Fairness in Reinforcement Learning

Matthew Joseph, UPenn

Fairness in Reinforcement Learning

Joint work with


Shahin Jabbari


Michael Kearns


Jamie Morgenstern


Aaron Roth

Matthew Joseph, UPenn

Penn

Good News

Good News

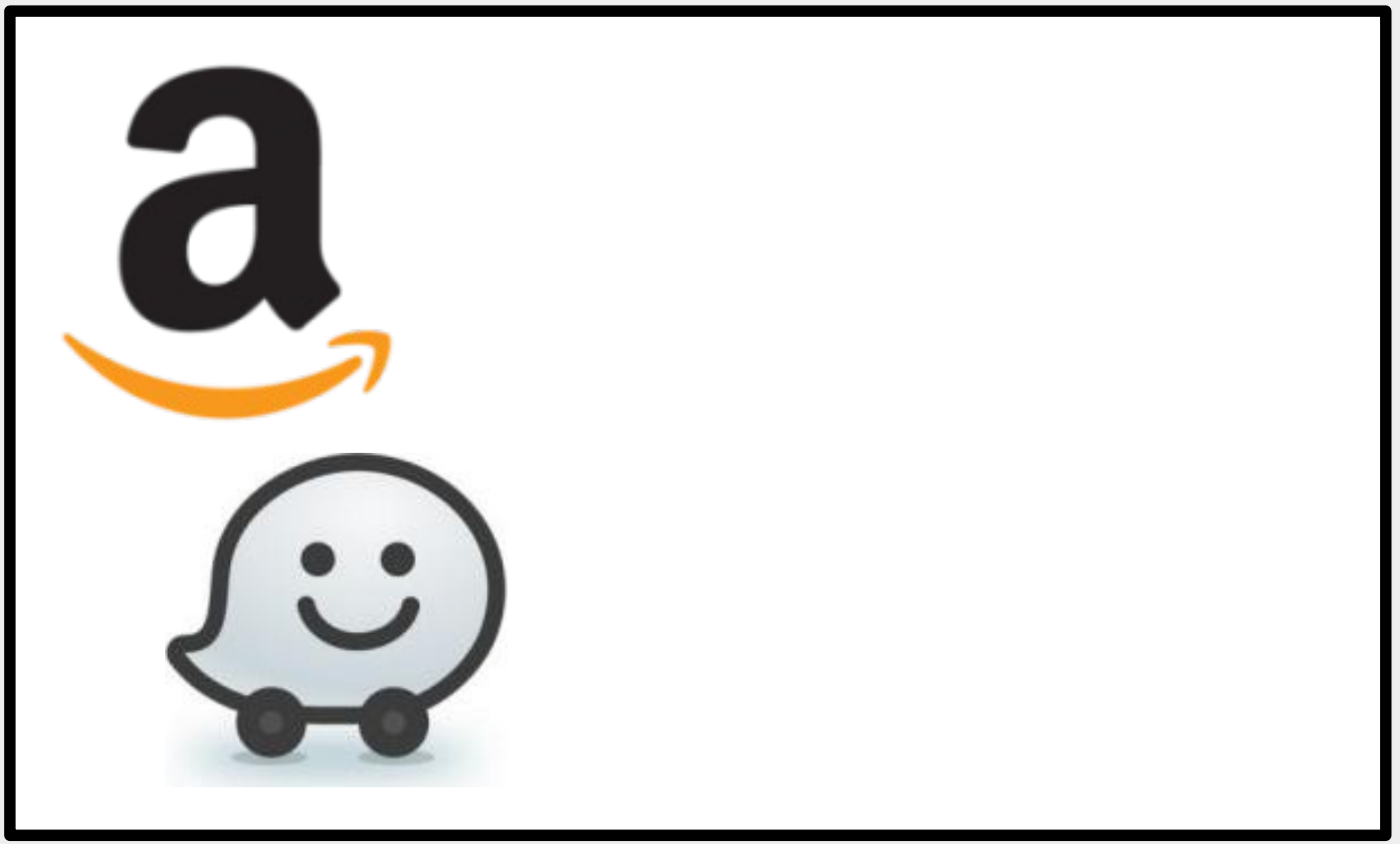
Machine learning has 'made it'

Good News

Machine learning has 'made it'

Good News

Machine learning has 'made it'



Good News

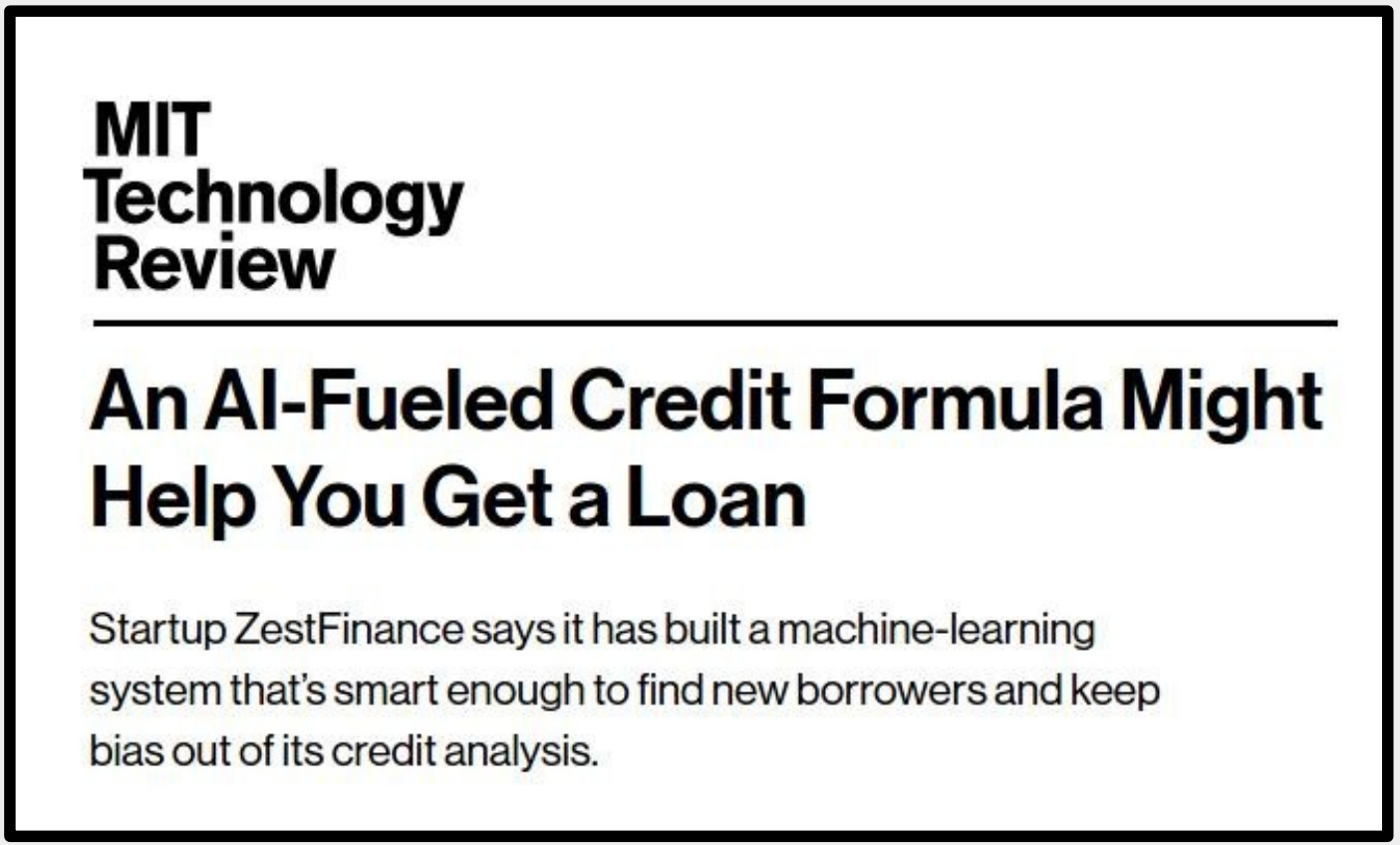
Machine learning has 'made it'

Good News

Machine learning has 'made it'

Good News

Machine learning has 'made it'



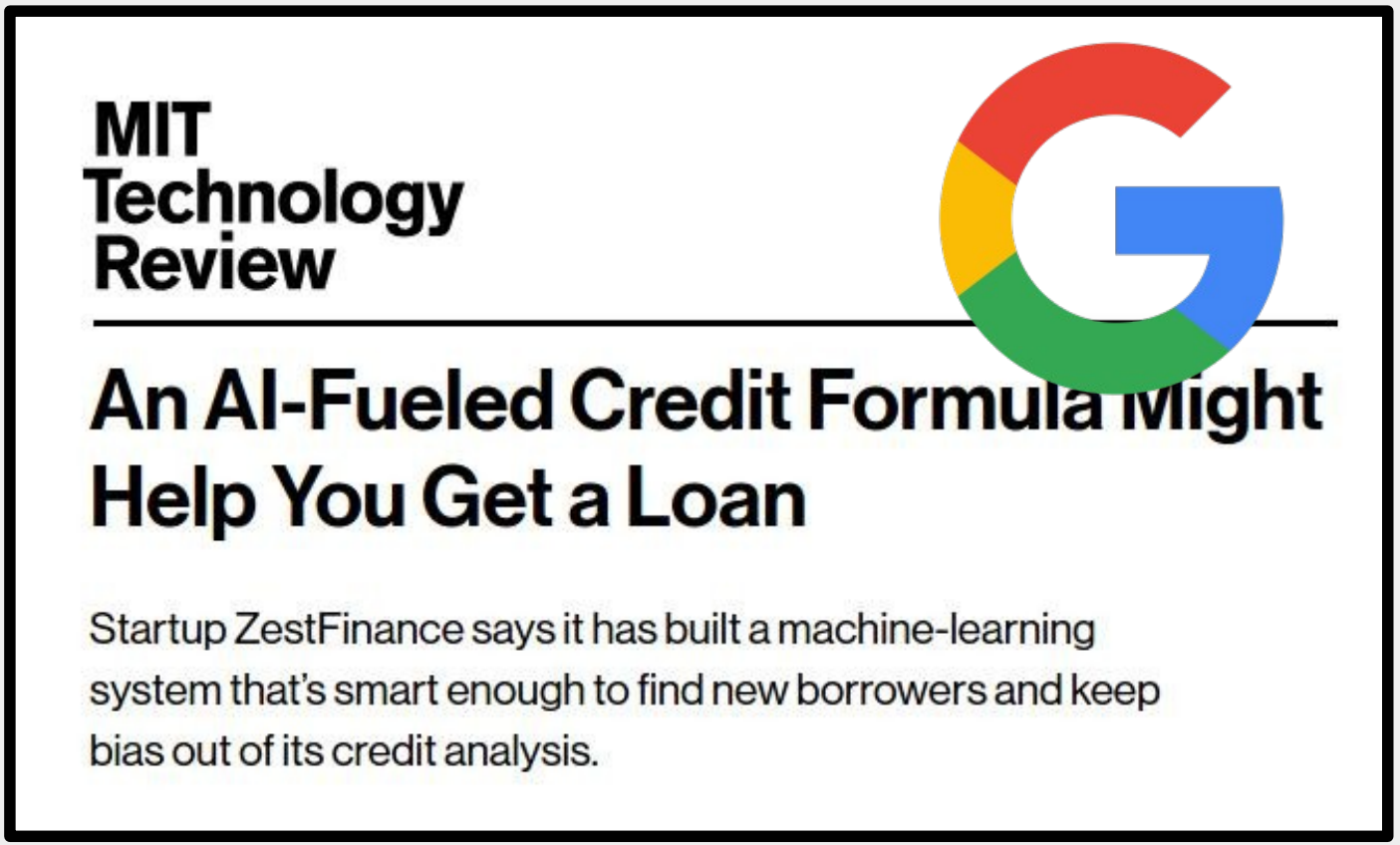
MIT Technology Review

An AI-Fueled Credit Formula Might Help You Get a Loan

Startup ZestFinance says it has built a machine-learning system that's smart enough to find new borrowers and keep bias out of its credit analysis.

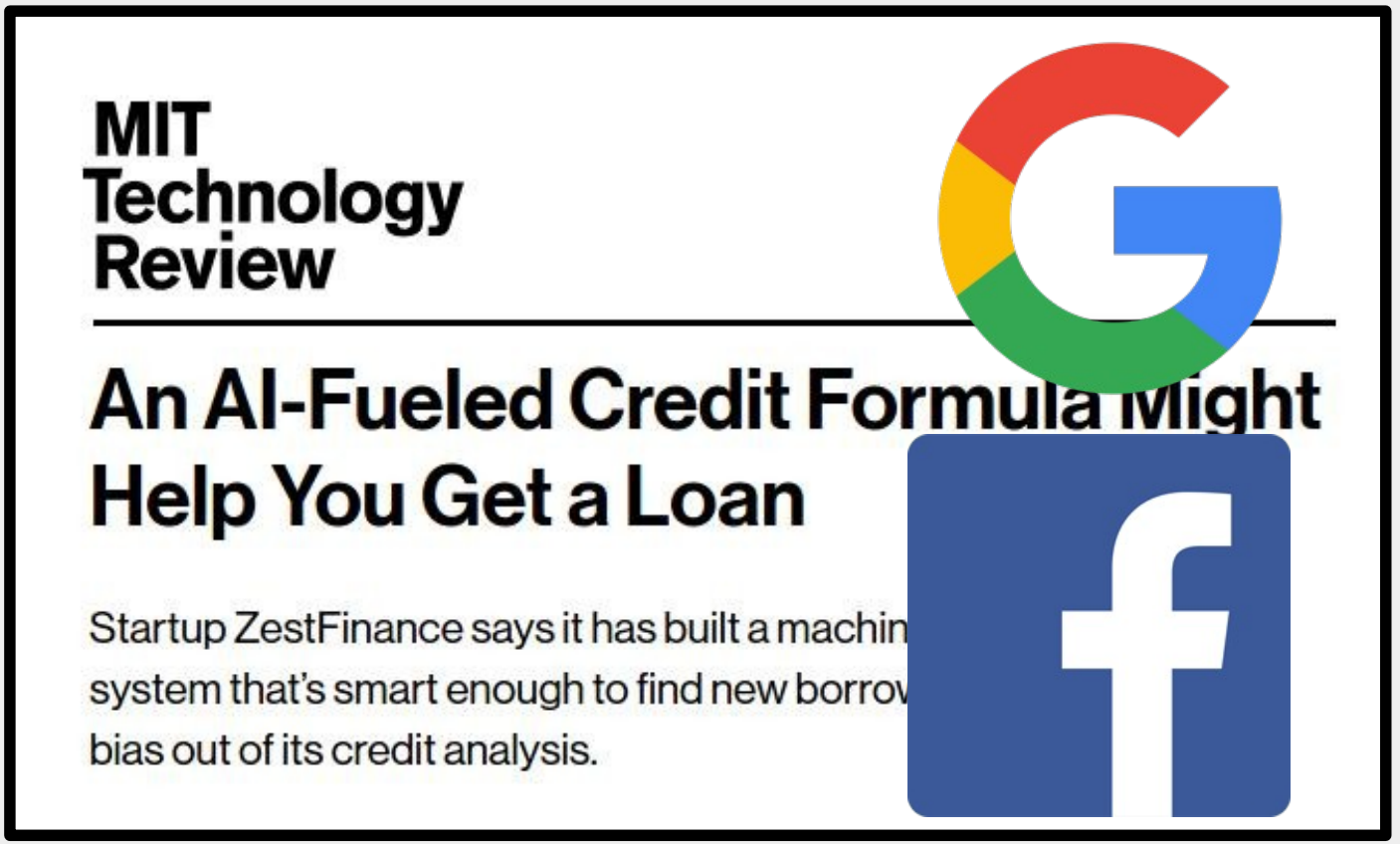
Good News

Machine learning has 'made it'



Good News

Machine learning has 'made it'



Not So Good News

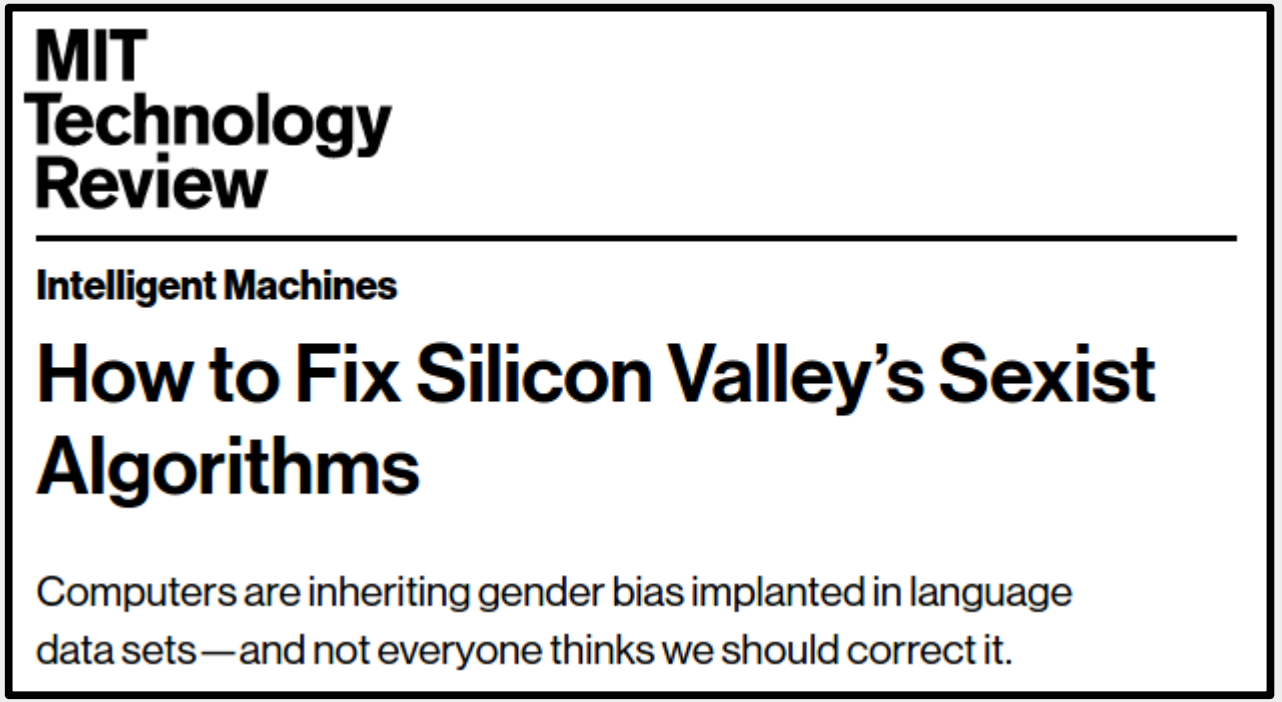
Not So Good News

Not So Good News

Responsibility lagging behind power

Not So Good News

Responsibility lagging behind power

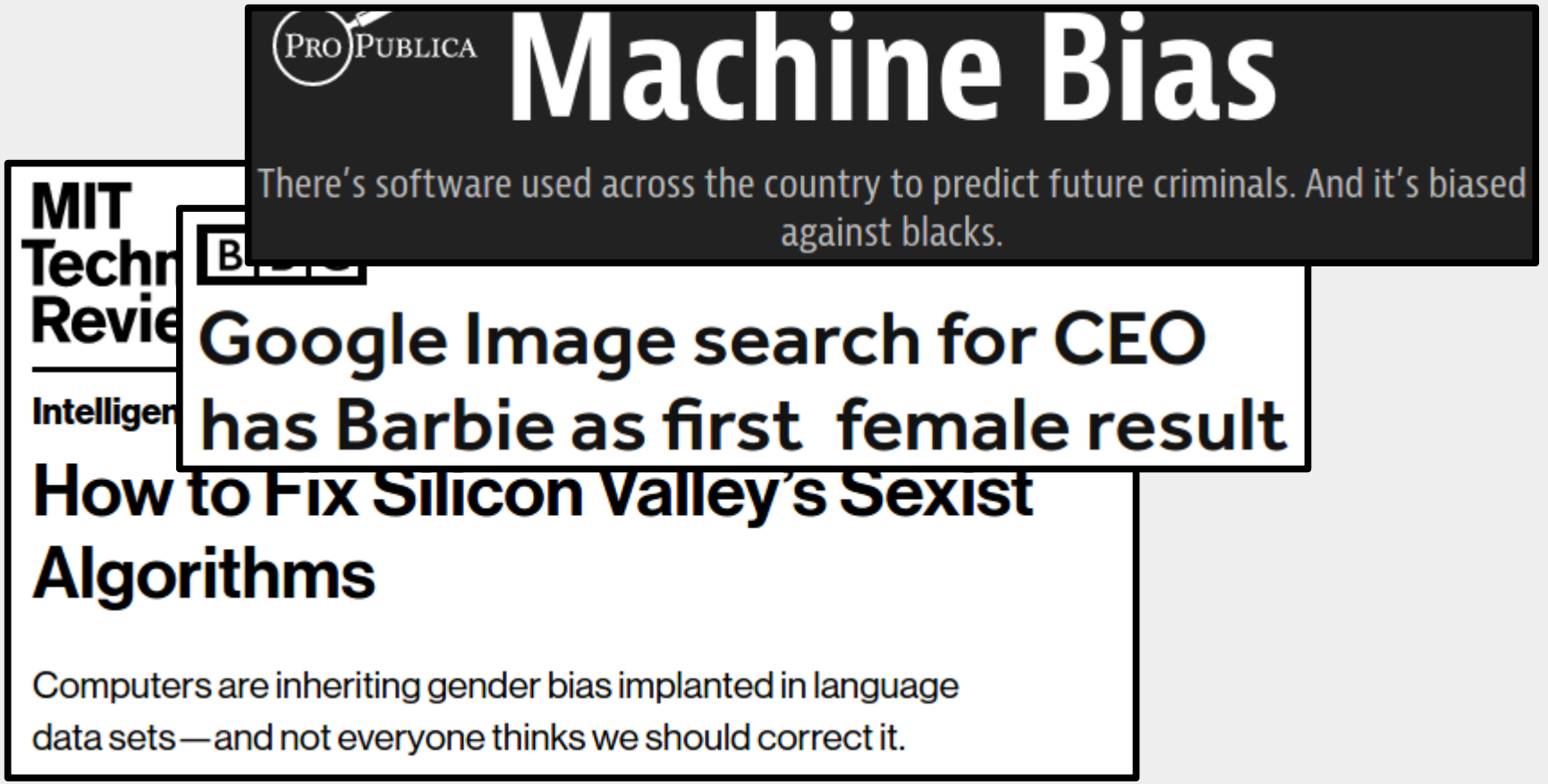


Not So Good News

Responsibility lagging behind power

Not So Good News

Responsibility lagging behind power



Not So Good News

Responsibility lagging behind power

A Dilemma

A Dilemma

Machine learning deserves wide use

A Dilemma

Machine learning deserves wide use

More use means higher stakes

A Dilemma

Machine learning deserves wide use

More use means higher stakes

How to balance?

Our Motivation

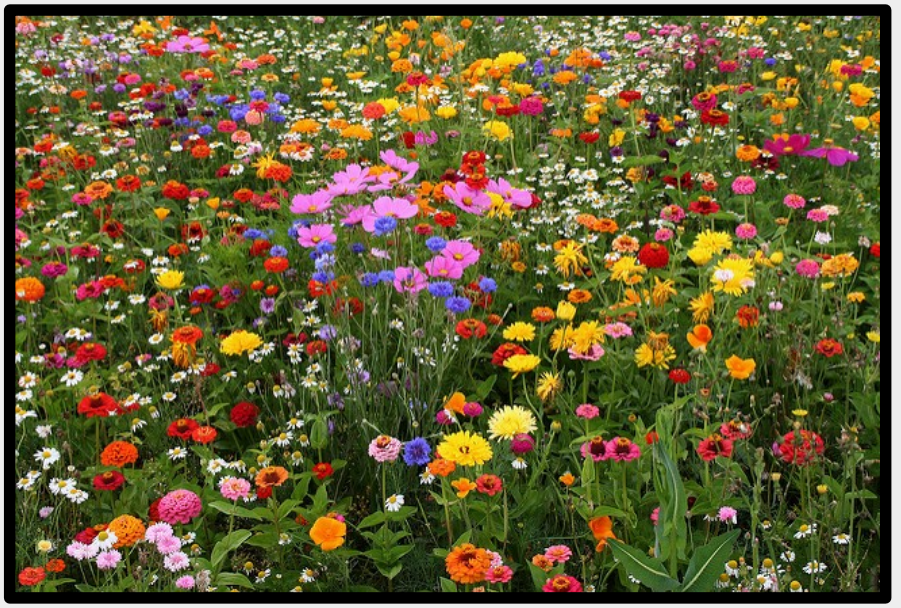
Our Motivation

Study how machine learning can be efficient, accurate, and fair

But what is 'fairness'?

But what is 'fairness'?

Many reasonable settings and definitions!



Outline For This Talk

Outline For This Talk

1. Specify a setting

Outline For This Talk

1. Specify a setting

2. Specify a fairness definition

Outline For This Talk

1. Specify a setting

2. Specify a fairness definition

3. See what guarantees we can make

Outline For This Talk

1. Specify a setting

2. Specify a fairness definition

3. See what guarantees we can make

A General Setting

A General Setting

Learning through feedback
from sequential choices

An Example

An Example

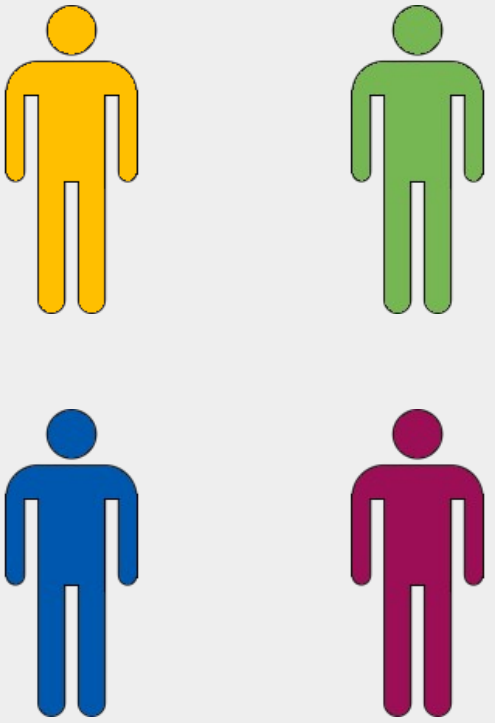
Learning how to grant loans

An Example

Learning how to grant loans

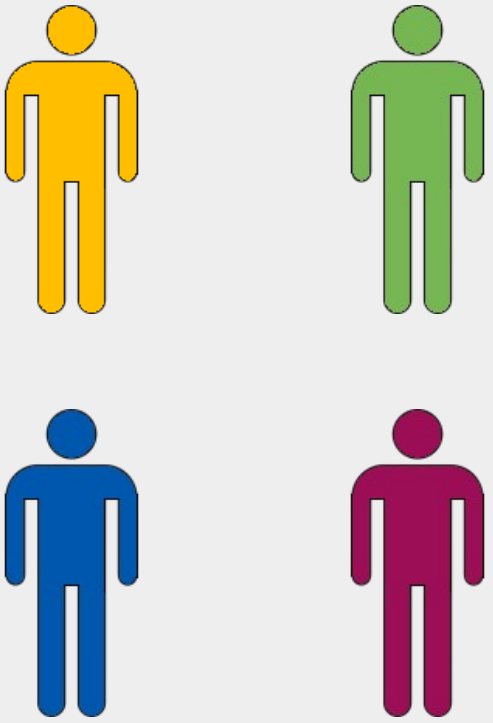
An Example

Learning how to grant loans



An Example

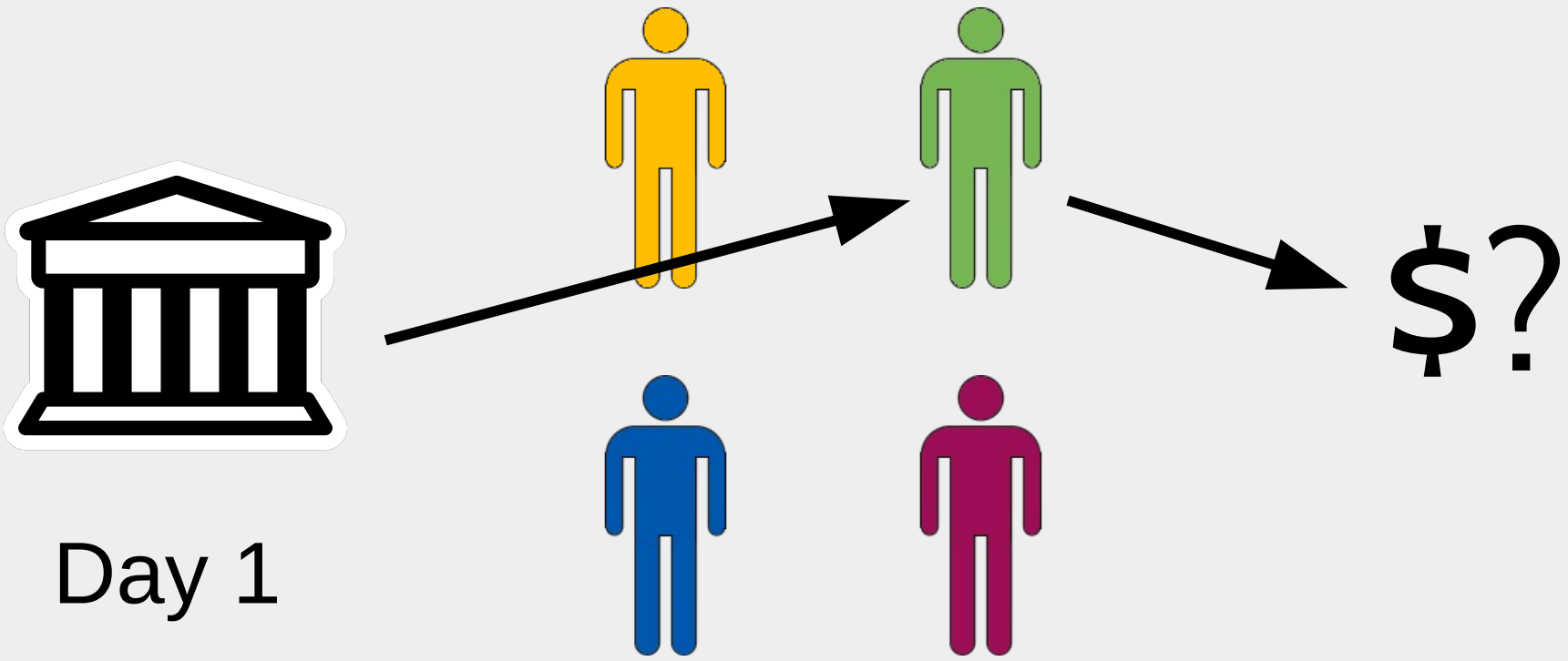
Learning how to grant loans



$?

An Example

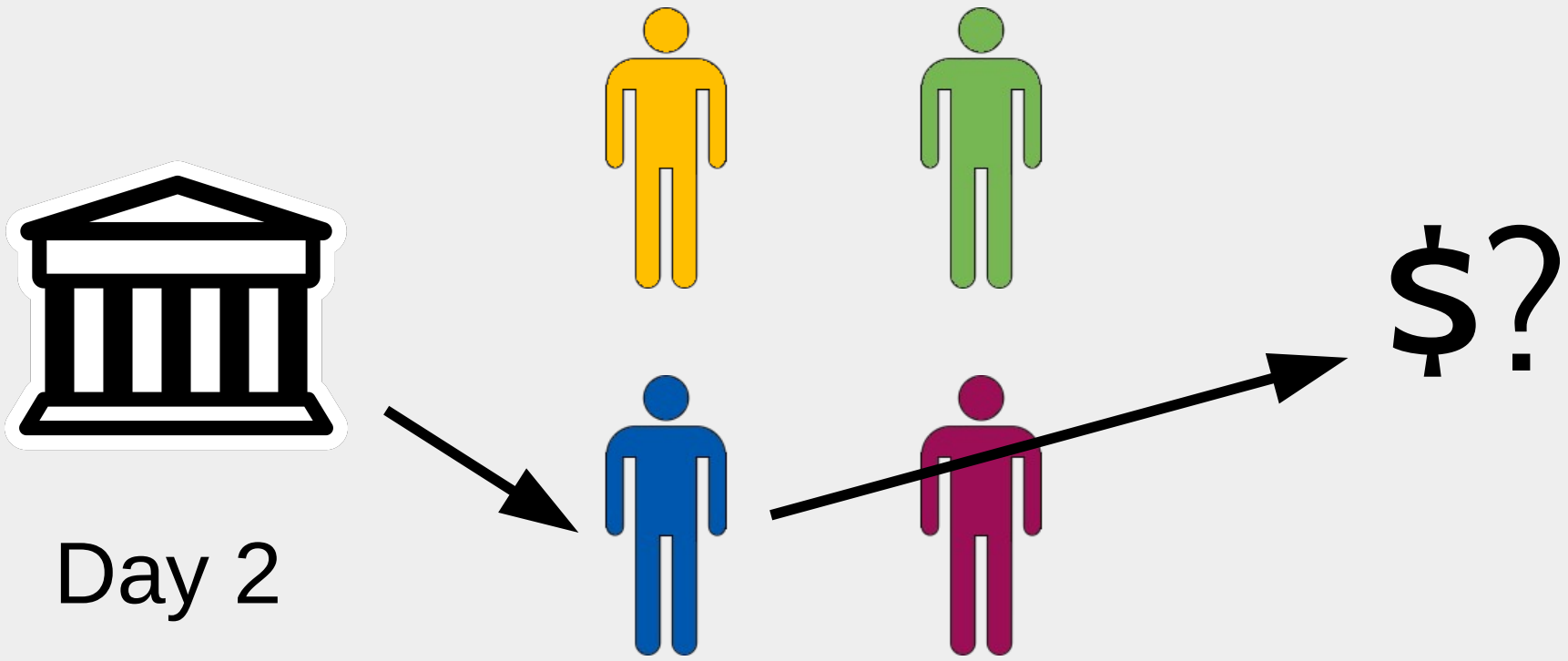
Learning how to grant loans



Day 1

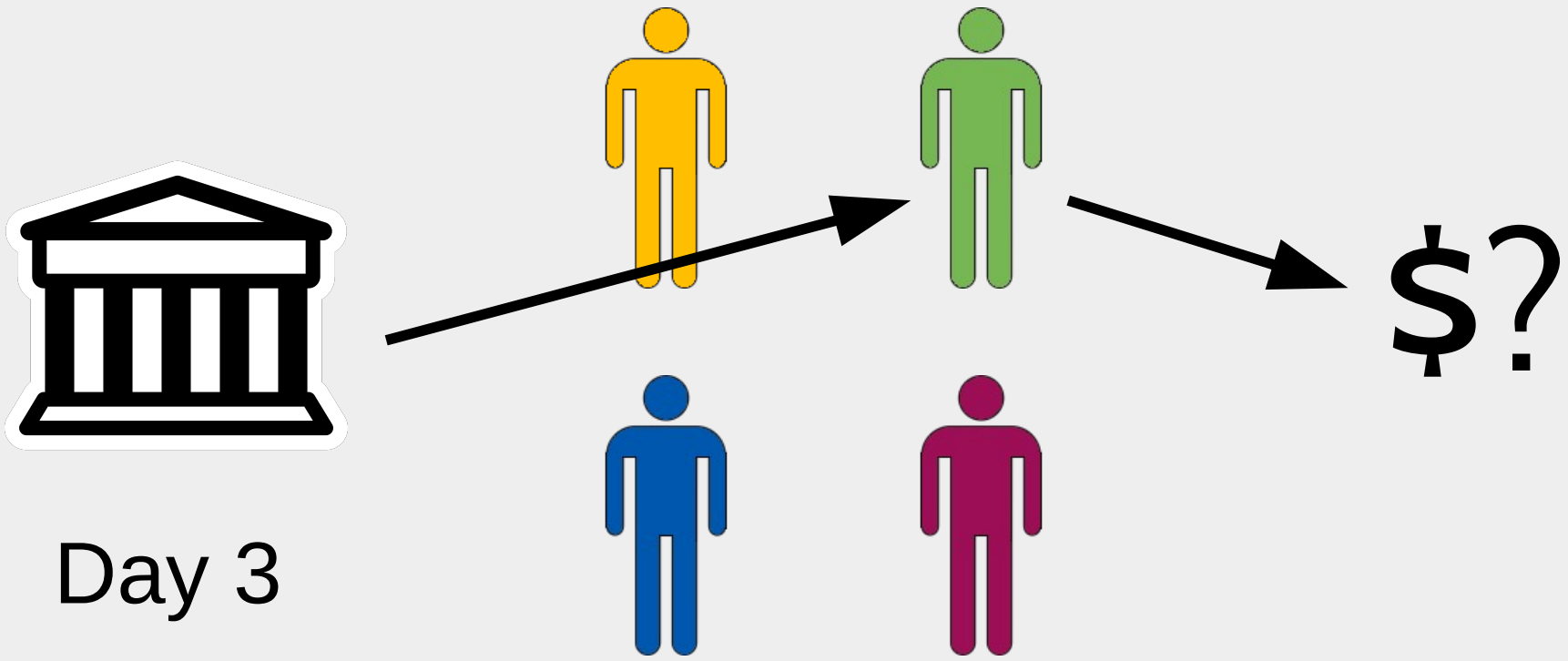
An Example

Learning how to grant loans



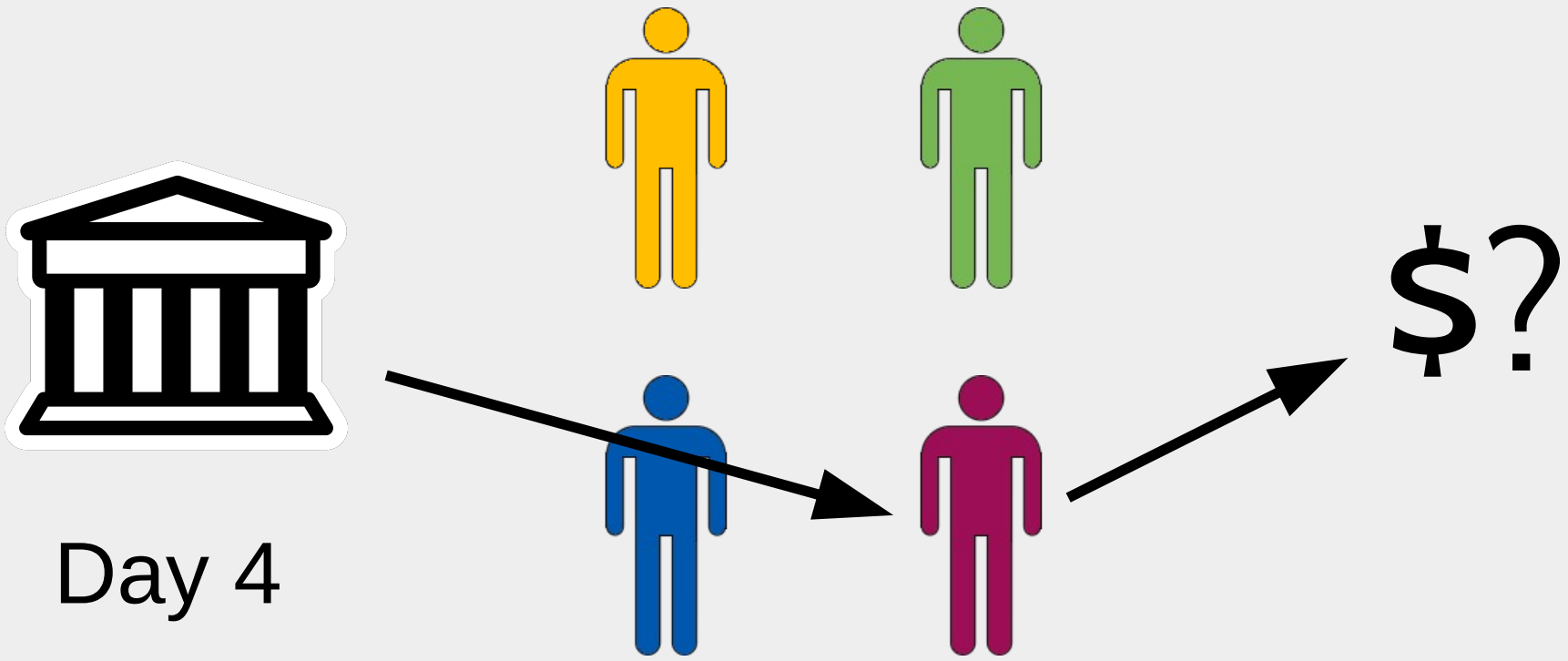
An Example

Learning how to grant loans



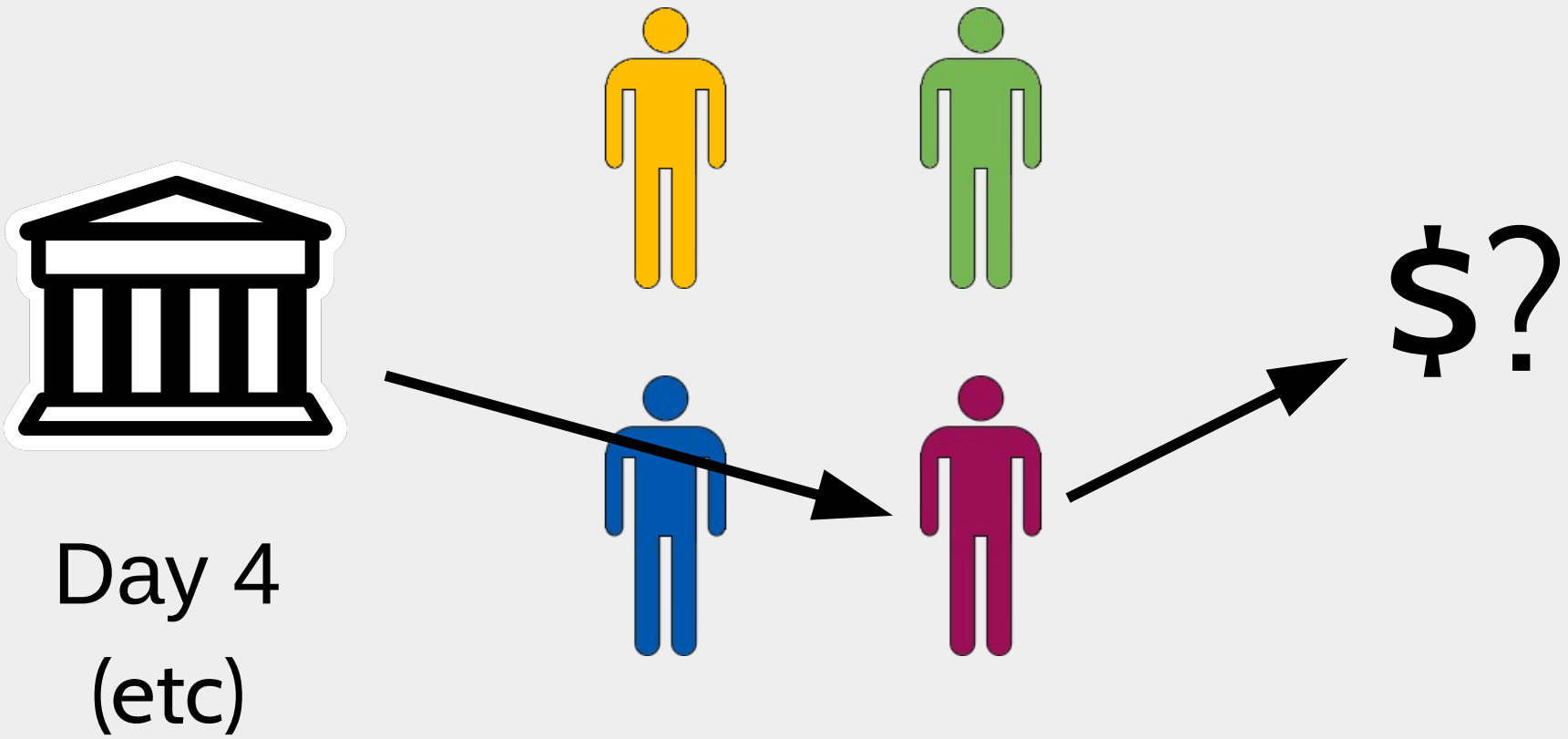
An Example

Learning how to grant loans



An Example

Learning how to grant loans



Previous work: bandits setting [JKMR16]

Previous work: bandits setting [JKMR16]

Fairness (TBD!) adds cost

Previous work: bandits setting [JKMR16]

Fairness (TBD!) adds cost (performance lower bound)...

Previous work: bandits setting [JKMR16]

Fairness (TBD!) adds cost (performance lower bound)...

...but cost often not too bad

Previous work: bandits setting [JKMR16]

Fairness (TBD!) adds cost (performance lower bound)...

...but cost often not too bad (performance upper bound)

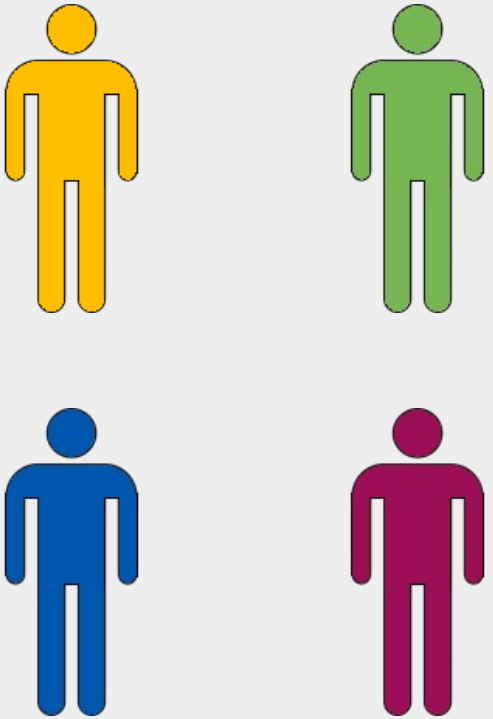
However...

However...

Bandits setting is limited – choices don't affect environment

However...

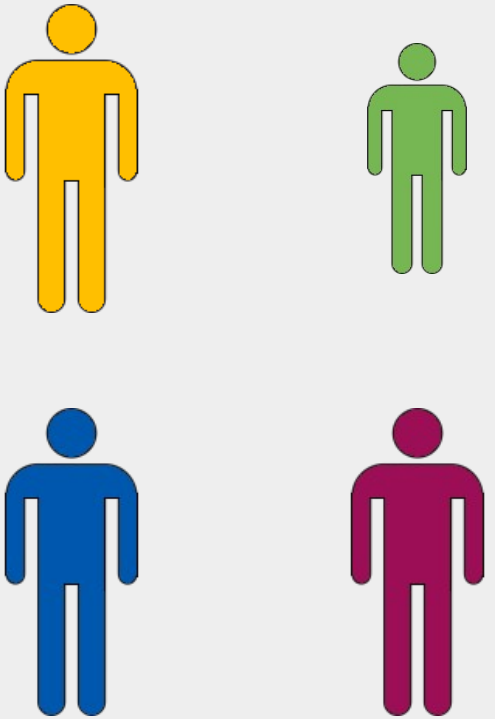
Bandits setting is limited – choices don't affect environment



$?

However...

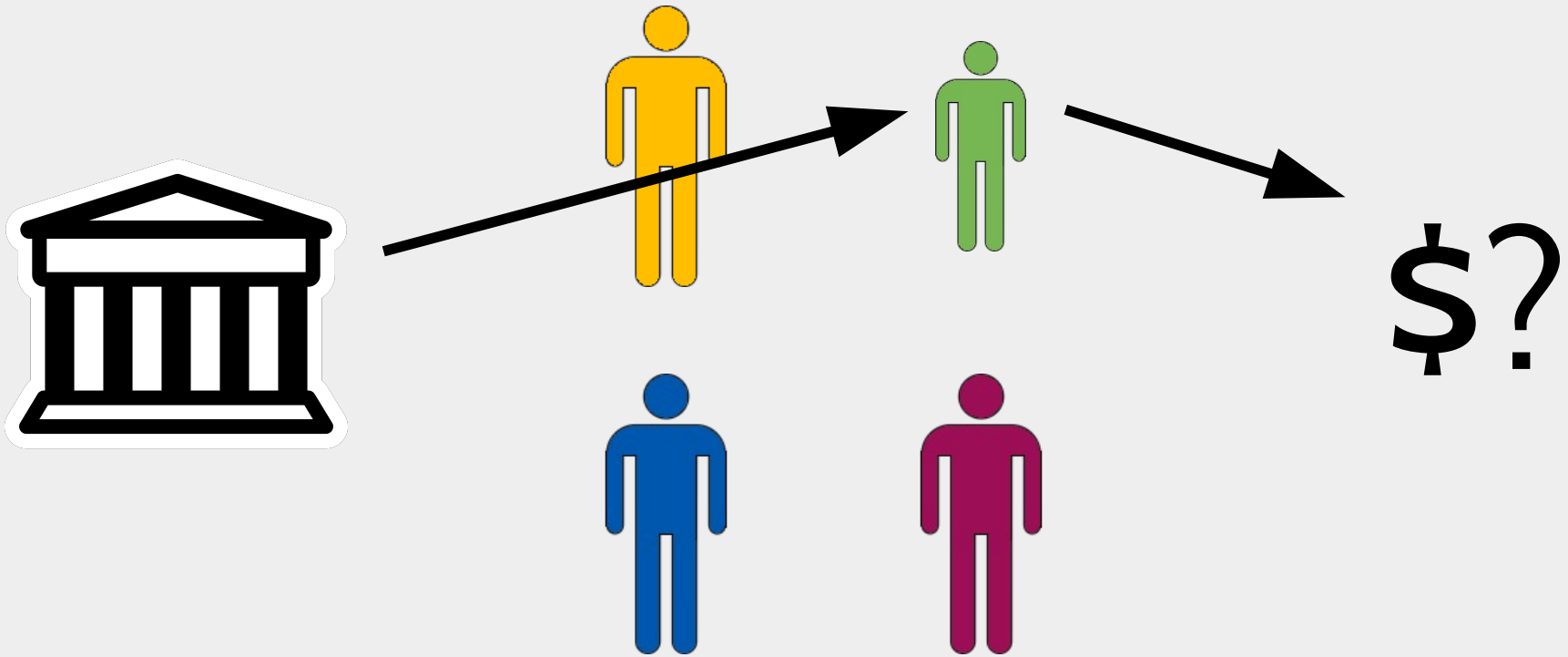
Bandits setting is limited – choices don't affect environment



$?

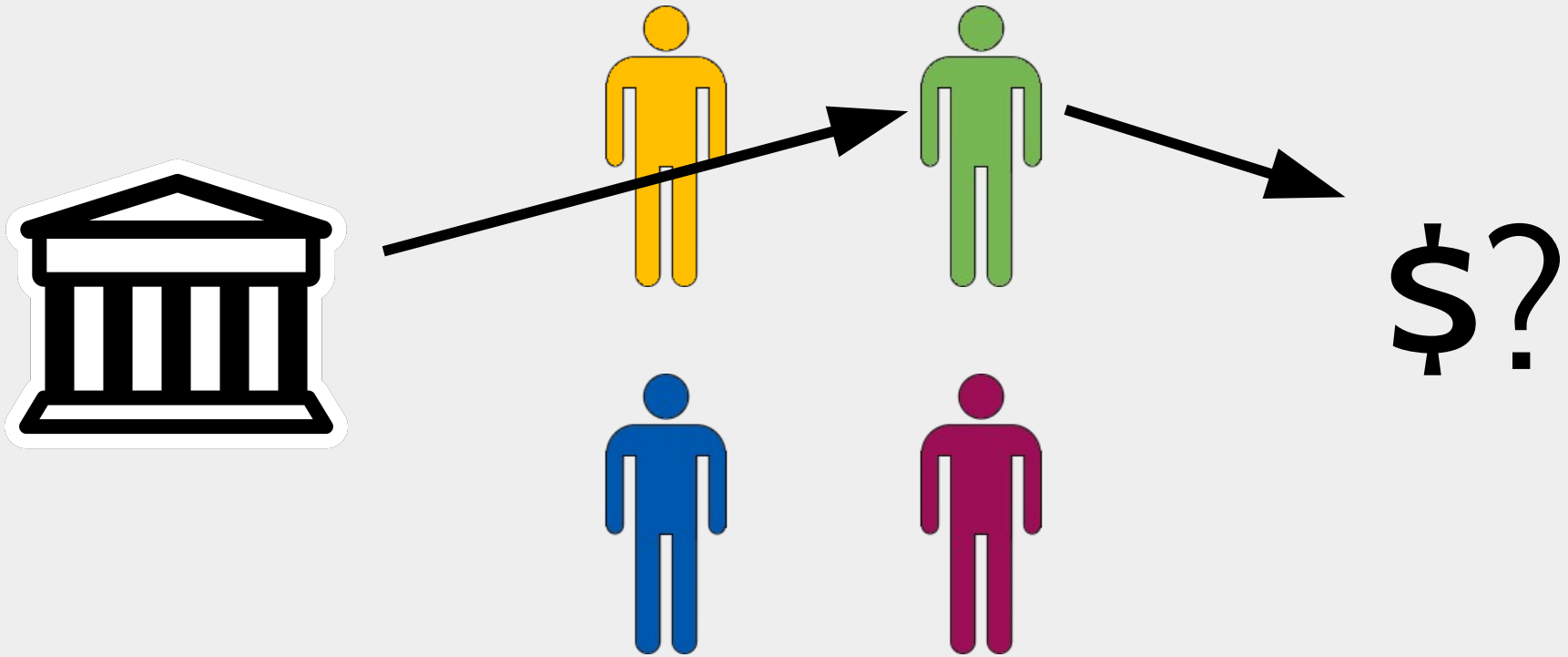
However...

Bandits setting is limited – choices don't affect environment



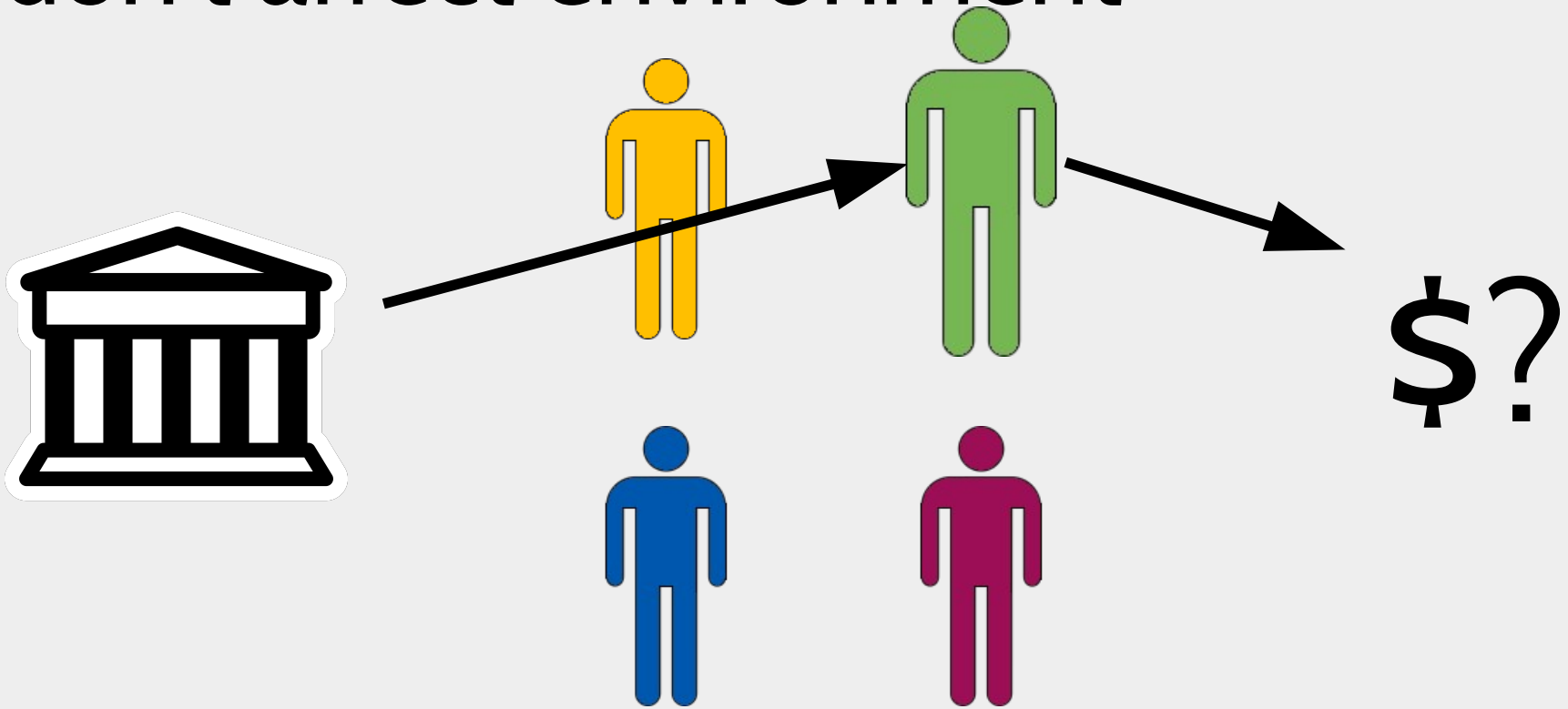
However...

Bandits setting is limited – choices don't affect environment



However...

Bandits setting is limited – choices don't affect environment



Our Specific Setting

Our Specific Setting

Want to reason about how learning choices affect environment

Our Specific Setting

Want to reason about how learning choices affect environment → reinforcement learning (MDPs)

Outline For This Talk

1. Specify a setting
   Reinforcement learning
2. Specify a fairness definition

3. See what guarantees we can make

Outline For This Talk

1. Specify a setting
   Reinforcement learning
2. Specify a fairness definition

3. See what guarantees we can make

A General Fairness Def.

A General Fairness Def.

Meritocratic fairness [JKMR16]:

A General Fairness Def.

Meritocratic fairness [JKMR16]:
(whp), in every choice learning algorithm makes, for options a and b

A General Fairness Def.

Meritocratic fairness [JKMR16]:
(whp), in every choice learning algorithm makes, for options a and b

$$\text{if quality}(a) \geq \text{quality}(b),$$

A General Fairness Def.

Meritocratic fairness [JKMR16]:
(whp), in every choice learning algorithm makes, for options a and b

if quality(a) ≥ quality(b),
then P(choose a) ≥ P(choose b)

In Our Example...

In Our Example...

 is fair if (whp) every day:

In Our Example...

 is fair if (whp) every day:

if quality() ≥ quality()

In Our Example...

 is fair if (whp) every day:

if quality( ) ≥ quality( )

then P(choose) ≥ P(choose)

In Our Example...

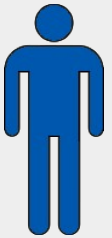
 is fair if (whp) every day:

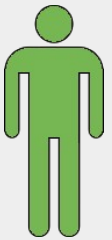
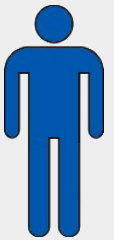
if quality( ) ≥ quality( )

then P(choose) ≥ P(choose)

In Our Example...

 is fair if (whp) every day:

if quality() ≥ quality()

then P(choose ) ≥ P(choose )

In Our Example...

 is fair if (whp) every day:

if quality( ) ≥ quality( )

then P(choose) ≥ P(choose)

Previous Fairness Def.

if quality(a) ≥ quality(b),
then P(choose a) ≥ P(choose b)

Previous Fairness Def.

if quality(a) ≥ quality(b),
then P(choose a) ≥ P(choose b)

Previous Fairness Def.

if quality(a) ≥ quality(b),
then P(choose a) ≥ P(choose b)

if reward(a) ≥ reward(b),

Previous Fairness Def.

if quality(a) ≥ quality(b),
then P(choose a) ≥ P(choose b)

if reward(a) ≥ reward(b),
then P(choose a) ≥ P(choose b)

Previous Fairness Def.

if quality(a) ≥ quality(b),
then P(choose a) ≥ P(choose b)

Short-term

if reward(a) ≥ reward(b),
then P(choose a) ≥ P(choose b)

Our Specific Fairness Def.

if quality(a) ≥ quality(b),
then P(choose a) ≥ P(choose b)

Our Specific Fairness Def.

if quality(a) ≥ quality(b),
then P(choose a) ≥ P(choose b)

Our Specific Fairness Def.

if quality(a) ≥ quality(b),
then P(choose a) ≥ P(choose b)

if $Q^*(s, a) \geq Q^*(s, b)$,
then $P(s, a) \geq P(s, b)$

Our Specific Fairness Def.

if quality(a) ≥ quality(b),
then P(choose a) ≥ P(choose b)

Long-term

if $Q^*(s, a) \geq Q^*(s, b)$,
then $P(s, a) \geq P(s, b)$

Limitations

Limitations

- Minimal – more “necessary” than “sufficient”

Limitations

- Minimal – more "necessary" than "sufficient"
- Assumes feedback reflects quality

Benefits

Benefits

- Minimal – more “necessary” than “sufficient”

Benefits

- Minimal – more "necessary" than "sufficient"
- Holds *throughout* learning process

Benefits

- Minimal – more "necessary" than "sufficient"
- Holds *throughout* learning process
- Aligned with optimality!

How to be fair?

How to be fair?

if quality(a) ≥ quality(b),
then P(choose a) ≥ P(choose b)

How to be fair?

Uniformly random exploration is fair

if quality(a) ≥ quality(b),
then P(choose a) ≥ P(choose b)

How to be fair?

(since P(choose a) = P(choose b))

Uniformly random exploration is fair

if quality(a) ≥ quality(b),
then P(choose a) ≥ P(choose b)

How to be fair?

Uniformly random exploration is fair

if quality(a) ≥ quality(b),
then P(choose a) ≥ P(choose b)

How to be fair?

Uniformly random exploration is fair

if quality(a) ≥ quality(b),
then P(choose a) ≥ P(choose b)

Optimal exploitation is fair

How to be fair?

Uniformly random exploration is fair

if quality(a) ≥ quality(b),
then P(choose a) ≥ P(choose b)

Optimal exploitation is fair

(since quality(best) ≥ quality(rest))

How to be fair?

Uniformly random exploration is fair

Optimal exploitation is fair

How to be fair?

Uniformly random exploration is fair

Need fair path between

Optimal exploitation is fair

Outline For This Talk

1. Specify a setting
   Reinforcement learning
2. Specify a fairness definition
   Meritocratic fairness
3. See what guarantees we can make

Outline For This Talk

1. Specify a setting
   Reinforcement learning
2. Specify a fairness definition
   Meritocratic fairness
3. See what guarantees we can make

Our Performance Metric

steps required before
average Q^*-value of visited states
is near-optimal

Without Fairness

Without Fairness

Near-optimality takes
poly(MDP parameters) steps

Without Fairness

Near-optimality takes
poly(MDP parameters) steps

What does fairness cost?

Lower Bound

Lower Bound

Theorem: No fair algorithm can guarantee near-optimality in under exponential(# states) steps.

Lower Bound Sketch

Lower Bound Sketch

if $Q^*(s, a) \geq Q^*(s, b)$,
then $P(s, a) \geq P(s, b)$

Lower Bound Sketch

if $Q^*(s, a) \geq Q^*(s, b)$,
then $P(s, a) \geq P(s, b)$

Fairness → must explore randomly to learn Q^* values...

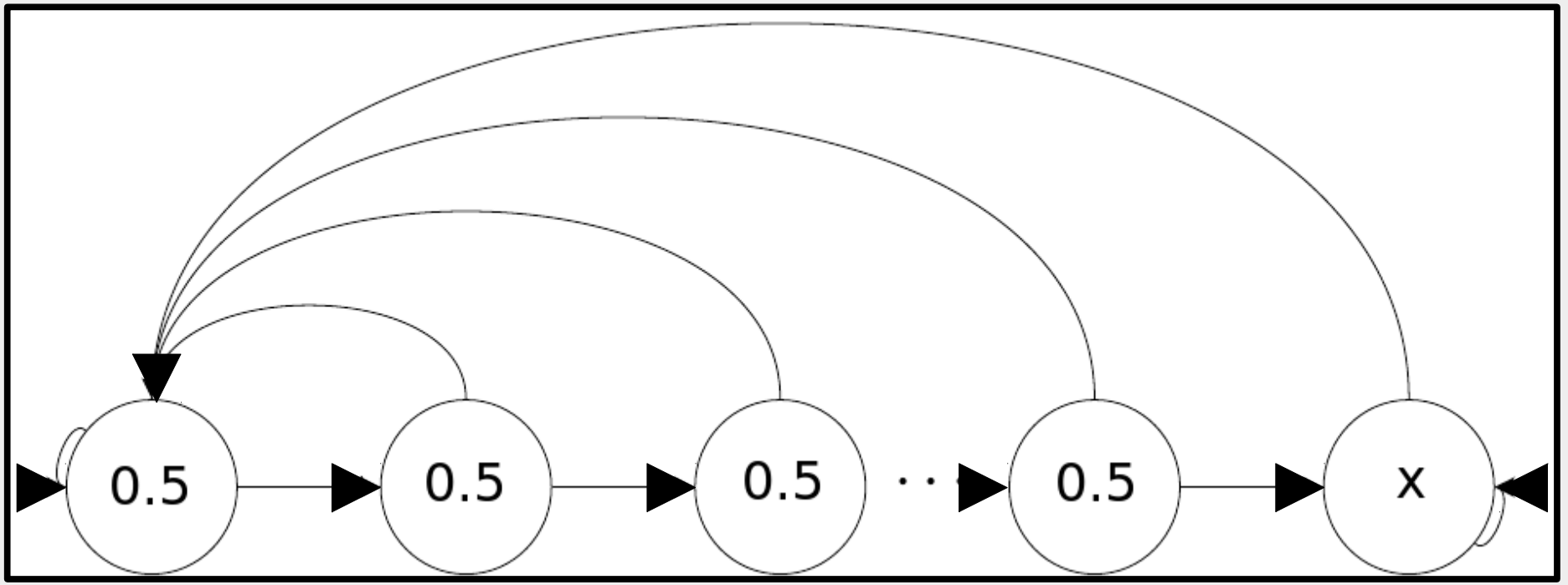
Lower Bound Sketch

if $Q^*(s, a) \geq Q^*(s, b)$,
then $P(s, a) \geq P(s, b)$

Fairness → must explore randomly to learn Q^* values...

...but sometimes random exploration does poorly

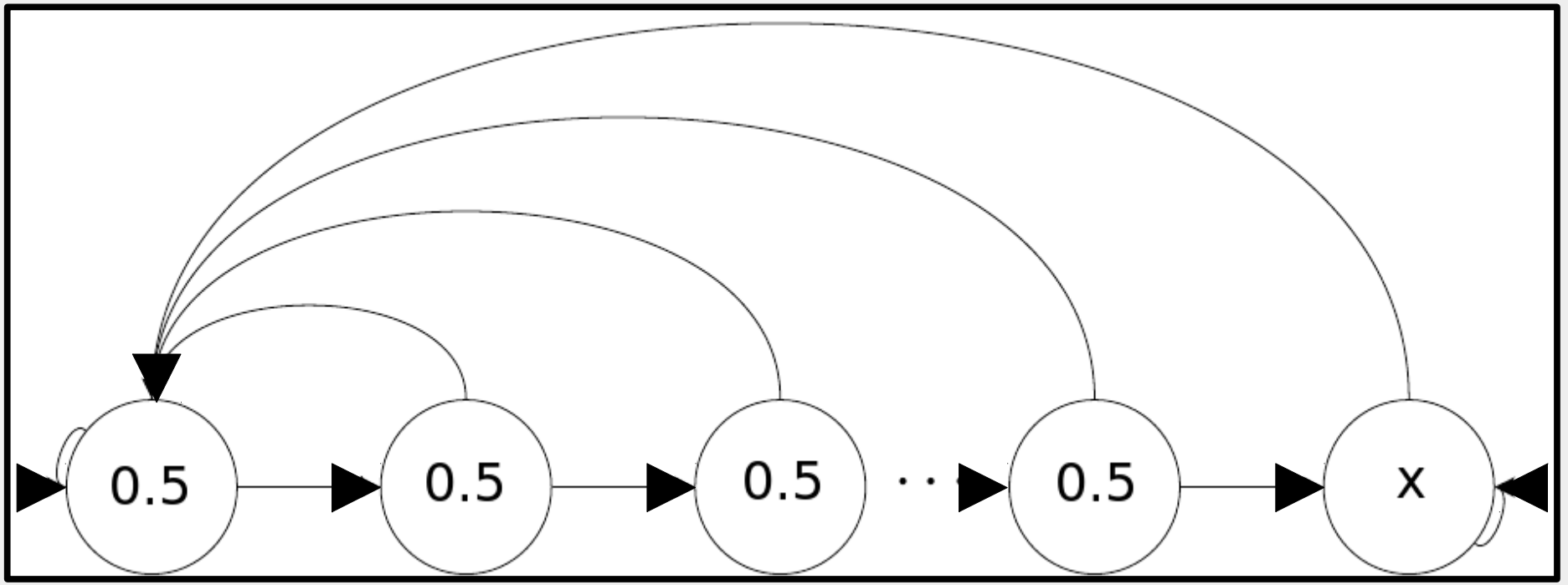
Lower Bound Sketch



Matthew Joseph, UPenn

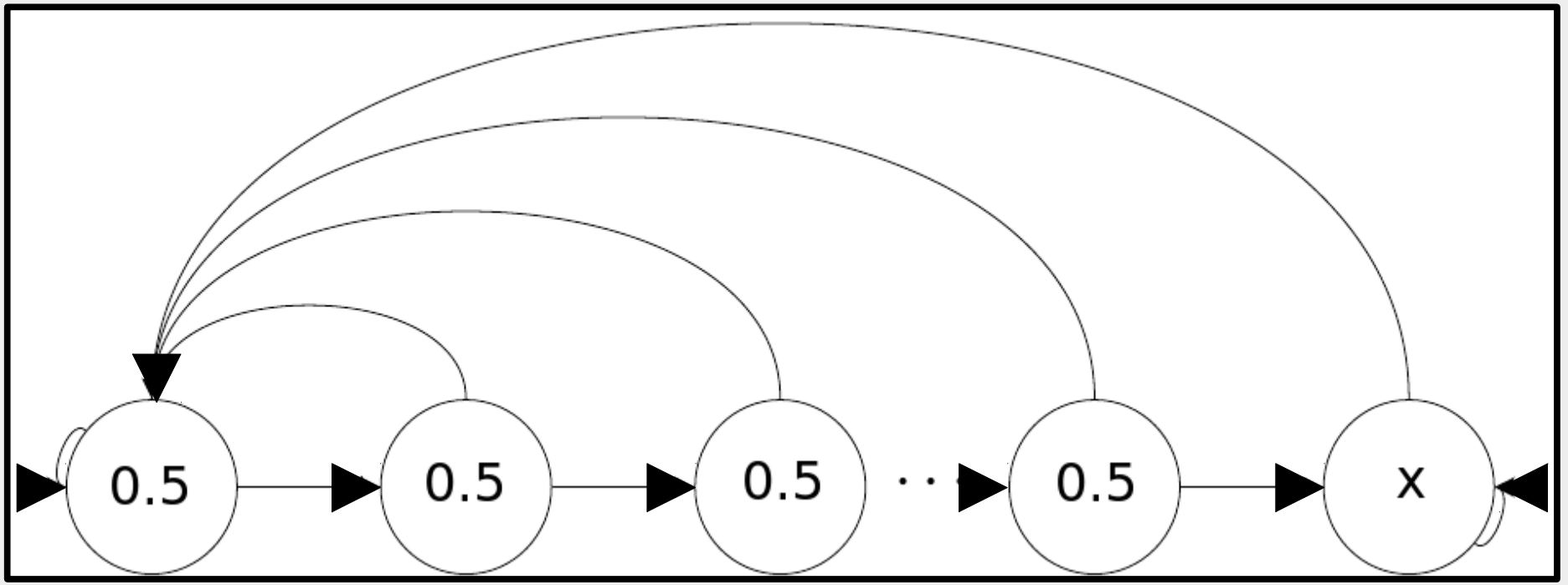
Lower Bound Sketch

“Combination lock” MDP



Lower Bound Sketch

"Combination lock" MDP



Exponential in # states

How to get around this?

How to get around this?

How to get around this?

Idea: relax to approximate fairness

Approximate Fairness

if $Q^*(s, a) \geq Q^*(s, b)$,

then $\mathcal{L}(s, a) \geq \mathcal{L}(s, b)$

Approximate Fairness

if $Q^*(s, a) \geq Q^*(s, b)$,

then $\mathcal{L}(s, a) \geq \mathcal{L}(s, b)$

Approximate Fairness

if $Q^*(s, a) \geq Q^*(s, b)$,
then $\mathcal{L}(s, a) \geq \mathcal{L}(s, b)$

if $Q^*(s, a) + \alpha \geq Q^*(s, b)$,
then $\mathcal{L}(s, a) \geq \mathcal{L}(s, b)$

Approximate Fairness

if $Q^*(s, a) \geq Q^*(s, b)$,

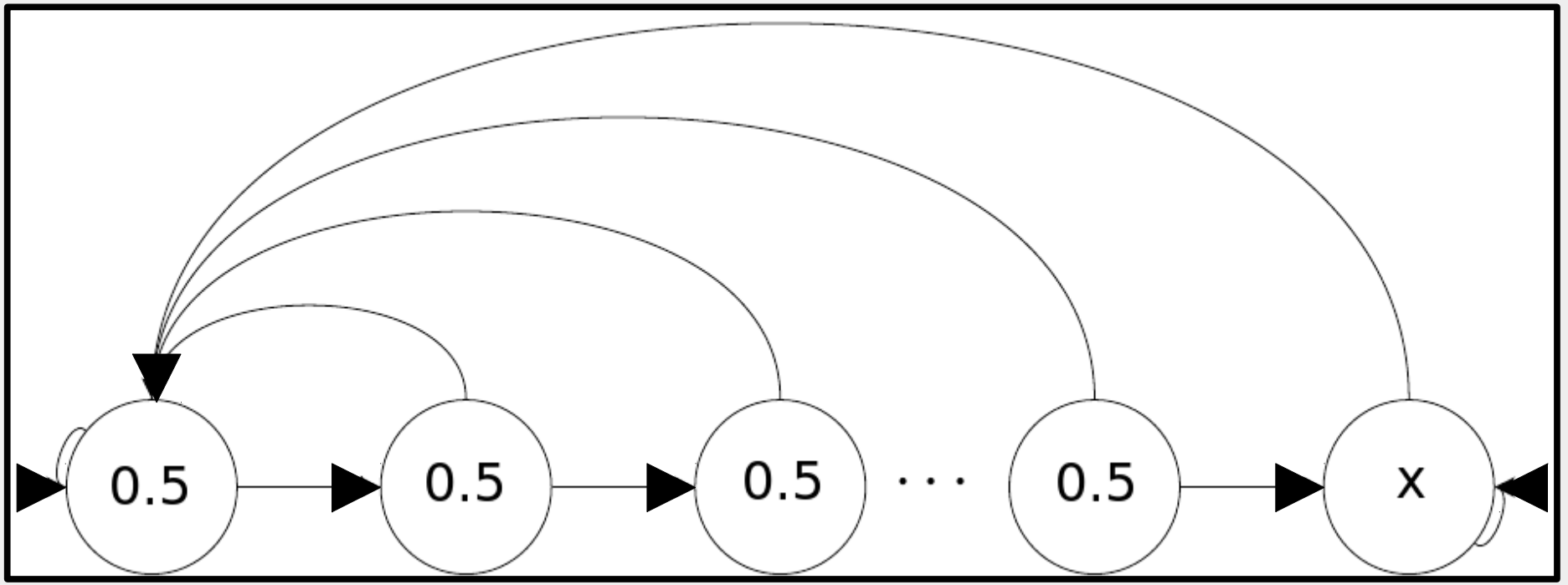
then $\mathcal{L}(s, a) \geq \mathcal{L}(s, b)$

Approximate "action" fairness

if $Q^*(s, a) + \alpha \geq Q^*(s, b)$,

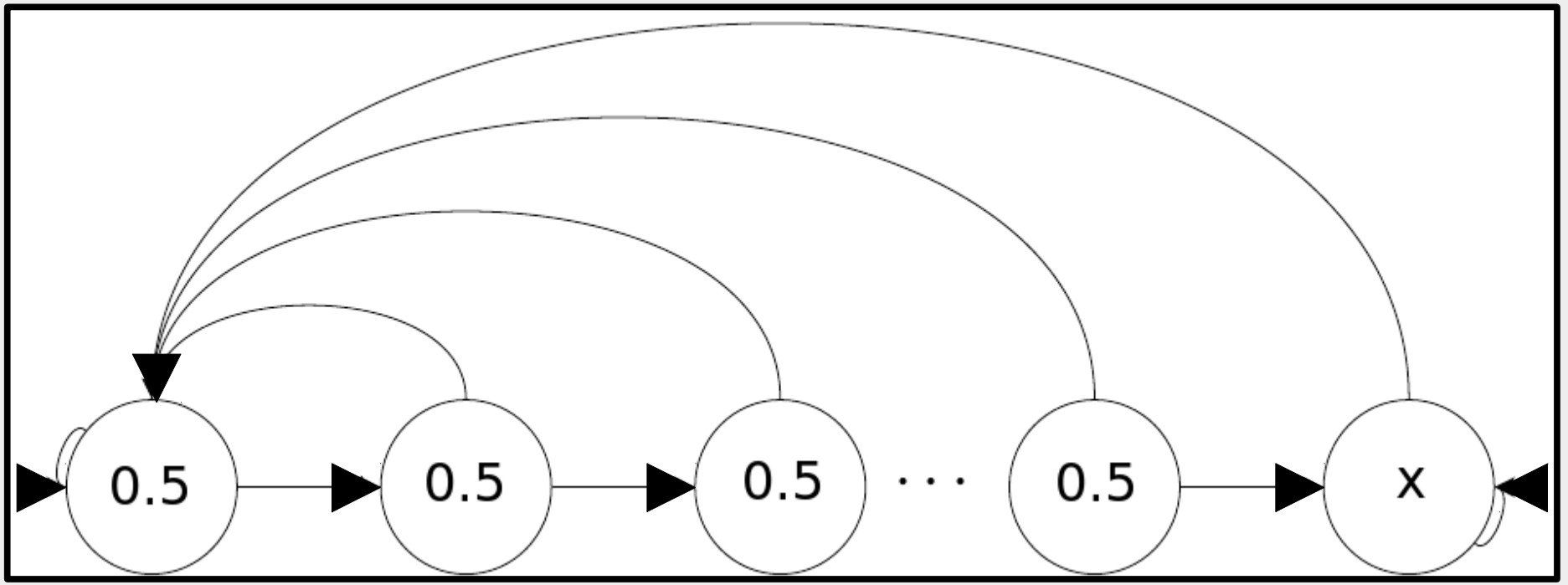
then $\mathcal{L}(s, a) \geq \mathcal{L}(s, b)$

Better!



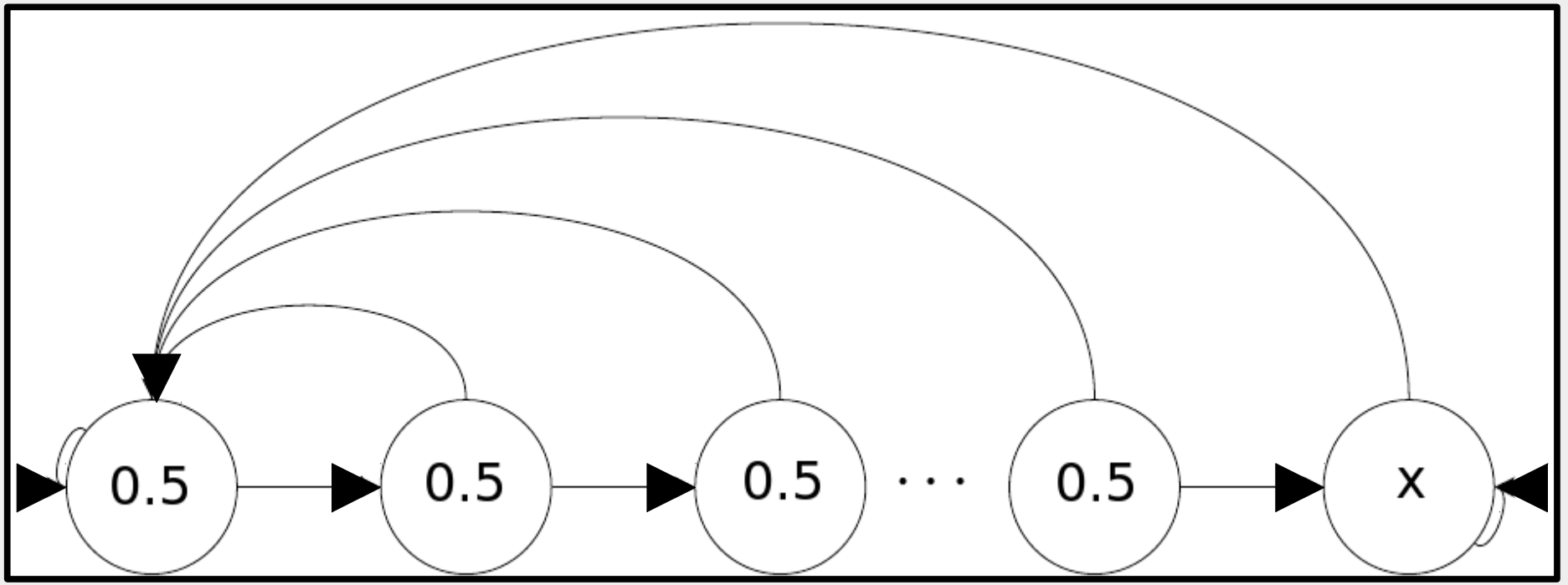
Better!

No longer exponential in # states



Better!

No longer exponential in # states



(Still) exponential in $1/(1-\gamma)$

An Algorithm: Fair-E^3

An Algorithm: Fair-E^3

Start from E^3 [KS98]

An Algorithm: Fair-E^3

Start from E^3 [KS98]

Adapt to satisfy approximate-action fairness

Fair-E^3: Sketch

Fair-E^3: Sketch

Organize world into “known” and “unknown” states

Fair-E^3: Sketch

Organize world into "known" and "unknown" states

"Known": good estimates of transitions, rewards, Q^* values . . .

Fair-E^3: Sketch

Fair-E^3: Sketch

In unknown state: take random walk
→ state more known

Fair-E^3: Sketch

Fair-E^3: Sketch

In known state:

Fair-E^3: Sketch

In known state:

Either *fairly* exploit in known states for good reward

Fair-E^3: Sketch

In known state:

Either *fairly* exploit in known states for good reward

Or *fairly* explore to unknown quickly

$E^3 \rightarrow$ Fair-E^3

$E^3 \rightarrow$ Fair-E^3

Approximate fairness makes everything trickier

$E^3 \rightarrow$ Fair-E^3

Approximate fairness makes everything trickier

"Known" must be stronger

$E^3 \rightarrow$ Fair-E^3

Approximate fairness makes everything trickier

"Known" must be stronger

Computing fair policies more delicate

Upper Bound

Upper Bound

Theorem: Fair-E^3 is approximate action fair and near-optimal in poly(all MDP parameters but γ), exp(1/1-γ) steps

Collected Results

Collected Results

Without fairness: near-optimality in poly(MDP parameters)

Collected Results

Without fairness: near-optimality in poly(MDP parameters)

With fairness: exp(# states)

Collected Results

Without fairness: near-optimality in poly(MDP parameters)

With fairness:
exp(# states)

With approximate fairness:
exp(discount factor)

Summing Up

Summing Up

- Fair ML matters!

Summing Up

- Fair ML matters!
- Proposed meritocratic fairness, studied in RL setting

Summing Up

- Fair ML matters!
- Proposed meritocratic fairness, studied in RL setting
- Proved separations between "unfair", fair, and approximately fair RL

Summing Up

- Fair ML matters!
- Proposed meritocratic fairness, studied in RL setting
- Proved separations between "unfair", fair, and approximately fair RL
- Can we do better?

Summing Up

- Fair ML matters!
- Proposed meritocratic fairness, studied in RL setting
- Proved separations between "unfair", fair, and approximately fair RL
- Can we do better?

Thanks!

Paper: arxiv.org/abs/1611.03071

References

[JKMR16] "Fairness in Learning: Classic and Contextual Bandits"
Matthew Joseph, Michael Kearns, Jamie Morgenstern, and Aaron Roth.
NIPS 2016

[KS02] "Near-Optimal Reinforcement Learning in Polynomial Time"
Michael Kearns and Satinder Singh.
ICML 1998