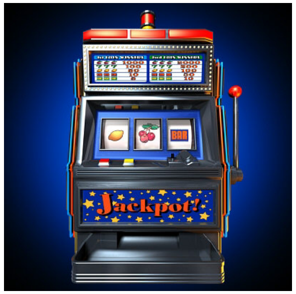
$r = 1$

i	m_i	n_i	$\bar{x}_i$
1	6.0	1	6.0
2	7.0	1	7.0

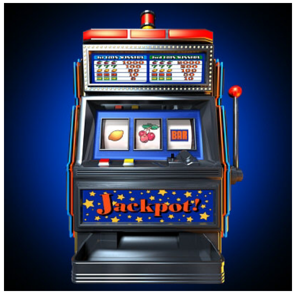
- Teacher Arm$_1$ expected value:
  - Define η: probability Arm$_1$ returns > 8
  - Assume: $\eta > \frac{1}{2}$
  - EV: $\mu_1 + \eta\mu_1 + (1-\eta)\mu_2 > 9 + \frac{9}{2} + \frac{5}{2} = \mathbf{16}$

Teacher should consider Arm_1

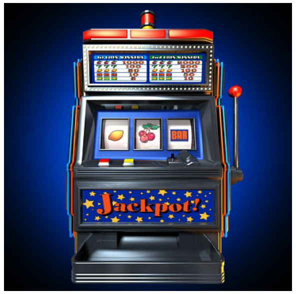
$\mu_* = 10.0$



$\mu_1 = 9.0$



$\mu_2 = 5.0$

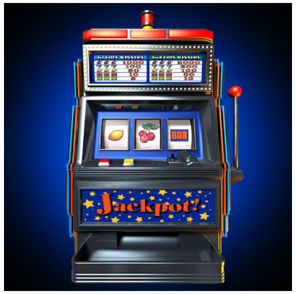


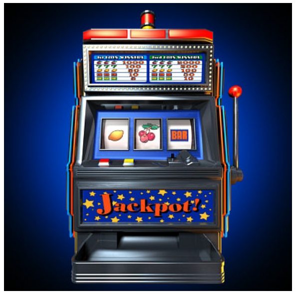
$r = 1$

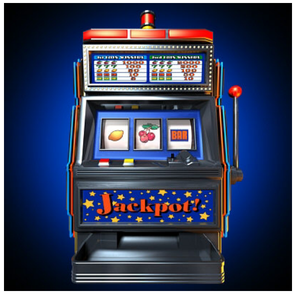
i	m_i	n_i	$\bar{x}_i$
1	6.0	1	6.0
2	7.0	1	7.0

- Teacher Arm_1 expected value:
  - Define η: probability Arm_1 returns > 8
  - Assume: $\eta > \frac{1}{2}$
  - EV: $\mu_1 + \eta\mu_1 + (1-\eta)\mu_2 > 9 + \frac{9}{2} + \frac{5}{2} = \mathbf{16}$
- Teacher Arm_* expected value:
  - EV: $\mu_* + \mu_2$

Teacher should consider Arm$_1$

$\mu_* = 10.0$ 

$\mu_1 = 9.0$ 

$\mu_2 = 5.0$ 

$r = 1$

i	m_i	n_i	$\bar{x}_i$
1	6.0	1	6.0
2	7.0	1	7.0

- Teacher Arm$_1$ expected value:
  - Define η: probability Arm$_1$ returns > 8
  - Assume: $\eta > \frac{1}{2}$
  - EV: $\mu_1 + \eta\mu_1 + (1-\eta)\mu_2 > 9 + \frac{9}{2} + \frac{5}{2} = \mathbf{16}$
- Teacher Arm$_*$ expected value:
  - EV: $\mu_* + \mu_2 = \mathbf{15}$

Should teacher consider Arm$_2$?

- $\bar{x}_1 < \bar{x}_2 \Longrightarrow$ **no**

Should teacher consider Arm$_2$?

- $\bar{x}_1 < \bar{x}_2 \Longrightarrow$ **no**
  - Sequence of values from Arm$_2$: $u_0, u_1, u_2, \ldots$

Should teacher consider Arm$_2$?

- $\bar{x_1} < \bar{x_2} \Longrightarrow$ **no**
  - Sequence of values from Arm$_2$: $u_0, u_1, u_2, \ldots$
  - Optimal from Arm$_2$: u_0

Should teacher consider Arm$_2$?

- $\bar{x_1} < \bar{x_2} \implies$ **no**
  - Sequence of values from Arm$_2$: $u_0, u_1, u_2, \ldots$
  - Optimal from Arm$_2$: u_0, a

Should teacher consider Arm_2?

- $\bar{x_1} < \bar{x_2} \implies$ **no**
  - Sequence of values from Arm_2: $u_0, u_1, u_2, \ldots$
  - Optimal from Arm_2: $u_0, a, b, c, d, e, \ldots w, x, y, z$

Should teacher consider Arm_2?

- $\bar{x_1} < \bar{x_2} \Longrightarrow$ **no**
  - Sequence of values from Arm_2: $u_0, u_1, u_2, \ldots$
  - Optimal from Arm_2: $u_0, a, b, c, d, e, \ldots w, x, y, z$
  - Also possible: μ_*

Should teacher consider Arm$_2$?

- $\bar{x_1} < \bar{x_2} \Longrightarrow$ **no**
  - Sequence of values from Arm$_2$: $u_0, u_1, u_2, \ldots$
  - Optimal from Arm$_2$: $u_0, a, b, c, d, e, \ldots w, x, y, z$
  - Also possible: μ_*, u_0, μ_*

Should teacher consider Arm$_2$?

- $\bar{x_1} < \bar{x_2} \implies$ **no**
  - Sequence of values from Arm$_2$: $u_0, u_1, u_2, \ldots$
  - Optimal from Arm$_2$: $u_0, a, b, c, d, e, \ldots w, x, y, z$
  - Also possible: μ_*, u_0, μ_*, a

Should teacher consider Arm$_2$?

- $\bar{x_1} < \bar{x_2} \implies$ **no**
  - Sequence of values from Arm$_2$: $u_0, u_1, u_2, \ldots$
  - Optimal from Arm$_2$: $u_0, a, b, c, d, e, \ldots w, x, y, z$
  - Also possible: $\mu_*, u_0, \mu_*, a, b, c, d, e, \ldots, w, x$

Should teacher consider Arm$_2$?

- $\bar{x_1} < \bar{x_2} \Longrightarrow$ **no**
  - Sequence of values from Arm$_2$: $u_0, u_1, u_2, \ldots$
  - Optimal from Arm$_2$: $u_0, a, b, c, d, e, \ldots w, x, y, z$
  - Also possible: $\mu_*, u_0, \mu_*, a, b, c, d, e, \ldots, w, x$
- $\bar{x_1} > \bar{x_2} \Longrightarrow$ **?**

Should teacher consider Arm_2?

- $\bar{x_1} < \bar{x_2} \Longrightarrow$ **no**
  - Sequence of values from Arm_2: $u_0, u_1, u_2, \ldots$
  - Optimal from Arm_2: $u_0, a, b, c, d, e, \ldots w, x, y, z$
  - Also possible: $\mu_*, u_0, \mu_*, a, b, c, d, e, \ldots, w, x$
- $\bar{x_1} > \bar{x_2} \Longrightarrow$ **?**
  - Subtle, but still **no**

Should teacher consider Arm_2?

- $\bar{x_1} < \bar{x_2} \implies$ **no**
  - Sequence of values from Arm_2: $u_0, u_1, u_2, \ldots$
  - Optimal from Arm_2: $u_0, a, b, c, d, e, \ldots w, x, y, z$
  - Also possible: $\mu_*, u_0, \mu_*, a, b, c, d, e, \ldots, w, x$
- $\bar{x_1} > \bar{x_2} \implies$ **?**
  - Subtle, but still **no**
  - Challenge: prove it!

Never teach when $\bar{x_1} > \bar{x_2}$

- Same proof

Never teach when $\bar{x_1} > \bar{x_2}$

- Same proof
  - Sequence of values from Arm$_1$: $v_0, v_1, v_2, \ldots$
  - Optimal from Arm$_1$: $v_0, a, b, c, d, e, \ldots w, x, y, z$
  - Also possible: $\mu_*, v_0, \mu_*, a, b, c, d, e, \ldots, w, x$

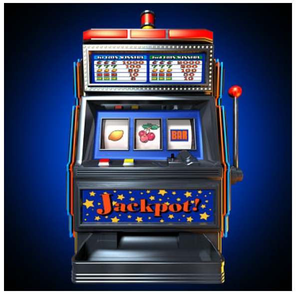
Never teach when $\bar{x_1} > \bar{x_2}$

- Same proof
  - Sequence of values from Arm_1: $v_0, v_1, v_2, \ldots$
  - Optimal from Arm_1: $v_0, a, b, c, d, e, \ldots w, x, y, z$
  - Also possible: $\mu_*, v_0, \mu_*, a, b, c, d, e, \ldots, w, x$
- Only need to consider Arm_1 when $\bar{x_1} < \bar{x_2}$
  - Depends on distributions

Never teach when $\bar{x_1} > \bar{x_2}$

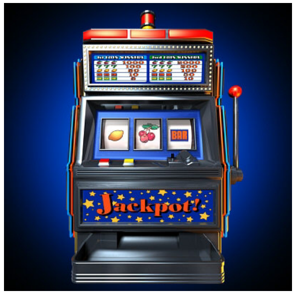
- Same proof
  - Sequence of values from Arm$_1$: $v_0, v_1, v_2, \ldots$
  - Optimal from Arm$_1$: $v_0, a, b, c, d, e, \ldots w, x, y, z$
  - Also possible: $\mu_*, v_0, \mu_*, a, b, c, d, e, \ldots, w, x$
- Only need to consider Arm$_1$ when $\bar{x_1} < \bar{x_2}$
  - Depends on distributions
  - Consider binary and normal

Arms with Binary Distributions



$$\longrightarrow \begin{cases} 1 & \text{with probability } p \\ 0 & \text{with probability } 1-p \end{cases}$$

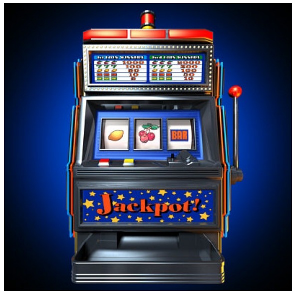
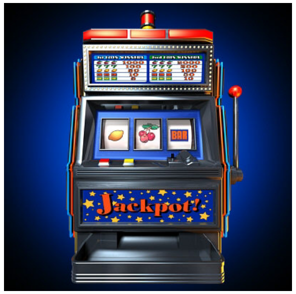
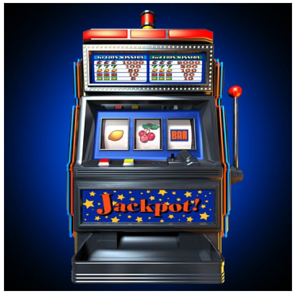
Arms with Binary Distributions



$$\longrightarrow \begin{cases} 1 & \text{with probability } p \\ 0 & \text{with probability } 1-p \end{cases}$$

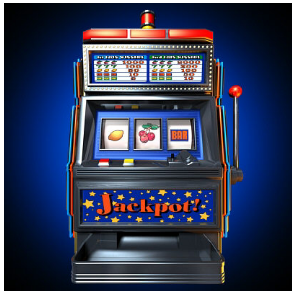
$\mu_i = p_i \quad m_i =$ number of 1's so far

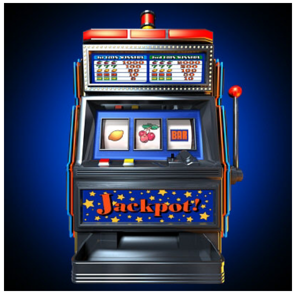
Arms with Binary Distributions, $r = 1$

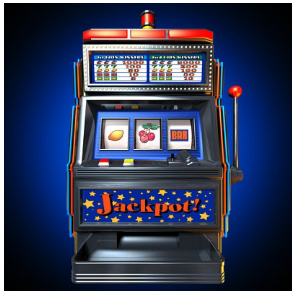




$$p_* > p_1 > p_2$$

Arms with Binary Distributions, $r = 1$





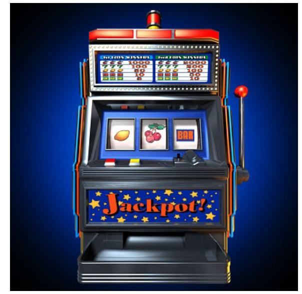


$p_* \quad > \quad p_1 \quad > \quad p_2$

- Consider teaching if

  1. $\bar{x_1} < \bar{x_2}$

Arms with Binary Distributions, $r = 1$

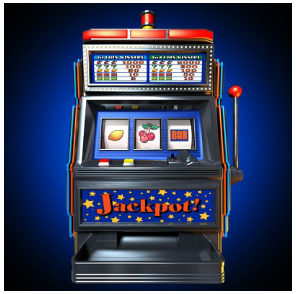
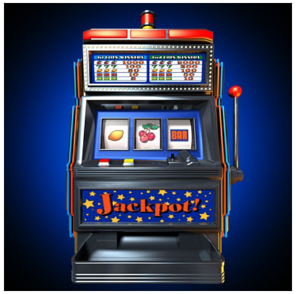
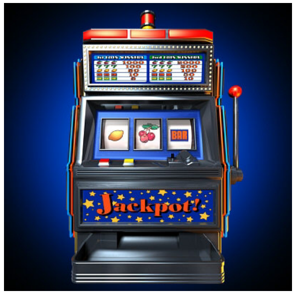
  

$$p_* \quad > \quad p_1 \quad > \quad p_2$$

- Consider teaching if

   1. $\bar{x}_1 < \bar{x}_2 \quad \equiv \quad \frac{m_1}{n_1} < \frac{m_2}{n_2}$

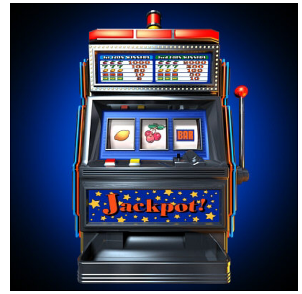
Arms with Binary Distributions, $r = 1$

$p_* > p_1 > p_2$

- Consider teaching if
  1. $\bar{x}_1 < \bar{x}_2 \equiv \frac{m_1}{n_1} < \frac{m_2}{n_2}$
  2. It could help: $\frac{m_1+1}{n_1+1} > \frac{m_2}{n_2}$

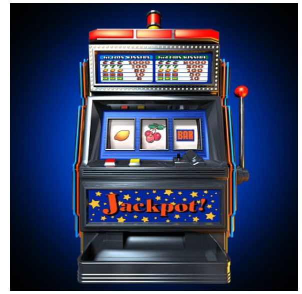
Arms with Binary Distributions, $r = 1$



$p_* \quad > \quad p_1 \quad > \quad p_2$

- Consider teaching if
  1. $\bar{x_1} < \bar{x_2} \quad \equiv \quad \frac{m_1}{n_1} < \frac{m_2}{n_2}$
  2. It could help: $\frac{m_1+1}{n_1+1} > \frac{m_2}{n_2}$
- Teacher Arm$_*$ expected value: $p_* + p_2$

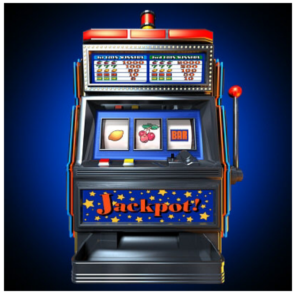
Arms with Binary Distributions, $r = 1$

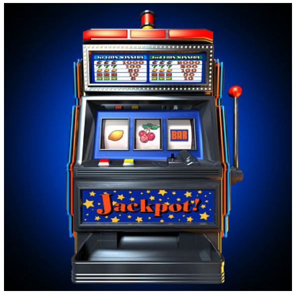



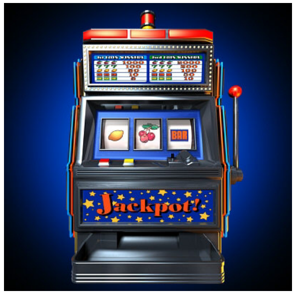

$$p_* \quad > \quad p_1 \quad > \quad p_2$$

- Consider teaching if
  1. $\bar{x_1} < \bar{x_2} \quad \equiv \quad \frac{m_1}{n_1} < \frac{m_2}{n_2}$
  2. It could help: $\frac{m_1+1}{n_1+1} > \frac{m_2}{n_2}$
- Teacher Arm$_*$ expected value: $p_* + p_2$
- Teacher Arm$_1$ expected value: p_1

Arms with Binary Distributions, $r = 1$



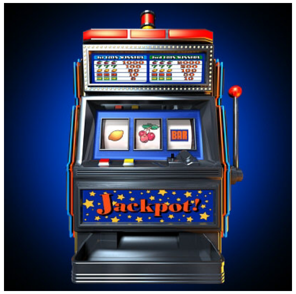


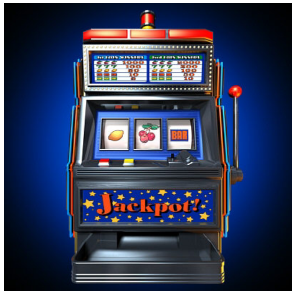


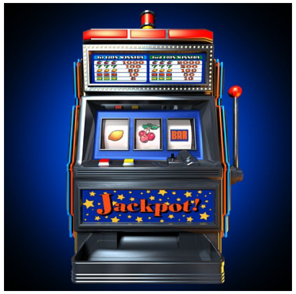
$$p_* \quad > \quad p_1 \quad > \quad p_2$$

- Consider teaching if
  1. $\bar{x_1} < \bar{x_2} \quad \equiv \quad \frac{m_1}{n_1} < \frac{m_2}{n_2}$
  2. It could help: $\frac{m_1+1}{n_1+1} > \frac{m_2}{n_2}$
- Teacher Arm$_*$ expected value: $p_* + p_2$
- Teacher Arm$_1$ expected value: $p_1 + p_1 * p_1$

Arms with Binary Distributions, $r = 1$



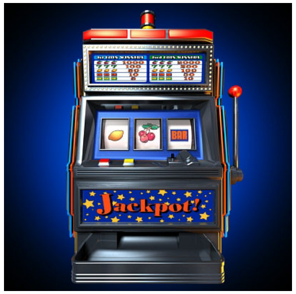
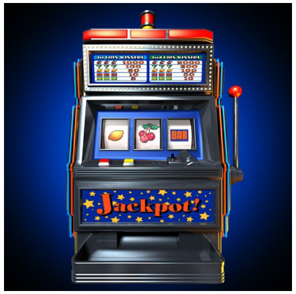
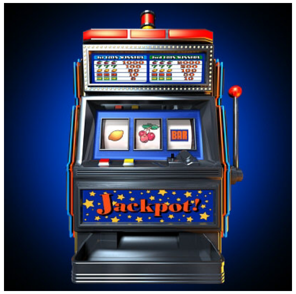




$p_* \quad > \quad p_1 \quad > \quad p_2$

- Consider teaching if
  1. $\bar{x_1} < \bar{x_2} \quad \equiv \quad \frac{m_1}{n_1} < \frac{m_2}{n_2}$
  2. It could help: $\frac{m_1+1}{n_1+1} > \frac{m_2}{n_2}$
- Teacher Arm$_*$ expected value: $p_* + p_2$
- Teacher Arm$_1$ expected value: $p_1 + p_1 * p_1 + (1 - p_1)p_2$

Arms with Binary Distributions, $r = 1$

$p_* \quad > \quad p_1 \quad > \quad p_2$

- Consider teaching if
  1. $\bar{x_1} < \bar{x_2} \quad \equiv \quad \frac{m_1}{n_1} < \frac{m_2}{n_2}$
  2. It could help: $\frac{m_1+1}{n_1+1} > \frac{m_2}{n_2}$
- Teacher Arm$_*$ expected value: $p_* + p_2$
- Teacher Arm$_1$ expected value: $p_1 + p_1 * p_1 + (1 - p_1)p_2$

Teach iff conditions 1, 2, and $p_* - p_1 < p_1(p_1 - p_2)$

Algorithm for Optimal Teacher Action

- Polynomial algorithm finds optimal teacher action
  - Takes starting values M_1, N_1, M_2, N_2 and R

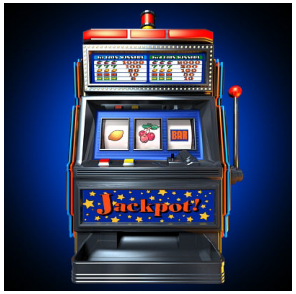
Algorithm for Optimal Teacher Action

- Polynomial algorithm finds optimal teacher action
  - Takes starting values M_1, N_1, M_2, N_2 and R
- Dynamic programming
  - Works backwards from $r = 1$
  - Considers all reachable values of m_1, n_1, m_2, n_2

Algorithm for Optimal Teacher Action

- Polynomial algorithm finds optimal teacher action
  - Takes starting values M_1, N_1, M_2, N_2 and R
- Dynamic programming
  - Works backwards from $r = 1$
  - Considers all reachable values of m_1, n_1, m_2, n_2
- $O(r^5)$ in both memory and runtime

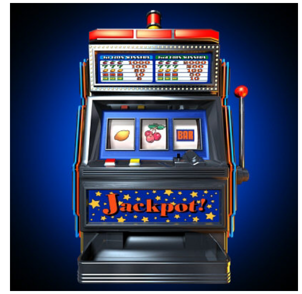
Arms with Normal Distributions



$$\longrightarrow \quad \mathrm{N}(\mu, \sigma)$$

Arms with Normal Distributions, $r = 1$

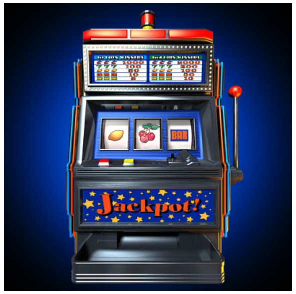
(μ_*, σ_*)



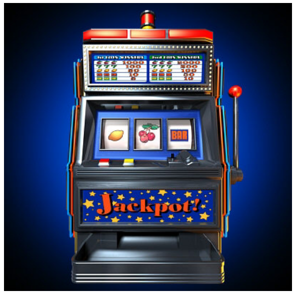
(μ_1, σ_1)

(μ_2, σ_2)

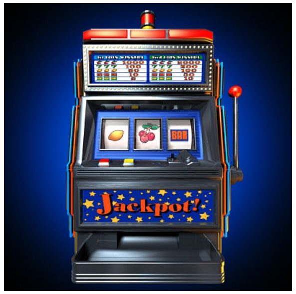
Arms with Normal Distributions, $r = 1$



(μ_*, σ_*)



(μ_1, σ_1)

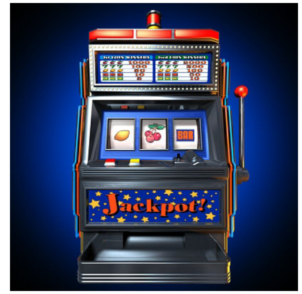


(μ_2, σ_2)

- Cost of teaching: $\mu_* - \mu_1$

Arms with Normal Distributions, $r = 1$

(μ_*, σ_*)

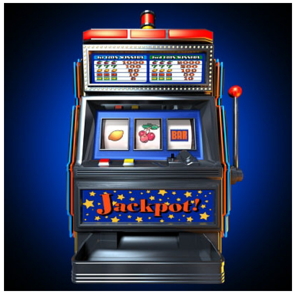


(μ_1, σ_1)

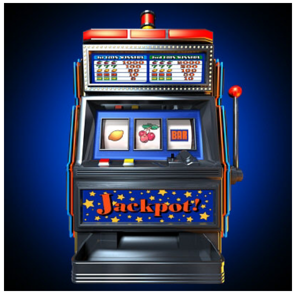
(μ_2, σ_2)

- Cost of teaching: $\mu_* - \mu_1$
- Benefit of teaching if successful: $\mu_1 - \mu_2$

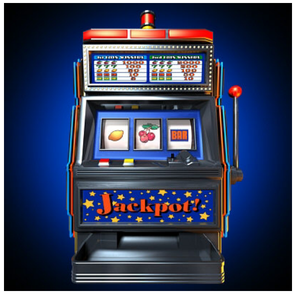
Arms with Normal Distributions, $r = 1$



(μ_*, σ_*)



(μ_1, σ_1)

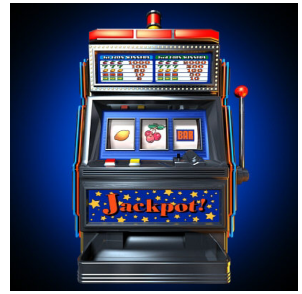


(μ_2, σ_2)

- Cost of teaching: $\mu_* - \mu_1$
- Benefit of teaching if successful: $\mu_1 - \mu_2$ $(\bar{x_1} < \bar{x_2})$

Arms with Normal Distributions, $r = 1$

(μ_*, σ_*)

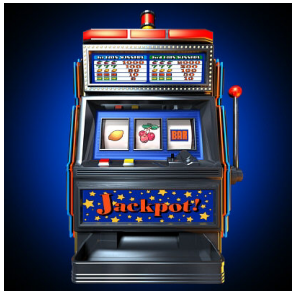


(μ_1, σ_1)

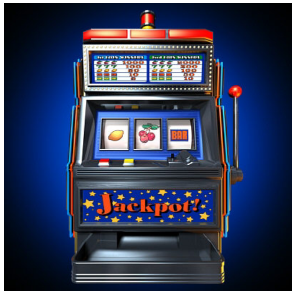
(μ_2, σ_2)

- Cost of teaching: $\mu_* - \mu_1$
- Benefit of teaching if successful: $\mu_1 - \mu_2$ $(\bar{x_1} < \bar{x_2})$
- Probability it's successful: $1 - \Phi_{\mu_1, \sigma_1}(\bar{x_2}(n_1 + 1) - \bar{x_1} n_1)$
  - Cumulative probability that pulling Arm$_1$ causes $\bar{x_1} > \bar{x_2}$

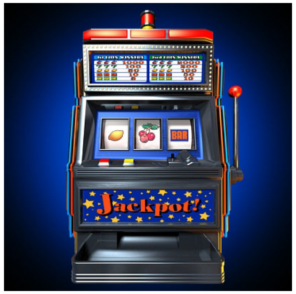
Arms with Normal Distributions, $r = 1$



(μ_*, σ_*)



(μ_1, σ_1)

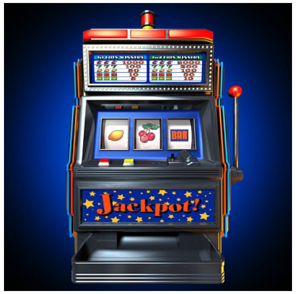


(μ_2, σ_2)

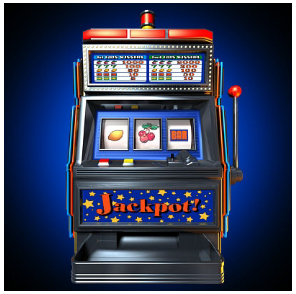
- Cost of teaching: $\mu_* - \mu_1$
- Benefit of teaching if successful: $\mu_1 - \mu_2$ $(\bar{x_1} < \bar{x_2})$
- Probability it's successful: $1 - \Phi_{\mu_1, \sigma_1}(\bar{x_2}(n_1 + 1) - \bar{x_1} n_1)$
  - Cumulative probability that pulling Arm_1 causes $\bar{x_1} > \bar{x_2}$

Teach iff $1 - \Phi_{\mu_1, \sigma_1}(\bar{x_2}(n_1 + 1) - \bar{x_1} n_1) > \frac{\mu_* - \mu_1}{\mu 1 - \mu_2}$

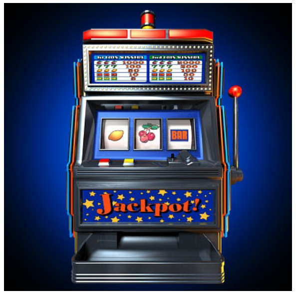
Arms with Normal Distributions, $r \geq 2$



(μ_*, σ_*)



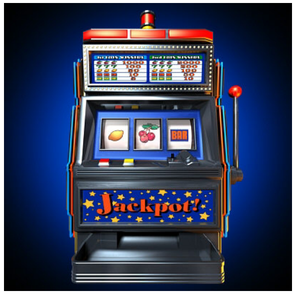
(μ_1, σ_1)



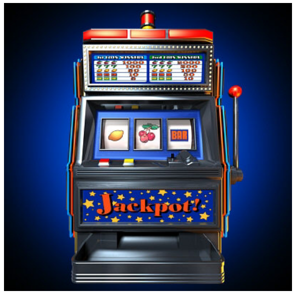
(μ_2, σ_2)

- Can solve computationally — nested integral
- Not exactly, nor efficiently

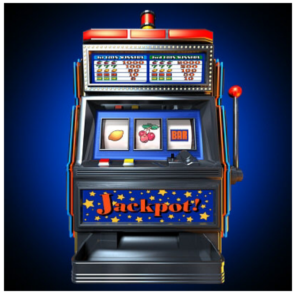
Arms with Normal Distributions, $r \geq 2$



(μ_*, σ_*)



(μ_1, σ_1)



(μ_2, σ_2)

- Can solve computationally — nested integral
- Not exactly, nor efficiently
- Can you find an efficient algorithm?

Experiments

- Evaluating teacher heuristics

Experiments

- Evaluating teacher heuristics

  1. Never teach
  2. Teach iff $\bar{x_1} < \bar{x_2}$
  3. Teach iff it would be optimal to teach if $r = 1$

Experiments

- Evaluating teacher heuristics

  1. Never teach
  2. Teach iff $\bar{x_1} < \bar{x_2}$
  3. Teach iff it would be optimal to teach if $r = 1$

  – None dominates

Experiments

- Evaluating teacher heuristics

  1. Never teach
  2. Teach iff $\bar{x_1} < \bar{x_2}$
  3. Teach iff it would be optimal to teach if $r = 1$

  – None dominates

- Looking for patterns in optimal action as a function of r

Experiments

- Evaluating teacher heuristics

  1. Never teach
  2. Teach iff $\bar{x_1} < \bar{x_2}$
  3. Teach iff it would be optimal to teach if $r = 1$

  – None dominates

- Looking for patterns in optimal action as a function of r

  – Conjecture: teach when $r = 1 \Longrightarrow$ teach when $r = 2$

Experiments

- Evaluating teacher heuristics
  1. Never teach
  2. Teach iff $\bar{x_1} < \bar{x_2}$
  3. Teach iff it would be optimal to teach if $r = 1$

  – None dominates

- Looking for patterns in optimal action as a function of r
  – Conjecture: teach when $r = 1 \Longrightarrow$ teach when $r = 2$
  – **False!**

Experiments

- Evaluating teacher heuristics

  1. Never teach
  2. Teach iff $\bar{x_1} < \bar{x_2}$
  3. Teach iff it would be optimal to teach if $r = 1$

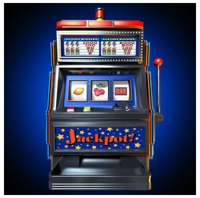
  – None dominates

- Looking for patterns in optimal action as a function of r

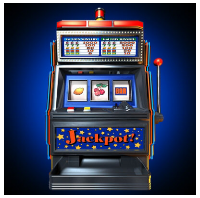
  – Conjecture: teach when $r = 1 \Longrightarrow$ teach when $r = 2$
  – **False!** (binary and normal)

More than 3 arms

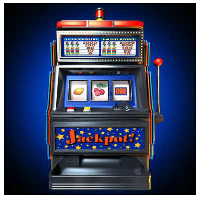
Arm_{*1}



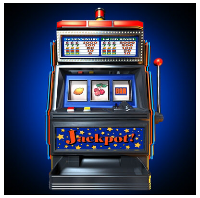
Arm_{*2}



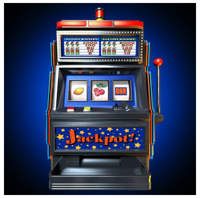
Arm_{*3}



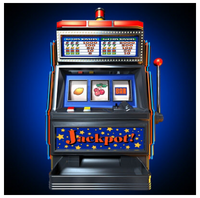
Arm_1



Arm_2



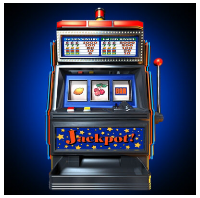
Arm_3



$\cdots$

$\cdots$

Arm_z

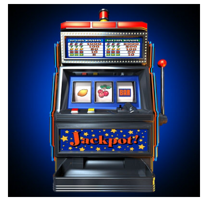


More than 3 arms

Arm_{*1}

Arm_{*2}

Arm_{*3}



Arm_1

Arm_2

Arm_3

$\cdots$

Arm_z

- Additional arms for teacher make no difference

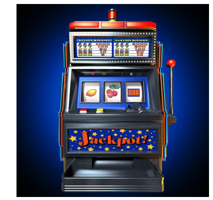
More than 3 arms

Arm_*

Arm_1

Arm_2

Arm_3



$\cdots$

Arm_z

- Additional arms for teacher make no difference
  - Ignore all but the best

More than 3 arms

Arm_*

Arm_1

Arm_2

Arm_3

$\cdots$

Arm_z

- Additional arms for teacher make no difference
  - Ignore all but the best
- Additional learner arms: most results generalize naturally

More than 3 arms

Arm_* Arm_1 Arm_2 Arm_3 $\cdots$ Arm_z

$\cdots$

- Additional arms for teacher make no difference
  - Ignore all but the best
- Additional learner arms: most results generalize naturally
  - Never teach with Arm_z

More than 3 arms

Arm_* Arm_1 Arm_2 Arm_3 $\cdots$ Arm_z





$\ldots$


- Additional arms for teacher make no difference
  - Ignore all but the best
- Additional learner arms: most results generalize naturally
  - Never teach with Arm_z (Arm_1–Arm_{z-1} possible)

More than 3 arms

Arm_*

Arm_1

Arm_2

Arm_3

$\cdots$

Arm_z

- Additional arms for teacher make no difference
  - Ignore all but the best

- Additional learner arms: most results generalize naturally
  - Never teach with Arm_z (Arm_1–Arm_{z-1} possible)
  - Never teach with Arm_i when $\bar{x}_i > \bar{x}_j, \forall j \neq i$

More than 3 arms

Arm$_*$

Arm$_1$

Arm$_2$

Arm$_3$

$\cdots$

$\cdots$

Arm$_z$

- Additional arms for teacher make no difference
  - Ignore all but the best
- Additional learner arms: most results generalize naturally
  - Never teach with Arm$_z$ (Arm$_1$–Arm$_{z-1}$ possible)
  - Never teach with Arm$_i$ when $\bar{x}_i > \bar{x}_j, \forall j \neq i$
  - **Surprising:** May be best to teach with Arm$_j$ for $j > i$

More than 3 arms

Arm_*

Arm_1

Arm_2

Arm_3

$\cdots$

Arm_z

- Additional arms for teacher make no difference
  - Ignore all but the best
- Additional learner arms: most results generalize naturally
  - Never teach with Arm_z (Arm_1–Arm_{z-1} possible)
  - Never teach with Arm_i when $\bar{x}_i > \bar{x}_j, \forall j \neq i$
  - **Surprising:** May be best to teach with Arm_j for $j > i$ (teach with Arm_2, even though $\bar{x}_1 > \bar{x}_2 > \bar{x}_3$)

Sample Open Questions

- What if the teacher doesn't know the distributions?

Sample Open Questions

- What if the teacher doesn't know the distributions?
  - Exploration vs. exploitation

Sample Open Questions

- What if the teacher doesn't know the distributions?
  - Exploration vs. exploitation vs. teaching

Sample Open Questions

- What if the teacher doesn't know the distributions?
  - Exploration vs. exploitation vs. teaching
- What if the learner isn't greedy: explores on its own?

Sample Open Questions

- What if the teacher doesn't know the distributions?
  - Exploration vs. exploitation vs. teaching
- What if the learner isn't greedy: explores on its own?
- How does this extend to the infinite (discounted) case?

Sample Open Questions

- What if the teacher doesn't know the distributions?
  - Exploration vs. exploitation vs. teaching
- What if the learner isn't greedy: explores on its own?
- How does this extend to the infinite (discounted) case?
- What if there are multiple learners?

Ad Hoc Teams

- Ad hoc team player is an individual
  - Unknown teammates (programmed by others)
- May or may not be able to communicate
- Teammates likely sub-optimal: no control

Goal: Create a good team player

Ad Hoc Teams

- Ad hoc team player is an individual
  - Unknown teammates (programmed by others)
- May or may not be able to communicate
- Teammates likely sub-optimal: no control

Goal: Create a good team player

– **So far:** Minimal representative scenarios

Ad Hoc Teams

- Ad hoc team player is an individual
  - Unknown teammates (programmed by others)
- May or may not be able to communicate
- Teammates likely sub-optimal: no control

Goal: Create a good team player

- **So far:** Minimal representative scenarios
- **Future:** Unknown teammate behavior, communication, incomplete teacher knowledge,. . .

Ad Hoc Teams

- Ad hoc team player is an individual
  - Unknown teammates (programmed by others)
- May or may not be able to communicate
- Teammates likely sub-optimal: no control

Goal: Create a good team player

– **So far:** Minimal representative scenarios
– **Future:** Unknown teammate behavior, communication, incomplete teacher knowledge,. . . **much more!**

Acknowledgements

- Yonatan Aumann, Michael Littman, Reshef Meir, Jeremy Stober, Daniel Stronger
- Fulbright and Guggenheim Foundations
- Israel Science Foundation