Impossibilities in Computing

Lecture 2

Unit 2

ML and Society (Spring 2024)

Atri Rudra

Pass Phrase for today: William Isaac

WILLIAM ISAAC

COMPUTATIONAL SOCIAL SCIENTIST

IN DETAIL

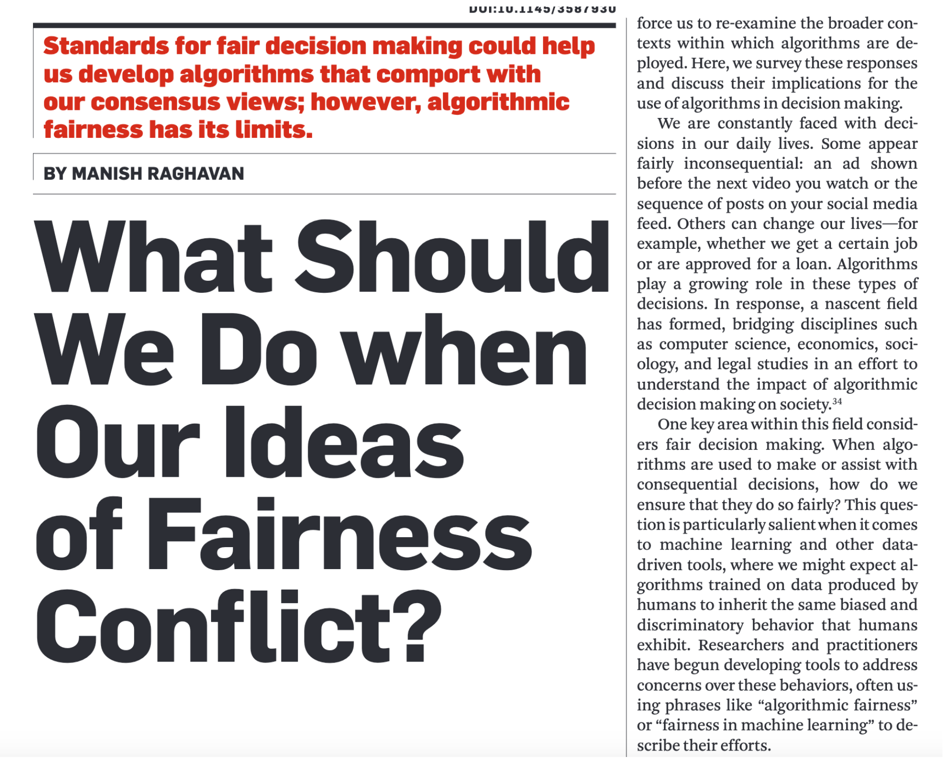
To predict and serve?

Predictive policing systems are used increasingly by law enforcement to try to prevent crime before it occurs. But what happens when these systems are trained using biased data? **Kristian Lum** and **William Isaac** consider the evidence – and the social consequences

Checking In

Who has started on the unit 2 group submission?

This paper will be crucial for Unit 2 mid-point submission:


DOI:10.1145/3587930

Standards for fair decision making could help us develop algorithms that comport with our consensus views; however, algorithmic fairness has its limits.

BY MANISH RAGHAVAN

What Should We Do when Our Ideas of Fairness Conflict?

force us to re-examine the broader contexts within which algorithms are deployed. Here, we survey these responses and discuss their implications for the use of algorithms in decision making.

We are constantly faced with decisions in our daily lives. Some appear fairly inconsequential: an ad shown before the next video you watch or the sequence of posts on your social media feed. Others can change our lives—for example, whether we get a certain job or are approved for a loan. Algorithms play a growing role in these types of decisions. In response, a nascent field has formed, bridging disciplines such as computer science, economics, sociology, and legal studies in an effort to understand the impact of algorithmic decision making on society.[34]

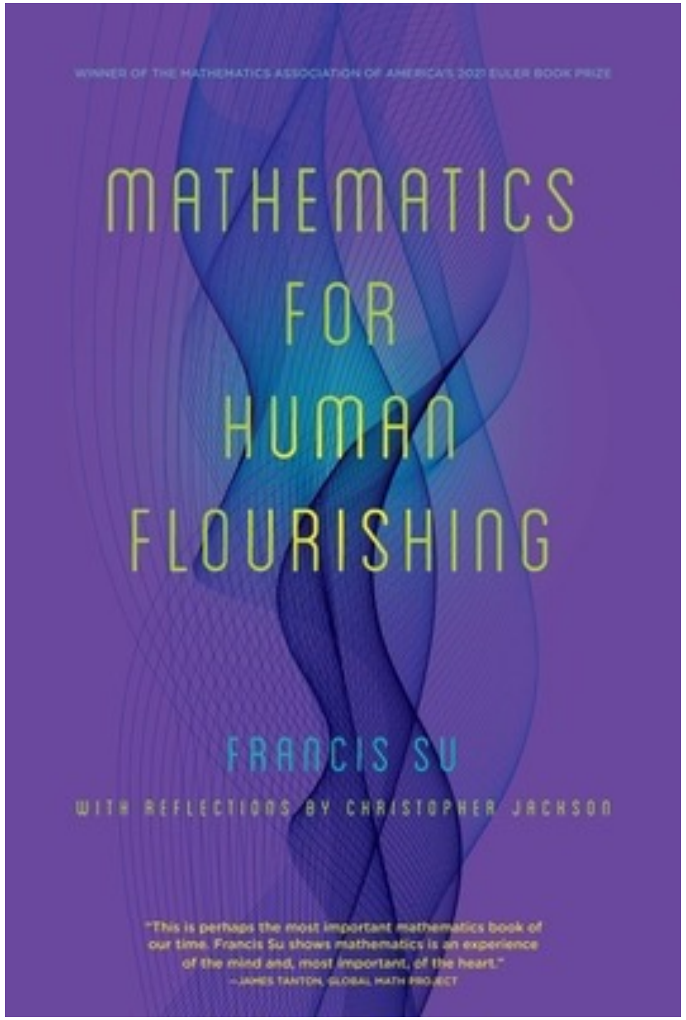
One key area within this field considers fair decision making. When algorithms are used to make or assist with consequential decisions, how do we ensure that they do so fairly? This question is particularly salient when it comes to machine learning and other data-driven tools, where we might expect algorithms trained on data produced by humans to inherit the same biased and discriminatory behavior that humans exhibit. Researchers and practitioners have begun developing tools to address concerns over these behaviors, often using phrases like "algorithmic fairness" or "fairness in machine learning" to describe their efforts.

Readings this week are a bit disjointed


IN DETAIL

To predict and serve?

Predictive policing systems are used increasingly by law enforcement to try to prevent crime before it occurs. But what happens when these systems are trained using biased data? **Kristian Lum** and **William Isaac** consider the evidence – and the social consequences



Reminder

What do you think when you hear impossible?

Discuss!

In the context of computational problems, what does an impossible problem (that is defined *mathematically*) mean to you?

First interpretation

Essentially not possible to come up with a precise mathematical description of a problem

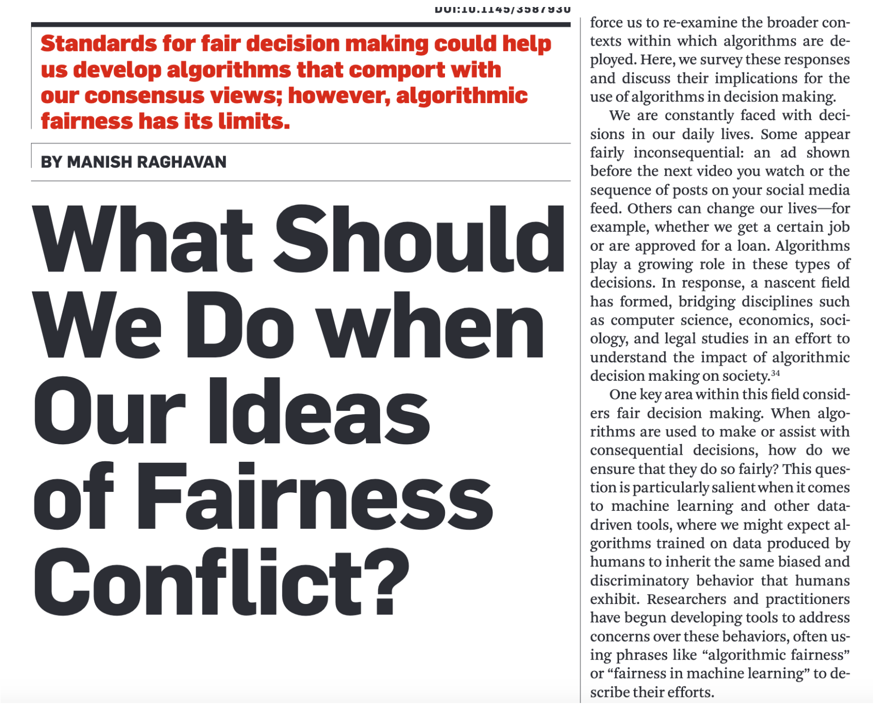
At least not in the sense of being able to write the math formulation down

Try to learn the problem from data itself!

Second interpretation

It is possible to precisely define the problem but there does not exist *any* solution

Solve an approximate version of the impossible problem


DOI:10.1145/3587930

Standards for fair decision making could help us develop algorithms that comport with our consensus views; however, algorithmic fairness has its limits.

BY MANISH RAGHAVAN

What Should We Do when Our Ideas of Fairness Conflict?

force us to re-examine the broader contexts within which algorithms are deployed. Here, we survey these responses and discuss their implications for the use of algorithms in decision making.

We are constantly faced with decisions in our daily lives. Some appear fairly inconsequential: an ad shown before the next video you watch or the sequence of posts on your social media feed. Others can change our lives—for example, whether we get a certain job or are approved for a loan. Algorithms play a growing role in these types of decisions. In response, a nascent field has formed, bridging disciplines such as computer science, economics, sociology, and legal studies in an effort to understand the impact of algorithmic decision making on society.[34]

One key area within this field considers fair decision making. When algorithms are used to make or assist with consequential decisions, how do we ensure that they do so fairly? This question is particularly salient when it comes to machine learning and other data-driven tools, where we might expect algorithms trained on data produced by humans to inherit the same biased and discriminatory behavior that humans exhibit. Researchers and practitioners have begun developing tools to address concerns over these behaviors, often using phrases like "algorithmic fairness" or "fairness in machine learning" to describe their efforts.

Case 3.1

It is possible to precisely define the problem that has a solution but COMPUTING the solution is impossible (period)

Solve the problem for "real world" cases

Model checking

文A 13 languages

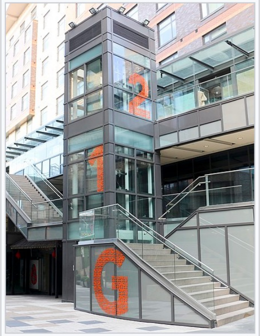
Article Talk Read Edit View history Tools

From Wikipedia, the free encyclopedia

This article is about checking of models in computer science. For the checking of models in statistics, see statistical model validation.

In computer science, **model checking** or **property checking** is a method for checking whether a finite-state model of a system meets a given specification (also known as correctness). This is typically associated with hardware or software systems, where the specification contains liveness requirements (such as avoidance of livelock) as well as safety requirements (such as avoidance of states representing a system crash).

In order to solve such a problem algorithmically, both the model of the system and its specification are formulated in some precise mathematical language. To this end, the problem is formulated as a task in logic, namely to check whether a structure satisfies a given logical formula. This general concept applies to many kinds of logic and many kinds of structures. A simple model-checking problem consists of verifying whether a formula in the propositional logic is satisfied by a given structure.



Elevator control software can be model-checked to verify both safety properties, like *"The cabin never moves with its door open",*[1] and liveness properties, like *"Whenever the n^{th} floor's call button is pressed, the cabin will eventually stop at the n^{th} floor and open the door".*

Overview [edit]

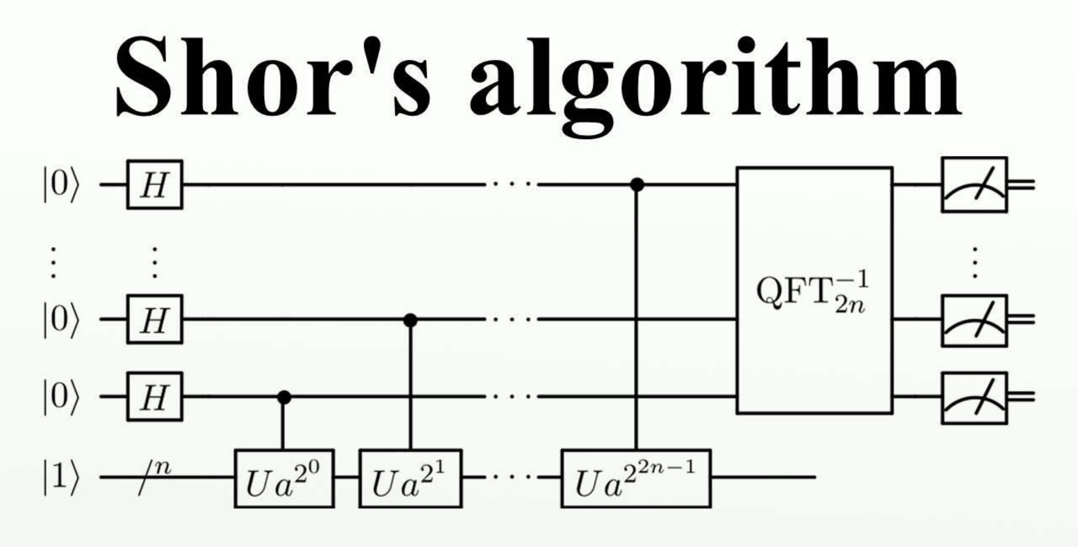
Property checking is used for verification when two descriptions are not equivalent. During refinement, the specification is complemented with details that are unnecessary in the higher-level specification. There is no need to verify the newly introduced properties against the original specification since this is not possible. Therefore, the strict bi-directional equivalence check is relaxed to a one-way property check. The implementation or design is regarded as a model of the system, whereas the specifications are properties that the model must satisfy.[2]

An important class of model-checking methods has been developed for checking models of hardware and software designs where the specification is given by a temporal logic formula. Pioneering work in temporal logic specification was done by Amir Pnueli, who received the 1996 Turing award for "seminal work introducing temporal logic into computing science".[3] Model checking began with the pioneering work of E. M. Clarke, E. A. Emerson,[4][5][6] by J. P. Queille, and J. Sifakis.[7] Clarke, Emerson, and Sifakis shared the 2007 Turing Award for their seminal work founding and developing the field of model checking.[8][9]

Case 3.2.1

It is possible to precisely define the problem that has a solution but COMPUTING the solution efficiently with **current** technology is very hard

Solve the problem mathematically



https://en.wikipedia.org/wiki/File:Shor's_algorithm.svg

Shtetl-Optimized

The Blog of Scott Aaronson

If you take nothing else from this blog: quantum computers won't solve hard problems instantly by just trying all solutions in parallel.

And also: deliberately gunning down Jewish (or any) children is wrong.

PSPACE PostBQP NP BQP P

« NAND now for something completely different

Quantum Computing Since Democritus Lecture 10: Quantum Computing »

Shor, I'll do it

I've been talking a lot recently about how quantum algorithms *don't* work. But last week JR Minkel, an editor at *Scientific American*, asked me to write a brief essay about how quantum algorithms *do* work, which he could then link to from *SciAm*'s website."OK!" I replied, momentarily forgetting about the $10^{10^{5000}}$ quantum algorithm tutorials that are already on the web. So, here's the task I've set for myself: to explain Shor's algorithm without using a single ket sign, or for that matter any math beyond arithmetic.

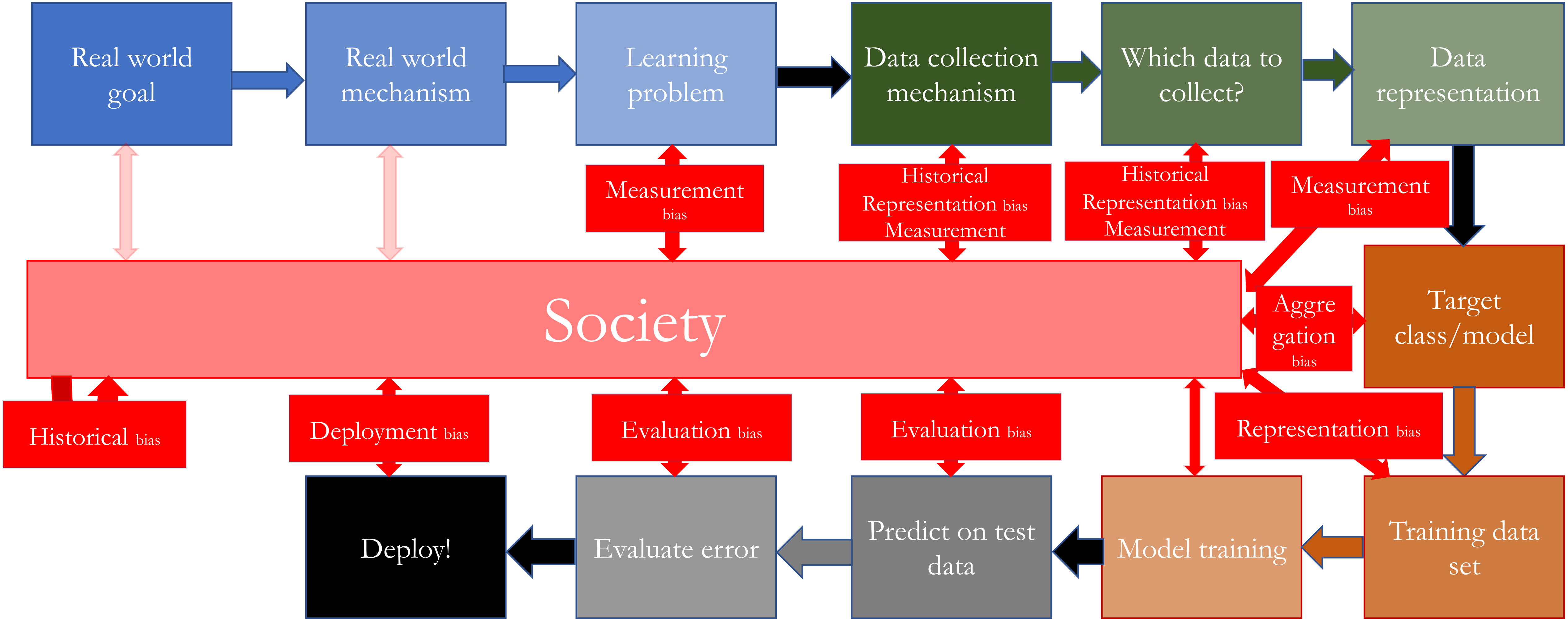
Alright, so let's say you want to break the RSA cryptosystem, in order to rob some banks, read your ex's email, whatever. We all know that breaking RSA reduces to finding the prime factors of a large integer N. Unfortunately, we also know that "trying all possible divisors in parallel," and then instantly picking the right one, isn't going to work. Hundreds of popular magazine articles notwithstanding, trying everything in parallel just isn't the sort of thing that a quantum computer can do. Sure, in some sense you can "try all possible divisors" — but if you then measure the outcome, you'll get a *random* divisor, which almost certainly won't be the one you

Case 3.2.2

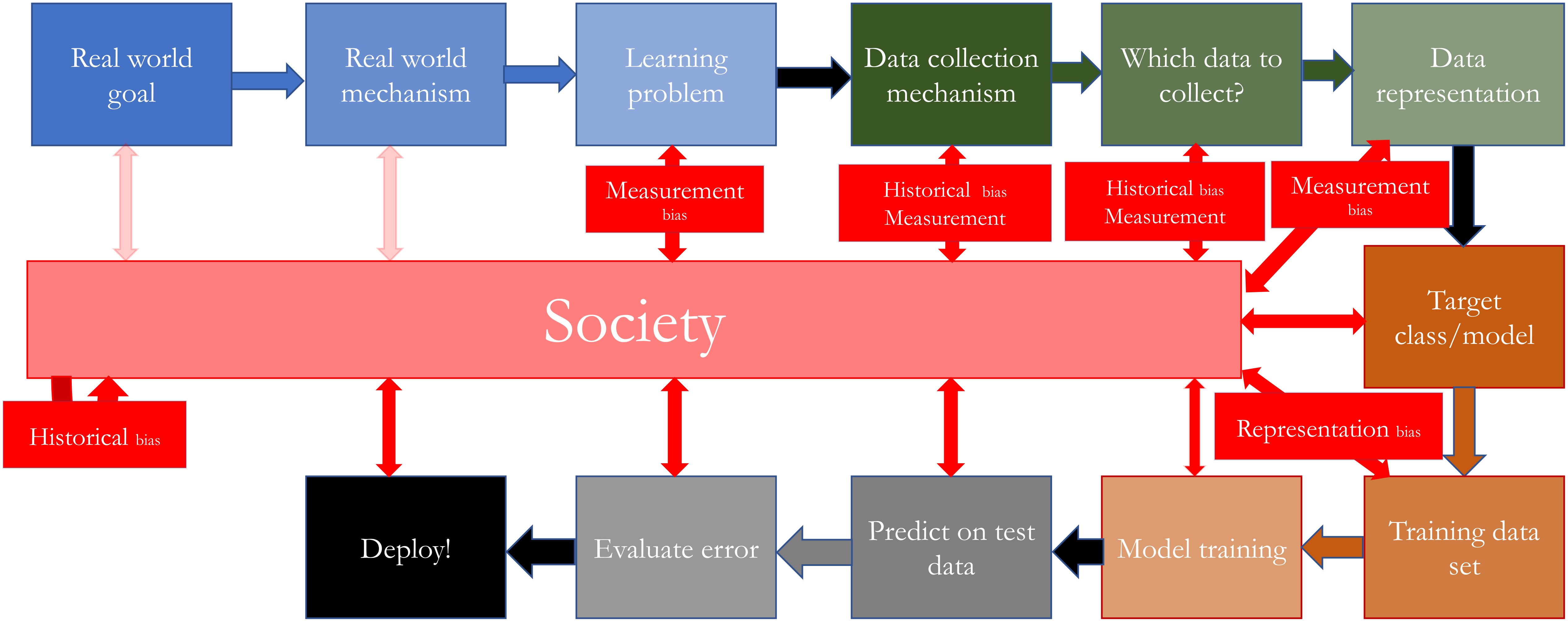
It is possible to precisely define the problem that has a solution but implementing the solution efficiently in **current** world is hard

Any example?

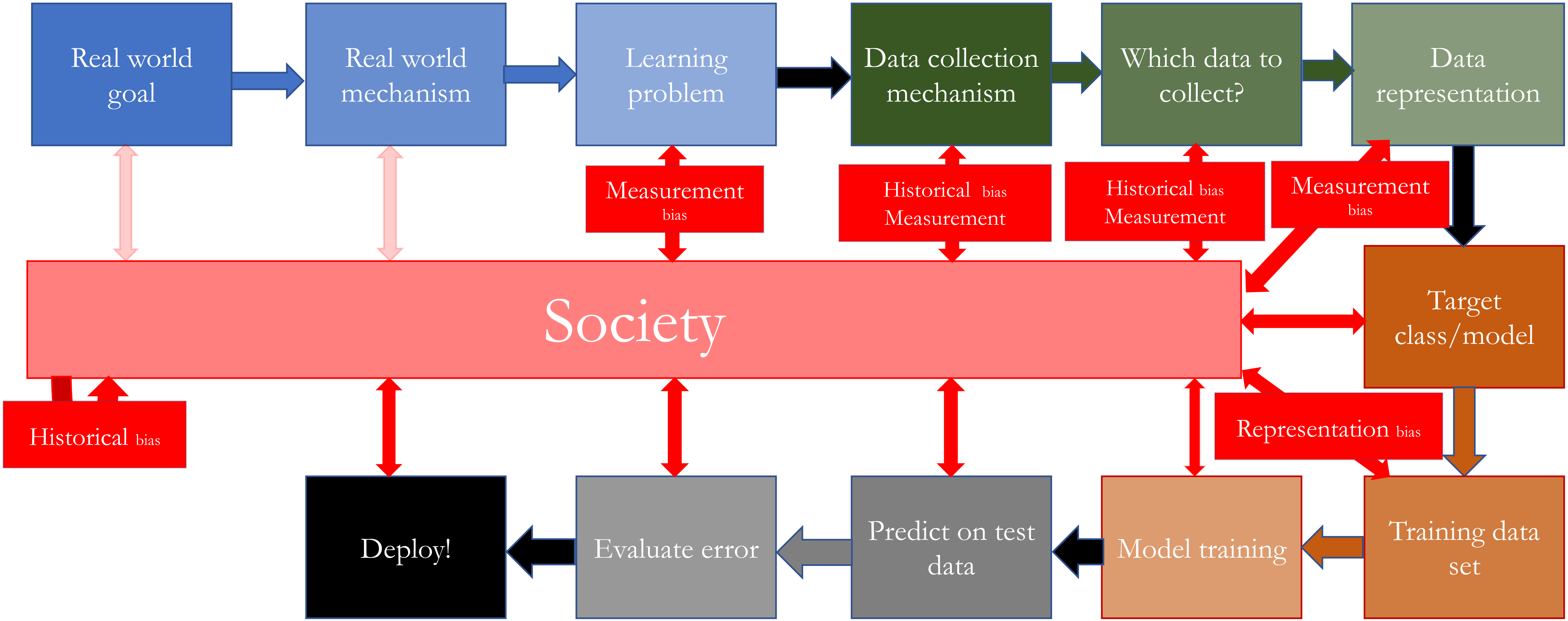
Different biases



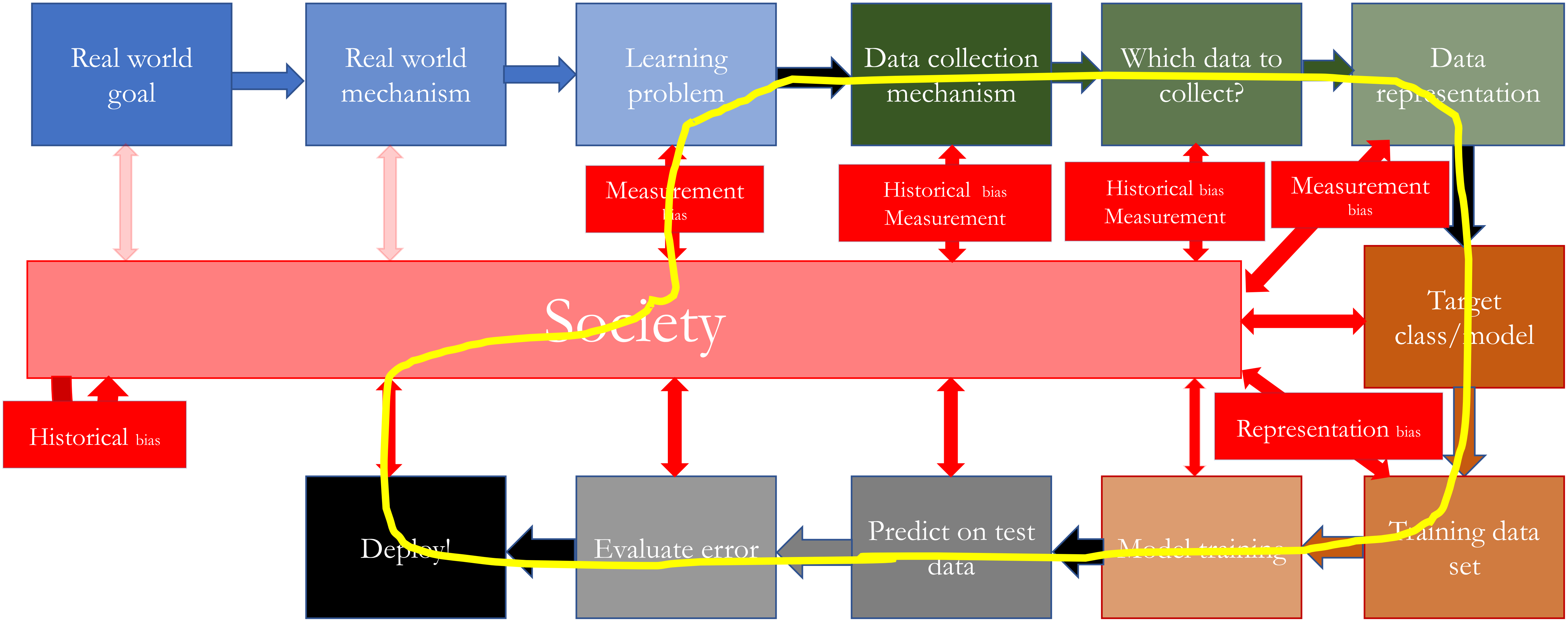
Let us assume…



What is a feedback loop



The “loop” in feedback loop



Case 3.2.2

It is possible to precisely define the problem that has a solution but implementing the solution efficiently in **current** world is hard

How do you “prove” that a specific AI system will result in a feedback loop?

Model the world mathematically and show a feedback loop in your model


IN DETAIL

To predict and serve?

Predictive policing systems are used increasingly by law enforcement to try to prevent crime before it occurs. But what happens when these systems are trained using biased data? **Kristian Lum** and **William Isaac** consider the evidence – and the social consequences

Proceedings of Machine Learning Research 81:1–12, 2018 Conference on Fairness, Accountability, and Transparency

Runaway Feedback Loops in Predictive Policing*

Danielle Ensign DANIPHYE@GMAIL.COM
University of Utah
Sorelle A. Friedler SORELLE@CS.HAVERFORD.EDU
Haverford College
Scott Neville DROP.SCOTT.N@GMAIL.COM
University of Utah
Carlos Scheidegger CSCHEID@CSCHEID.NET
University of Arizona
Suresh Venkatasubramanian† SURESH@CS.UTAH.EDU
University of Utah

Editors: Sorelle A. Friedler and Christo Wilson

Abstract

Predictive policing systems are increasingly used to determine how to allocate police across a city in order to best prevent crime. Discovered crime data (e.g., arrest counts) are used to help update the model, and the process is repeated. Such systems have been empirically shown to be susceptible to runaway feedback loops, where police are repeatedly sent back to the same neighborhoods regardless of the true crime rate.

In response, we develop a mathematical model of predictive policing that proves the predictive policing algorithm) interact: in brief, while reported incidents can attenuate the degree of runaway feedback, they cannot entirely remove it without the interventions we suggest.

Keywords: Feedback loops, predictive policing, online learning.

1. Introduction

Machine learning models are increasingly being used to make real-world decisions, such as who

Case 3.3

It is possible to precisely define the problem that has a solution but COMPUTING the solution *efficiently* is hard/not known with current technology

Problems that are hard because no one has been able to show that it is "easy"

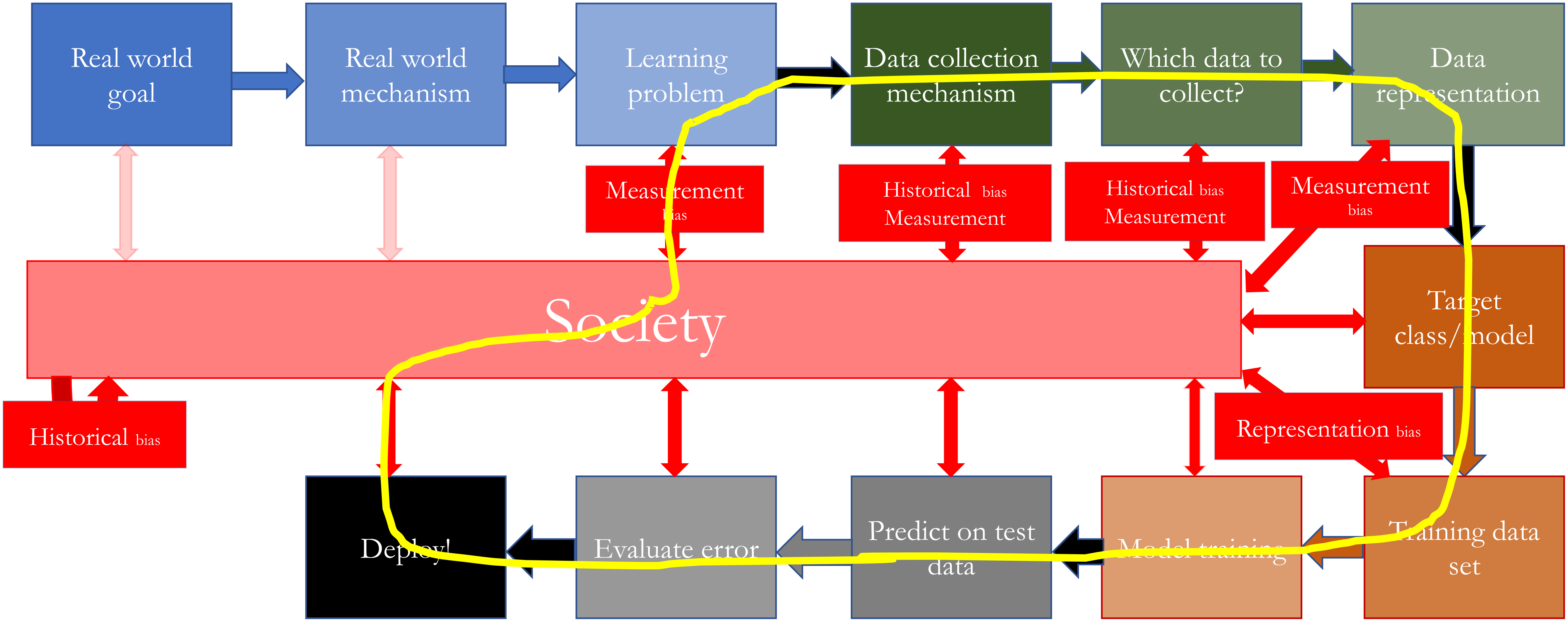
Note the *human* angle!

The "work around"

If something is impossible, it might make something *else* possible

Cryptography!

The “loop” in feedback loop



Predictive policing

Potential biases in predictive policing

Exercise

Exercise

Figure out how predictive policing can lead to a feedback loop.

Aside 1: DAGs in causal diagrams

Consider a world that can be represented as a causal DAG

Can we have a feedback loop in this world?

Going forward your causal diagrams can have cycles!

Community organization can get results!

Kate Crawford
@katecrawford

Big news: LAPD will end the use of the broken predictive policing system known as PredPol, citing budget concerns under COVID-19. This is thanks in large part to community groups like @stoplapdspying pushing back against its use.

LAPD will end controversial program that aimed to predict where crimes woul...
Chief Moore says, due to financial constraints caused by the pandemic, the LAPD will end a program that predicts where property crimes could occur.
latimes.com

Can we formalize this intuition?

How do we “prove” that feedback loops can exist in predictive policing?

Simulation results

Theoretical modeling results

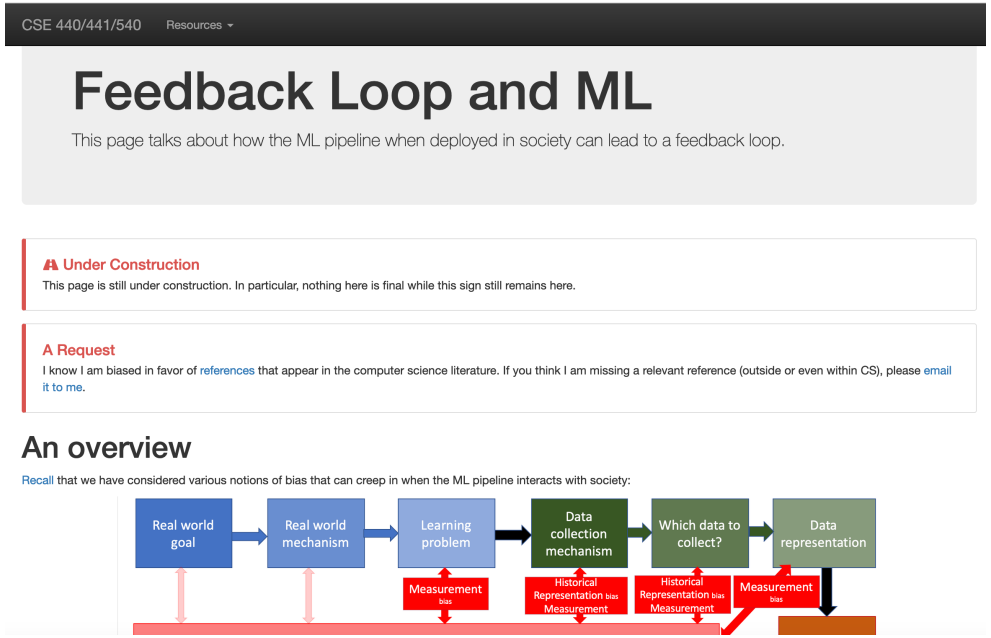
A simulation result

IN DETAIL

To predict and serve?

Predictive policing systems are used increasingly by law enforcement to try to prevent crime before it occurs. But what happens when these systems are trained using biased data? **Kristian Lum** and **William Isaac** consider the evidence – and the social consequences

Theoretical modeling: Ensign et al.


CSE 440/441/540 Resources

Feedback Loop and ML

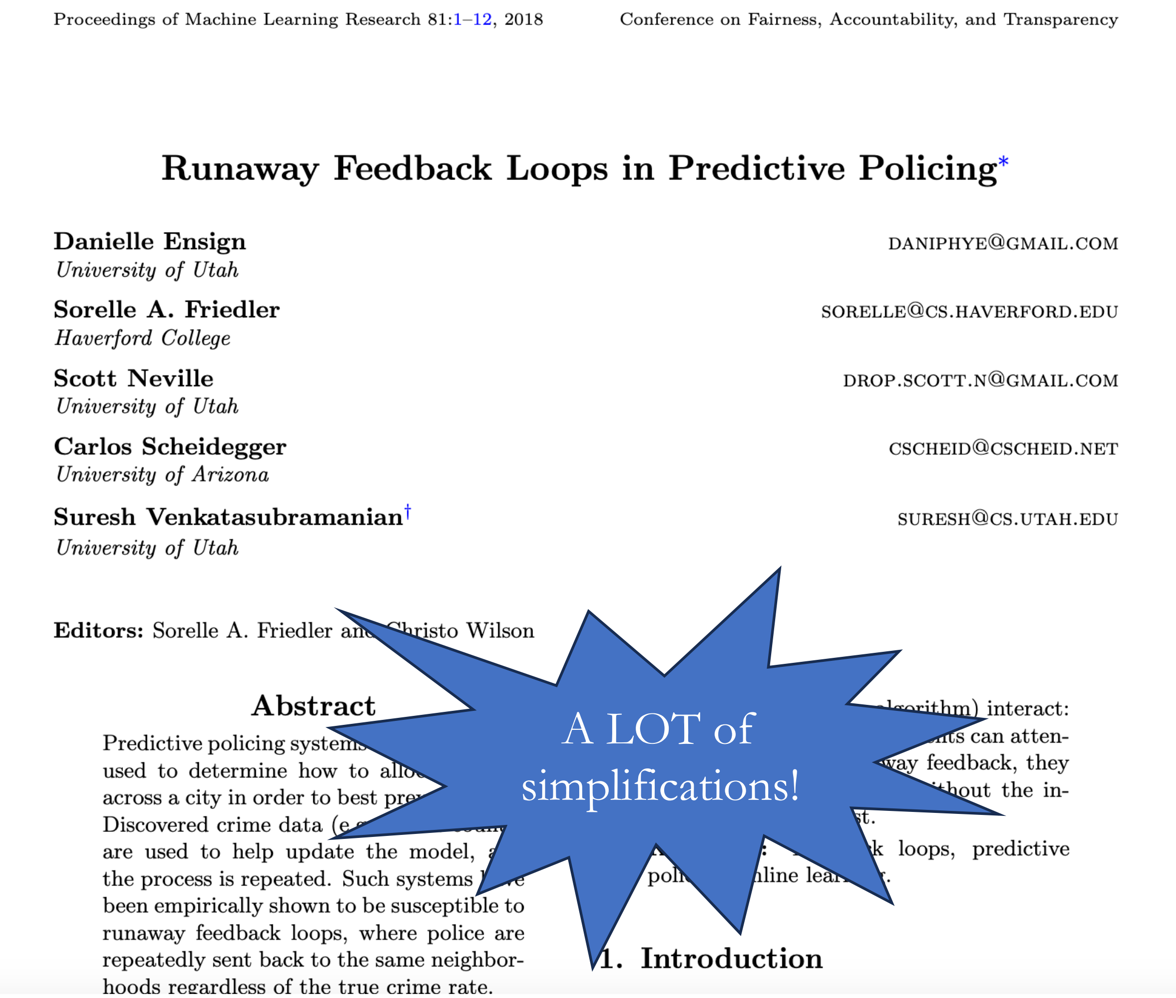
This page talks about how the ML pipeline when deployed in society can lead to a feedback loop.

Under Construction
This page is still under construction. In particular, nothing here is final while this sign still remains here.

A Request
I know I am biased in favor of references that appear in the computer science literature. If you think I am missing a relevant reference (outside or even within CS), please email it to me.

An overview

Recall that we have considered various notions of bias that can creep in when the ML pipeline interacts with society:


Proceedings of Machine Learning Research 81:1–12, 2018 Conference on Fairness, Accountability, and Transparency

Runaway Feedback Loops in Predictive Policing*

Danielle Ensign DANIPHYE@GMAIL.COM
University of Utah
Sorelle A. Friedler SORELLE@CS.HAVERFORD.EDU
Haverford College
Scott Neville DROP.SCOTT.N@GMAIL.COM
University of Utah
Carlos Scheidegger CSCHEID@CSCHEID.NET
University of Arizona
Suresh Venkatasubramanian† SURESH@CS.UTAH.EDU
University of Utah

Editors: Sorelle A. Friedler an[d] Christo Wilson

Abstract

Predictive policing system[s] ... used to determine how to allo[cate] ... across a city in order to best pre[dict] ... Discovered crime data (e ... are used to help update the model, a[nd] ... the process is repeated. Such systems h[ave] been empirically shown to be susceptible to runaway feedback loops, where police are repeatedly sent back to the same neighborhoods regardless of the true crime rate.

...[alg]orithm) interact: ...[eve]nts can atten- ...[runa]way feedback, they ...[w]ithout the in- ...[s]t.

...k loops, predictive poli[cing] ... [o]nline lear[ning] ...

1. Introduction

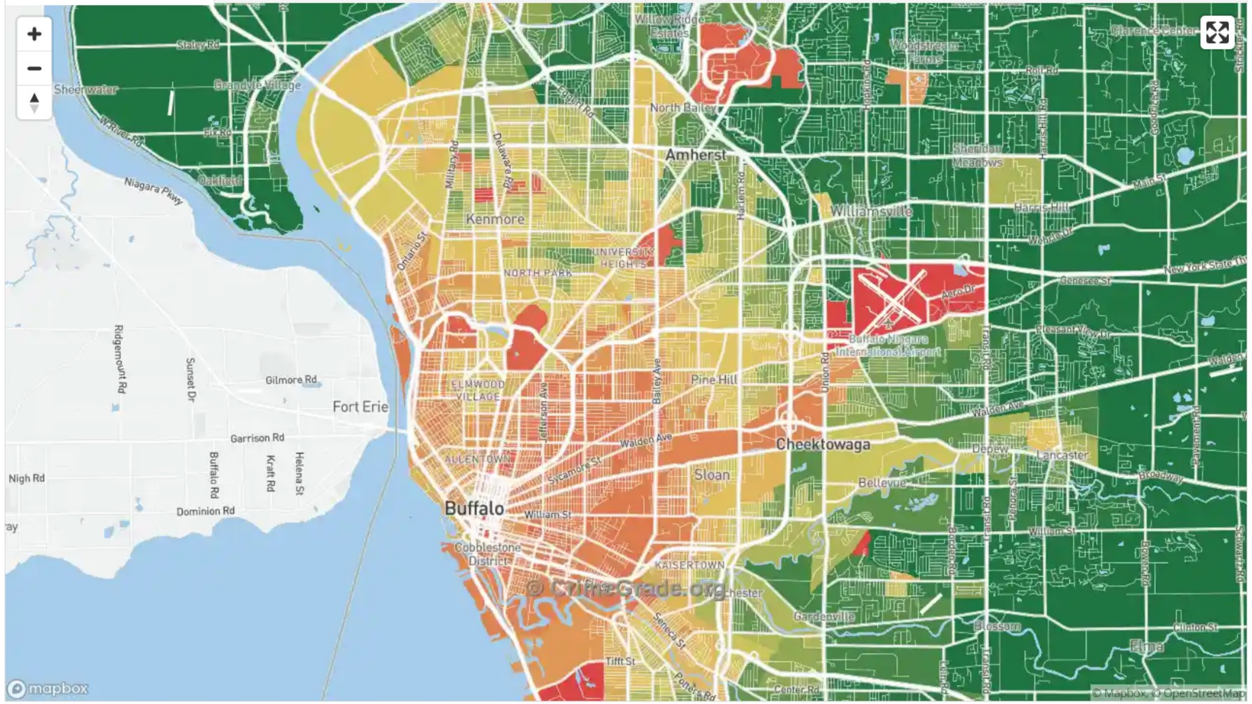
Basic Setup

Only ONE cop patrol E and W

Crime per Capita in Buffalo

The map below shows crime per 1,000 Buffalo residents.

Overall Crime | Property Crime | Violent Crime | More

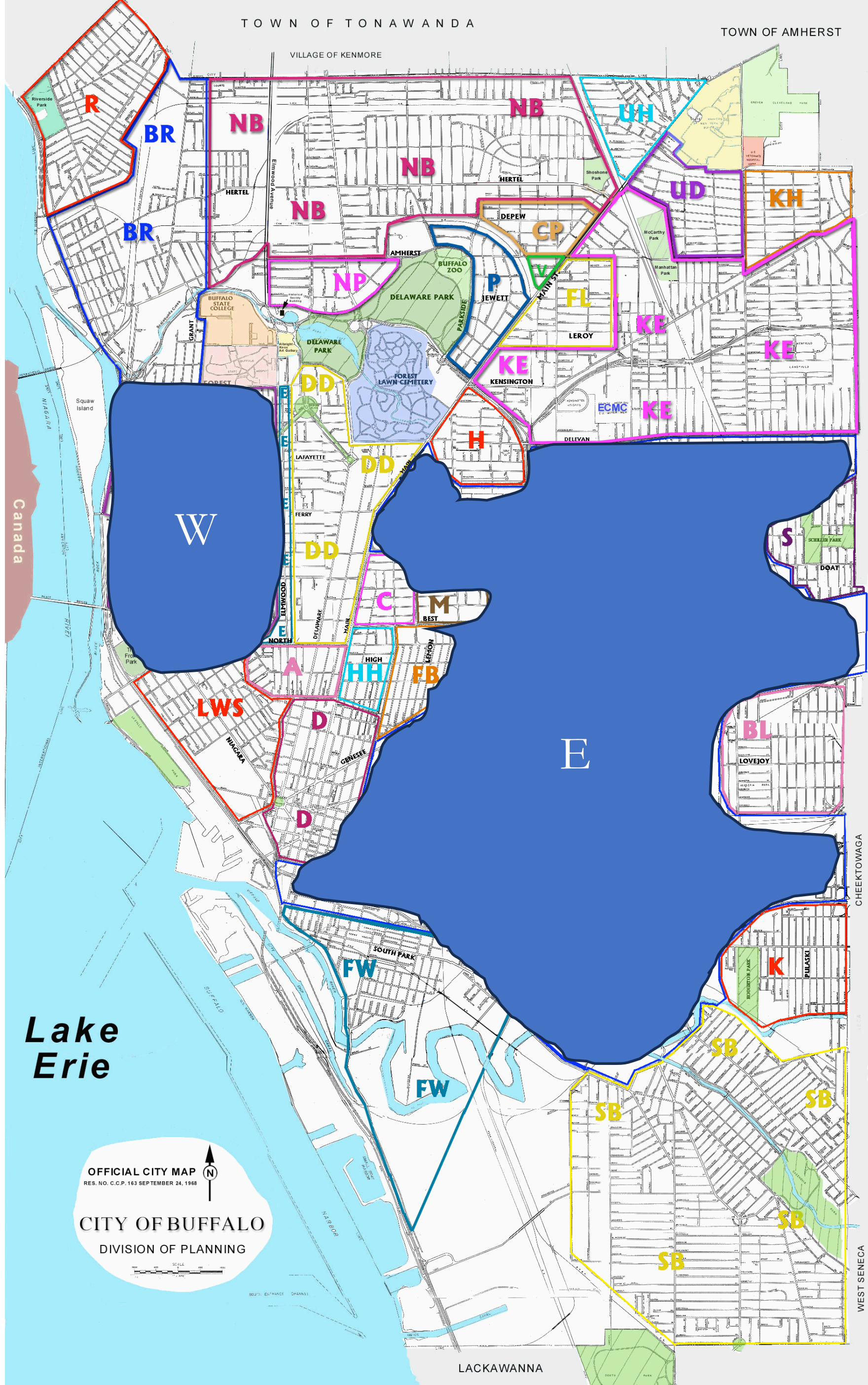


Crime Grades

https://crimegrade.org/safest-places-in-buffalo-ny/

A+	C	F

A+ (dark green) areas are safest



https://library.buffalo.edu/maps/buffalo-wnymaps/buffalo-neighborhoods.html

Assumption 1: One region/day

Cop can only go to one of E or W region per day

Cop will go to E or W with probability proportional to the number of crimes reported in that region

Notation alert

For any given day t, we will use $n_E^{(t)}$ and $n_W^{(t)}$ to denote the number of observed crimes in E and W respectively from day 0 to day t.

Cop will visit E on t+1 with probability $$\frac{n_E^{(t)}}{n_E^{(t)} + n_W^{(t)}}$$

Cop will visit W on t+1 with probability $$\frac{n_W^{(t)}}{n_E^{(t)} + n_W^{(t)}}.$$

Assumption 2: Unequal crime rates

E and W unequal crime rates (which are known)

Notation alert

The crime rate for E is λ_E and the crime rate for W is λ_W.

Mathematically: $\lambda_E \neq \lambda_W$.

Do NOT need λ_E and λ_W to be far apart

$\lambda_E = 10.5\%$ and $\lambda_W = 11\%$

Assumption 3: Observed = actual crime rate

Cop discovers crime at *exactly* the same rate as the actual crime rate in either region

Is this a reasonable assumption?

If cop goes to E:

Discovers one crime with probability λ_E and no crime with probability $1 - \lambda_E$

If cop goes to W:

Discovers one crime with probability λ_W and no crime with probability $1 - \lambda_W$

An exercise

If cop goes to E:

Discovers one crime with probability λ_E and no crime with probability $1 - \lambda_E$

If cop goes to W:

Discovers one crime with probability λ_W and no crime with probability $1 - \lambda_W$

Notation alert

For any given day t, we will use $n_E^{(t)}$ and $n_W^{(t)}$ to denote the number of observed crimes in E and W respectively from day 0 to day t.

Exercise

Given the above what are the relationships of $n_E^{(t+1)}$ with $n_E^{(t)}$ (and similarly the relationship of $n_W^{(t+1)}$ with $n_W^{(t)}$)?

Solution to exercise

Exercise

Given the above what are the relationships of $n_E^{(t+1)}$ with $n_E^{(t)}$ (and similarly the relationship of $n_W^{(t+1)}$ with $n_W^{(t)}$)?

If the cop visits E with probability λ_E the cop will discover/report one crime and not crime otherwise. In other words,

$$n_E^{(t+1)} = \begin{cases} n_E^{(t)} + 1 & \text{with probability } \lambda_E \\ n_E^{(t)} & \text{with probability } 1 - \lambda_E \end{cases}.$$

And we have a similar result **If the cop visits** W:

$$n_W^{(t+1)} = \begin{cases} n_W^{(t)} + 1 & \text{with probability } \lambda_W \\ n_W^{(t)} & \text{with probability } 1 - \lambda_W \end{cases}.$$

Finally, we have our model…

The evolution of the number of observed crimes

- The process starts with initial values $n_E^{(0)}$ and $n_W^{(0)}$.
- For $t = 1, 2, \ldots$

```
//The process repeats "forever"
```

  - With probability $\frac{n_E^{(t)}}{n_E^{(t)}+n_W^{(t)}}$ do:

    ```
    //Cop visits E
    ```

    - With probability λ_E set $n_E^{(t+1)} = n_E^{(t)} + 1$
    - Else with probability $1 - \lambda_E$ set $n_E^{(t+1)} = n_E^{(t)}$
  - Otherwise with probability $\frac{n_W^{(t)}}{n_E^{(t)}+n_W^{(t)}}$ do:

    ```
    //Cop visits W
    ```

    - With probability λ_W set $n_W^{(t+1)} = n_W^{(t)} + 1$
    - Else with probability $1 - \lambda_W$ set $n_W^{(t+1)} = n_W^{(t)}$

Next exercise…

Exercise

To make things concrete assume that $n_E^{(0)} = n_W^{(0)} = 100$ and $\lambda_E = 10.5\%$ and $\lambda_W = 11\%$. What would you consider to be a manifestation of feedback loop as the process above runs?

`Hint`: Think about how the ratios $\frac{n_E^{(t)}}{n_E^{(t)}+n_W^{(t)}}$ and $\frac{n_W^{(t)}}{n_E^{(t)}+n_W^{(t)}}$ evolve as t grows larger. (Side question: why are these ratios something worth monitoring?)

Cop will visit E with probability $\quad \dfrac{n_E^{(t)}}{n_E^{(t)} + n_W^{(t)}}$

Cop will visit W with probability $\quad \dfrac{n_W^{(t)}}{n_E^{(t)} + n_W^{(t)}}.$

Solution to exercise

Exercise

To make things concrete assume that $n_E^{(0)} = n_W^{(0)} = 100$ and $\lambda_E = 10.5\%$ and $\lambda_W = 11\%$. What would you consider to be a manifestation of feedback loop as the process above runs?

`Hint`: Think about how the ratios $\frac{n_E^{(t)}}{n_E^{(t)}+n_W^{(t)}}$ and $\frac{n_W^{(t)}}{n_E^{(t)}+n_W^{(t)}}$ evolve as t grows larger. (Side question: why are these ratios something worth monitoring?)

Cop will visit E with probability $\frac{n_E^{(t)}}{n_E^{(t)} + n_W^{(t)}}$ $\approx \lambda_E$ $\ll \lambda_E$ $\gg \lambda_E$

Cop will visit W with probability $\frac{n_W^{(t)}}{n_E^{(t)} + n_W^{(t)}}$. $\approx \lambda_W$ $\gg \lambda_W$ $\ll \lambda_W$

Does this model have a feedback loop?

The evolution of the number of observed crimes

- The process starts with initial values $n_E^{(0)}$ and $n_W^{(0)}$.
- For $t = 1, 2, \ldots$

```
//The process repeats "forever"
```

  - With probability $\frac{n_E^{(t)}}{n_E^{(t)}+n_W^{(t)}}$ do:

```
//Cop visits E
```

    - With probability λ_E set $n_E^{(t+1)} = n_E^{(t)} + 1$
    - Else with probability $1 - \lambda_E$ set $n_E^{(t+1)} = n_E^{(t)}$
  - Otherwise with probability $\frac{n_W^{(t)}}{n_E^{(t)}+n_W^{(t)}}$ do:

```
//Cop visits W
```

    - With probability λ_W set $n_W^{(t+1)} = n_W^{(t)} + 1$
    - Else with probability $1 - \lambda_W$ set $n_W^{(t+1)} = n_W^{(t)}$

$n_E^{(0)} = n_W^{(0)} = 100$ and $\lambda_E = 10.5\%$ and $\lambda_W = 11\%$.

What do you think will happen?

Answer is yes and in the most extreme sense.

			$\lambda_E > \lambda_W$	$\lambda_E < \lambda_W$
Cop will visit E with probability	$\frac{n_E^{(t)}}{n_E^{(t)} + n_W^{(t)}}$	$\approx \lambda_E$	$= 1$	$= 0$
Cop will visit W with probability	$\frac{n_W^{(t)}}{n_E^{(t)} + n_W^{(t)}}.$	$\approx \lambda_W$	$= 0$	$= 1$

How the heck do you prove such a thing?

Pólya urn model

6 languages

Article Talk Read Edit View history Tools

From Wikipedia, the free encyclopedia

In statistics, a **Pólya urn model** (also known as a **Pólya urn scheme** or simply as **Pólya's urn**), named after George Pólya, is a family of urn models that can be used to interpret many commonly used statistical models.

The model represents objects of interest (such as atoms, people, cars, etc.) as colored balls in an urn. In the basic Pólya urn model, the experimenter puts *x* white and *y* black balls into an urn. At each step, one ball is drawn uniformly at random from the urn, and its color observed; it is then returned in the urn, and an additional ball of the same color is added to the urn.

If by random chance, more black balls are drawn than white balls in the initial few draws, it would make it more likely for more black balls to be drawn later. Similarly for the white balls. Thus the urn has a self-reinforcing property ("the rich get richer"). It is the opposite of sampling without replacement, where every time a particular value is observed, it is less likely to be observed again, whereas in a Pólya urn model, an observed value is *more* likely to be observed again. In a Pólya urn model, successive acts of measurement over time have less and less effect on future measurements, whereas in sampling without replacement, the opposite is true: After a certain number of measurements of a particular value, that value will never be seen again.

It is also different from sampling with replacement, where the ball is returned to the urn but without adding new balls. In this case, there is neither self-reinforcing nor anti-self-reinforcing.

Lemma 3 (Renlund (2010)) *Suppose we are given a Pólya urn with replacement matrix of the form*

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix}$$

with a positive number of balls of each kind to start with. Assume that $a, b, c, d \geq 0$ and at least one entry is strictly positive. Then the limit of the fraction of balls of each type exists almost surely. The fraction p of A-colored balls can be characterized as follows:

- *If $a = d, c = b = 0$, then p tends towards a beta distribution.*
- *If not, then p tends towards a single point distribution x^*, where $x^* \in [0, 1]$ is a root of the quadratic polynomial*

$$(c + d - a - b)x^2 + (a - 2c - d)x + c.$$

If two such roots exist, then it is the one such that $f'(x^*) < 0$.

Now let’s prove this “lemma 3”!

Just kidding ☺

Now let's prove this "lemma 3"!

Just kidding ☺

Is there a (mathematical) "fix"?

A potential fix

In the above model, Ensign, Friedler, Neville, Scheidegger and Venkatasubramanian suggest the following fix (which they can mathematically prove that it works) is based roughly on the following idea. If the cop visits a specific region most of the time, then it should not be a surprise if they discover a crime in the region and in such a case they should "discount" the crime discovery by not recording such discovery most of the time. On the other hand, if the cop visits region infrequently and they discover a crime, they have learned something "new" and hence would record such crime discoveries most of the time.

Discussion Summary due Sat 11:59pm!

IN DETAIL

To predict and serve?

Predictive policing systems are used increasingly by law enforcement to try to prevent crime before it occurs. But what happens when these systems are trained using biased data? **Kristian Lum** and **William Isaac** consider the evidence – and the social consequences