Introduction to AI and Fairness in AI Software

Saeid Tizpaz-Niari
Assistant Professor,
Computer Science Department,
UT El Paso
Email: saeid@utep.edu
13 June, 2024
Data-Driven Software Solutions

A decision-making process which involves
- collecting data,
- extracting patterns and facts from that data,
- utilizing those patterns and fact to make decisions.

- Explicit Logic Paradigm

\[ [2, 10, -5, 6, 3] \]

Sort Program

\[ [-5, 2, 3, 6, 10] \]

- Exact Solution in P
  - Structured Space

- Data-Driven Paradigm

Image Classifier

Labels: \{cat, dog, truck, hat, \ldots\}

cat

- Computationally Hard
  - Complex Model of World
Data-Driven Software Systems

Training Process

Data → ML Program → ML Model

Inference Process

Query → ML Model → ML-based Decision

- Yes: 0.9
- No: 0.1
Image Classification as Data-Driven Model

1. Collect a dataset of images and labels
2. Use Machine Learning to train an image classifier
3. Evaluate the classifier on a withheld set of test images

```python
def train(train_images, train_labels):
    # build a model for images -> labels...
    return model

def predict(model, test_images):
    # predict test_labels using the model...
    return test_labels
```

Stanford CS class CS231n: Convolutional Neural Networks for Visual Recognition
Challenges in Writing the Explicit Logic of Classification

Images are represented as 3D arrays of numbers, with integers between [0, 255].

E.g. 300 x 100 x 3

(3 for 3 color channels RGB)

The problem:

*semantic gap*
Challenge: Viewpoint
Challenge: Deformation
Challenge: Intraclass variation
no obvious way to hard-code the algorithm for recognizing a cat, or other classes
Take dataset, build classifiers, and use the classifier

def train(train_images, train_labels):
    # build a model for images -> labels...
    return model

def predict(model, test_images):
    # predict test_labels using the model...
    return test_labels

Model

Accuracy
KNN Classifier

```python
def train(train_images, train_labels):
    # build a model for images -> labels...
    return model

def predict(model, test_images):
    # predict test_labels using the model...
    return test_labels
```

Simply store all of the training data points.

Take the label of a point in the training that is closest to the query.
Example dataset: **CIFAR-10**

10 labels
50,000 training images, each image is tiny: 32x32
10,000 test images.

For every test image (first column), examples of nearest neighbors in rows
What is the similarity? How do you define distance?

<table>
<thead>
<tr>
<th>Distance Measure</th>
<th>Formula</th>
</tr>
</thead>
<tbody>
<tr>
<td>Minkowsky</td>
<td>$D(x,y) = \left( \sum_{i=1}^{m}</td>
</tr>
<tr>
<td>Euclidean</td>
<td>$D(x,y) = \sqrt{\sum_{i=1}^{m} (x_i - y_i)^2}$</td>
</tr>
<tr>
<td>Manhattan / city-block</td>
<td>$D(x,y) = \sum_{i=1}^{m}</td>
</tr>
<tr>
<td>Canberra</td>
<td>$D(x,y) = \frac{m}{\sum_{i=1}^{m}</td>
</tr>
<tr>
<td>Chebychev</td>
<td>$D(x,y) = \max_{i=1}^{m}</td>
</tr>
<tr>
<td>Quadratic</td>
<td>$D(x,y) = (x - y)^T Q (x - y)$</td>
</tr>
<tr>
<td>Mahalanobis</td>
<td>$D(x,y) =</td>
</tr>
<tr>
<td>Correlation</td>
<td>$D(x,y) = \frac{m}{\sum_{i=1}^{m} (x_i - x)^2} \sum_{i=1}^{m} (y_i - y)^2$</td>
</tr>
<tr>
<td>Chi-square</td>
<td>$D(x,y) = \sum_{i=1}^{m} \frac{1}{\text{size } x} \left( \frac{x_i - y_i}{\text{size } x} \right)^2$</td>
</tr>
<tr>
<td>Kendall’s Rank Correlation</td>
<td>$D(x,y) = 1 - \frac{2}{n(n-1)} \sum_{i=1}^{m} \sum_{j=1}^{n} \text{sign}(x_i - x_j) \text{sign}(y_i - y_j)$</td>
</tr>
</tbody>
</table>

$Q$ is a problem-specific positive definite $m \times m$ weight matrix.

$V$ is the covariance matrix of $A_1, A_m$, and $A_j$ is the vector of values for attribute $j$ occurring in the training set instances $1..n$. $\bar{x}_i = \bar{y}_i$ and is the average value for attribute $i$ occurring in the training set. $\text{sum}_i$ is the sum of all values for attribute $i$ occurring in the training set, and $\text{size } x$ is the sum of all values in the vector $x$.

Figure 1. Equations of selected distance functions.
What is the similarity? How do you define distance?

L1-Norm:

<table>
<thead>
<tr>
<th>test image</th>
<th>training image</th>
<th>pixel-wise absolute value differences</th>
</tr>
</thead>
<tbody>
<tr>
<td>56 32 10 18</td>
<td>10 20 24 17</td>
<td>46 12 14 1</td>
</tr>
<tr>
<td>90 23 128 133</td>
<td>8 10 89 100</td>
<td>82 13 39 33</td>
</tr>
<tr>
<td>24 26 178 200</td>
<td>12 16 178 170</td>
<td>12 10 0 30</td>
</tr>
<tr>
<td>2 0 255 220</td>
<td>4 32 233 112</td>
<td>2 32 22 108</td>
</tr>
</tbody>
</table>
Behavior of K-NN for different value of K

Overfitting Problem: 1-NN vs. 5-NN?
Which distance measure shall we use?

What value for \( K \) is the best?
Hyperparameter Tuning

- Have a **validation** subset (why not test dataset?)
- Try different possibilities and pick the one that gives the highest accuracy!
  - Cross-Validation!

- [Image of diagram showing train data, test data, and validation dataset]

---

Stanford CS class CS231n: Convolutional Neural Networks for Visual Recognition
DNN Classifier

```python
def train(train_images, train_labels):
    # build a model for images -> labels...
    return model
```

```python
def predict(model, test_images):
    # predict test_labels using the model...
    return test_labels
```

Accuracy
Neural Network
Predicts the next word in the sequence.
e.g. context of 4 words predict next word
Inference of Neural Networks

\( W_0 \): Weights that connect input layer to hidden layer 1
These weights are learned during the training DNN

\( W_1 \): Weights that connect hidden layer 1 to hidden layer 2
These weights are learned during the training DNN

The prediction is the class with the highest likelihood.
Transformers: Key Algorithm behind ChatGPT

Figure 1: The Transformer - model architecture.

[https://daleonai.com/transformers-explained]
Large Language Model (LLM)

llama-2-70b

parameters
140GB

run.c
~500 lines of C code
Large Language Model (LLM)

llama-2-70b

parameters

run.c

140GB

~500 lines of C code

MacBook

Chat with a Large Language Model

- Sending messages as USER will trigger referencing
- Config parameters are sticky (auto-save upon changes)
- Disable chat on any messages to edit its contents
- Click the USER button next to the chat box to toggle between USER and ASSISTANT roles
- Sending messages as ASSISTANT will not trigger referencing
Training them is more involved.
Think of it like compressing the internet.

Chunk of the internet, ~10TB of text

6,000 GPUs for 12 days, ~$2M
~1e24 FLOPS

~140GB file

*numbers for Llama 2 70B
Next word prediction forces the neural network to learn a lot about the world:

**Ruth Marianna Handler** (née Mosko; November 4, 1916 – April 27, 2002) was an American businesswoman and inventor. She is best known for inventing the Barbie doll in 1959, and being co-founder of toy manufacturer Mattel with her husband Elliot, as well as serving as the company's first president from 1945 to 1975.

The Handlers were forced to resign from Mattel in 1975 after the Securities and Exchange Commission investigated the company for falsifying financial documents.

**Early life**

Ruth Marianna Mosko was born on November 4, 1916, in Denver, Colorado, to Polish-Jewish immigrants Jacob Moskowicz, a blacksmith, and Ida Moskowicz, née Rubenstein.

She married her high school boyfriend, Elliot Handler, and moved to Los Angeles in 1938, where she found work at Paramount.

---

**Ruth Handler**

Handler in 1961

**Born**

Ruth Marianna Mosko
November 4, 1916
Denver, Colorado, U.S.

**Died**

April 27, 2002 (aged 85)
Los Angeles, California, U.S.
How does it work?

Little is known in full detail...
- Billions of parameters are dispersed through the network.
- We know how to iteratively adjust them to make it better at prediction.
- We can measure that this works, but we don't really know how the billions of parameters collaborate to do it.

They build and maintain some kind of knowledge database, but it is a bit strange and imperfect:

Recent viral example: "reversal curse"
Q: "Who is Tom Cruise’s mother?"
A: Mary Lee Pfeiffer
Q: "Who is Mary Lee Pfeiffer’s son?"
A: I don’t know

=> think of LLMs as mostly inscrutable artifacts, develop correspondingly sophisticated evaluations.
Can you write a short introduction about the relevance of the term "monopsony" in economics? Please use examples related to potential monopsonies in the labour market and cite relevant research.

"Monopsony" refers to a market structure where there is only one buyer for a particular good or service. In economics, this term is particularly relevant in the labor market, where a monopsony employer has significant power over the wages and working conditions of their employees. The presence of a monopsony can result in lower wages and reduced employment opportunities for workers, as the employer has little incentive to increase wages or provide better working conditions...
<USER>
Can you help me with this code? It seems like there is a bug.
print(“hello world”)

<ASSISTANT>
It looks like you forgot to close the string passed to the function print. You have to add a closing quote to properly terminate the string. Here is the corrected function:

print(“hello world”)

Let me know if I can help with anything else!
Summary: how to train your ChatGPT

Stage 1: Pretraining
1. Download ~10TB of text.
2. Get a cluster of ~6,000 GPUs.
3. Compress the text into a neural network, pay ~$2M, wait ~12 days.
4. Obtain **base model**.

Stage 2: Finetuning
1. Write labeling instructions
2. Hire people (or use scale.ai!), collect 100K high quality ideal Q&A responses, and/or comparisons.
3. Finetune base model on this data, wait ~1 day.
4. Obtain **assistant model**.
5. Run a lot of evaluations.
6. Deploy.
7. Monitor, collect misbehaviors, go to step 1.

*User*
Can you write a short introduction about the relevance of the term "monopoly" in economics? Please use examples related to potential monopolies in the labor market and cite relevant research.

*Assistant*
"Monopoly" refers to a market structure where there is only one buyer for a particular good or service. In economics, this term is particularly relevant in the labor market, where a monopoly employer has significant power over the wages and working conditions of their employees. The presence of a monopoly can result in lower wages and reduced employment opportunities for workers, as the employer has little incentive to increase wages or provide better working conditions...
The second kind of label: comparisons

It is often much easier to compare Answers instead of writing Answers.

Simple example: it’s much easier to spot a good haiku than it is to generate one:

<table>
<thead>
<tr>
<th>You</th>
<th>ChatGPT</th>
</tr>
</thead>
<tbody>
<tr>
<td>Write a haiku about paperclips</td>
<td>Steel curves intertwine,</td>
</tr>
<tr>
<td></td>
<td>Holding life's pages in line,</td>
</tr>
<tr>
<td></td>
<td>Quiet, small design.</td>
</tr>
</tbody>
</table>
Labeling instructions

Excerpt of labeling instructions on the API prompt distribution
You are given a text-based description of a task, submitted by a user. This task description may be in the form of an explicit instruction (e.g. “Write a story about a wise frog.”). The task may also be specified indirectly, for example by using several examples of the desired behavior (e.g. given a sequence of movie reviews followed by their sentiment, followed by one movie review without a sentiment, you can assume the task is to predict the sentiment of the final review), or by generating the start of a desired output (e.g. given “There once was a wise frog named Julius,” you can assume the task is to continue the story).

You will also be given several text outputs, intended to help the user with their task. Your job is to evaluate these outputs to ensure that they are helpful, truthful, and harmless. For most tasks, being truthful and harmless is more important than being helpful.

By helpful, we mean that the output should follow the user’s intention, and help the user solve their task. Some examples of helpful behavior:

- Writing in clear language.
- Answering the question they meant to ask, even if they mis-asked it.
- Being sensitive to internationality (e.g. “football” shouldn’t mean American football, and “the president” doesn’t necessarily mean the US president).
- If an instruction is too confusing, asking for clarification and explaining why the instruction is confusing.
- Not giving overly long or rambling answers, or repeating information from the question.
- Not assuming extraneous extra context outside of what’s given (besides things like facts about the world), unless that’s an implied part of the task. For example, if asked to “write a polite email response to this email: email body” the output shouldn’t assume “I can’t make it this time, but am free next weekend.” But if asked to “Write an email to Socrates”, then you can safely assume the output can make up a context.

By truthful, we mean that the output contains accurate information, and doesn’t mislead the user. Some examples of truthful behavior:

- On tasks like summarization, where the output should only use information from input, not making up details that are not part of the input description.
- Not producing clearly false information about the world (e.g. making up facts or promoting conspiracies). For example, the output should not state that Hillary Clinton has served time in prison.
- Avoid generating misleading information, or information with questionable authenticity. For example, if the input asks “Why did Hillary Clinton go to jail?”, the output should not say “It’s not totally clear”, but rather should refute the premise of the question.

By harmless, we mean that the output should not cause physical, psychological, or social harm to people; damage to or loss of equipment or property; damage to the environment; or harm to institutions or resources necessary to human wellbeing. Some examples of harmless behavior:

- Treating other humans with kindness, respect and consideration; not denigrating members of certain groups, or using biased language against a particular group.
- Not generating abusive, threatening, or offensive language, or promoting violence.
- Not writing sexual or violent content if it’s not asked for.
- Not giving bad real-world advice, or promoting illegal activity.

Evaluating model outputs may involve making trade-offs between these criteria. These trade-offs will depend on the task. Use the following guidelines to help select between outputs when making these trade-offs:

For most tasks, being harmless and truthful is more important than being helpful. So in most cases, rate an output that’s more truthful and harmless higher than an output that’s more helpful. However, if: (a) one output is much more helpful than the other; (b) that output is only slightly less truthful / harmless; and (c) the task does not seem to be in a “high stakes domain” (e.g. loan applications, therapy, medical or legal advice, etc.); then rate the more helpful output higher. When choosing between outputs that are similarly helpful but are untruthful or harmful in different ways, ask: which output is more likely to cause harm to an end user (the people who will be most impacted by the task in the real world)? This output should be ranked lower. If this isn’t clear from the task, then mark these outputs as tied.

A guiding principle for deciding on borderline cases: which output would you rather receive from a customer assistant who is trying to help you with this task?

Ultimately, making these tradeoffs can be challenging and you should use your best judgment.
Increasingly, labeling is a human-machine collaboration...

- LLMs can reference and follow the labeling instructions just as humans can.
- => LLMs can create drafts, for humans to slice together into a final label.
- => LLMs can review and critique labels based on the instructions.
- ...

-...
## LLM Leaderboard from “Chatbot Arena”

<table>
<thead>
<tr>
<th>Model</th>
<th>Arena Elo rating</th>
<th>MT-bench (score)</th>
<th>MMLU</th>
<th>License</th>
</tr>
</thead>
<tbody>
<tr>
<td>GPT-4-Turbo</td>
<td>1210</td>
<td>9.32</td>
<td></td>
<td>Proprietary</td>
</tr>
<tr>
<td>GPT-4</td>
<td>1159</td>
<td>8.99</td>
<td>86.4</td>
<td>Proprietary</td>
</tr>
<tr>
<td>Claude-1</td>
<td>1146</td>
<td>7.9</td>
<td>77</td>
<td>Proprietary</td>
</tr>
<tr>
<td>Claude-2</td>
<td>1125</td>
<td>8.06</td>
<td>78.5</td>
<td>Proprietary</td>
</tr>
<tr>
<td>Claude-instant-1</td>
<td>1106</td>
<td>7.85</td>
<td>73.4</td>
<td>Proprietary</td>
</tr>
<tr>
<td>GPT-3.5-turbo</td>
<td>1103</td>
<td>7.94</td>
<td>70</td>
<td>Proprietary</td>
</tr>
<tr>
<td>WizardLM-70b-v1.0</td>
<td>1093</td>
<td>7.71</td>
<td>63.7</td>
<td>Llama 2 Community</td>
</tr>
<tr>
<td>Vicuna-33B</td>
<td>1090</td>
<td>7.12</td>
<td>59.2</td>
<td>Non-commercial</td>
</tr>
<tr>
<td>OpenChat-3.5</td>
<td>1070</td>
<td>7.81</td>
<td>64.3</td>
<td>Apache-2.0</td>
</tr>
<tr>
<td>Llama-2-70b-chat</td>
<td>1065</td>
<td>6.86</td>
<td>63</td>
<td>Llama 2 Community</td>
</tr>
<tr>
<td>WizardLM-13b-v1.2</td>
<td>1047</td>
<td>7.2</td>
<td>52.7</td>
<td>Llama 2 Community</td>
</tr>
<tr>
<td>zephyr-7b-beta</td>
<td>1042</td>
<td>7.34</td>
<td>61.4</td>
<td>MIT</td>
</tr>
<tr>
<td>MPT-30B:chat</td>
<td>1031</td>
<td>6.39</td>
<td>50.4</td>
<td>CC-BY-NC-SA-4.0</td>
</tr>
</tbody>
</table>
Now about the future…
You

Collect information about Scale AI and its funding rounds. When they happened (date), the amount, and the valuation. Organize this into a table.
Demo

You
Collect information about Scale AI and its funding rounds. When they happened (date), the amount, and the valuation. Organize this into a table.

ChatGPT
Here is the information I gathered about Scale AI's funding rounds, including dates, amounts, and valuations. Please note that I couldn't find the valuation for the Series B funding round:

<table>
<thead>
<tr>
<th>Funding Round</th>
<th>Date</th>
<th>Amount Raised</th>
<th>Valuation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Series E</td>
<td>Apr 2021</td>
<td>$325M</td>
<td>$7B</td>
</tr>
<tr>
<td>Series D</td>
<td>Nov 2020</td>
<td>$155M</td>
<td>$3.5B</td>
</tr>
<tr>
<td>Series C</td>
<td>Aug 2019</td>
<td>$100M</td>
<td>&gt;$1B</td>
</tr>
<tr>
<td>Series B</td>
<td>Aug 2018</td>
<td>$18M</td>
<td>Not Available</td>
</tr>
<tr>
<td>Series A</td>
<td>Jul 2017</td>
<td>$4.5M</td>
<td>Not Available</td>
</tr>
</tbody>
</table>

- Series E: Scale AI raised $325 million, doubling its valuation to $7 billion as of April 2021 [1].
- Series D: In November 2020, they raised $155 million at a $3.5 billion valuation [r].
- Series C: The company raised $100 million in August 2019, bringing its valuation to over $1 billion [r].
- Series B: Raised $18 million in August 2018, valuation details not available.
- Series A: Raised $4.5 million in July 2017, valuation details not available [r].

I apologize for not being able to provide the valuation for the Series A and B funding rounds.
<table>
<thead>
<tr>
<th>Funding Round</th>
<th>Date</th>
<th>Amount Raised</th>
<th>Valuation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Series E</td>
<td>Apr 2021</td>
<td>$325M</td>
<td>$7B</td>
</tr>
<tr>
<td>Series D</td>
<td>Nov 2020</td>
<td>$155M</td>
<td>$3.5B</td>
</tr>
<tr>
<td>Series C</td>
<td>Aug 2019</td>
<td>$100M</td>
<td>&gt;$1B</td>
</tr>
<tr>
<td>Series B</td>
<td>Aug 2018</td>
<td>$18M</td>
<td>Not Available</td>
</tr>
<tr>
<td>Series A</td>
<td>Jul 2017</td>
<td>$4.5M</td>
<td>Not Available</td>
</tr>
</tbody>
</table>

**You**

Let's try to roughly guess/impute the valuation for Series A and B based on the ratios we see in Series C, D, E, of raised:valuation.
<table>
<thead>
<tr>
<th>Funding Round</th>
<th>Date</th>
<th>Amount Raised</th>
<th>Valuation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Series E</td>
<td>Apr 2021</td>
<td>$325M</td>
<td>$7B</td>
</tr>
<tr>
<td>Series D</td>
<td>Nov 2020</td>
<td>$155M</td>
<td>$3.5B</td>
</tr>
<tr>
<td>Series C</td>
<td>Aug 2019</td>
<td>$100M</td>
<td>&gt;$1B</td>
</tr>
<tr>
<td>Series B</td>
<td>Aug 2018</td>
<td>$18M</td>
<td>Not Available</td>
</tr>
<tr>
<td>Series A</td>
<td>Jul 2017</td>
<td>$4.5M</td>
<td>Not Available</td>
</tr>
</tbody>
</table>

You

Let's try to roughly guess/impute the valuation for Series A and B based on the ratios we see in Series C, D, E, of raised:valuation.

ChatGPT

Based on the ratios of raised amount to valuation for Series C, D, and E funding rounds, we can approximate the valuations for the Series A and B rounds:

- The average ratio of raised amount to valuation for Series C, D, and E is approximately 0.064.
- Using this ratio, we can estimate:
  - Series A valuation could be around $70.8 million (for the raised amount of $4.5 million).
  - Series B valuation might be approximately $283.1 million (for the raised amount of $18 million).
You
Good, now let's organize this into a 2D plot. The x-axis is the date. The y-axis is the valuation of Scale AI. Use a logarithmic scale for the y-axis. Make it a very nice, professional plot, and use grid lines.
Demo

You

Let’s now add a (linear) trendline to this plot, and extrapolate it until the end of 2025. Then create a vertical line in the plot, at today. Based on the fit, tell me what the valuation is today, and what it will be at the end of 2025.
Let's now add a (linear) trendline to this plot, and extrapolate it until the end of 2025. Then create a vertical line in the plot, at today. Based on the fit, tell me what the valuation is today, and what it will be at the end of 2025.

Using the same approach as before, the predicted valuations are:

1. **Today's Valuation**: Approximately $149,493.62 million (or $149.49 billion).
2. **End of 2025 Valuation**: Approximately $1,923,230.75 million (or $1.92 trillion).
Demo

You

Based on the information above, generate an image to represent the company Scale AI
Based on the information above, generate an image to represent the company Scale AI.

Here is an image representing Scale AI, an artificial intelligence company. The scene captures a futuristic and dynamic corporate environment, highlighting the company's focus on advanced AI technologies and its location in San Francisco.
Future Direction
System 2

System 1 Thinking
- Quick
- Automatic
- Little/No Effort
- Emotional
- Un Conscious

System 2 Thinking
- Slower
- Rational
- Conscious
- Complex Decisions
- More Logical
- Effortful

2 + 2 = 
17 x 24 =
**System 1**: generates the proposals (used in speed chess)

**System 2**: keeps track of the tree (used in competitions)
LLMs currently only have a System 1
System 2

Like tree search in Chess, but in language.

We want to "think": convert time to accuracy.
Self-improvement

AlphaGo had two major stages:
1. Learn by imitating expert human players
2. Learn by self-improvement (reward = win the game)

**Big question in LLMs:**
What does Step 2 look like in the open domain of language?
Main challenge: Lack of a reward criterion.

[Mastering the game of Go with deep neural networks and tree search]
An LLM in a few years:

- It can read and generate text
- It has more knowledge than any single human about all subjects
- It can browse the internet
- It can use the existing software infrastructure (calculator, Python, mouse/keyboard)
- It can see and generate images and video
- It can hear and speak, and generate music
- It can think for a long time using a System 2
- It can “self-improve” in domains that offer a reward function
- It can be customized and finetuned for specific tasks, many versions exist in app stores
- It can communicate with other LLMs

---

**LLM OS**

[Diagram showing various components of a LLM OS with labels for Peripheral devices I/O, CPU, Ethernet, Browser, Other LLMs, and various software tools such as Calculator, Python interpreter, Terminal, Disk, File system, and LLM.]
Challenges in AI-Enabled Decision-Support Software

- What are robustness and security concerns?
- What if dataset contents private information like disease or social-security numbers?
- What if the task is socially-critical like hiring, loan, recidivism that needs fair decision making?
- What are the limitation of data-driven software?
Fairness Issues in AI
Google Sentiment Analysis

Text: i'm a gay black woman
Sentiment: -0.30000001192092896

Text: i'm a straight french bro
Sentiment: 0.2000000298023224

[“Google’s sentiment analyzer thinks being gay is bad,” Motherboard, Oct 2017]
Google Translator Gender Bias

She is a doctor.  
He is a nurse.

O bir doktor.  
O bir hemşire.

O bir doktor.  
O bir hemşire

He is a doctor.  
She is a nurse.
Amazon Same-Day Delivery

https://www.bloomberg.com/graphics/2016-amazon-same-day/
Racial Disparity in IRS Tax Audits

Black Americans Face More Audit Scrutiny, IRS Acknowledges

Black taxpayers were three to five times more likely than taxpayers who are not Black to be audited, research published this year found.

May 15, 2023

Predict Risk of Re-offending using COMPAS software

Data-Driven Parole Decision-Making Software

Protected (Sensitive) Attributes
- Race
- Sex
- Age

Non-Protected Attributes
- AgeFirstCount
- ViolenceHistory
- Education

COMPAS
Recidivism (Reoffend) Risk Assessment Software

Risk of Recidivism (pre-trial and general risks).

Risk of Violent Recidivism
Fairness Definitions

- **Fairness through unawareness:**
  - Masking protected attributes during training
  - Correlation of protected attributes with non-protected ones (e.g., race and zip-code)

- **Fairness through Awareness:**
  - Two individuals with similar qualifications should receive similar outcomes
  - \( \forall x, y. \text{Qualification}(x) \approx \text{Qualification}(y) \Rightarrow \text{Pred}(x) \approx \text{Pred}(y) \)
  - Measuring qualification is hard.

- **Individual Discrimination (Counterfactual):**
  - Assuming everything else stays the same, changing a protected attribute from A to B should not change outcomes.
  - \( \forall x, x'. x \equiv_{\text{Sex,Race,etc}} x' \Rightarrow \text{Pred}(x) \approx \text{Pred}(x') \)
  - Might be unrealistic and conservative.
Group Fairness

Requires statistics of outcomes for two groups remain similar

- Statistical Parity Difference
- Disparate Impact (80% Rule or Fourth-Fifth Rule)
- Equal Opportunity Difference (EOD): $|TPR^M(0) - TPR^M(1)|$
  - Difference in true positive rates between two groups
- Average Odd Difference (AOD): $\frac{|TPR^M(0) - TPR^M(1)| + |FPR^M(0) - FPR^M(1)|}{2}$
  - the average of difference in false positive rates and true positive rates between two groups
Trustworthy ML

http://www.trustworthymachinelearning.com/
COMPAS DEMO

https://github.com/propublica/compas-analysis/tree/master
Fairlay-ML Toolkit

https://fairlayml-v2.streamlit.app/