Legal

This material is based upon work funded and supported by the Department of Defense under Contract No. FA8702-15-D-0002 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center.

DM19-0537

Introduction to AI/ML

April Galyardt

Machine Learning Research Scientist

Software Engineering Institute
Carnegie Mellon University
Pittsburgh, PA 15213

The SEI is a DoD R&D Federally Funded Research and Development Center

Established in 1984 at Carnegie Mellon University

~650 employees (ft + pt), of whom about 70% are engaged in technical work

Initiated CERT cybersecurity program in 1988

Offices in Pittsburgh and DC, with several locations near customer facilities

~$140M in annual funding (~$20M DoD Line funding for research)

Objectives

Be able to navigate the jargon

Understand some of the key principles involved

Be able to ask useful questions

When people say “AI”, they might be referring to either a narrow AI or a general AI.

Narrow AI

An algorithm to carry out one particular task.

Examples:

Google Translate

Autonomous Vehicles

Spam Filters

General AI

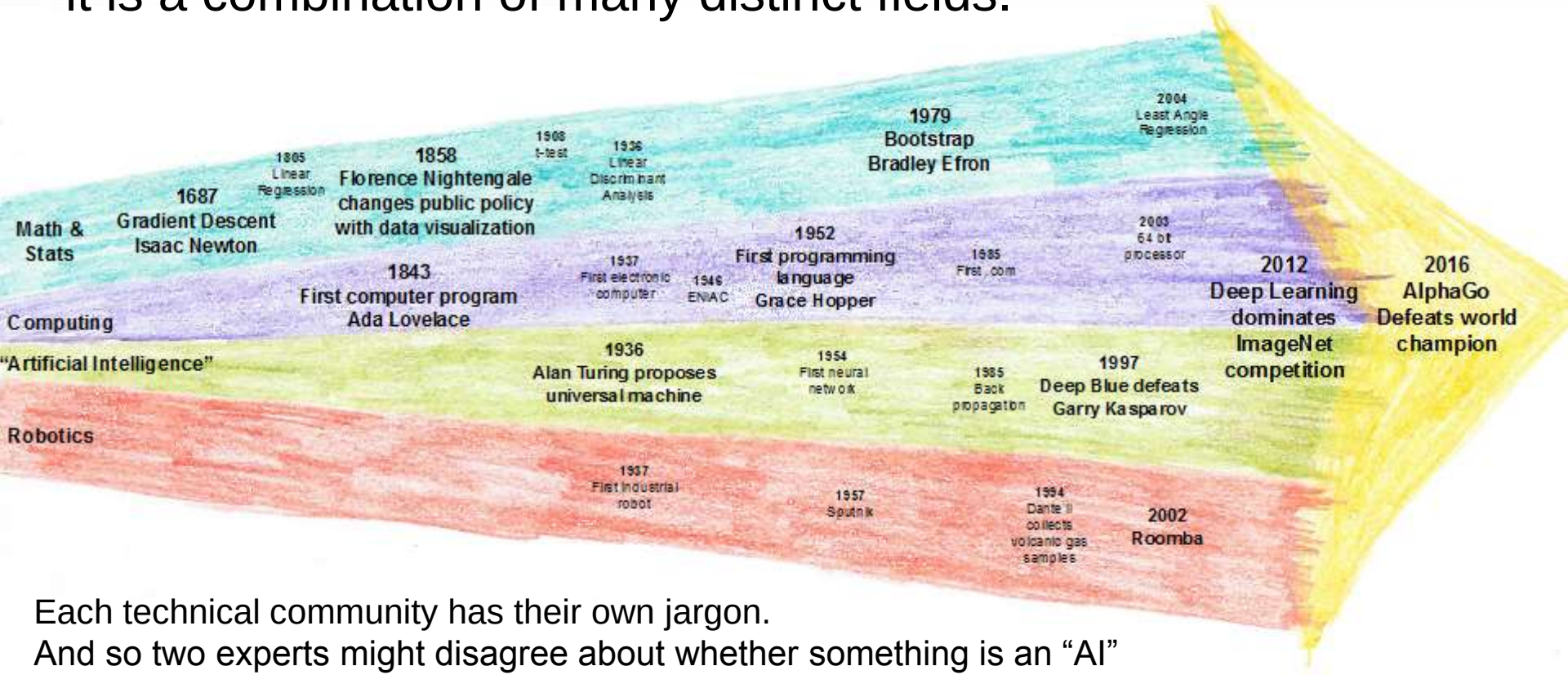
A machine that exhibits human intelligence.

Examples:

Doesn’t exist yet.

E.g., One goal should be to be able to monitor the understanding of its audience and make adjustments. This is something human children can generally do by the age of 5.

Modern Artificial Intelligence is not a monolith, it is a combination of many distinct fields.



Each technical community has their own jargon.
And so two experts might disagree about whether something is an "AI"

Example of language confusion

“I want to use _____ to build a speech recognition algorithm.”

Different people would fill in the blank differently, and mean the same thing:

“Artificial Intelligence”	– In this example, probably refers to using ML to set up a narrow AI.
“Machine Learning”	– A large set of methods for extracting information from data, including neural networks.
“Neural Network”	– is one type of machine learning algorithm, originally patterned after how humans were thought to think.
“Deep Learning”	– A strategy for building neural networks, that are easy to write down, and can model complex behavior.

Recommendation: When somebody says “AI,” ask them to be more specific and define what they mean.

Related definitions: Statistics & Data Science

Statistics is the art and science of learning from data.

Data science refers to managing and analyzing large amounts of data.

Statistics + Data Management

Machine Learning

If you ask a **Computer Scientist** to define **Machine Learning** you might get:

The science of getting computers to act without being explicitly programmed, by extracting information from data.

Statistics + Automation

If you ask a **Statistician** to define **Machine Learning**, you might get:

A set of models & algorithms for extracting information from data, developed primarily after 1980 that require intense computational power.

Statistics + Computing Power

Implication: Definitions matter. Different things fit in these two definitions.

ML and AI are not the same thing, but there is overlap.

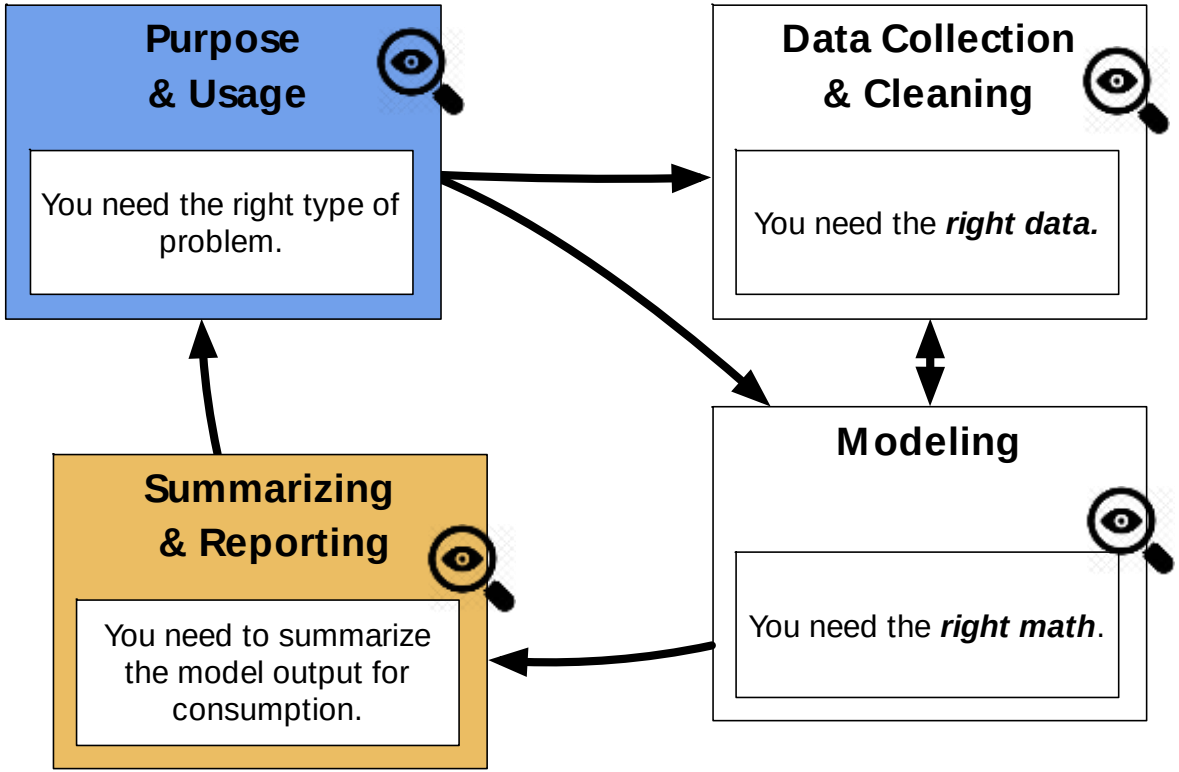
These could all be considered narrow AI, but do not use any ML:

- Navigation (Google Maps) – Calculates optimum path based on an algorithm.
- Roomba – uses sensors to map the room, then follows a pre-programmed algorithm to most efficiently cover it.
- Non-player Characters (NPCs) in video games. – Dialog and interactions between player characters & NPCs are pre-programmed and constrained for game play.

ML is currently one of the best ways to create a successful narrow AI from data:

- Image recognition (used in autonomous vehicles)
- Speech recognition (used in Alexa, Siri, & Echo)
- Recommendation Engines (used by Netflix & Amazon)

Process for an ML/Data Science Project



For most projects, data collection & cleaning requires a majority of the time and effort.

Checklist to identify good candidates for ML projects.

Can you state your problem as either:

- I would like to use ___ data to predict ___.
- I would like to understand the structure of the features recorded in ___ data.
- I would like to optimize ____ well defined process.

Is it a large scale problem?

Have you already done exploratory analysis on available data?

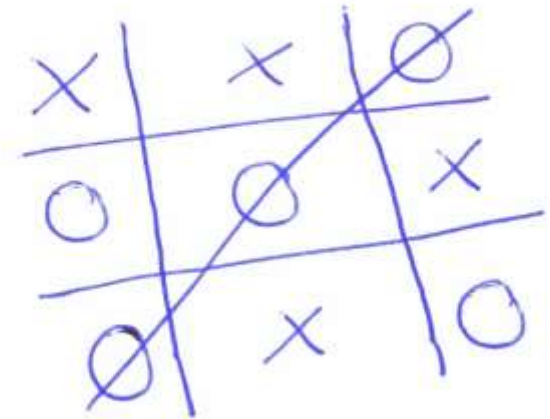
Have you considered the broader context?

There are different types of ML and they are good at different things.

	Supervised Machine Learning	Unsupervised Machine Learning	Reinforcement Learning
Useful for	Making A->B Predictions	Discovering previously unknown patterns in data	Optimization in complex, but constrained tasks
Example Uses	Determine whether an image contains a ship. Determine whether a set of financial documents indicate fraud. From a baseball players prior performance, predict performance in the next game.	Discover customer profiles Identify clusters of malware Identify anomalous network activity	Optimizing logistics chain management Optimizing strategy in a game
Common Methods	**Regression** (Linear, Regression Trees, Kernel Regression, …) **Classification** (Support Vector Machines, Logistic Regression, Discriminant Analysis) Neural Networks, Ensemble Methods…	**Clustering** (K-means, DBSCAN, Mixture modeling) Association Rule Mining Anomaly Detection Neural Networks	Q-Learning Policy Optimization State-Action-Reward-State-Action Deep Deterministic Policy Gradient
Notes	By far the most common	Data widely available, implementation & verification are tricky	Only beginning to move into commercial space, still largely academic.

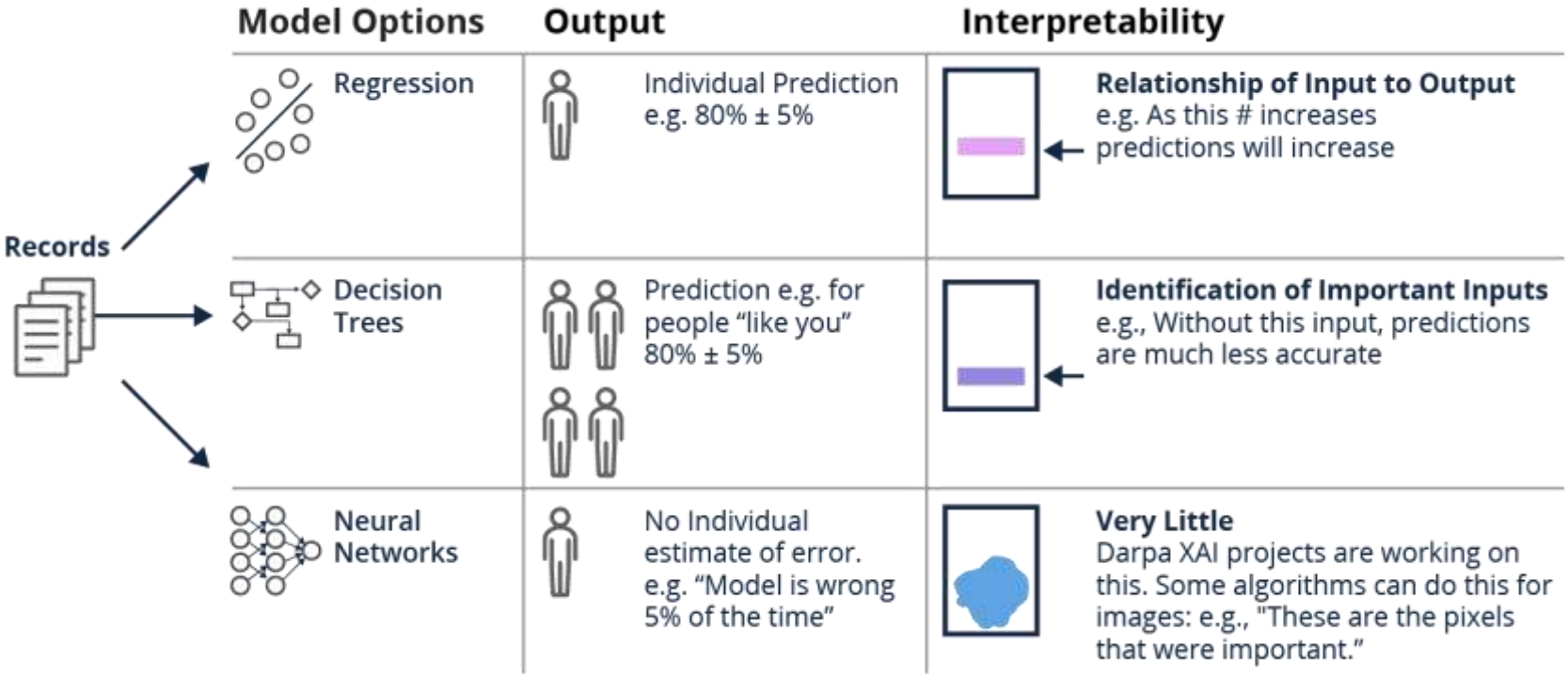
If you use the wrong math, you can get pure nonsense.

An algorithm that always takes the “best” move on the next turn will never lose at ***tic-tac-toe.*** It will either win or draw.



An algorithm that always takes the next “best” move in ***chess***, will always fall into any trap set by it’s opponent. In order to see the trap and avoid it, the algorithm must be able to consider more than one move ahead.

There are usually several options of “right math,” but only some of them will give you the summary you need.



Lots of performance metrics can seem similar, even if they're quite different.

Different kinds of errors may have different kinds of implications.

Metric	What it measures
Error Rate	How often is the algorithm wrong?
False Positive	Algorithm predicted "yes", But the truth is "no"
False Negative	Algorithm predicted "no", But the truth is "yes"
Positive Predictive Value	Of the "yes's" that were predicted, How many are actually "yes"

Consider a system designed to predict whether a defendant eligible for probation will reoffend.

Metric	White Defendants	Black Defendants
Error Rate	39%	39%
Labeled High Risk Did not reoffend.	23.5%	**44.9%**
Labeled Low Risk Did reoffend.	**47.7%**	28.0%

The metrics you choose for an ML project are a policy statement about what kind of systematic errors are acceptable, and which should be minimized.

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

ML provides the greatest benefit when applied at scale

Scale could mean:

- A task that is executed repeatedly.
- There is more data to analyze than a human can handle.

First things first

Before starting an ML project, you should be able to answer basic questions about your data, such as

- “How many?”
- “How often?”
- “What kind?”

Broader Context

Every ML system is different, but there are two issues that will be omnipresent.

- What are the consequences if my data or my model were leaked?
- What are the consequences of an incorrect decision based on my model?

Useful questions to begin an oversight discussion

What policy is this algorithm implementing?

- What are the intended consequences of a policy?
- What unintended consequences can be anticipated?

What checks and balances are in place?

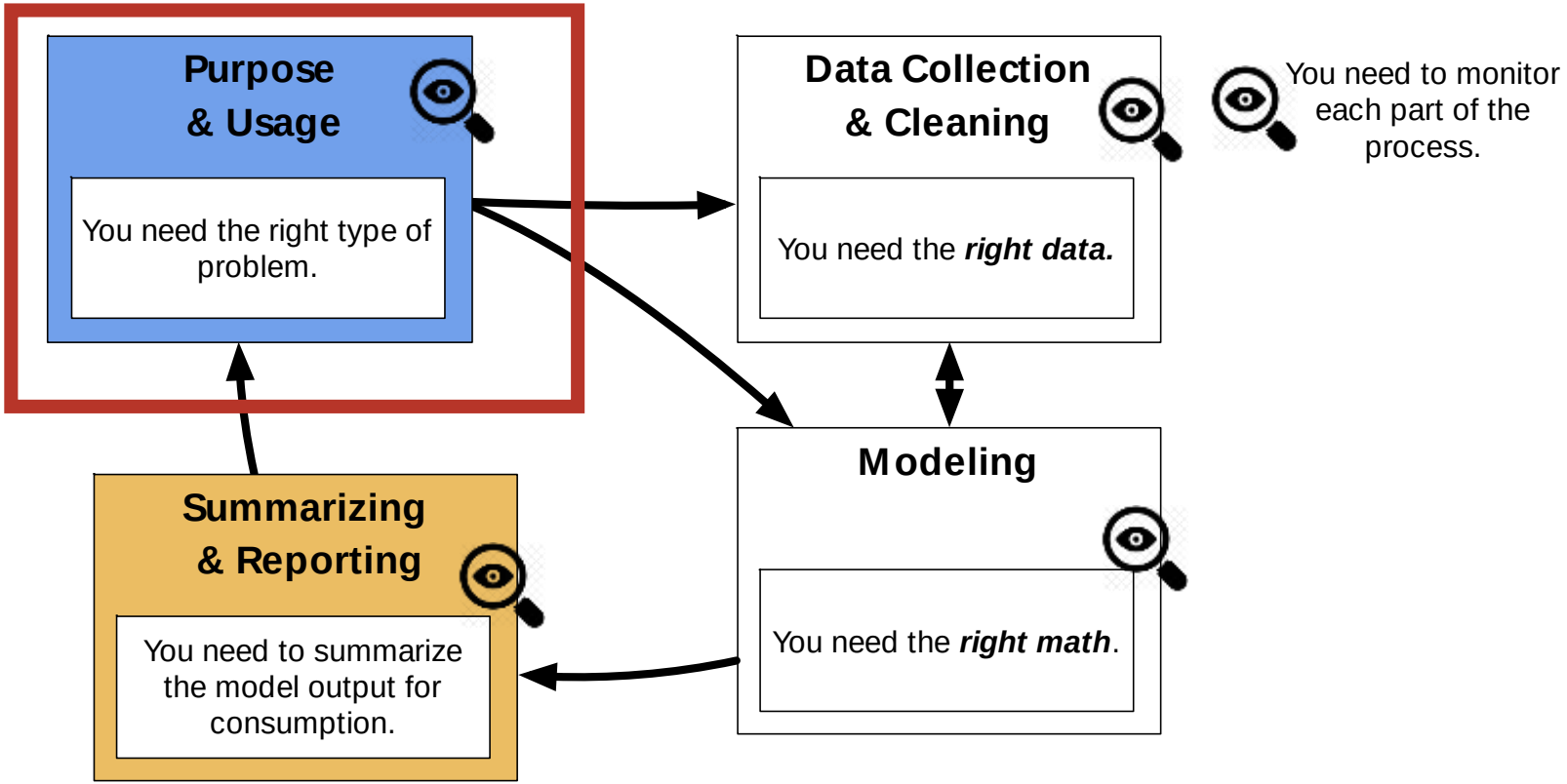
- How will field performance be evaluated?
- What is the procedure for monitoring and validation? Who will be doing the monitoring?
- Are there historic problems (e.g. racial bias) in this area that could be perpetuated?

What procedures are in place for handling inherent uncertainty?

- How is uncertainty communicated to the end user?
- How can the end user check or verify a prediction? (e.g., If you're uncertain about a rain forecast, you might look at a radar map.)
- How does the end user make a decision when told a prediction has low confidence? (e.g., the ML system only has 60% confidence in it's prediction.)

Questions?

Purpose & Usage



Translating a real world problem into something tractable is not always straightforward.

Example: What does it mean to “jump”?

Researchers were trying to have an AI design creatures that were optimized for “jumping”

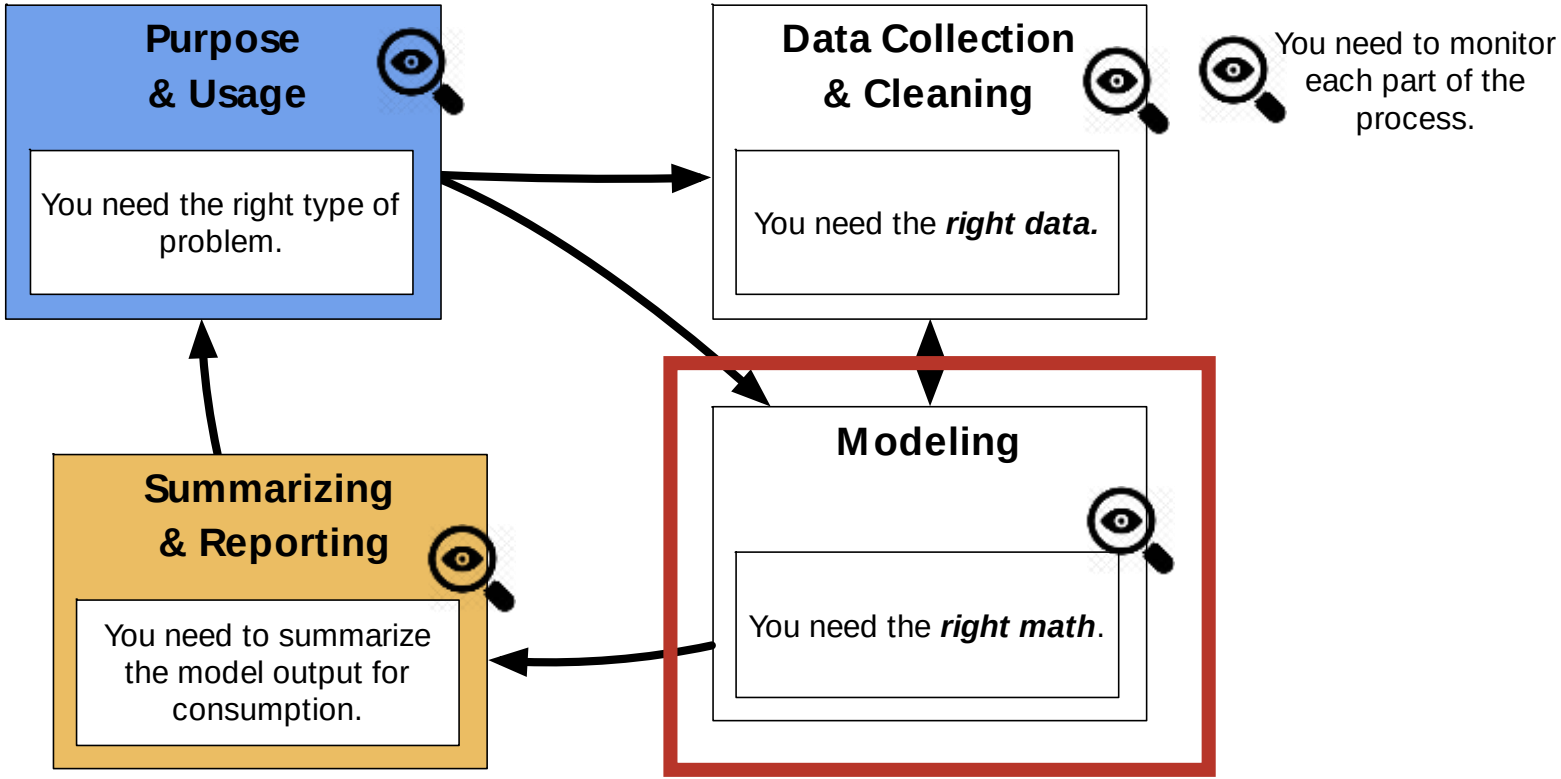
When they defined jumping as “maximum elevation reached by the center of gravity of the creature during the test,” They got tall skinny creatures with enormous heads (top).

When they defined jumping as "furthest distance from the ground of the block that was originally closest to the ground” they got tall skinny creatures that flipped over (bottom).

Recommendation: Ask people about the issues they’ve run into.

arXiv:1803.03453

Modeling

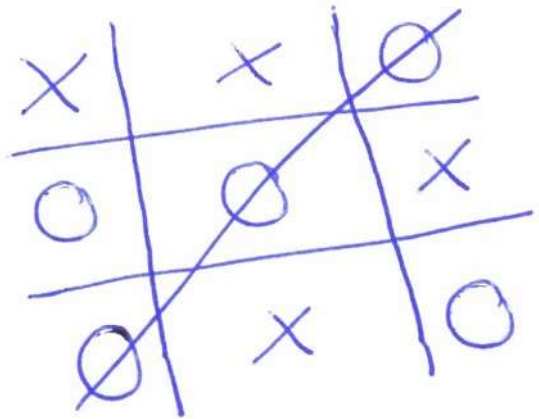


There are different types of ML and they are good at different things.

	Supervised Machine Learning	Unsupervised Machine Learning	Reinforcement Learning
Useful for	Making A->B Predictions	Discovering previously unknown patterns in data	Optimization in complex, but constrained tasks
Example Uses	Determine whether an image contains a ship. Determine whether a set of financial documents indicate fraud. From a baseball players prior performance, predict performance in the next game.	Discover customer profiles Identify clusters of malware Identify anomalous network activity	Optimizing logistics chain management Optimizing strategy in a game
Common Methods	**Regression** (Linear, Regression Trees, Kernel Regression, …) **Classification** (Support Vector Machines, Logistic Regression, Discriminant Analysis) Neural Networks, Ensemble Methods…	**Clustering** (K-means, DBSCAN, Mixture modeling) Association Rule Mining Anomaly Detection Neural Networks	Q-Learning Policy Optimization State-Action-Reward-State-Action Deep Deterministic Policy Gradient
Notes	By far the most common	Data widely available, implementation & verification are tricky	Only beginning to move into commercial space, still largely academic.

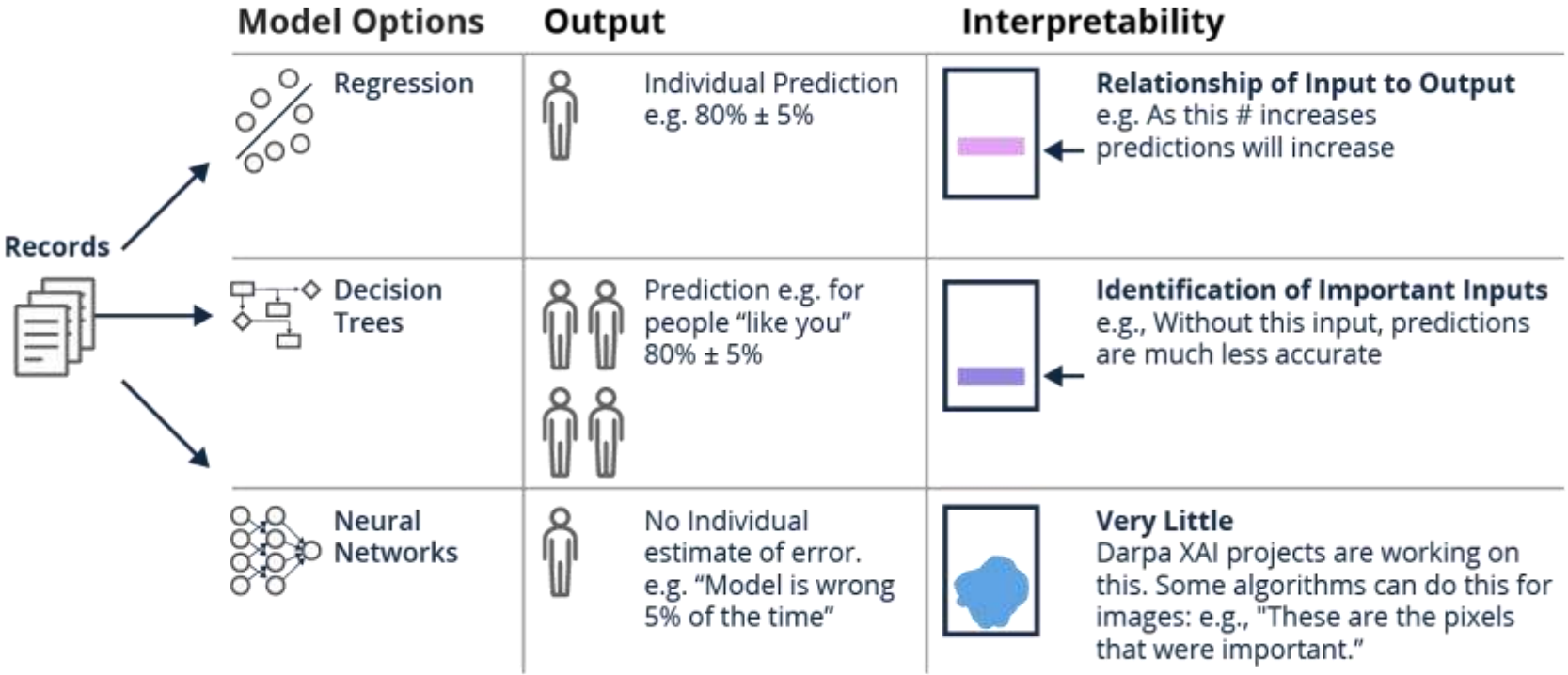
If you use the wrong math, you can get pure nonsense.

An algorithm that always takes the “best” move on the next turn will never lose at ***tic-tac-toe.*** It will either win or draw.



An algorithm that always takes the next “best” move in ***chess***, will always fall into any trap set by it’s opponent. In order to see the trap and avoid it, the algorithm must be able to consider more than one move ahead.

There are usually several options of “right math,” but only some of them will give you the summary you need.



Lots of performance metrics can seem similar, even if they're quite different.

Different kinds of errors may have different kinds of implications.

Metric	What it measures
Error Rate	How often is the algorithm wrong?
False Positive	Algorithm predicted "yes", But the truth is "no"
False Negative	Algorithm predicted "no", But the truth is "yes"
Positive Predictive Value	Of the "yes's" that were predicted, How many are actually "yes"

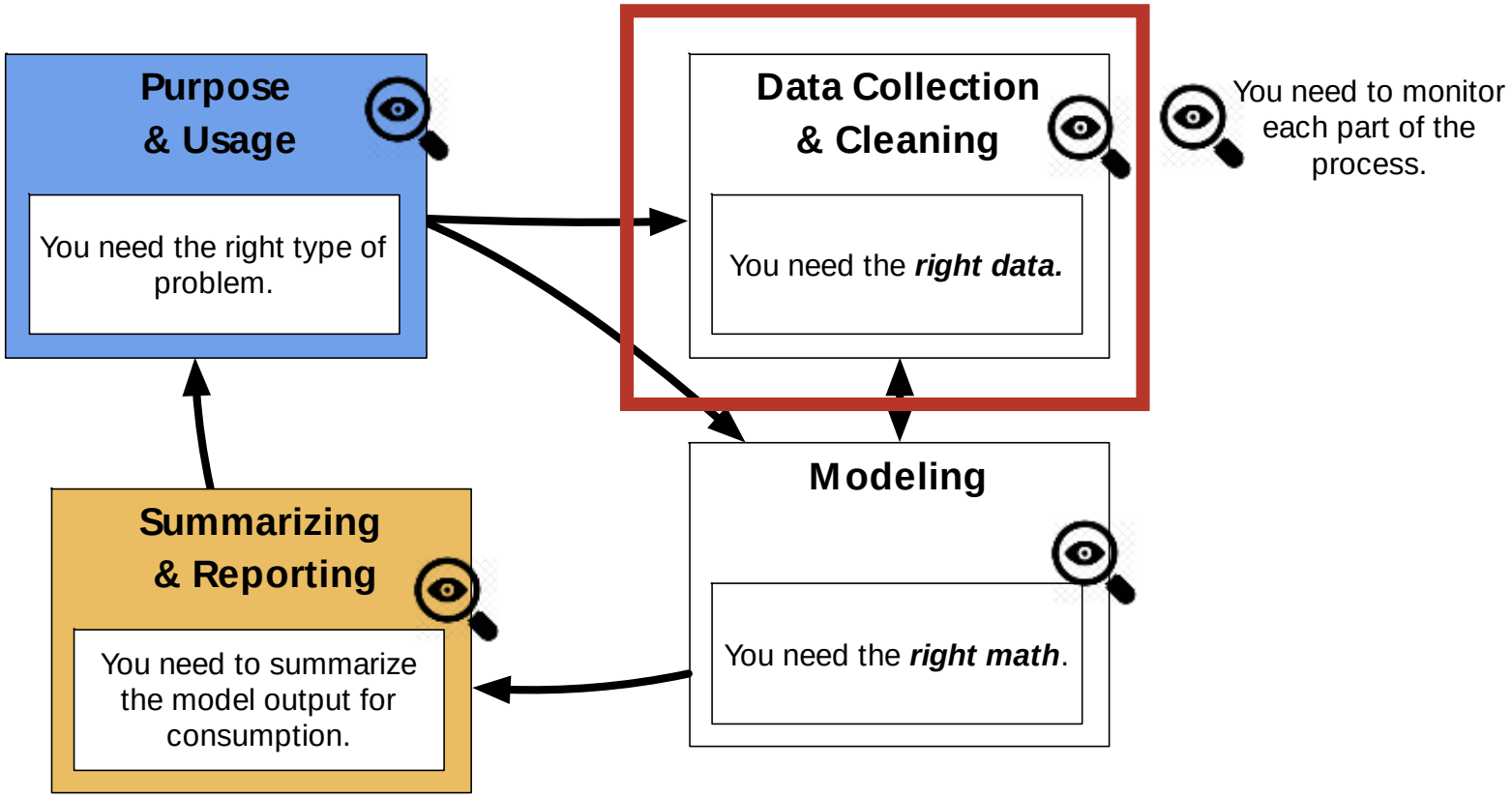
Consider a system designed to predict whether a defendant eligible for probation will reoffend.

Metric	White Defendants	Black Defendants
Error Rate	39%	39%
Labeled High Risk Did not reoffend.	23.5%	**44.9%**
Labeled Low Risk Did reoffend.	**47.7%**	28.0%

The metrics you choose for an ML project are a policy statement about what kind of systematic errors are acceptable, and which should be minimized.

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

Data Collection & Cleaning



Your data has to contain the right information, otherwise, it doesn't matter how much you have.

If you want to be able to input a pile of financial documents and detect fraud, that's a supervised learning problem.

You need to train your model on financial documents that have been labeled

contains fraud

no identified fraud

Without this labeled data you cannot do ***this*** analysis.

(There are other analyses that can still move you towards that goal, but they're more complex and less certain produce useful results.)

Data Collection must reflect Usage Conditions

ML algorithms learn what's in the data.

So, "what's in the data" must match the conditions where the results will be used.

Train (your model) the way you fight.

Google Flu Trends

Designed to predict severity of flu outbreak, during real time.

In 2008, it worked very well, correctly predicting the CDC results weeks earlier.

In 2013, the predictions were off by 140%.

The application conditions changed over time, reducing the accuracy of predictions.

Project was discontinued.

https://www.wired.com/2015/10/can-learn-epic-failure-google-flu-trends/

Dog/Wolf Images

(a) Husky classified as wolf

(b) Explanation

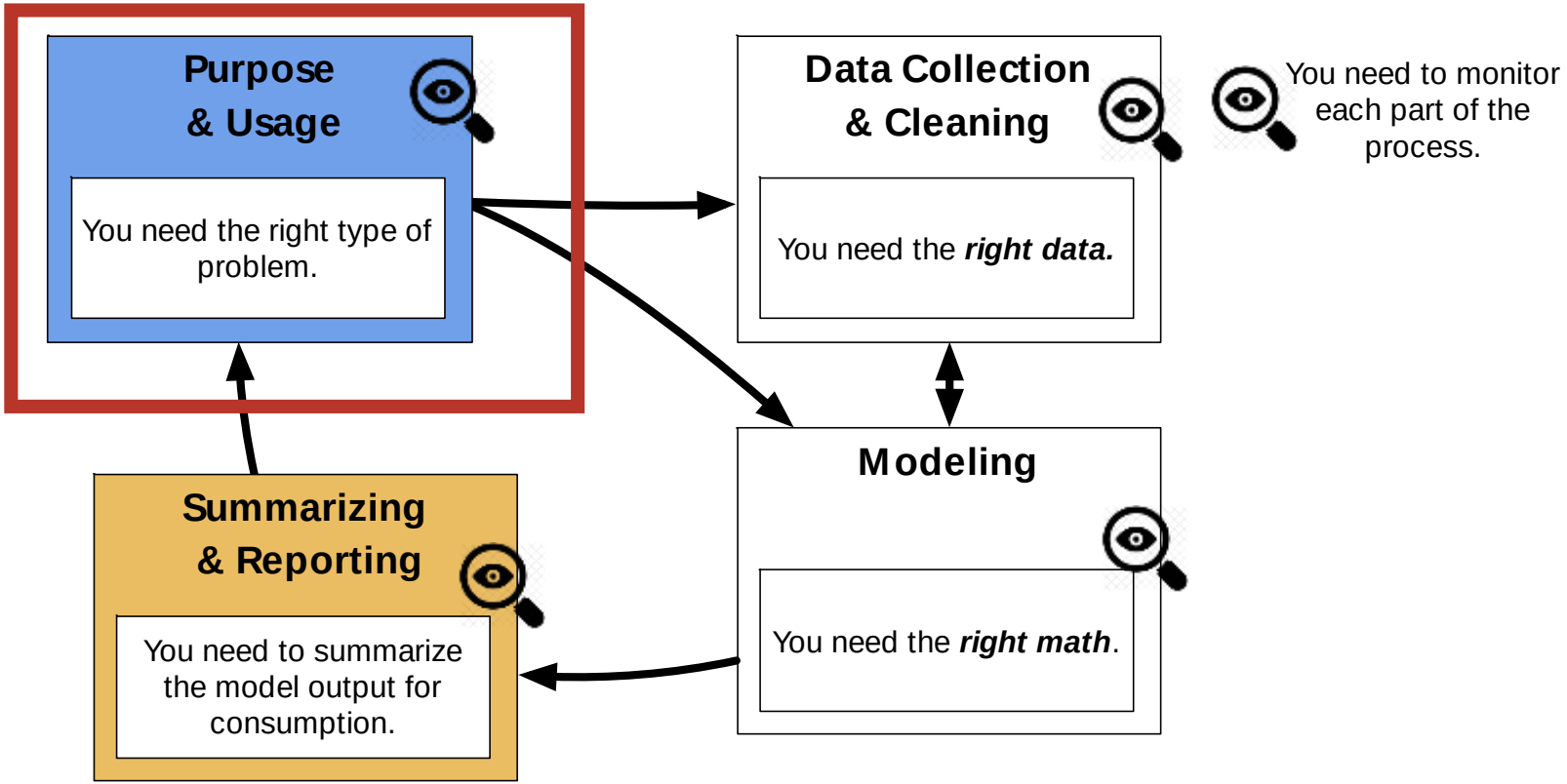
Figure 11: Raw data and explanation of a bad model's prediction in the "Husky vs Wolf" task.

This model learned to distinguish between wolves & dogs by looking at the background, because a snow background indicated "wolf."

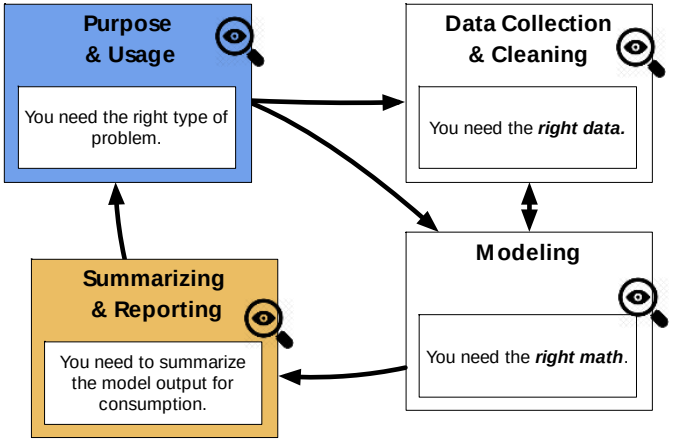
Implications for training data in Project Maven.

https://arxiv.org/pdf/1602.04938.pdf

Purpose & Usage



If your use-case changes…



You definitely need a new model.

Because the math must match the use.

You very likely need new data.

Because the data must match the use.

Possible Mitigations:

- Get end user involved from the beginning.
- Do feasibility studies.
- Collect extra data.

Usability, Interpretability & Explainability

Cautionary Tale:

Flint Michigan developed an ML system to predict which homes had lead pipes that needed to be replaced.

Overall, performance was reasonably good, with about 80% accuracy.

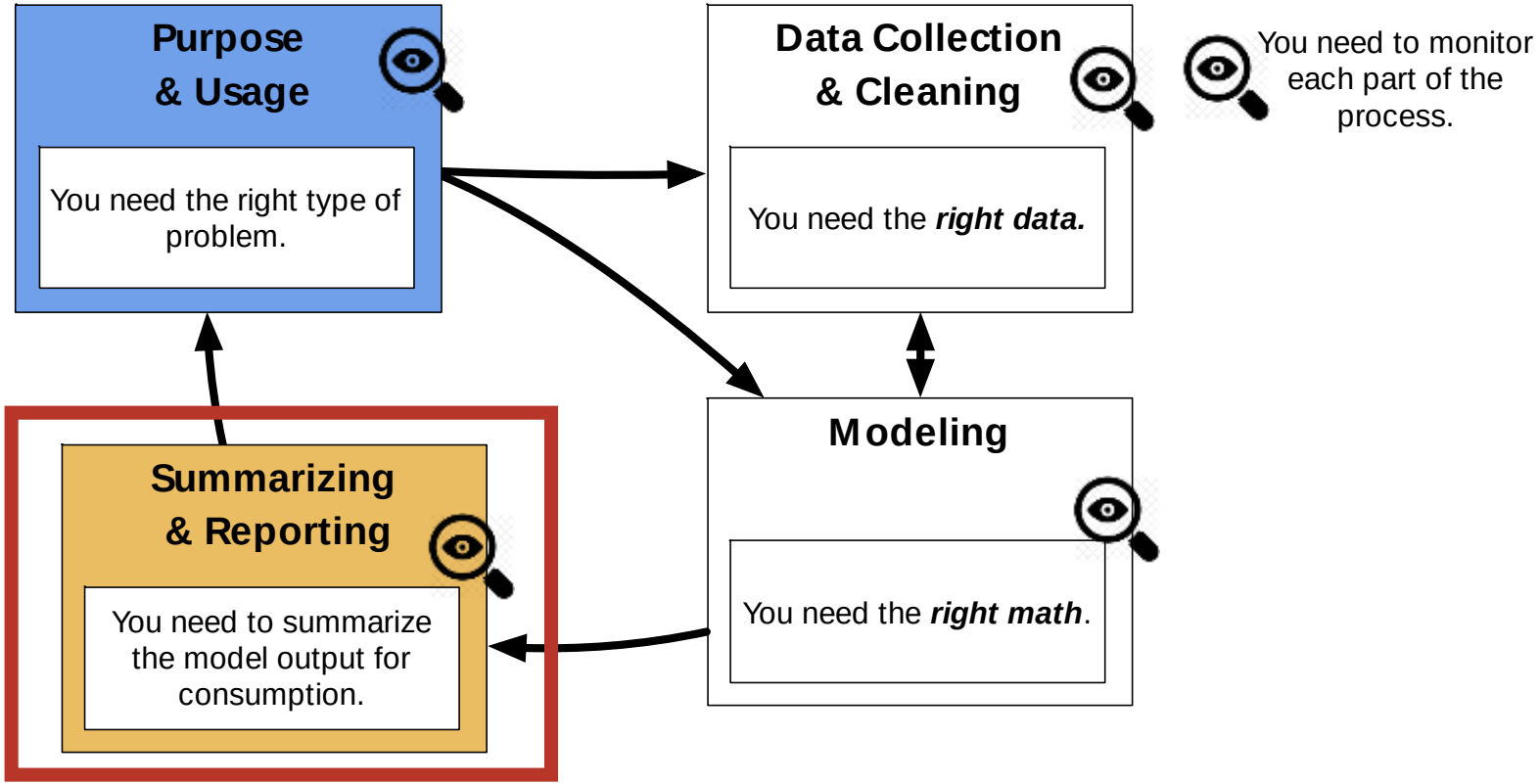
The city scrapped the system. One of the primary reasons was that they could not convey to the public how the algorithm worked, and why some houses were prioritized over others.

“When we started this, people would say, ‘You did my neighbor’s house and you didn’t do mine,’” [Mayor] Weaver said.

Recommendation: Usability, Interpretability & Explainability must be designed into an ML/AI system. It cannot be added as an afterthought.

https://www.theatlantic.com/technology/archive/2019/01/how-machine-learning-found-flints-lead-pipes/578692/

Purpose & Usage



ML system user interfaces must communicate inherent uncertainty.

ML systems are based on probability.

There is inherent uncertainty in any probabilistic system.

Output from an ML system must communicate this uncertainty to the users.

Hypothetical:
Her mom is 5’10” and her dad is 6’1”. Can you predict how tall she will be as an adult?
You can make a good guess, but with some uncertainty.
e.g., *She’ll probably be about 5’10”*

This is the kind of prediction (supervised) ML is doing. It’s doing it faster, with more data, and more precise math. So there’s (usually) less uncertainty than a human prediction.

e.g., *There’s a 80% probability she’ll be between 5’8” and 5’10”.*

ML systems must be designed to work within a larger system.

Risks are different in different contexts.

- Low risk: An ML system makes an error in predicting which new movie you will like.
- Medium risk: An ML system makes an error in predicting whether an individual is an insider threat.

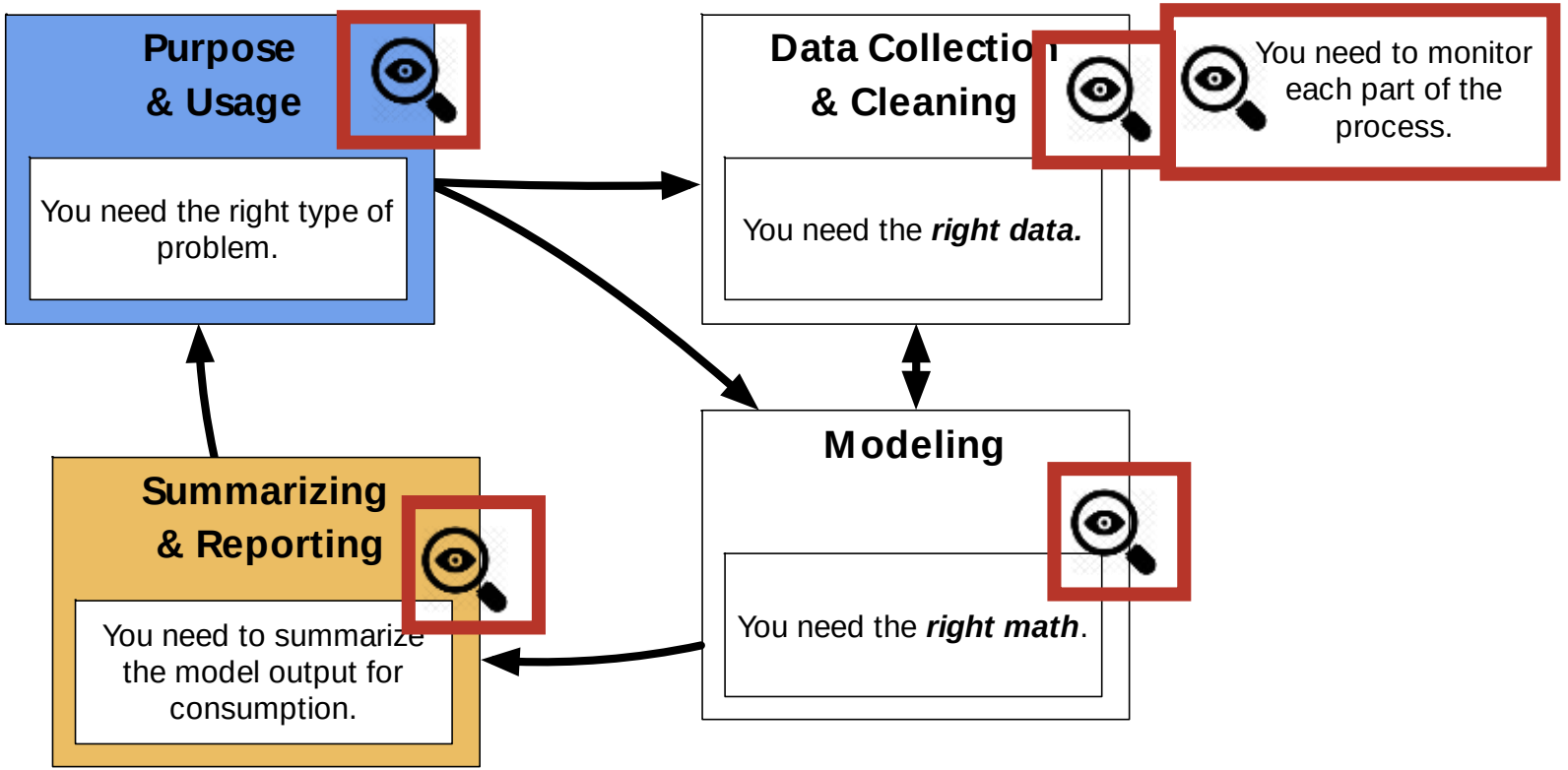
Risk evaluations should be considered in system design

- Performance metrics must be chosen to measure context dependent risks.
- If the risks are acceptable, an autonomous system might be appropriate.
- If the risks of an autonomous system are too high, then there need to be appropriate procedures in place to verify a decision and escalate as appropriate. (checks & balances)

Tactics, techniques and procedures must be carefully designed to mitigate risk.

- A human-in-the-loop is often suggested as one way to mitigate risk, but this may not be sufficient if there are incentives to “just go with what the algorithm says”

Monitoring



AI/ML implement policy

Examples of where algorithms are implementing policy:

- Determining whether applicants are eligible for Medicare.
- Identifying and locating men delinquent in paying child support.
- Identifying and removing citizens registered to vote in multiple locations.
- Predicting whether an individual is a threat and should be detained.

In each of these examples (and many more), we must be able to verify that the algorithm is implementing policy as intended, and that policy is not being set by software developers.

Danielle Keats Citron, Technological Due Process, 85 Wash. U. L. Rev. 1249 (2008).

Any deployed AI/ML system needs checks & balances

In the same way that we require inspections for food safety, automobile safety and aviation safety, we must be able to determine whether a deployed AI/ML system is functioning as designed.

We must be able to validate and adjust:

- Training data
- Model choice
- Model implementation
- The data pipeline for the deployed model
- Performance in the field

Anybody can download Tensorflow and throw a pile of data at it. That does not mean the output is useful or correct.

Validation metrics don't really exist right now. This is one place to invest in research.

Good monitoring and validation practices help ensure good performance, making the system more secure against both adversaries and poor implementation.

Adversarial Issues are

1. An active area of research – we don't even fully understand what kinds of attacks might be possible right now.
2. Every neural network ML system is susceptible to attack in infinite variations on a theme.
3. Adversarial attacks are becoming more widespread, we are starting to see them "in the wild."
4. Current best practices for mitigating adversarial risks are ongoing monitoring and validation of the ML system, and having well designed procedures around it.

Many Adversarial AI applications leverage "usage doesn't match training" in different ways.

This 3D printed turtle was built to be classified as a rifle from any angle.

It was created to fit into a gap in what the algorithm had learned a rifle looked like. The turtle is not from the same population of images that the algorithm was trained on.

https://www.theverge.com/2017/11/2/16597276/google-ai-image-attacks-adversarial-turtle-rifle-3d-printed

Useful questions to begin an oversight discussion

What policy is this algorithm implementing?

- What are the intended consequences of a policy?
- What unintended consequences can be anticipated?

What checks and balances are in place?

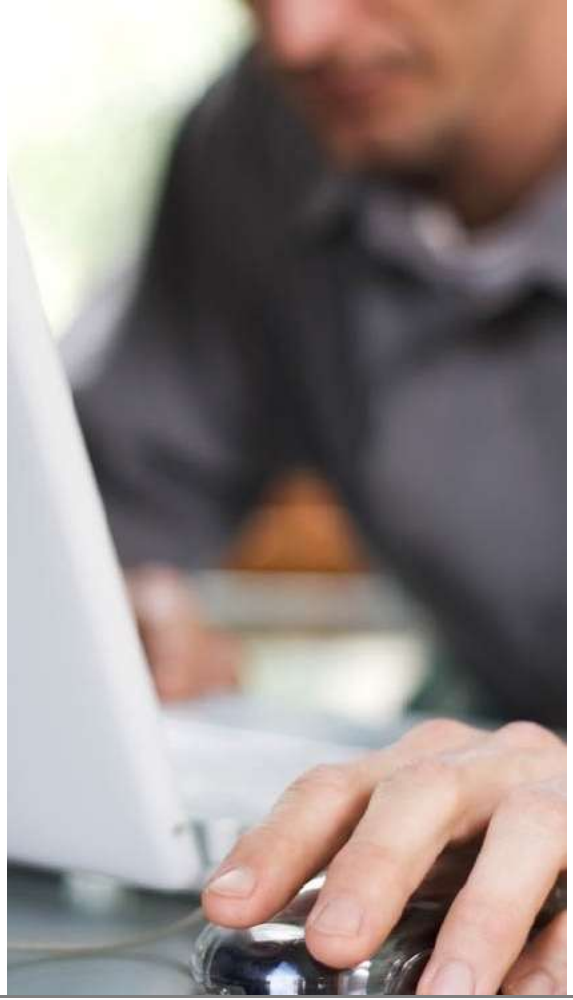
- How will field performance be evaluated?
- What is the procedure for monitoring and validation? Who will be doing the monitoring?
- Are there historic problems (e.g. racial bias) in this area that could be perpetuated?

What procedures are in place for handling inherent uncertainty?

- How is uncertainty communicated to the end user?
- How can the end user check or verify a prediction? (e.g., If you're uncertain about a rain forecast, you might look at a radar map.)
- How does the end user make a decision when told a prediction has low confidence? (e.g., the ML system only has 60% confidence in it's prediction.)

SEI PowerPoint Template

Extra Slides



AI in China

Artificial Intelligence Development Plan: *"AI has become a new focus of international competition. AI is a strategic technology that will lead in the future; the world's major developed countries are taking the development of AI as a major strategy to enhance national competitiveness and protect national security."*

No privacy or warrant constraints on collecting data.
Social Credit System - will assign each citizen with a social credit score that will determine a person's ability to travel overseas, get a home loan, or even access the internet.
Monitoring piloted in schools
Foreign & industrial espionage – cases recently brought by FBI.

https://thedisconnect.co/three/camera-above-the-classroom

Technical Details - thoughts

Tom: We might include basic concepts on gradient decent at a high level to talk a bit about how ML works. I might have some material for this if you don't have any handy.

April: I was also thinking about how we turn pictures into "data" and feature extraction problems. Gradient decent ties in with the optimization theme, feature extraction ties in with the "what's in the data" theme. So might be useful to have back up slides on both technical areas.

Processing Data

Choices in the art of processing textual (and other unstructured) data.

Is the signal strong enough?

It’s currently pretty easy to detect and interpret street signs from images.

It’s still difficult (impossible?) to interpret gestures from images.

Overfitting & Underfitting

A useful analogy

A narrow AI is about as smart as an ant.

It's very good at doing a single thing ***in a specific context***. And can often do that one thing, better/faster/cheaper than humans.

But! The same way ants get easily confused when they lose their trail, an AI can exhibit unanticipated and undesirable behavior outside of their intended context.