# CSCI 567: Machine Learning

Vatsal Sharan

Spring 2026

Lecture 13, Apr 21

USC University of Southern California

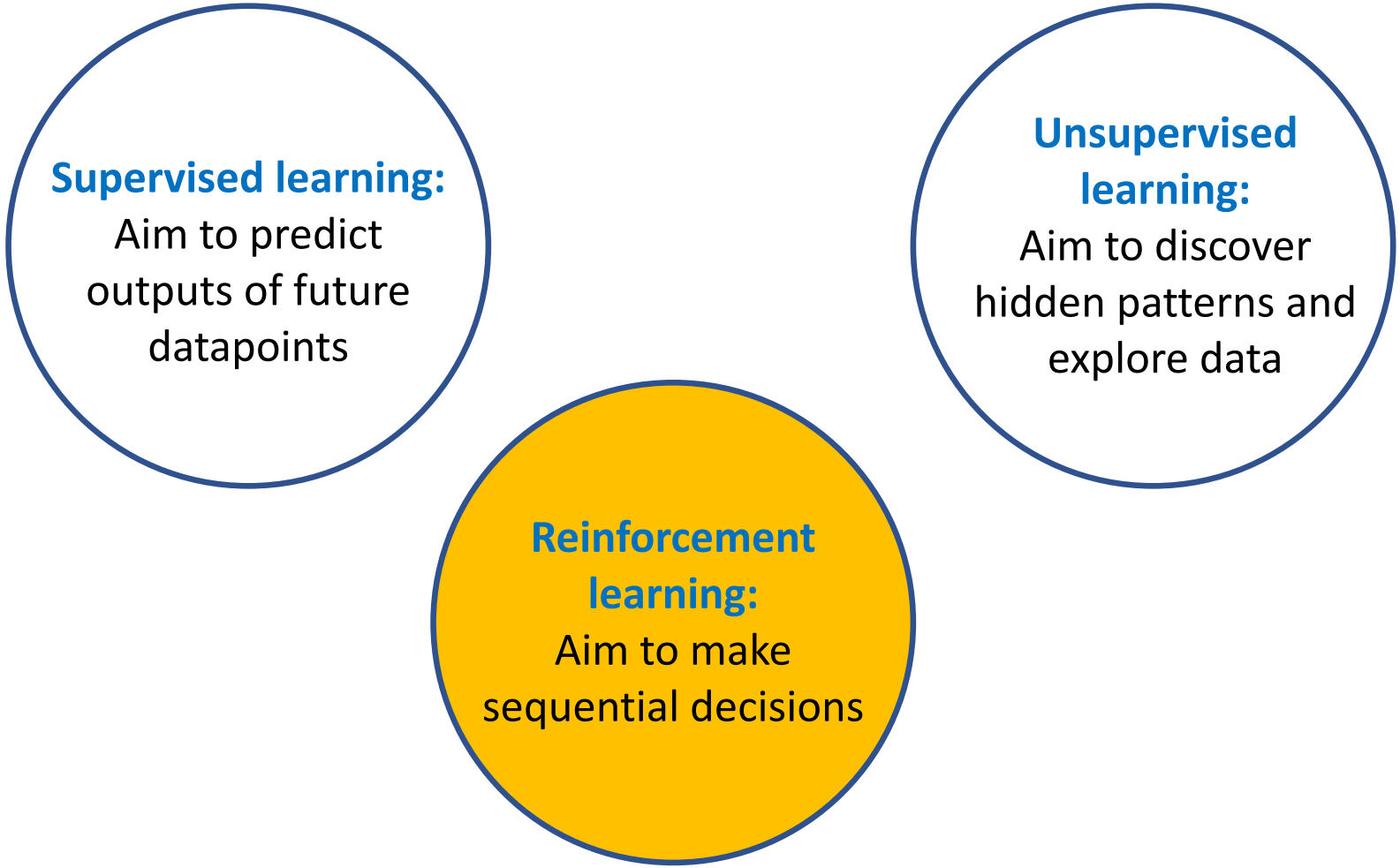
# Administrivia

- Exam 2 is on May 1 from 1pm-3:20pm
  - Similar instructions as Exam 1
  - More info will be posted on Ed later

- Today's plan:
  - A bit on reinforcement Learning
  - Trustworthy ML


# Multiarmed bandits

# A simplistic taxonomy of ML



# Multi-armed bandits

- Motivation & setup
- Exploration vs. Exploitation

# Decision making

Problems we have discussed so far:

- start with a fixed training dataset
- learn a predictor from the data or discover some patterns in the data

But many real-life problems are about **learning continuously**:

- make a prediction/decision
- receive some feedback
- repeat

Broadly, these are called **online decision making problems**.

# Examples

Amazon/Netflix/Instagram **recommendation systems**:

- a user visits the website (or views a post etc.)
- the system recommends some products/movies/posts
- the system observes whether the user clicks on the recommendation

**Playing games** (Go/Atari/StarCraft/...) or **controlling robots**:

- make a move
- receive some reward (e.g. score a point) or loss (e.g. fall down)
- make another move...

# Multiarmed bandits: Motivation

Imagine going to a casino to play a slot machine

- it robs you, like a "bandit" with a single arm

Of course there are many slot machines in the casino

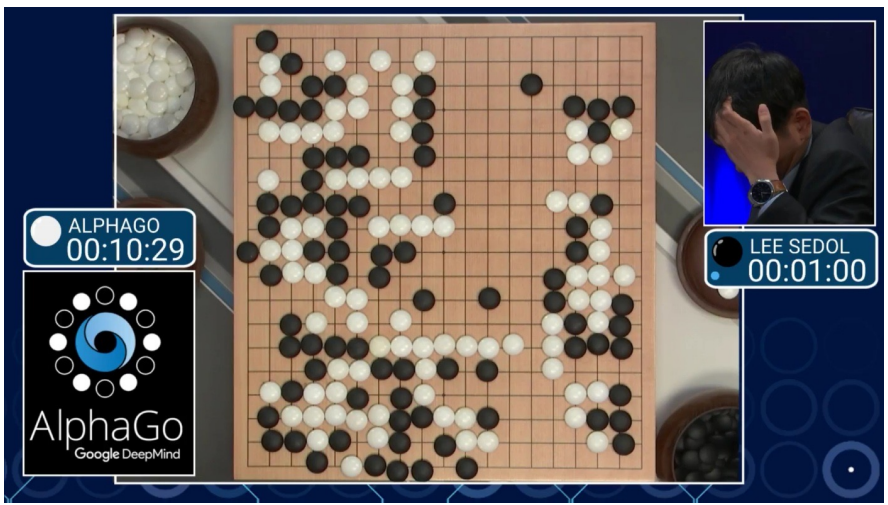
- like a bandit with multiple arms (hence the name)
- **if I can play 10 times, which machines should I play?**


# Applications

This simple model and its variants capture many real-life applications:

- recommendation systems, each product/movie/news story is an arm
  (Netflix employs a variant of bandit algorithm)

- game playing, each possible move is an arm
  (AlphaGo has a bandit algorithm as one of the components)

# Formal setup

There are $K$ **arms** (actions/choices/...)

The problem proceeds in rounds between the environment and a learner: for each time $t = 1, \ldots, T$

- the environment decides the reward for each arm $r_{t,1}, \ldots, r_{t,K}$
- the learner picks an arm $a_t \in [K]$
- the learner observes the reward for arm $a_t$, i.e., $r_{t,a_t}$

↖ reward for each arm at time $t$.

Importantly, *learner does not observe rewards for arms not selected!*

This kind of limited feedback is usually referred to as bandit feedback

# Evaluating performance

What should be the goal here?

Maximizing total rewards $\sum_{t=1}^{T} r_{t,a_t}$ seems natural.

But the absolute value of rewards is not meaningful, instead we should compare it to some *benchmark*. A classic benchmark is

$$\max_{a \in [K]} \sum_{t=1}^{T} r_{t,a}$$

i.e. the largest reward one can achieve by always playing a fixed arm

So we want to minimize

$$\max_{a \in [K]} \sum_{t=1}^{T} r_{t,a} - \sum_{t=1}^{T} r_{t,a_t}$$

This is called the **regret**: *how much I regret not sticking with the best fixed arm in hindsight?*

# Environments

**How are the rewards generated by the environments?**

- they could be generated via some fixed distribution
- they could be generated via some changing distribution
- they could be generated even completely arbitrarily/adversarially

We focus on a simple setting:

- rewards of arm $a$ are i.i.d. samples of $\text{Ber}(\mu_a)$, that is, $r_{t,a}$ is $1$ with prob. $\mu_a$, and $0$ with prob. $1-\mu_a$, independent of anything else.
- each arm has a different mean $(\mu_1, \ldots, \mu_K)$; the problem is essentially about finding the best arm $\text{argmax}_a \, \mu_a$ as quickly as possible

# Empirical means

Let $\hat{\mu}_{t,a}$ be the **empirical mean** of arm $a$ up to time $t$:

$$\hat{\mu}_{t,a} = \frac{1}{n_{t,a}} \sum_{\tau \leq t : a_\tau = a} r_{\tau,a}$$

where

$$n_{t,a} = \sum_{\tau \leq t} \mathbb{I}[a_\tau == a]$$

is the **number of times** we have picked arm $a$.

**Concentration**: $\hat{\mu}_{t,a}$ should be close to $\mu_a$ if $n_{t,a}$ is large

# Multi-armed bandits

- Motivation & setup
- Exploration vs. Exploitation

# Exploitation only

**Greedy:**

Pick each arm once for the first $K$ rounds.

For $t = K+1, \ldots, T$, pick $a_t = \text{argmax}_a \ \hat{\mu}_{t-1,a}$.

*What's wrong with this greedy algorithm?*

Consider the following example:

- $K = 2, \mu_1 = 0.6, \mu_2 = 0.5$ (so arm 1 is the best)
- suppose the algorithm first picks arm 1 and sees reward 0, then picks arm 2 and sees reward 1 (this happens with decent probability) ( $0.4 \times 0.5 = 0.2$)
- the algorithm will never pick arm 1 again!

# The key challenge

All bandit problems face the same **dilemma**:

**Exploitation vs. Exploration trade-off**

- on one hand we want to exploit the arms that we think are good
- on the other hand we need to explore all arms often enough in order to figure out which one is better
- so each time we need to ask: *do I explore or exploit? and how?*

We next discuss **three ways** to trade off exploration and exploitation for our simple multi-armed bandit setting.

# A natural first attempt

**Explore–then–Exploit:**

Input: a parameter $T_0 \in [T]$

**Exploration phase**: for the first $T_0$ rounds, pick each arm for $T_0/K$ times

**Exploitation phase**: for the remaining $T - T_0$ rounds, stick with the empirically best arm $\operatorname{argmax}_a \ \hat{\mu}_{T_0,a}$

Parameter $T_0$ clearly controls the exploration/exploitation trade-off

# Explore-then-Exploit: Issues

It's pretty reasonable, but the disadvantages are also clear:

- not clear how to tune the hyperparameter $T_0$
- in the exploration phase, even if an arm is clearly worse than others based on a few pulls, it's still pulled $T_0/K$ times
- clearly it won't work if the environment is changing

# A slightly better algorithm

**$\epsilon$-Greedy** Pick each arm once for the first $K$ rounds.

For $t = K+1, \ldots, T$,

- with probability $\epsilon$, explore: pick an arm uniformly at random
- with probability $1-\epsilon$, exploit: pick $a_t = \text{argmax}_a \ \hat{\mu}_{t-1,a}$

**Pros**

- always exploring and exploiting
- applicable to many other problems
- first thing to try usually

**Cons**

- need to tune $\epsilon$
- same uniform exploration

Is there a more adaptive way to explore?

# More adaptive exploration

A simple modification of "Greedy" leads to the well-known:

**Upper Confidence Bound (UCB) algorithm**

For $t = 1, \ldots, T$, pick $a_t = \text{argmax}_a \ \text{UCB}_{t,a}$ where

$$\text{UCB}_{t,a} := \hat{\mu}_{t-1,a} + 2\sqrt{\frac{\ln t}{n_{t-1,a}}}$$

estimate of rewards, initialize all to be 1.

if $n_{t-1,a}$ is small, then $\text{UCB}_{t,a}$ is large

- the first term in $\text{UCB}_{t,a}$ represents exploitation, while the second (bonus) term represents exploration
- the bonus term is large if the arm is not pulled often enough, which encourages exploration (adaptive due to the first term)
- a parameter-free algorithm, and *it enjoys optimal regret!*

# Upper confidence bound

*Why is it called upper confidence bound?*

One can prove that with high probability,

$$\mu_a \leq \text{UCB}_{t,a}$$

so $\text{UCB}_{t,a}$ is indeed an upper bound on the true mean.

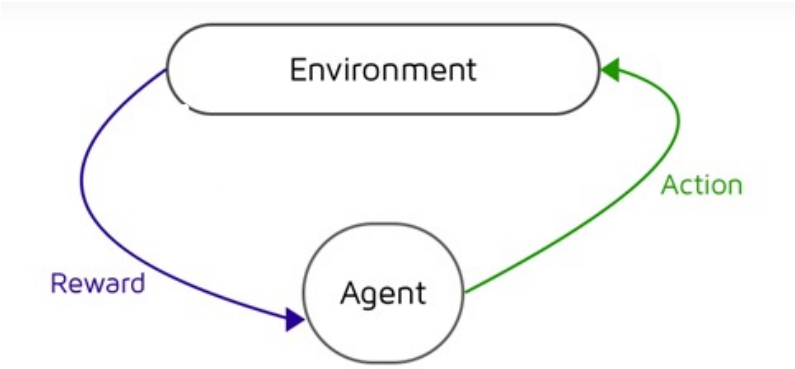
Another way to interpret UCB, "**optimism in face of uncertainty**":

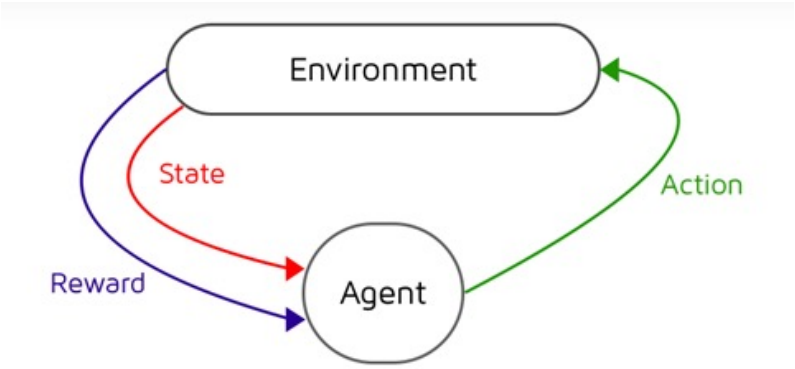
- true environment (best mean) is unknown due to randomness (**uncertainty**)
- have an upper bound (optimistic guess) on the expected reward of each environment, and pick best one according to upper bound (**optimism**)

This principle is useful for many other bandit problems.

# Limitations of multi-armed bandits

Multi-armed bandit is among the simplest decision making problems with limited feedback.



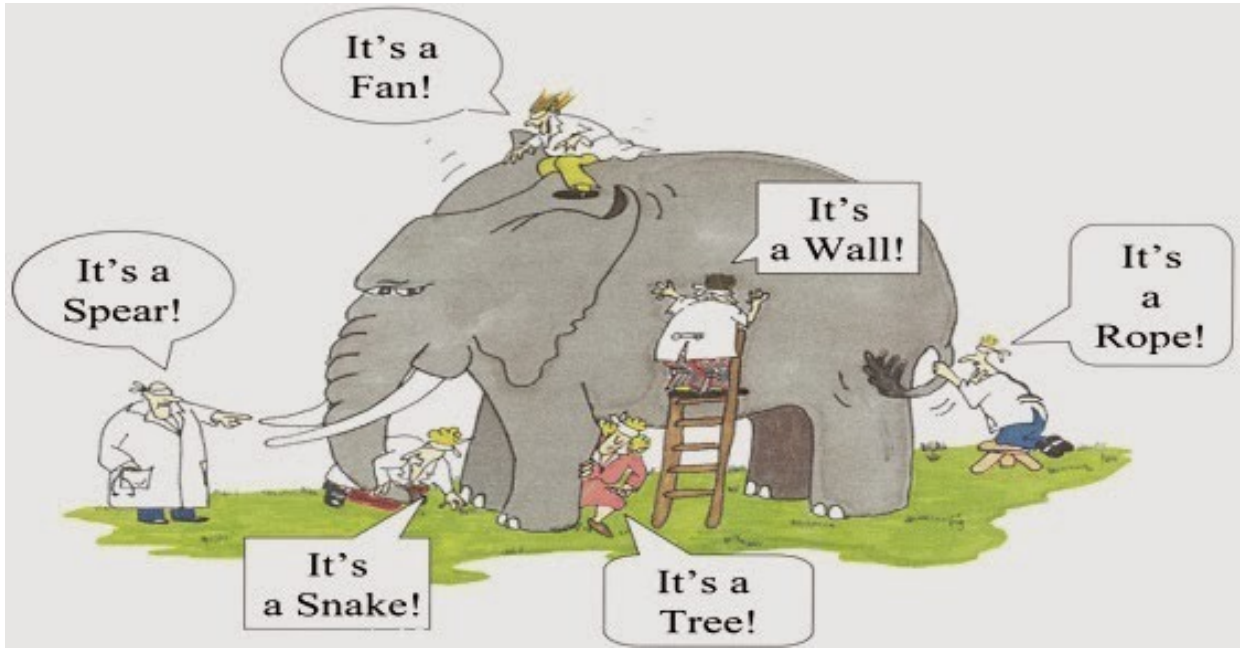

Often, it can be too simple to capture real-life problems. One important aspect it fails to capture is the "state" of the learning agent, which has impacts on the reward of each action.

- e.g. for Atari games, after making one move, the agent moves to a different state, with possible different rewards for each action

There are many other techniques and models in reinforcement learning (RL) which can deal with this issue.


# Trustworthy ML

# Machine Learning can be *brittle*




**The Blind Men and the Elephant**

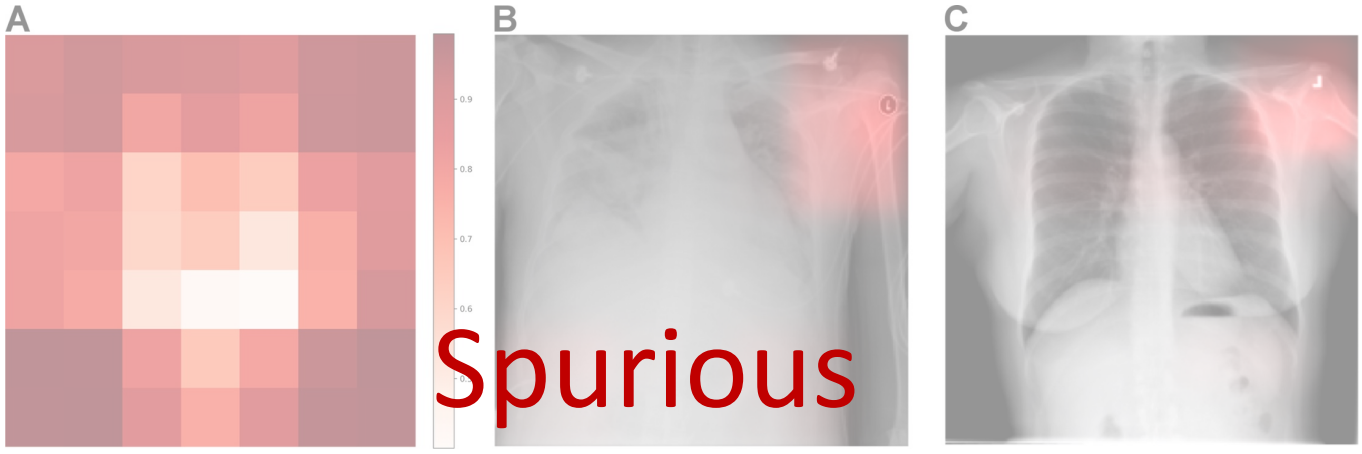
It was six men of Indostan
To learning much inclined,
Who went to see the Elephant
(Though all of them were blind),
That each by observation
Might satisfy his mind.

The First approached the Elephant,
And happening to fall
Against his broad and sturdy side,
At once began to bawl:
"God bless me! but the Elephant
Is very like a WALL!"

….

# Challenges in Trustworthy ML

- Spurious correlations and distributional shifts
- Biases in models and unfairness to demographics
- Adversarial examples
- Privacy, Interpretability, Ethics, …




# Spurious correlations and distributional shifts

# ML models can be very sensitive to changes in the data distribution

You saw a small example of this in the HW3 Bonus question:


# ML models can latch onto spurious features to make predictions

Consider the following task:

Waterbird

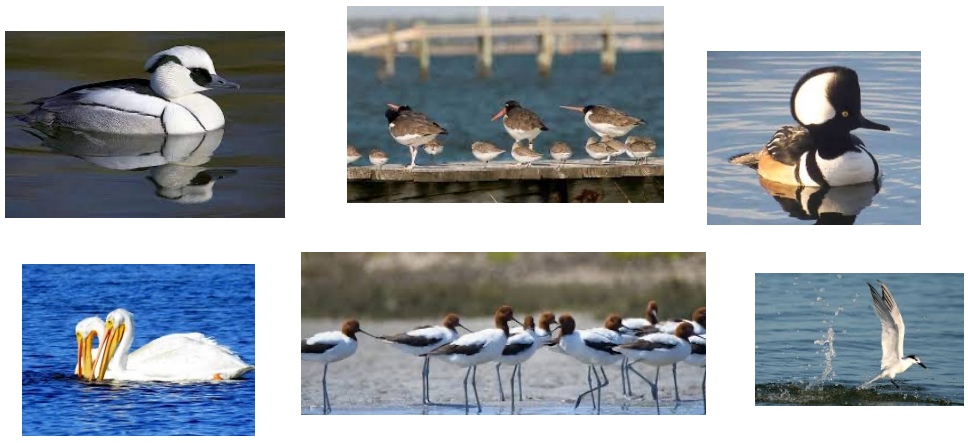
vs.

Landbird

Images courtesy of: University of Hawaii at Manoa, Greg Schechter (Flickr)

# ML models can latch onto spurious features to make predictions

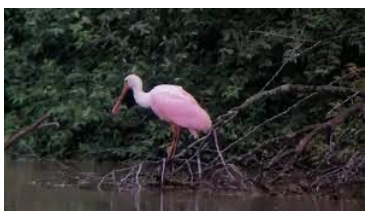
Most images of waterbirds are in water,
and landbirds are on land



Waterbirds vs. Landbirds

# ML models can latch onto spurious features to make predictions

But this isn't always true!


Waterbirds

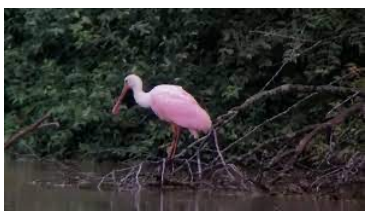
vs.

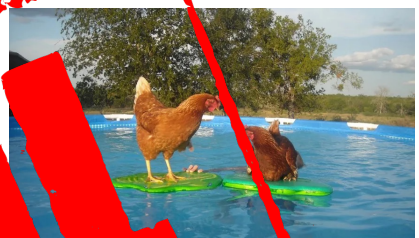


Landbirds

# ML models can latch onto spurious features to make predictions

**This is known as failure to distributional shifts**





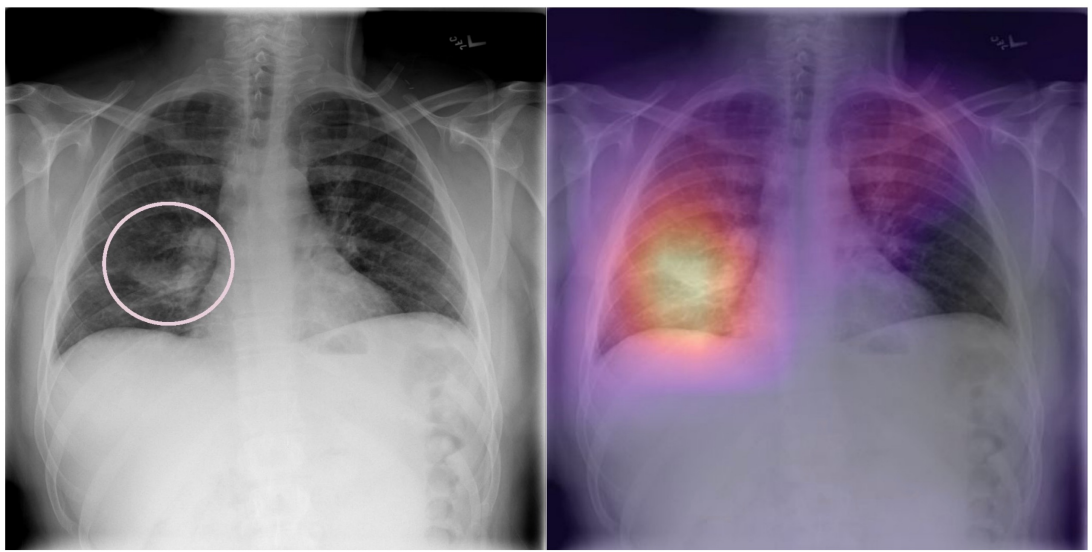
FAIL

Waterbirds vs. Landbirds

# A real-world example

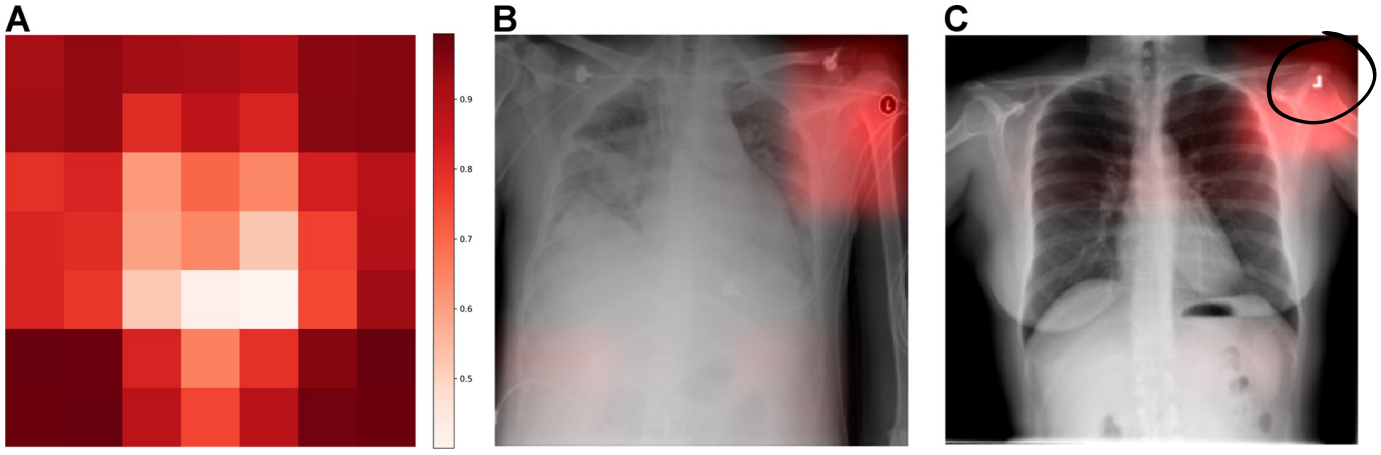
CNN models have obtained impressive results for diagnosing X-rays

E.g. *ChestX-ray8: Hospital-scale Chest X-ray Database and Benchmarks on Weakly-Supervised Classification and Localization of Common Thorax Diseases*, Wang et a;. 2017



Source: *Deep learning for chest radiograph diagnosis: A retrospective comparison of the CheXNeXt algorithm to practicing radiologists,* Rajpurkar et al. 2018

**But the models may not generalize as well to data from new hospitals because they can learn to pickup on spurious correlations such as the type of scanner and marks used by technicians in specific hospitals!**




*CNN to predict hospital system detects both general and specific image features.*
*(A) We obtained activation heatmaps from our trained model and averaged over a sample of images to reveal which subregions tended to contribute to a hospital system classification decision. Many different subregions strongly predicted the correct hospital system, with especially strong contributions from image corners. (B-C) On individual images, which have been normalized to highlight only the most influential regions and not all those that contributed to a positive classification, we note that the CNN has learned to detect a metal token that radiology technicians place on the patient in the corner of the image field of view at the time they capture the image. When these strong features are correlated with disease prevalence, models can leverage them to indirectly predict disease.*

Source: *Variable generalization performance of a deep learning model to detect pneumonia in chest radiographs: A cross-sectional study,* Zech et al. 2018

# How to make models robust to spurious correlations?

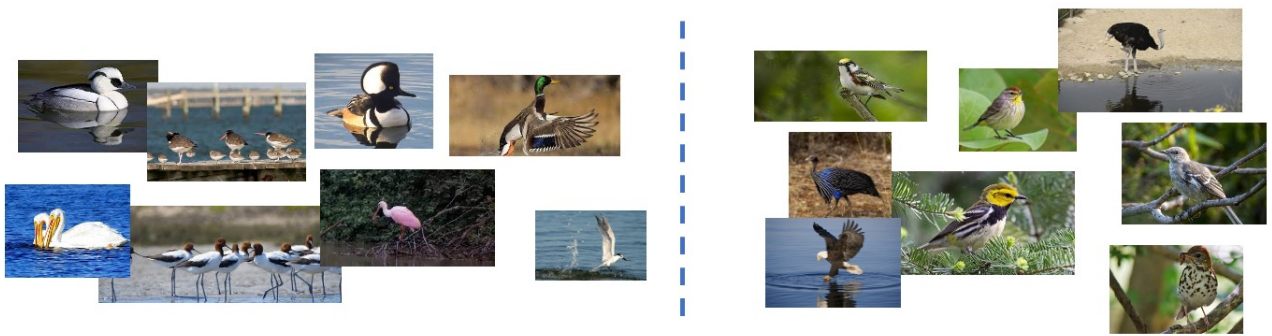
Very active research area, lots of algorithmic solutions.

- An example is Distributionally Robust Optimization. Here instead of minimizing the average loss (as we do with ERM), we minimize the worst loss across some known set of groups within the data.

(DRO)

$$\text{ERM: } \min_{\theta} \frac{1}{n} \sum_{i=1}^{n} \ell(f(x_i), y_i) \quad \Big| \quad \text{DRO: } \min_{\theta} \max_{\{\text{groups in data}\}} \frac{1}{|y_i|} \sum_{x \in y_i} \ell(f(x_i), y_i)$$

Usually, the best solution (if possible) is to collect more representative data.



Waterbirds vs. Landbirds

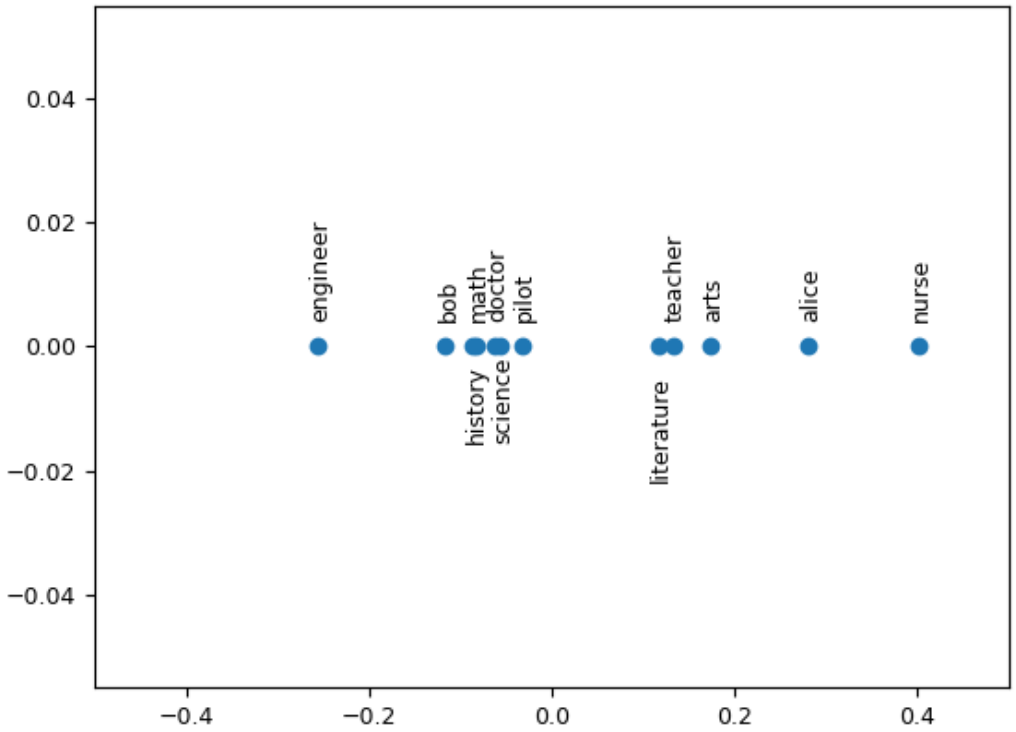
# Lesson: Don’t assume model is generalizing

- By now, you understand generalization when test distribution = train distribution
- However, this can be frequently violated for real-world applications
- **Important to test the model on different kinds of data, and understand limitations of models trained on certain data**


Fairness

# ML models can show biases against certain sub-populations

You saw a small example of this in the HW4 word embedding question:



# The ML loop

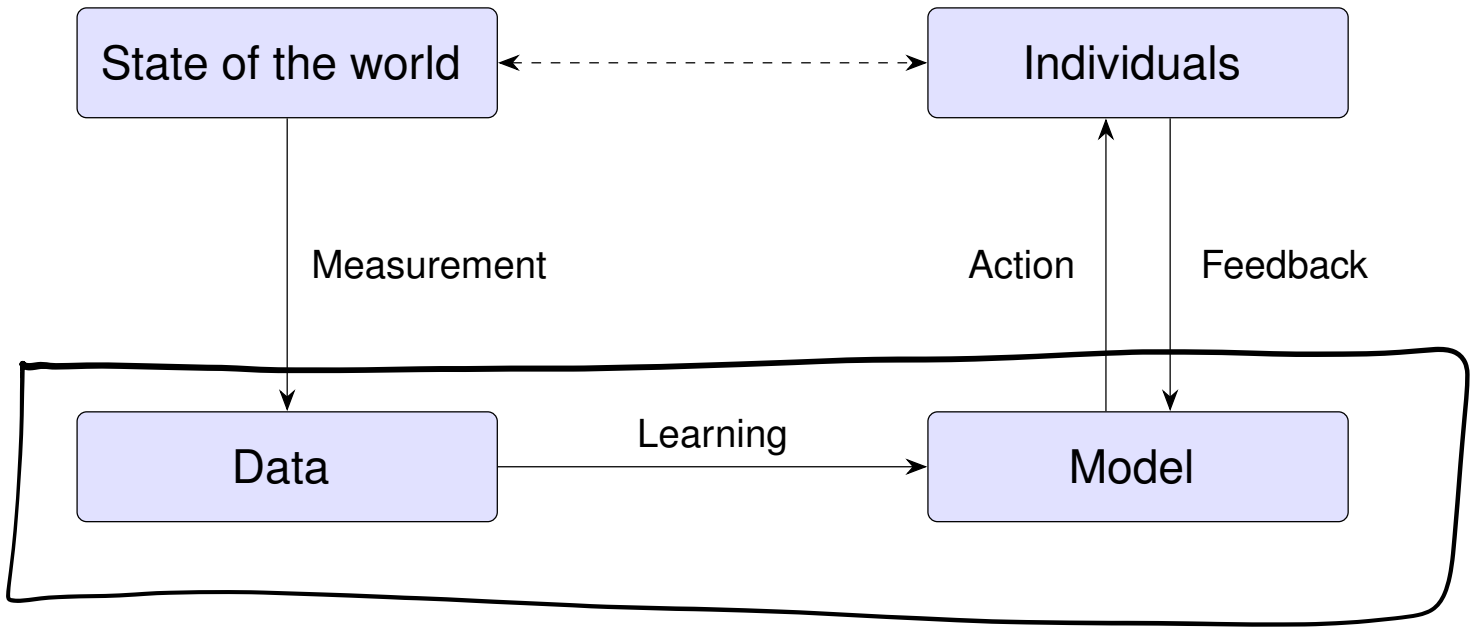



Fig. based on the book *Fairness And ML: Limitations and Opportunities*

# Unfairness could arise in various ways

- Unequal accuracy: The model may have poor performance on certain sub-populations or demographics
- Biased predictions: The predictions of the model could exhibit biases across different demographics
- Representation farm: The system may reinforce existing stereotype or biases
- …

# Unfairness could arise in various ways

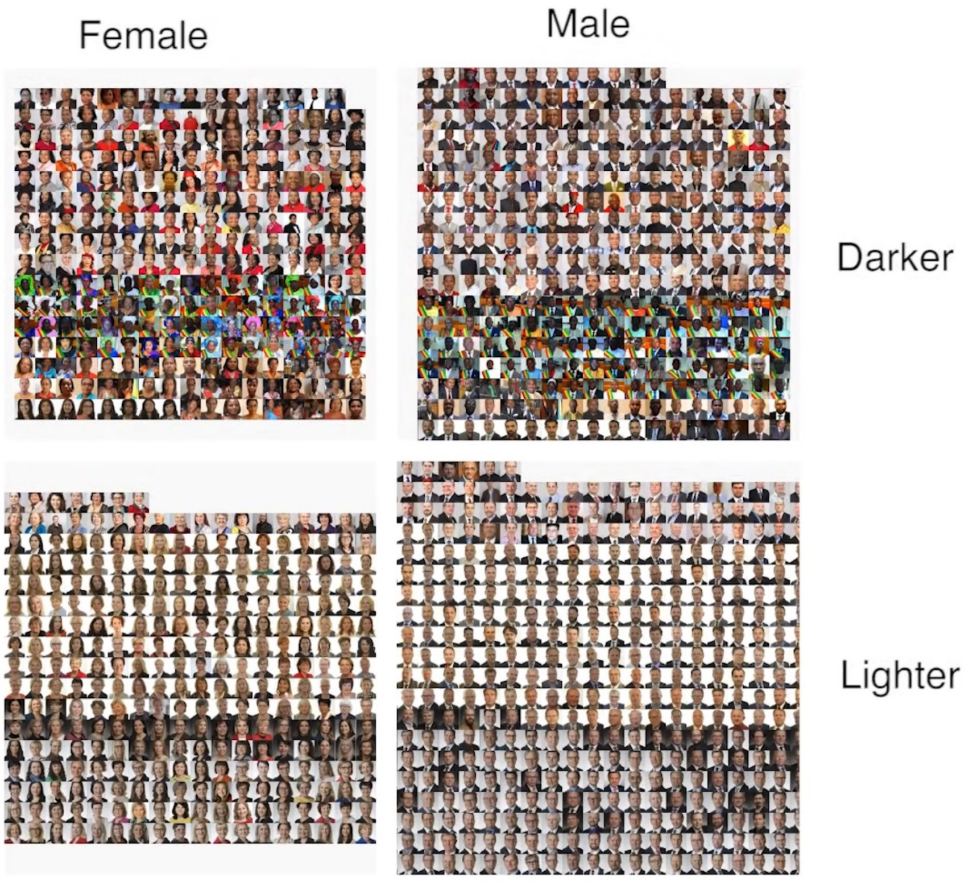
- Unequal accuracy: The model may have poor performance on certain sub-populations or demographics
- Biased predictions: The predictions of the model could exhibit biases across different demographics
- Representation farm: The system may reinforce existing stereotype or biases
- …

# Unequal accuracy: The GenderShades project

Models can do well on average but not on sub-populations

http://gendershades.org/

# How well do facial recognition tools perform on various demographics?

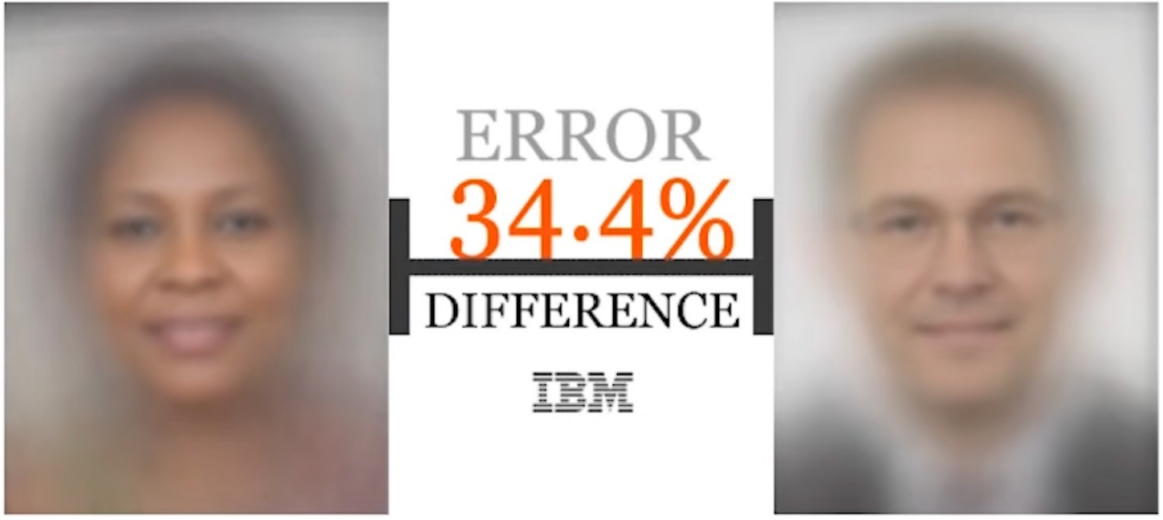



http://gendershades.org/

# Ans: *Not very well*

| | TYPE I | TYPE II | TYPE III | TYPE IV | TYPE V | TYPE VI |
|---|---|---|---|---|---|---|
| Microsoft | 1.7% | 1.1% | 3.3% | 0% | 23.2% | 25.0% |
| IBM | 5.1% | 7.4% | 8.2% | 8.3% | 33.3% | **46.8%** |
| FACE++ | 11.9% | 9.7% | 8.2% | 13.9% | 32.4% | **46.5%** |

# Ans: *Not very well*



# Mitigating harm due to unequal accuracy

- The problem of unequal accuracy of sub-groups bears similarities to the problem of ensuring the algorithm does well on distributional shifts (original distribution -> distribution with more weight on a particular demographic)
- As for distributional shifts and spurious correlations, getting more representative data is the best solution
- Algorithmic approaches also exist, similar to what we discussed for distributional shifts

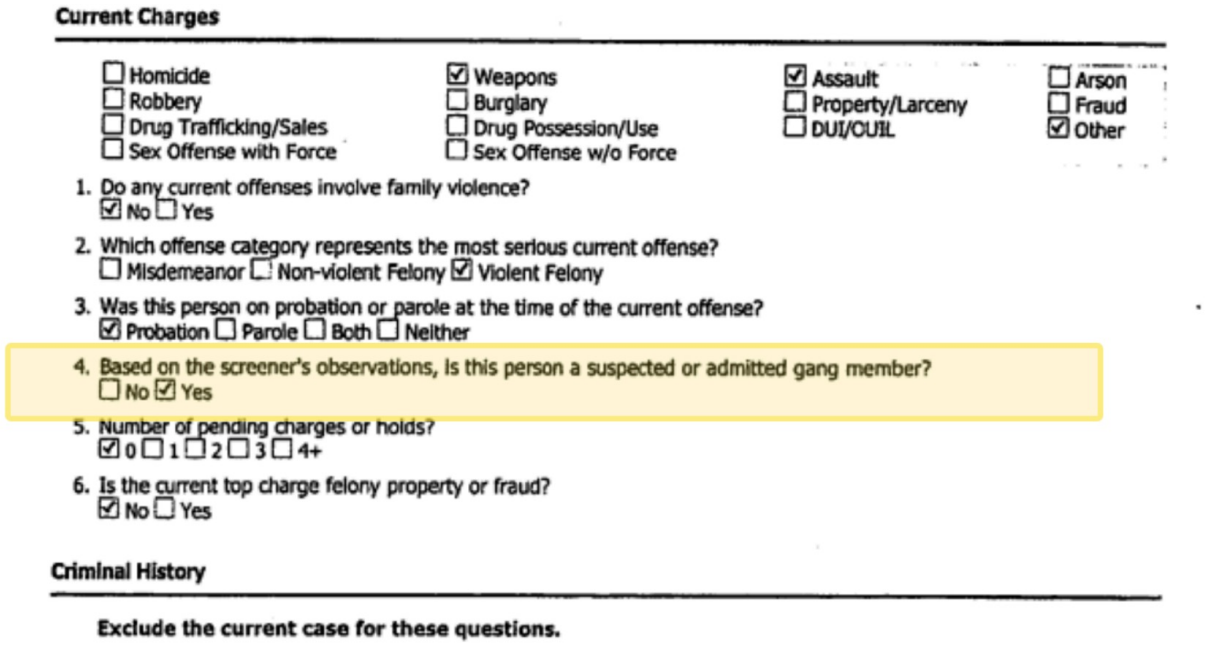
new dist.: uniform on darker skin women

# Unfairness could arise in various ways

- Unequal accuracy: The model may have poor performance on certain sub-populations or demographics
- Biased predictions: The predictions of the model could exhibit biases across different demographics
- Representation farm: The system may reinforce existing stereotype or biases
- …

# Bias in predictions: The COMPAS software

- COMPAS is a proprietary software used by many judicial systems to determine the risk that someone arrested for a crime again commits a crime in the future
- Used for decisions such as for deciding bail



**Current Charges**

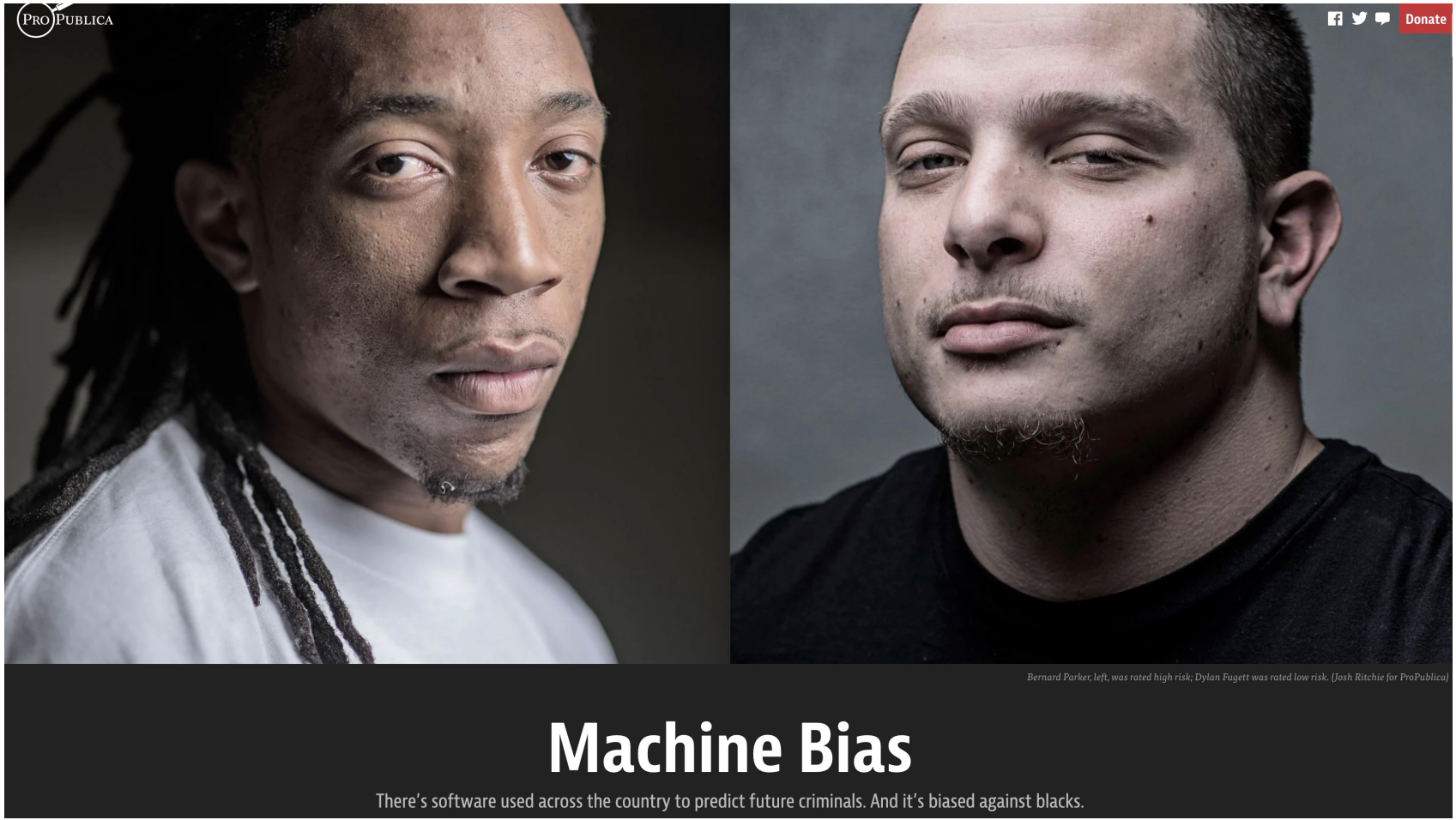
| | | | |
|---|---|---|---|
| ☐ Homicide | ☑ Weapons | ☑ Assault | ☐ Arson |
| ☐ Robbery | ☐ Burglary | ☐ Property/Larceny | ☐ Fraud |
| ☐ Drug Trafficking/Sales | ☐ Drug Possession/Use | ☐ DUI/CUIL | ☑ Other |
| ☐ Sex Offense with Force | ☐ Sex Offense w/o Force | | |

1. Do any current offenses involve family violence?
   ☑ No ☐ Yes
2. Which offense category represents the most serious current offense?
   ☐ Misdemeanor ☐ Non-violent Felony ☑ Violent Felony
3. Was this person on probation or parole at the time of the current offense?
   ☑ Probation ☐ Parole ☐ Both ☐ Neither
4. Based on the screener's observations, is this person a suspected or admitted gang member?
   ☐ No ☑ Yes
5. Number of pending charges or holds?
   ☑ 0 ☐ 1 ☐ 2 ☐ 3 ☐ 4+
6. Is the current top charge felony property or fraud?
   ☑ No ☐ Yes

**Criminal History**

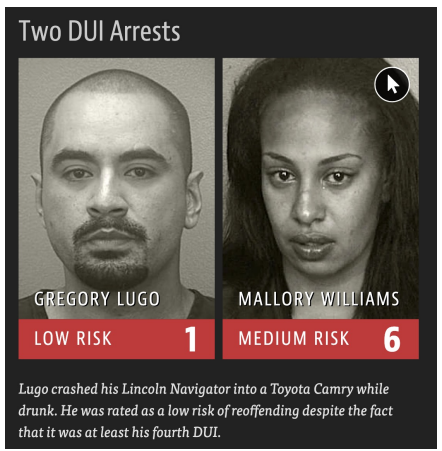
**Exclude the current case for these questions.**

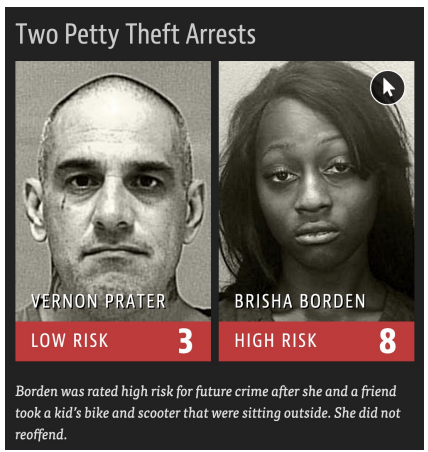
# Biases in COMPAS




https://www.propublica.org/article/machine-bias-risk-assessments-in-criminal-sentencing






*"In forecasting who would re-offend, the algorithm made mistakes with black and white defendants at roughly the same rate but in very different ways.*

- *The formula was particularly likely to falsely flag black defendants as future criminals, wrongly labeling them this way at almost twice the rate as white defendants.*
- *White defendants were mislabeled as low risk more often than black defendants."*

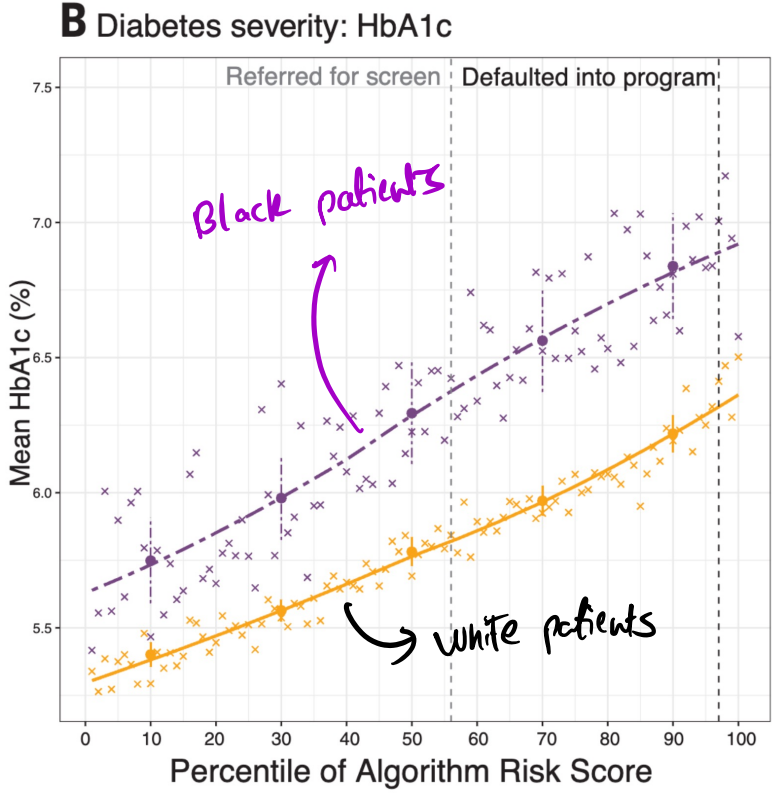
There are also inherent tensions here: the COMPAS algorithm is biased in one way and unbiased in another, and it may be impossible to simultaneously be unbiased in both.

https://www.propublica.org/article/machine-bias-risk-assessments-in-criminal-sentencing

# Bias in predictions: Predicting disease severity

Quoting from the paper:

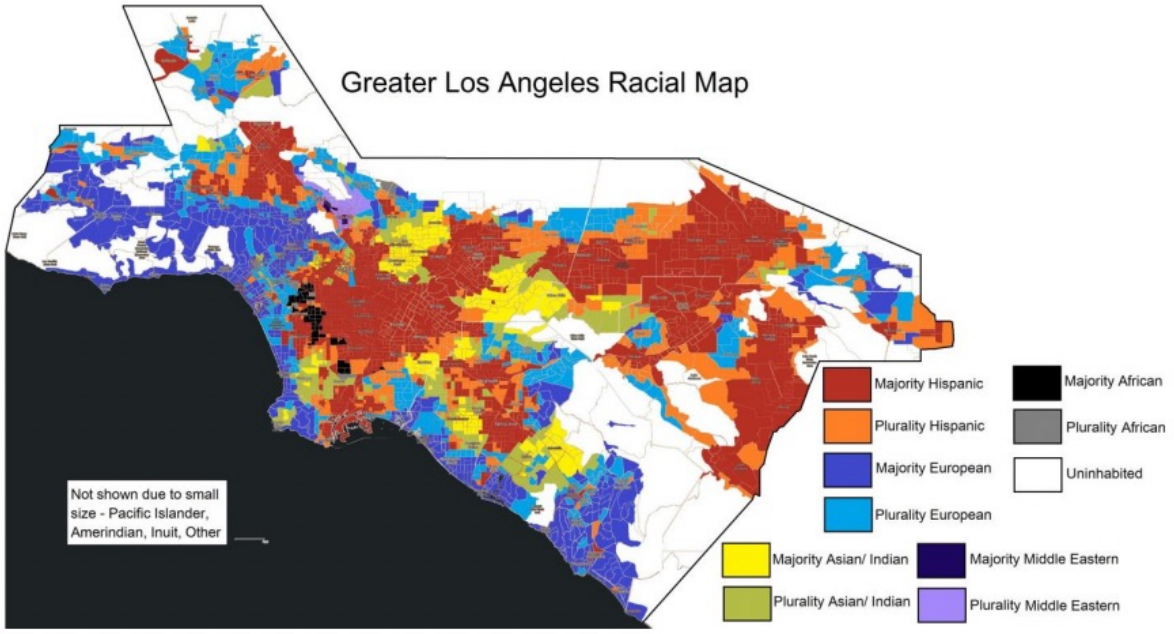
- Health systems rely on commercial prediction algorithms to identify and help patients with complex health needs.
- A widely used algorithm affecting millions of patients, exhibits significant racial bias: At a given risk score, Black patients are considerably sicker than White patients, as evidenced by signs of uncontrolled illnesses.
- Remedying this disparity would increase the percentage of Black patients receiving additional help from 17.7 to 46.5%.
- Bias arises because the algorithm predicts health care costs rather than illness, but unequal access to care means that on aggregate less money was spent caring for Black patients than for White patients.




Dissecting racial bias in an algorithm used to manage the health of populations, Obermeyer et al., Science 2019

# How to obtain fair classifiers?

Observation: No fairness by just excluding sensitive attributes
Why? Sensitive attribute can often be reconstructed from other features




Zip code has a lot of information about race

# Ensuring fairness in classification: **Group & Individual fairness notions**

Two broad classes of fairness notions in classification:

**Individual fairness:** Algorithm treats **similar individuals similarly**

**Group fairness:** Algorithm is **"unbiased" on protected groups** (such as race, gender etc.)

# Individual fairness

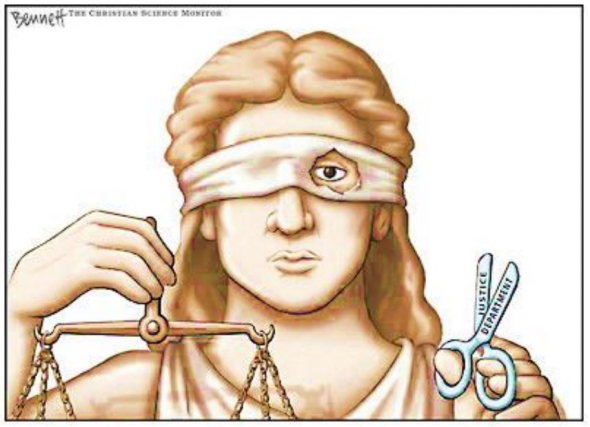
Define a ***metric*** $d(x, x')$ for the similarity between any two individuals $x$ and $x'$.

e.g.: $d(x, x') = \| x - x' \|_2$

If classifier predicts $p(x)$ as the probability of label being one for $x$, if

$$|p(x) - p(x')| \leq \mu \, d(x, x'),$$

then predictions of the classifier are individually fair with parameter $\mu$.





If these two individuals are similar, then their risk scores should be similar.

Fairness Through Awareness. Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, Richard Zemel. 2011

# Group fairness

Group fairness notions require that the models predictions obey certain properties over protected groups (e.g. by race, gender).

Many different notions have been proposed

- Statistical parity
- Equalized odds
- Calibration across groups

# Statistical parity

Binary classification setup (e.g. admitting a student to a degree program)

- Classifier $f$
- Datapoint $(x, y)$
- Sensitive attribute $a \in \{0,1\}$

Statistical parity: $\Pr_x[f(x) = 1 \mid a = 1] = \Pr_x[f(x) = 1 \mid a = 0]$

In words: **Predictions are independent of sensitive attribute**

E.g., admit equal fraction of men or women into program

Can be too strong if labels and sensitive attribute are not independent.

E.g. if women are more likely to be qualified for that degree program than men

# Equalized odds

Same binary classification setup (e.g. admitting student to degree program)

- Classifier $f$
- Datapoint $(x, y)$
- Sensitive attribute $a \in \{0,1\}$

Equalized odds:

$$\Pr_x[f(x) = 1 \mid a = 1, y = 1] = \Pr_x[f(x) = 1 \mid a = 0, y = 1]$$
$$\Pr_x[f(x) = 0 \mid a = 1, y = 0] = \Pr_x[f(x) = 0 \mid a = 0, y = 0]$$

In words: **Recall for both $y = 1$ and $y = 0$ is the same for both groups**

Also equivalent to saying: Conditioned on label, prediction is independent of sensitive attribute

*Recall* for class 1 = $\Pr_{x,y}[f(x) = 1 \mid y = 1]$

# Equalized odds

E.g. Professor Snape has to admit students to his Advanced Potions class.

100 students apply from Gryffindor (80% are qualified)

| | Qualified | Unqualified |
|---|---|---|
| Accepted | 60 | 5 |
| Rejected | 20 | 15 |
| Total | 80 | 20 |

100 students apply from Slytherin (40% are qualified)

| | Qualified | Unqualified |
|---|---|---|
| Accepted | 30 | 15 |
| Rejected | 10 | 45 |
| Total | 40 | 60 |

*Is Prof. Snape fair based on*
*(i) statistical parity,*
*(ii) equalized odds?*

Statistical parity :

$$\Pr_{x,y}[f(x) = 1 \mid a = 1] = \Pr_{x,y}[f(x) = 1 \mid a = 0]$$

Equalized odds:

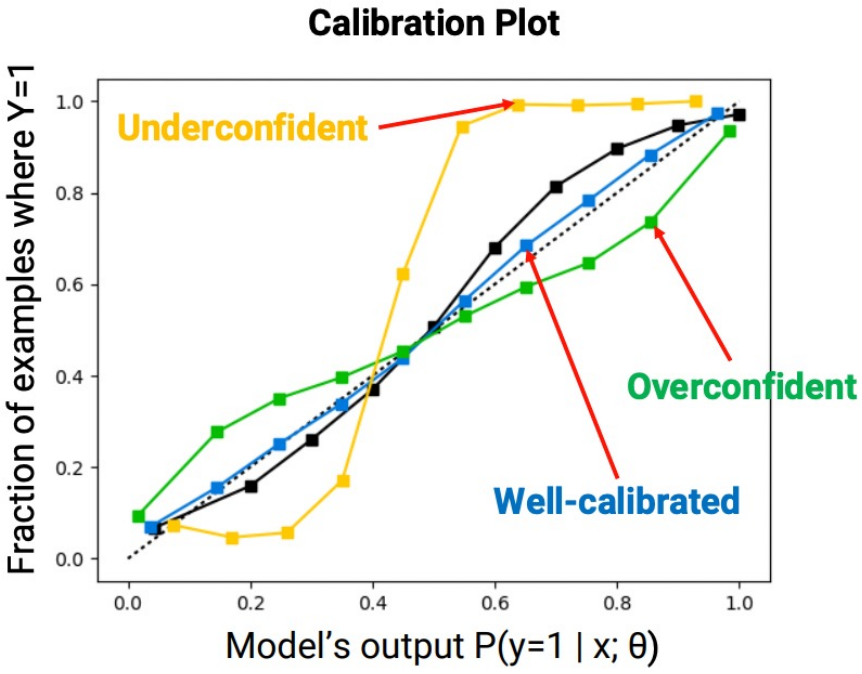
$$\Pr_{x}[f(x) = 1 \mid a = 1, y = 1] = \Pr_{x}[f(x) = 1 \mid a = 0, y = 1]$$
$$\Pr_{x}[f(x) = 0 \mid a = 1, y = 0] = \Pr_{x}[f(x) = 1 \mid a = 0, y = 0]$$

# Calibration across groups

Calibration: A model $f$ for binary classification is calibrated if

$$\Pr_{x,y}[y = 1 \mid f(x) = \alpha] = \alpha$$

Informally, this says that “predictions mean what they should”

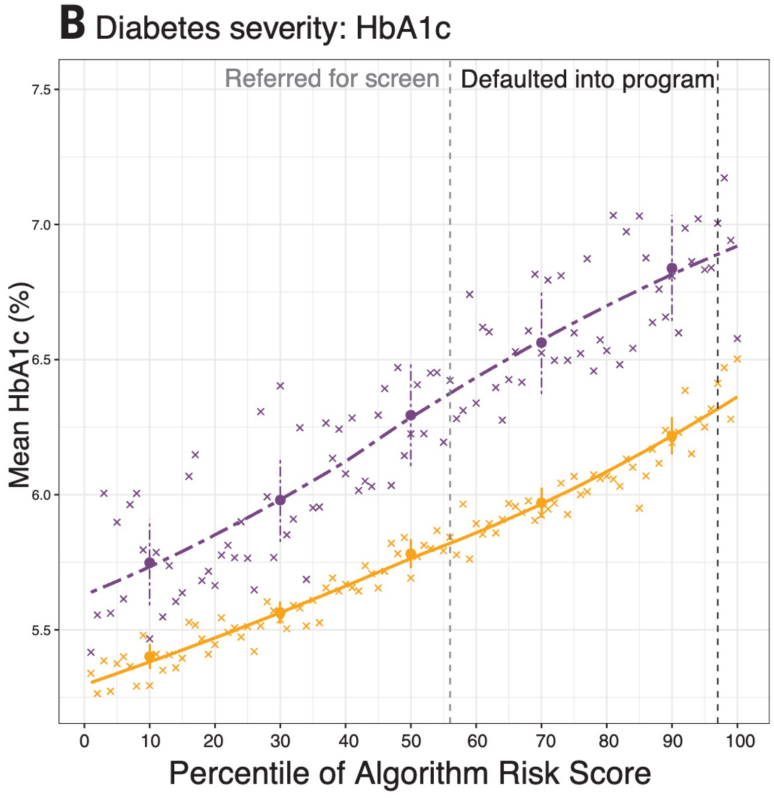


# Calibration across groups

Multi-calibration: A model $f$ for binary classification is calibrated for groups defined by sensitive attribute $a$ if

$$\Pr_{x,y}[y = 1 \mid f(x) = \alpha, a = 1] = \alpha,$$
$$\Pr_{x,y}[y = 1 \mid f(x) = \alpha, a = 0] = \alpha.$$

Informally, this says that "predictions mean what they should for each group"

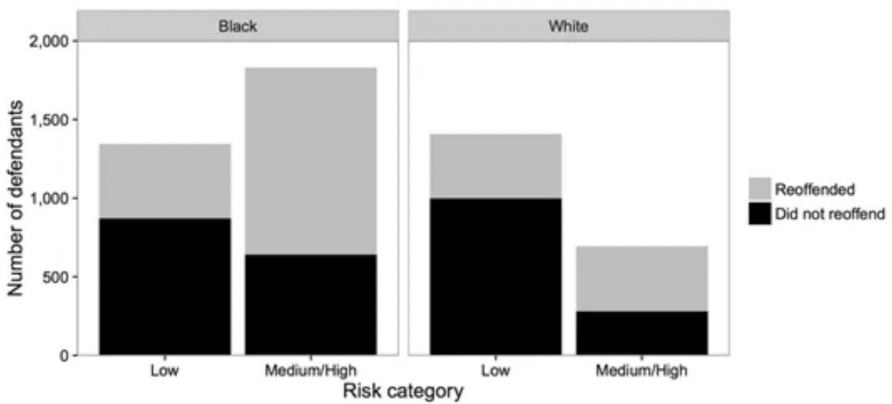


# Group fairness notions: Can we satisfy them all?

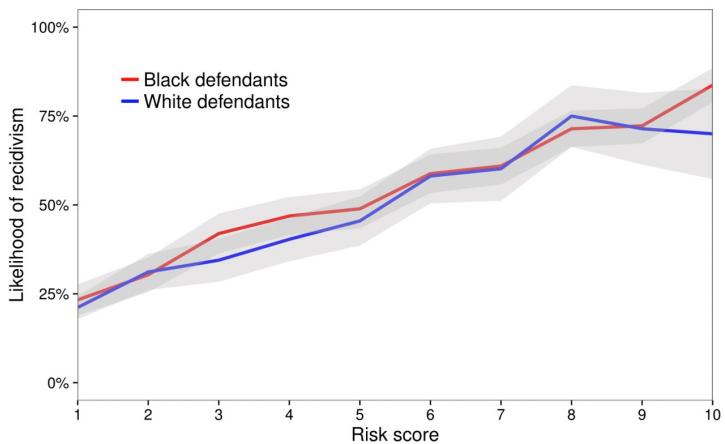
We saw three notions: statistical parity, equalized odds, calibration across groups
Can we satisfy all of them together? ***No!***

In our example from Hogwarts, Prof. Snape was fair in terms of equalized odds but unfair in terms of statistical parity. This tension between different notions arises in real data too.

**COMPAS: Unfair** because black defendants who did not recommit crime are assigned higher score (i.e. does not obey equalized odds)




**COMPAS: Fair** because probability of recommitting crime is similar for a given risk score, for both groups (i.e. is calibrated)

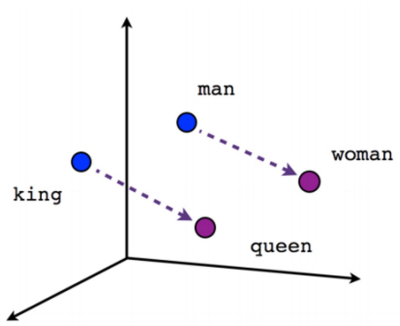



https://medium.com/soal-food/what-makes-an-algorithm-fair-6ad64d75dd0c

# Unfairness could arise in various ways

- Unequal accuracy: The model may have poor performance on certain sub-populations or demographics
- Biased predictions: The predictions of the model could exhibit biases across different demographics
- Representation farm: The system may reinforce existing stereotype or biases
- …

# Bias in representation: Word embeddings




Male-Female

Word analogy questions:

man: woman :: king : ??

$$\overrightarrow{\text{woman}} - \overrightarrow{\text{man}} \approx \overrightarrow{\text{queen}} - \overrightarrow{\text{king}}$$

$$\overrightarrow{\text{woman}} - \overrightarrow{\text{man}} \approx \overrightarrow{\text{homemaker}} - \overrightarrow{\text{computer programmer}}$$

**Gender stereotype *she-he* analogies.**

| | | |
|---|---|---|
| sewing-carpentry | register-nurse-physician | housewife-shopkeeper |
| nurse-surgeon | interior designer-architect | softball-baseball |
| blond-burly | feminism-conservatism | cosmetics-pharmaceuticals |
| giggle-chuckle | vocalist-guitarist | petite-lanky |
| sassy-snappy | diva-superstar | charming-affable |
| volleyball-football | cupcakes-pizzas | hairdresser-barber |

**Gender appropriate *she-he* analogies.**

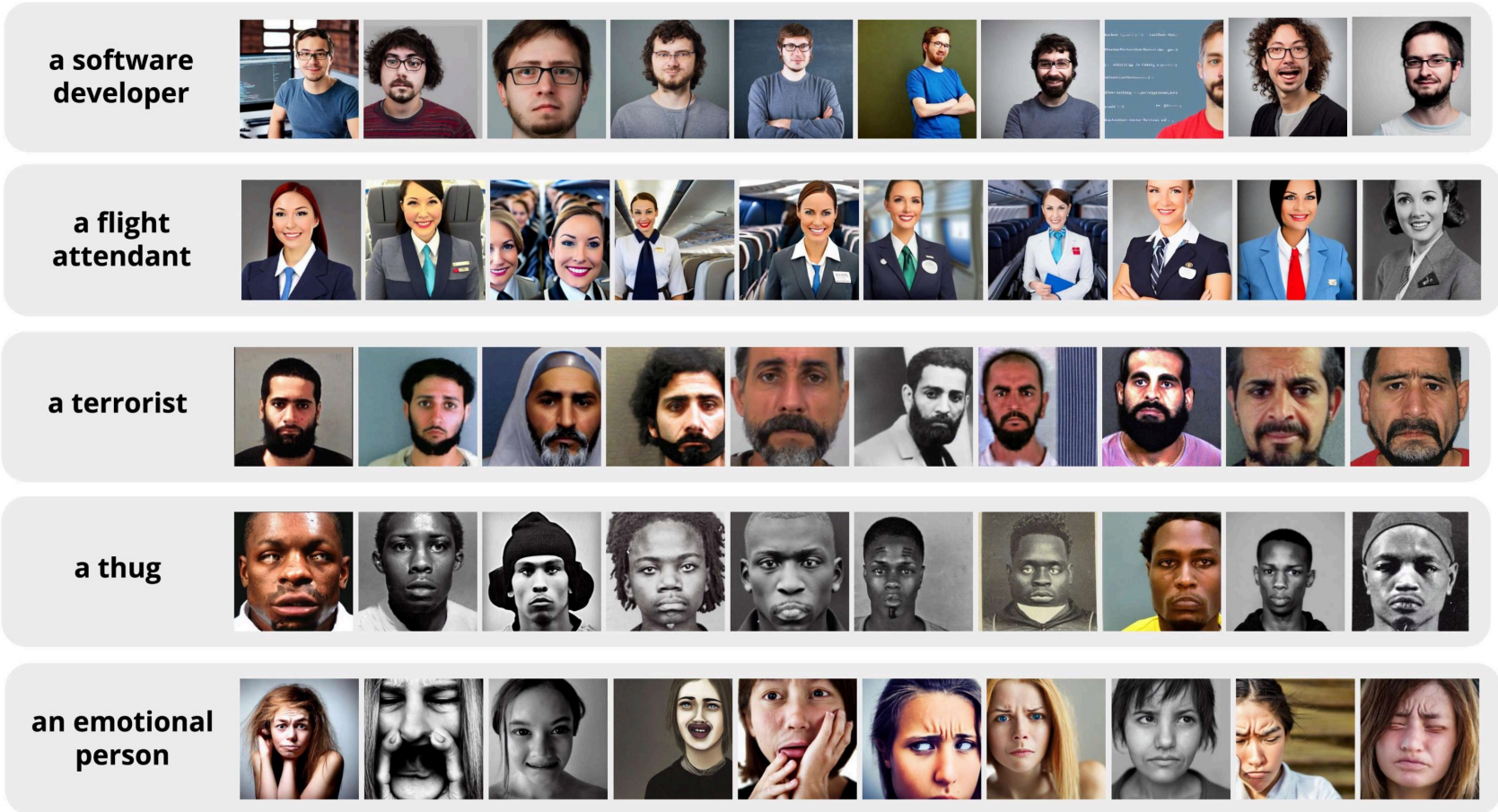
| | | |
|---|---|---|
| queen-king | sister-brother | mother-father |
| waitress-waiter | ovarian cancer-prostate cancer | convent-monastery |

From *Man is to Computer Programmer as Woman is to Homemaker?* Debiasing Word Embeddings, Bolukbasi et al. '16

# Bias in representation: Machine Translation

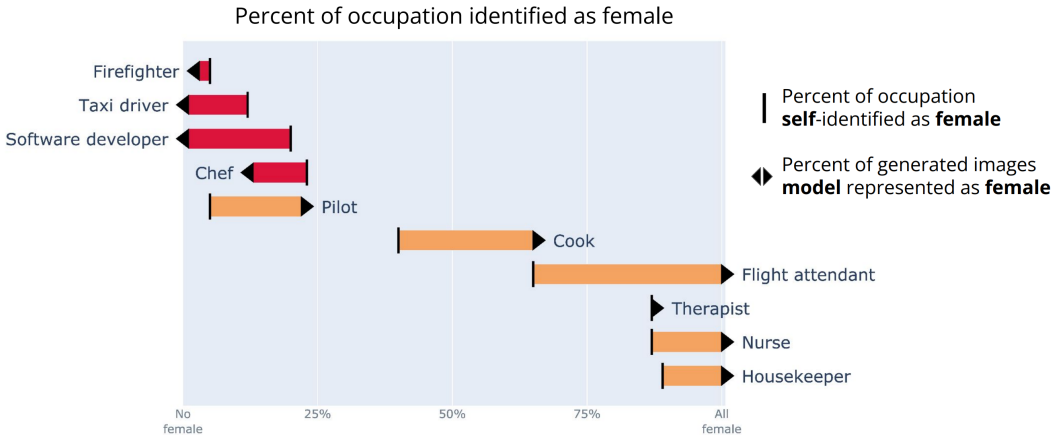

- Hindi does not have gendered pronouns
- Machine translation model seems to pick on existing stereotypes (likely from its training data), and rely on them
- Some efforts to mitigate such biases: https://research.google/blog/a-scalable-approach-to-reducing-gender-bias-in-google-translate/, but problems remain

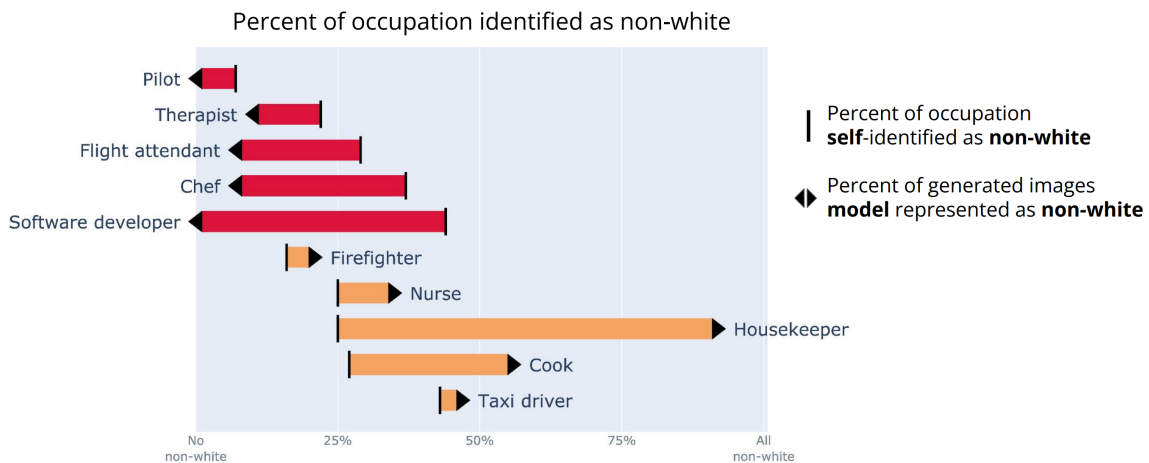
# Bias in representation: Image generation




Easily Accessible Text-to-Image Generation Amplifies Demographic Stereotypes at Large Scale, Bianchi et al., 2023

# Model amplifies existing biases




Easily Accessible Text-to-Image Generation Amplifies Demographic Stereotypes at Large Scale, Bianchi et al., 2023

# Some more instances of algorithmic bias

Aug 19, 2020 - Technology

## How an AI grading system ignited a national controversy in the U.K.

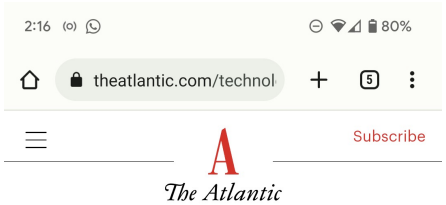


Bryan Walsh, author of Axios Future

Illustration: Eniola Odetunde/Axios

A huge controversy in the U.K. over an algorithm used to substitute for university-entrance exams highlights problems with the use of AI in the real world.

Link to article

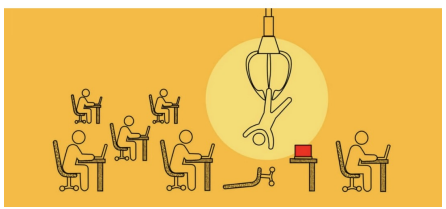



TECHNOLOGY

## It Was Supposed to Detect Fraud. It Wrongfully Accused Thousands Instead.

How Michigan's attempt to automate its unemployment system went horribly wrong

By Stephanie Wykstra and Undark



Link to article

# Some more instances of algorithmic bias

The New York Times

## *There Is a Racial Divide in Speech-Recognition Systems, Researchers Say*

Technology from Amazon, Apple, Google, IBM and Microsoft misidentified 35 percent of words from people who were black. White people fared much better.

Give this article

Amazon's Echo device is one of many similar gadgets on the market. Researchers say there is a racial divide in the usefulness of speech recognition systems. Grant Hindsley for The New York Times

Link to article

MIT Technology Review

Featured Topics Newsletters Events Podcasts Sign in Subscribe

ARTIFICIAL INTELLIGENCE

## LinkedIn's job-matching AI was biased. The company's solution? More AI.

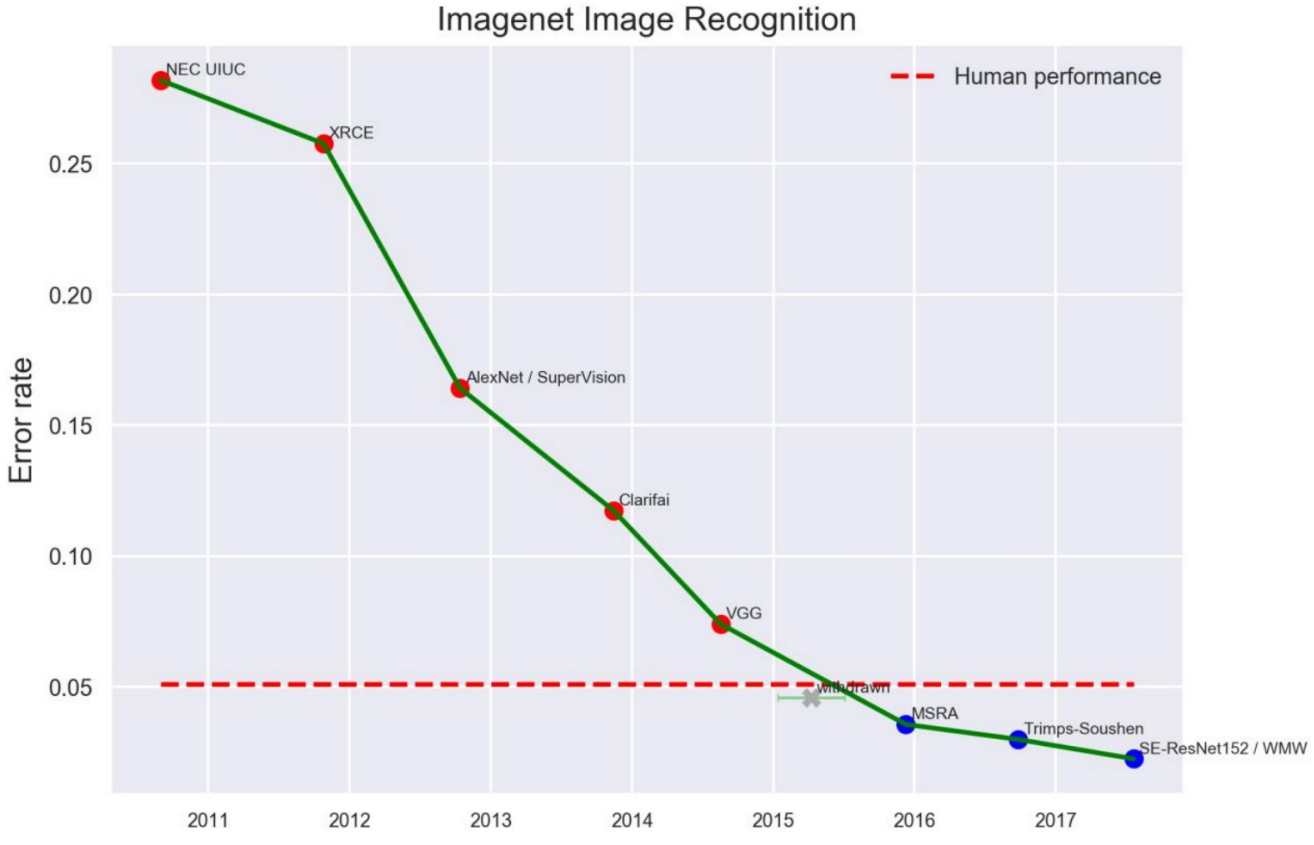
ZipRecruiter, CareerBuilder, LinkedIn—most of the world's biggest job search sites use AI to match people with job openings. But the algorithms don't always play fair.

By Sheridan Wall & Hilke Schellmann June 23, 2021

Link to article


# Adversarial examples

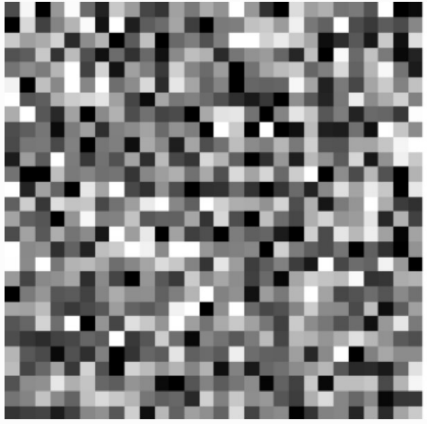
# Previously: CNNs are great at image classification



# However, ML can also be very sensitive to small variations in the input

Pig
(90% confidence)

+



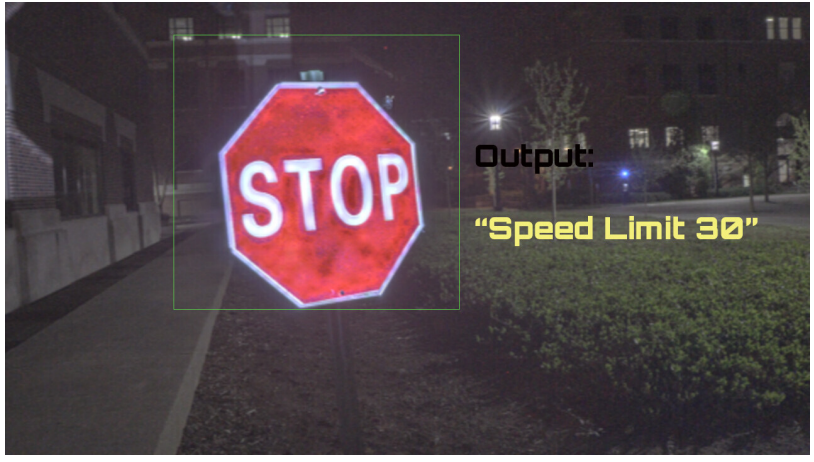
Small amount of
*adversarial* noise

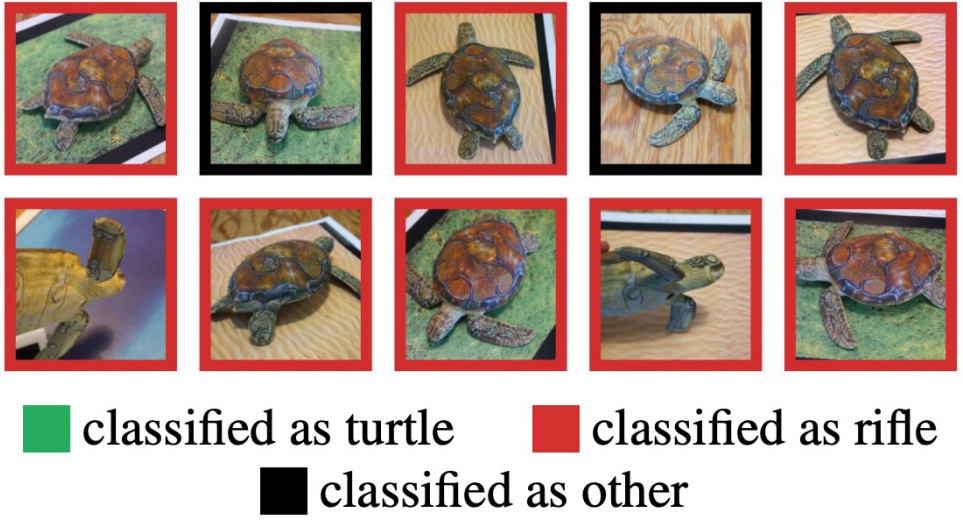
=

Airplane!
(99.9% confidence)

ML is so great, it can make pigs fly!!

# These are known as ***adversarial examples***



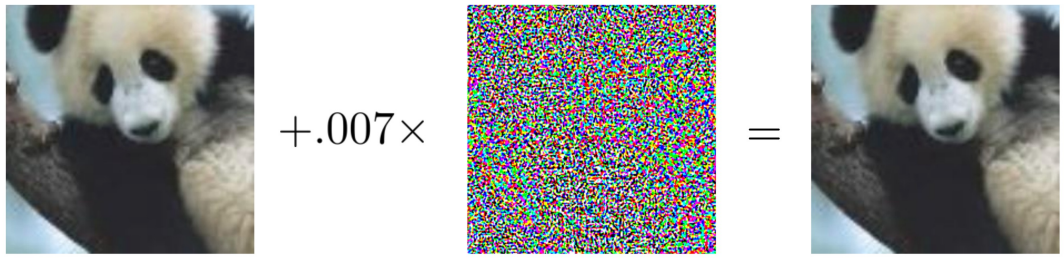




Adversarial examples have been shown to also hold for real-world tasks.

They are an issue because

1. Can pose potential security risks
2. Indicate that even though models are good, they don't quite work the same way as we do

# Adversarial examples: More formal setup




Panda
58% confidence

Nematode
8% confidence

Gibbon
99% confidence

Adversary: Given an image $x$ and classifier $f(x)$, comes up with some other image $x'$ which is "similar" to $x$, such that $f(x) \neq f(x')$.

How to define similarity? One notion is small perturbations based on some norm. Here let's consider the simple case of the $\ell_2$ norm, so for image $x$ find image $x'$ such that

$$\| x - x' \|_2 \leq \epsilon,$$

where $\epsilon$ is the allowed perturbation level.

# How should the adversary come up with an attack?

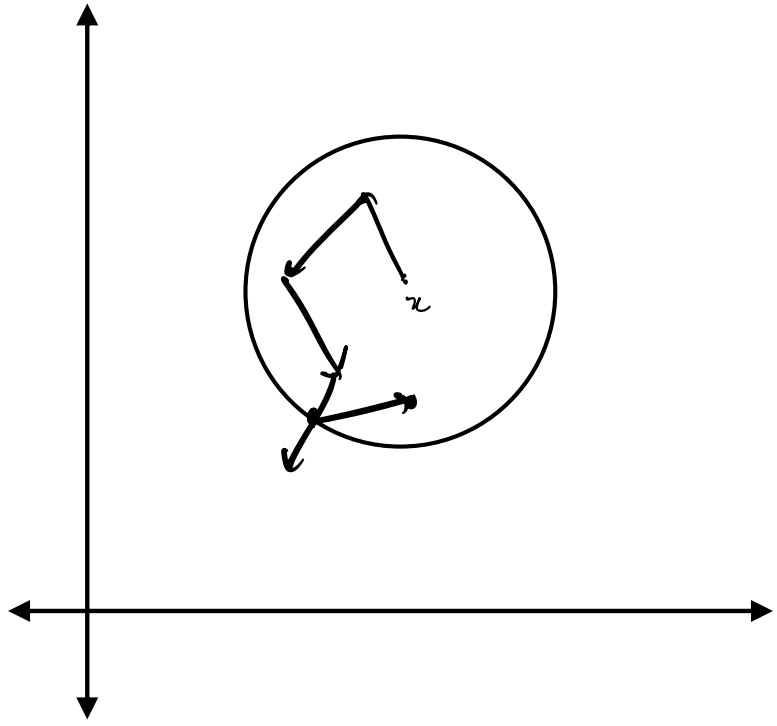
Adversary's formal goal: Given an image $x$ and classifier $f(x): x \rightarrow \{0,1\}$ , find some other image $x'$ such that

- $f(x) \neq f(x')$
- $x' \in B_\epsilon(x), B_\epsilon(x) = \{x' \; such \; that \; \| x - x' \|_2 \leq \epsilon\}$

One solution: Adversary finds the gradient *with respect to the input* $x$, and chooses the perturbation which changes the loss $\ell(f(x), y)$ the most locally.

Repeat some number of times:

1. Update $x_{new} = x + \eta \, \nabla_x \ell(f(x), y)$
2. If $x_{new}$ is outside the allowed perturbation region, "project" back into region.



# How to defend against adversarial examples?

Naïve strategy: **Do data augmentation by adding random noise to original inputs**

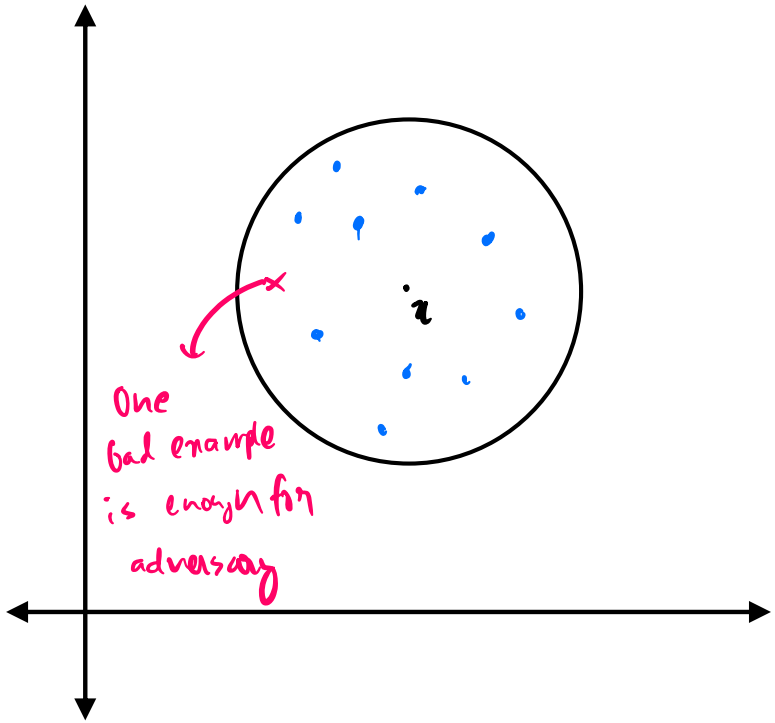
Issue: **Adversary might still be able to find one datapoint $x'$ within perturbation region such that $f(x) \neq f(x')$**

Better strategy:

**Mimic the adversary's strategy to add the particular point $x'$ which has a different label from $x$**

Training objective:

$$\min_{\theta} \sum_{all\ points\ x} \max_{x' \in B_\epsilon(x)} \ell(f(x'), y)$$



CONFIDENTIAL?

Privacy, interpretability, ethics …

# Privacy & Denonymization

Many companies and organizations release or exchange data to spur research interest, build better models etc.

Often, the data is “anonymized" before being released. But does anonymization actually work?

A story from the 90s:

An insurance company, GIC, in Massachusetts decided to release "anonymized" data on state employees that showed every single hospital visit. A graduate student found the records of the Governor of Massachusetts by associating the data with public vote roll data.

*“87 percent of all Americans can be uniquely identified using only three bits of information: ZIP code, birthdate, and sex.”*


https://arstechnica.com/tech-policy/2009/09/your-secrets-live-online-in-databases-of-ruin/

# Privacy & Denonymization

The Netflix prize:

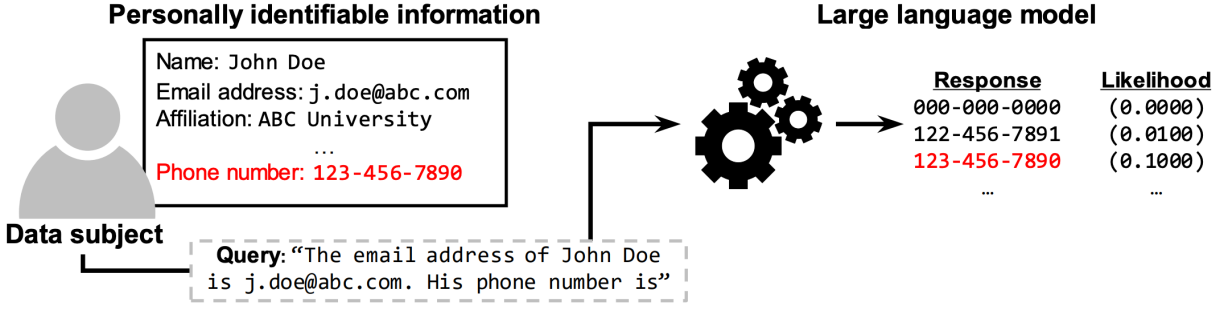
- Launched in 2006, $1M cash prize
- Dataset: 100 million movie ratings from nearly 500 thousand Netflix subscribers on a set of 17770 movies. Each data point corresponds to (anonymized user id, movie, date of rating, rating).
- Researchers were able to de-anonymize some of the subscribers by linking their rating with ratings on IMDB!
- Some Netflix subscribers had also publicly rated an overlapping set of movies on IMDB under their real identities.
- Lawsuit against Netflix, subsequent competition was cancelled.


From the book *Fairness And ML: Limitations and Opportunities*

# Privacy & Denonymization

In some cases, it is possible to recover some of the original training data of the model using only API access to the model. The following (left) is an example of an image recovered by an attacker who only knows the name of the person, and the original training image (right) from [1]

Some evidence that LLMs could also leak private information:




[1] Model Inversion Attacks that Exploit Confidence Information and Basic Countermeasures, Fredrikson et al., 2015
[2] ProPILE: Probing Privacy Leakage in Large Language Models, Kim et al., 2023,

# Interpretability and transparency: Why it is important

**Back to COMPAS:**

Glenn Rodríguez was denied parole because of a high risk score from COMPAS, despite being a “model of rehabilitation”.

However, there was an error in one of the entries to the COMPAS system.

Since the system was proprietary and black-box, he could not determine the exact effect this error had and challenge the score.

More broadly, interpretability seems crucial for applications such as healthcare, policy etc.

https://washingtonmonthly.com/2017/06/11/code-of-silence/

Also see: When a Computer Program Keeps You in Jail, NYTimes, Link

# Ethics in ML

``*Ethics is a study of what are good and bad ends to pursue in life and what it is right and wrong to do in the conduct of life",* Introduction to Ethics, John Deigh

Consider the following case-study on an application of ML.

**Goal:** Identify sexual orientation from facial features

**Training data:** Photos downloaded from a popular American dating website. All white, with gay and straight, male and female, all represented evenly

**Method:** A deep learning model was used to extract facial features + grooming features; then a logistic regression classifier to make prediction

**Result:** Accuracy: 81% for men, 74% for women

From Jieyu Zhao's class, "Ethics in NLP"

# Is this an ethical application of ML?

What are potential issues?

- **Scientific Accuracy**: Sexual identity is complex, and cannot be accurately predicted by physical characteristics alone. Also is subjective and can change over time.
- **Misuse and harm**: In many countries, being gay is punishable, in some places by death penalty
- **Cost of misclassification is high:** Could affect employment, relationships etc.
- **Data is likely biased:** Trained model could amplify these biases

From Jieyu Zhao's class, "Ethics in NLP"

# To conclude, going back to the beginning of Lecture 1..

## This class:

- Understand the fundamentals
- Understand when ML works, its limitations, think critically

In particular,

- Study fundamental statistical ML methods (supervised learning, unsupervised learning, etc.)
- Solidify your knowledge with hand-on programming tasks
- Prepare you for studying advanced machine learning techniques

1. Examine your task
2. Examine your data
3. Examine your model

ML/AI can be very powerful, but should be used responsibly