Practicing Trustworthy Machine Learning: A Tutorial

Subho Majumdar, Bias Buccaneers / AVID / TrustML
Matthew McAteer, Formic Labs
Yada Pruksachatkun, Infinitus Systems, Inc.

All material available at
https://github.com/shubhobm/ptml_tutorial

Summary

Trustworthy ML/ Responsible AI/ AI Ethics is on of the hottest buzzwords in ML/AI.

Summary

Trustworthy ML/ Responsible AI/ AI Ethics is on of the hottest buzzwords in ML/AI.

Easier said than done: operationalizing is hard.

Summary

Trustworthy ML/ Responsible AI/ AI Ethics is on of the hottest buzzwords in ML/AI.

Easier said than done: operationalizing is hard.

This tutorial is a summary of our collective work to address this need.

Summary

Trustworthy ML/ Responsible AI/ AI Ethics is on of the hottest buzzwords in ML/AI.

Easier said than done: operationalizing is hard.

This tutorial is a summary of our collective work to address this need.

- Fairness
- Explainability
- Privacy
- Robustness
- Systemic considerations

About Us

Subho Majumdar
Bias Buccaneers
AI Vulnerability Database
Trustworthy ML Initiative
Ex: Splunk, AT&T Labs,
IBM Research

Matthew McAteer
Formic Labs
Ex: 5cube Labs, Google,
MIT, Harvard

Matthew McAteer
Infinitus Systems, Inc
Ex: Amazon, Microsoft, MIT

Part I: Algorithmic Fairness

Fairness in Machine Learning: cautionary examples

Opportunities for bias exist across applications i
machine learning.

Sensitive features include

- Age, marital status, gender, race
- Religion, national origin, citizenship status, political opinion
- medical condition or disability, sexual orientation, military status, employment status, …

Types of Fairness

Group fairness

Idea**:** People from sensitive demographic groups or historically disadvantaged subpopulations and should not be disparately impacted.

Example**:** examples in previous slide.

Individual fairness

Idea**:** Individuals with similar attributes should be treated similarly by the ML algorithm, irrespective of their demographic background.

Example: Credit scoring

Metrics

Denote by $Y, X, S, \hat{Y}$ the random variables denoting respectively the binary output feature, input feature(s), binary sensitive feature and predicted output from a ML model.

Equalized Odds

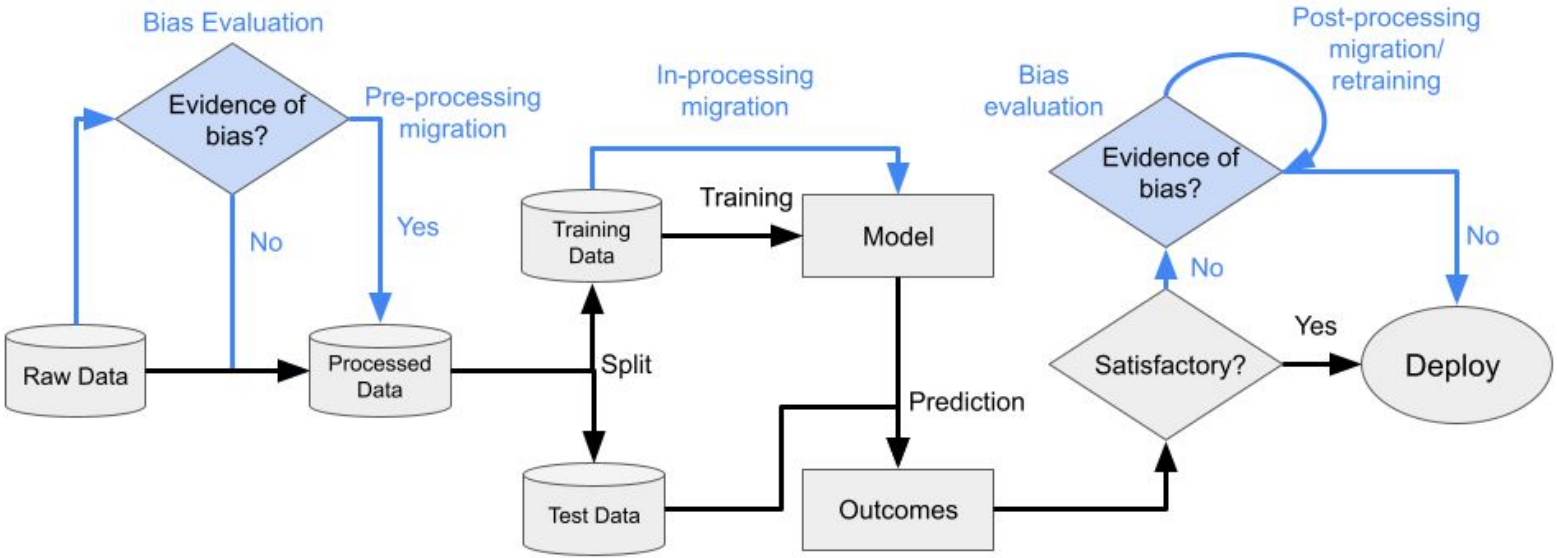
$$P(\hat{Y} = 1 \mid S = 0, Y = y) = P(\hat{Y} = 1 \mid S = 1, Y = y); \qquad y = 0, 1$$

Counterfactual Fairness

$$P(\hat{Y} = y \mid S = 0, X = x) = P(\hat{Y} = y \mid S = 1, X = x); \qquad y = 0, 1$$

Holds for all possible values x of X.

Detecting and Mitigating Bias



Deep Dive: Evaluating Language Models for Toxicity

- Evaluate toxicity of sentence completions.
- Supply prompts to complete sentences using a language model.
- Prompts are tagged with subgroups, e.g. religion, race.
- Calculate toxicity of completed sentences, and calculate mean and variance of predicted toxicity by subgroup.
- Notebook link

Tools

- IBM AI Fairness 360: good place to start
- Fairlearn: large community
- LinkedIn Fairness Toolkit (LiFT): scalable, scala/spark-based

Part II: Explainability and Interpretability

Why Explain?

Today’s ML models are complex and difficult to glean into. Explanations help elicit stakeholder trust into model decisions.

Why Explain?

Today's ML models are complex and difficult to glean into. Explanations help elicit stakeholder trust into model decisions.

Justification

Defend algorithmic decision-making comply with rules and regulations, e.g. GDPR Right to Explanation, or credit reporting reason codes

Why Explain?

Today's ML models are complex and difficult to glean into. Explanations help elicit stakeholder trust into model decisions.

Justification

Defend algorithmic decision-making comply with rules and regulations, e.g. GDPR Right to Explanation, or credit reporting reason codes

Discovery

Discover the limitations or errors in our decision-making and enrich human knowledge

Why Explain?

Today's ML models are complex and difficult to glean into. Explanations help elicit stakeholder trust into model decisions.

Justification

Defend algorithmic decision-making comply with rules and regulations, e.g. GDPR Right to Explanation, or credit reporting reason codes

Discovery

Discover the limitations or errors in our decision-making and enrich human knowledge

Control and improvement

Detect and troubleshoot model performance issues

Why Explain?

Today's ML models are complex and difficult to glean into. Explanations help elicit stakeholder trust into model decisions.

Justification

Defend algorithmic decision-making comply with rules and regulations, e.g. GDPR Right to Explanation, or credit reporting reason codes

Discovery

Discover the limitations or errors in our decision-making and enrich human knowledge

Control and improvement

Detect and troubleshoot model performance issues

Causation

A/B tests based on explanations can form and validate hypotheses on cause-effect relationships

Types of explanations: In-model vs. post-model

Inherently explainable models

Models that are explainable by design
e.g. linear/logistic regression, GAM, decision trees

Pros: quick to train, effect of features on outcome is directly known

Cons: underfit, less accurate on complex data

Post-hoc Explainability

Methods that explain outcomes of another model
e.g. LIME, SHAP

Pros: model-agnostic, account for local effects

Cons: fidelity issues, may be computationally expensive

Types of explanations: global vs. local

Global explainability

Aim to produce an overall comprehensible overview of a ML model

May take the form of

- Feature summaries (VarImp)
- Model internals (linear models, LASSO, decision trees)

Local explainability

Aim to explain one single sample or small groups of samples

Train simple, interpretable supervised models on tightly clustered synthetic data around the data-point to be explained, taking model predictions as labels, e.g. LIME

Deep Dive: Explaining Transformer Models

- Explain a sentiment analysis model using LIME.
- Notebook link

Tools

- AIX 360, InterpretML: good places to start, active community
- Python/R/Julia packages, e.g. lime in python

Part III: Privacy

What does it mean to Ensure Privacy?

Anonymization

Idea**:** Anonymize data of an individual, while still providing meaningful answers to aggregate queries to the data

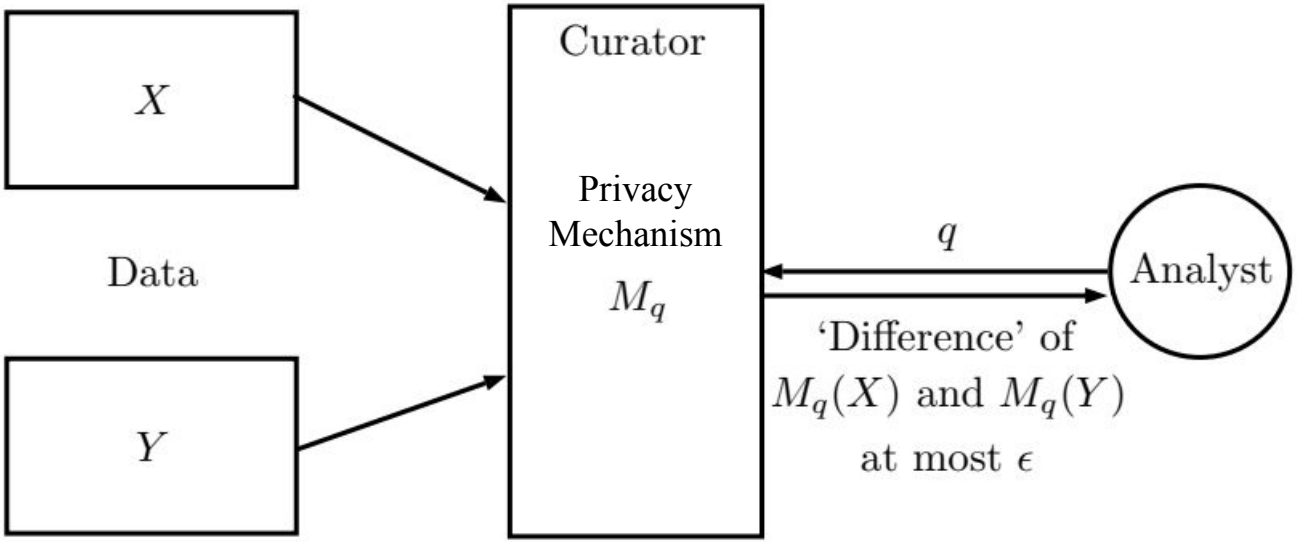
Example**:** differential privacy

Encryption

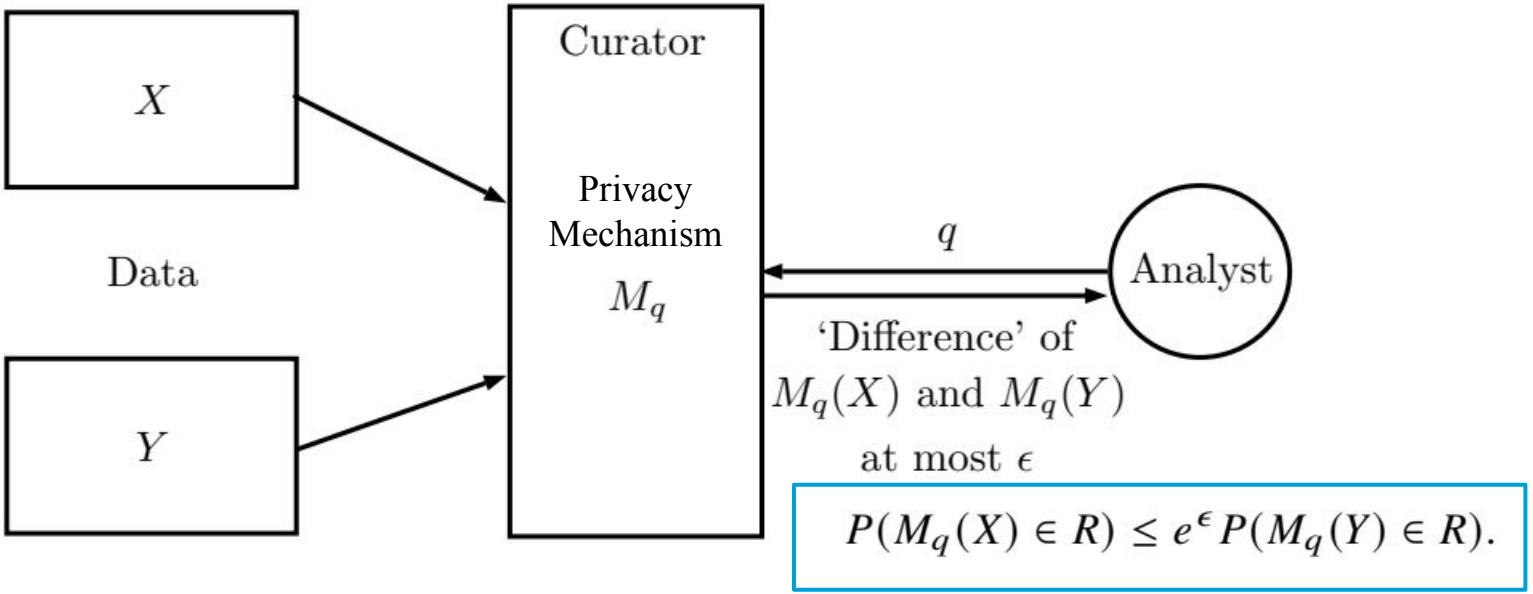
Idea**:** Encrypt the data itself to provide stronger protections against unwanted queries/access

Example: Homomorphic encryption

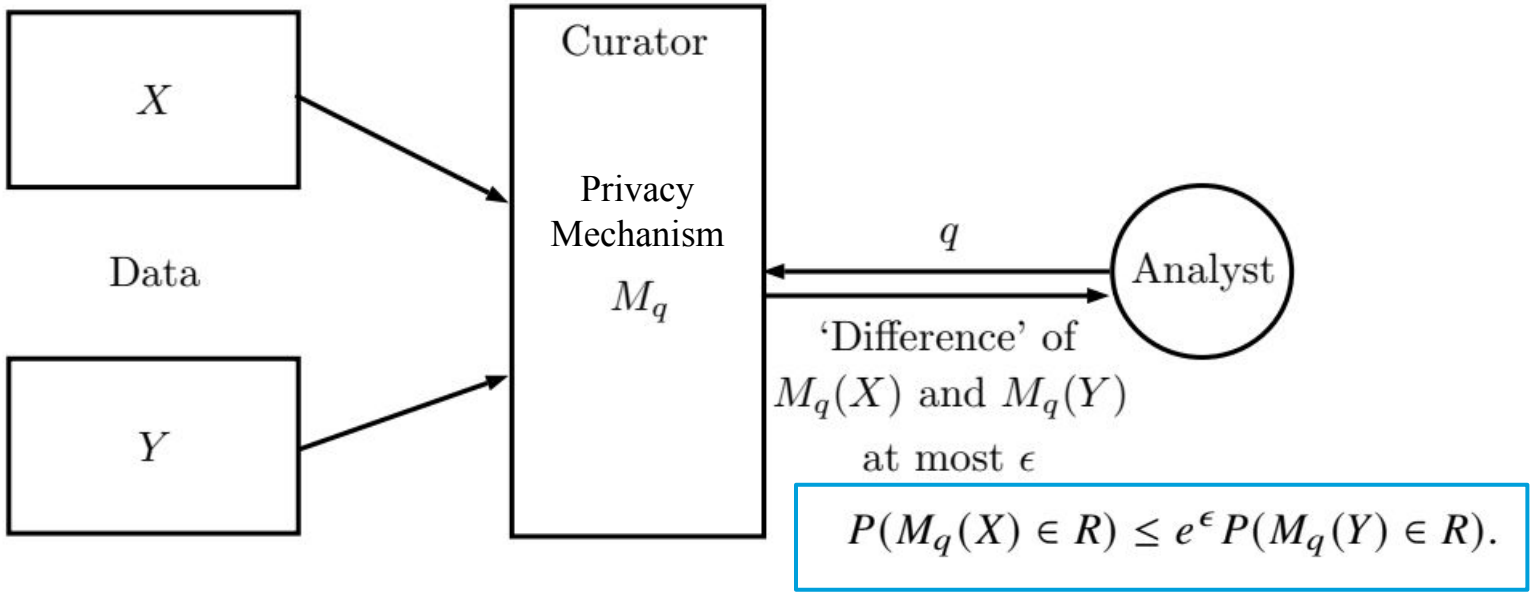
Differential Privacy (DP)



Differential Privacy (DP)



Differential Privacy (DP)



Global DP: above holds for all pairs of datasets X, Y that differ by only one record
Local DP at X: above holds for all datasets Y that differ from X by only one record

Homomorphic Encryption (HME)

While there are plenty of ways to lock up and secure your data to restrict access, the data itself usually needs to be exposed in some way to the model in order for it to learn.

Now, what if you could keep the data locked up or encrypted, while still letting the ML model learn the patterns it needs to.

This is the promise behind **homomorphically encrypted machine learning**.

Deep Dive: HME

- Basic definitions
- Code examples
- Links to resources
- Notebook link

Tools

- PyDP: good place to start, python implementation
- PipelineDP: scalable, spark-based
- OpenMined: THE open-source community for privacy in ML

Part IV: Robustness

Why do ML Models need to be Robust?

A model's ability to be resilient to variation in data is called **robustness**.

No matter how good your training data is, the model is going to encounter unexpected things in the real world, and robustness is about making sure it's ready for them.

Why do ML Models need to be Robust?

A model's ability to be resilient to variation in data is called **robustness**.

No matter how good your training data is, the model is going to encounter unexpected things in the real world, and robustness is about making sure it's ready for them.

There are two types of robustness:

Train-time: model's ability to generalize in spite of training data contamination

Test-time: model's ability to generalize beyond examples seen during training

Adversarial Robustness

Learned transformations that use ML models to modify and create inputs that fool the base ML model being attacked.

Motivation for novel methods to train base models robust to such attacks.

Adversarial Robustness

Learned transformations that use ML models to modify and create inputs that fool the base ML model being attacked.

Motivation for novel methods to train base models robust to such attacks.

Targeted Attacks

Designed to fool the base model into predicting a specific incorrect class

Untargeted Attacks

Designed to fool the base model into predicting any incorrect class

Adversarial Robustness

Learned transformations that use ML models to modify and create inputs that fool the base ML model being attacked.

Motivation for novel methods to train base models robust to such attacks.

Targeted Attacks

Designed to fool the base model into predicting a specific incorrect class

High potential of harm

Untargeted Attacks

Designed to fool the base model into predicting any incorrect class

Easier to craft

Deep Dive: HopSkipJump attack on ImageNet

- Adapted from IBM Adversarial Robustness Toolkit
- Starts from a base image and iteratively tries to add smaller perturbation to flip its label
- Difficult to predict ⇔ easy to attack
- Notebook link

Tools

- IBM Adversarial Robustness Toolbox
  - Great place to start
  - Active community
- AdvBox: python/command line-based

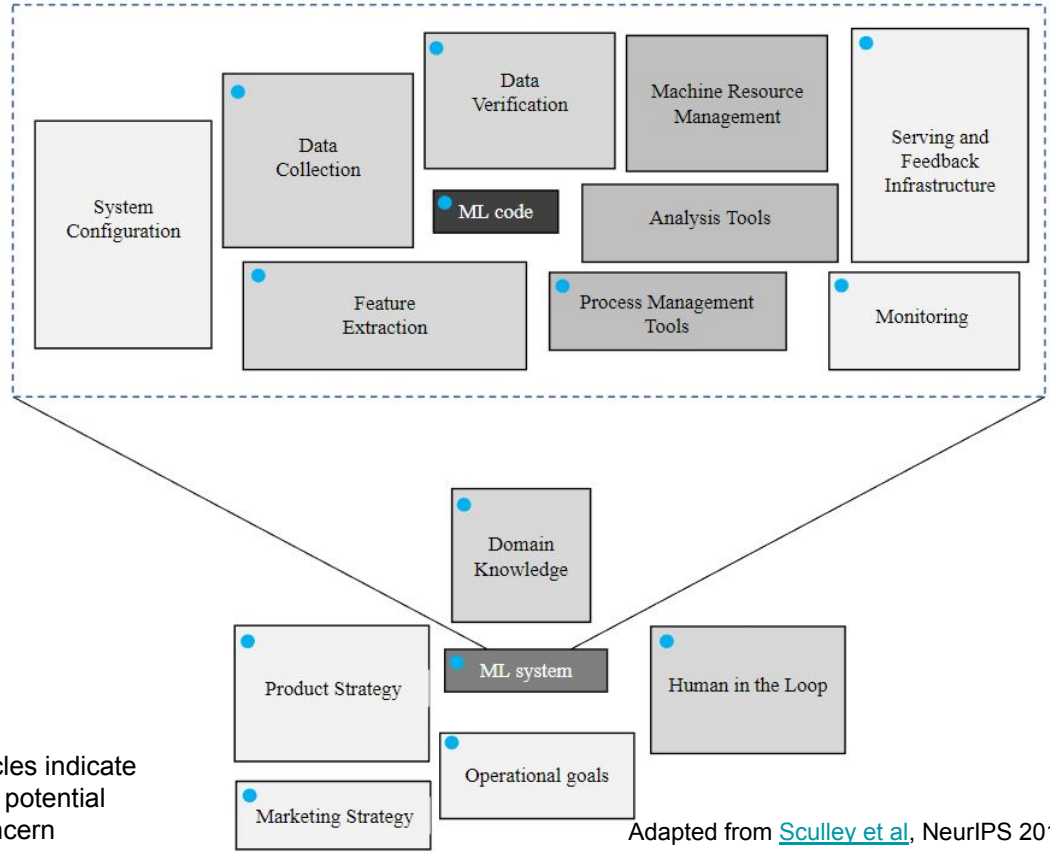
Part V: Systemic Considerations

Isn’t knowing about the tools enough?

No.

Only a small part of deployed ML systems is code.

Trust concerns can creep in from many external sources.



Blue circles indicate areas of potential trust concern

Adapted from Sculley et al, NeurIPS 2015.

Datasheets

Persist salient information about datasets for future use.

- Motivation
- Composition
- Collection
- Preprocessing
- Uses
- Distribution
- Maintenance

Model Cards

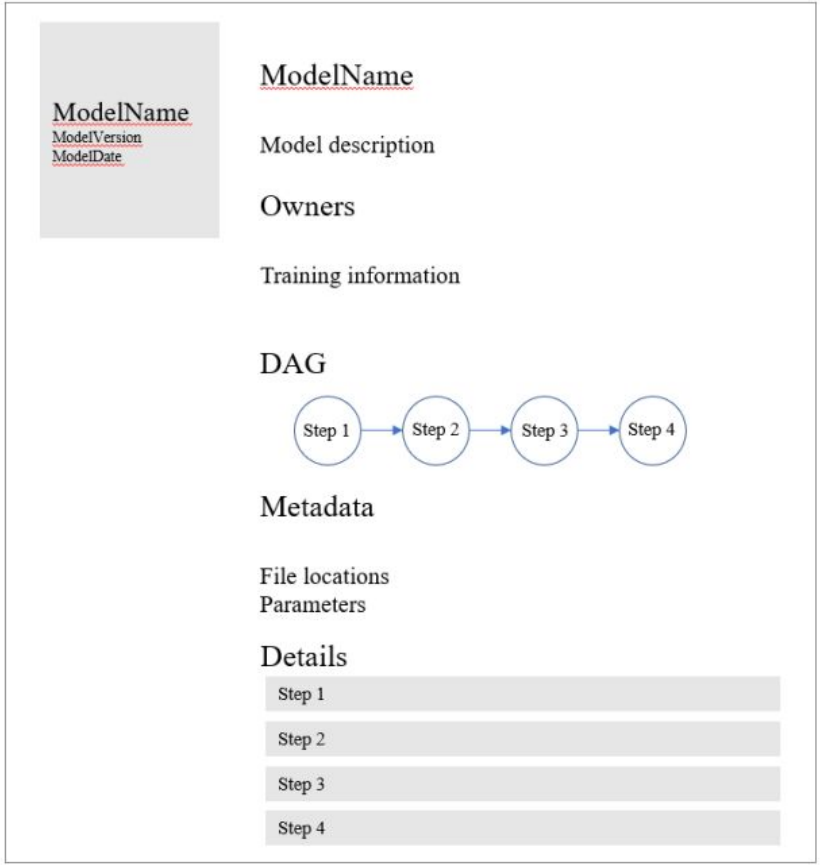
Persist salient information about models for future use.

- Model details
- Intended use
- Factors
- Metrics
- Evaluation data
- Training data
- Quantitative analyses
- Ethical considerations
- Caveats and recommendations

DAG Cards

Persist salient information about ML pipelines for future use.

- More than only datasheet or model cards.
- Higher-level abstraction of a model pipeline.
- Can be automated.



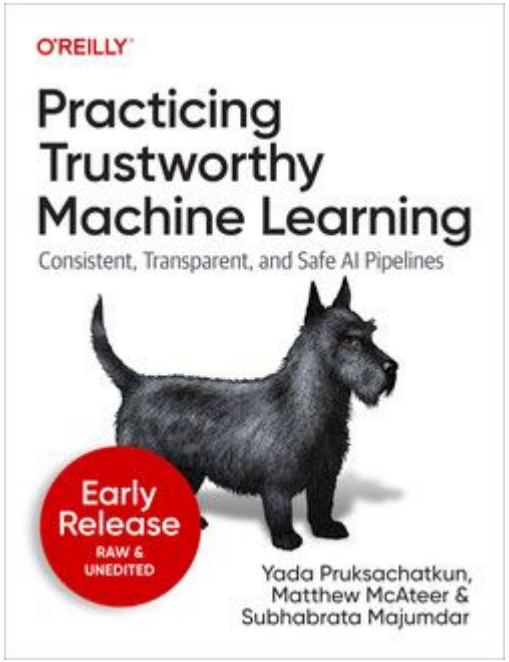
A few More Things to Keep in Mind

- Subject matter and stakeholder guidance
- Causality
- Sparsity and model compression
- Uncertainty quantification

THANK YOU!

All material available at
https://github.com/shubhobm/ptml_tutorial

For more details, concepts,
and deep dives, check out:



Worldwide Release Dec 2022

Available on early release
and for pre-order