Copyright Notice

DeepLearning.AI

Model Monitoring

Why Monitoring Matters

ML Lifecycle Revisited

Why Monitoring Matters

“An ounce of prevention is worth a pound of cure”

- *Benjamin Franklin*

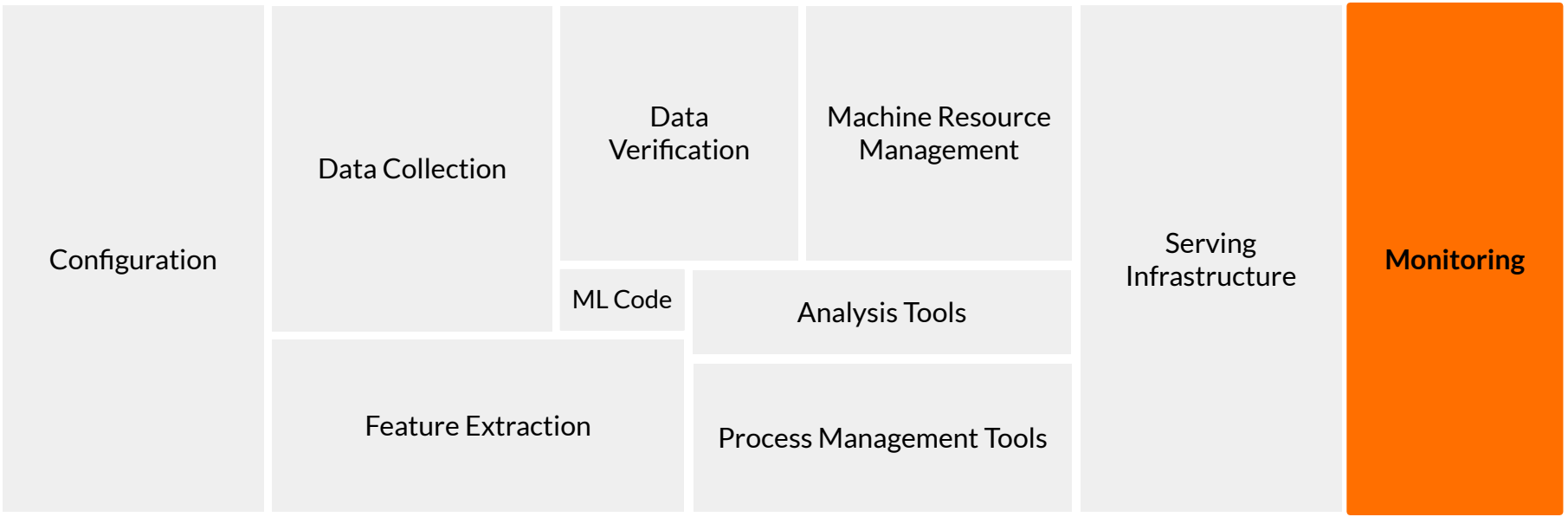
Why do you need monitoring?

- Immediate Data Skews
  - Training data is too old, not representative of live data
- Model Staleness
  - Environmental shifts
  - Consumer behaviour
  - Adversarial scenarios
- Negative Feedback Loops

Monitoring in ML Systems

ML Monitoring (functional monitoring)	System monitoring (non-functional monitoring)
Predictive performance	System performance
Changes in serving data	System status
Metrics used during training	System reliability
Characteristics of features	

Why is ML monitoring different?



DeepLearning.AI

Model Monitoring

Observability in ML

What is observability?

- *Observability* measures how well the internal states of a system can be inferred by knowing the inputs and outputs
- Observability comes from control system theory
- Observability and controllability are closely linked

Complexity of observing modern systems

- Modern systems can make observability difficult
  - Cloud-based systems
  - Containerized infrastructure
  - Distributed systems
  - Microservices

Deep observability for ML

- Not only top-level metrics
- Domain knowledge is important for observability
- TensorFlow Model Analysis (TFMA)
- Both supervised and unsupervised analysis

Goals of ML observability

- Alertable
  - Metrics and thresholds designed to make failures obvious
- Actionable
  - Root cause clearly identified

DeepLearning.AI

Model Monitoring

Monitoring Targets in ML

Basics: Input and output monitoring

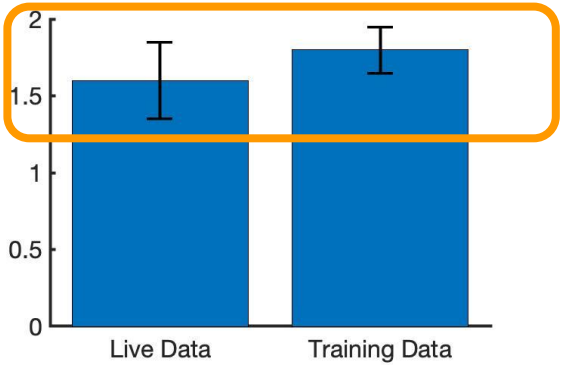
- Model input distribution
- Model prediction distribution
- Model versions
- Input/prediction correlation

Input Monitoring

Do these *check out*?	• Errors: Input values fall within an allowed set/range? • Changes: Distributions align with what you've seen in the past? • Per slice, e.g., marital status (single/married/widowed/divorced)

Prediction Monitoring

Statistical significance



- Unsupervised: Compare model prediction distributions with statistical tests
  - e.g., median, mean, standard deviation, min/max values
- Supervised: When labels are available

Operational Monitoring

ML engineering	Software engineering
Latency	Receiving an HTTP request
IO / Memory / Disk Utilisation	Entering/leaving a function
System Reliability (Uptime)	A user logging in
Auditability	Reading from net / writing to disk

DeepLearning.AI

Model Monitoring

Logging for ML Monitoring

Steps for building observability

- Start with the out-of-the-box logs, metrics and dashboards
- Add agents to collect additional logs and metrics
- Add logs-based metrics and alerting to create your own metrics and alerts
- Use aggregated sinks and workspaces to centralize your logs and monitoring

Logging

Log: An event log (usually just called “logs”) is an immutable, time-stamped record of discrete events that happened over time.

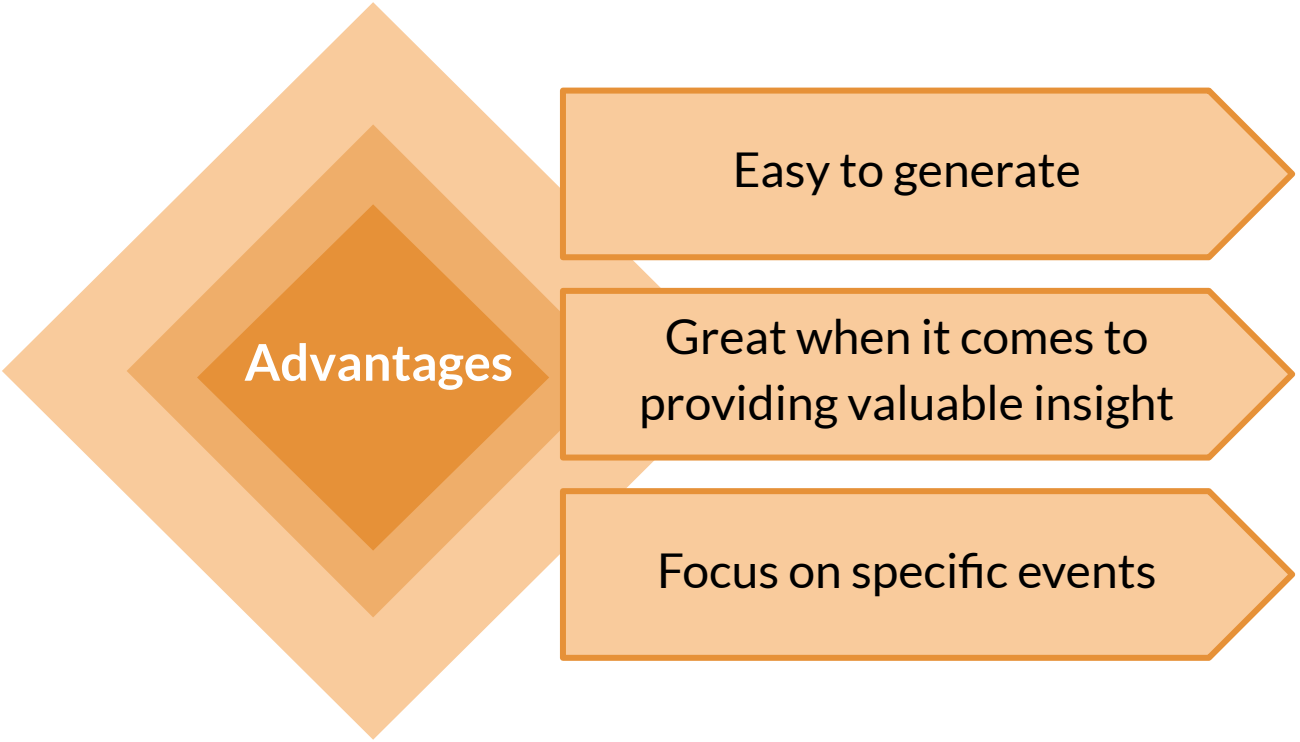
Tools for building observability

- Google Cloud Monitoring
- Amazon CloudWatch
- Azure Monitor

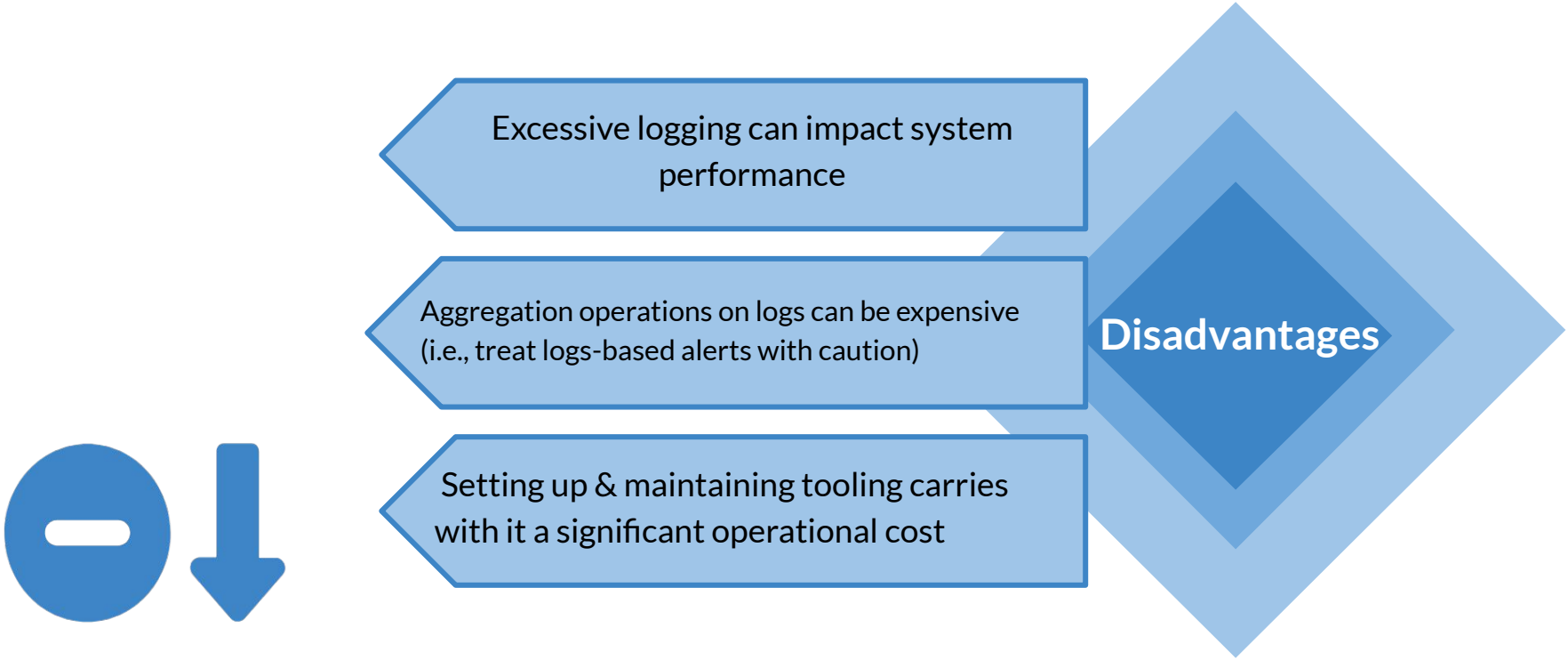
Google Cloud Monitoring

Amazon CloudWatch

Logging - Advantages



Logging - Disadvantages



Logging in Machine Learning

Key areas

➔ Use logs to keep track of the model inputs and predictions

Input red flags

➔ A feature becoming unavailable

➔ Notable shifts in the distributions

➔ Patterns specific to your model

Storing log data for analysis

- Basic log storage is often unstructured
- Parsing and storing log data in a queryable format enables analysis
  - Extracting values to generate distributions and statistics
  - Associating events with timestamps
  - Identifying the systems
- Enables automated reporting, dashboards, and alerting

New Training Data

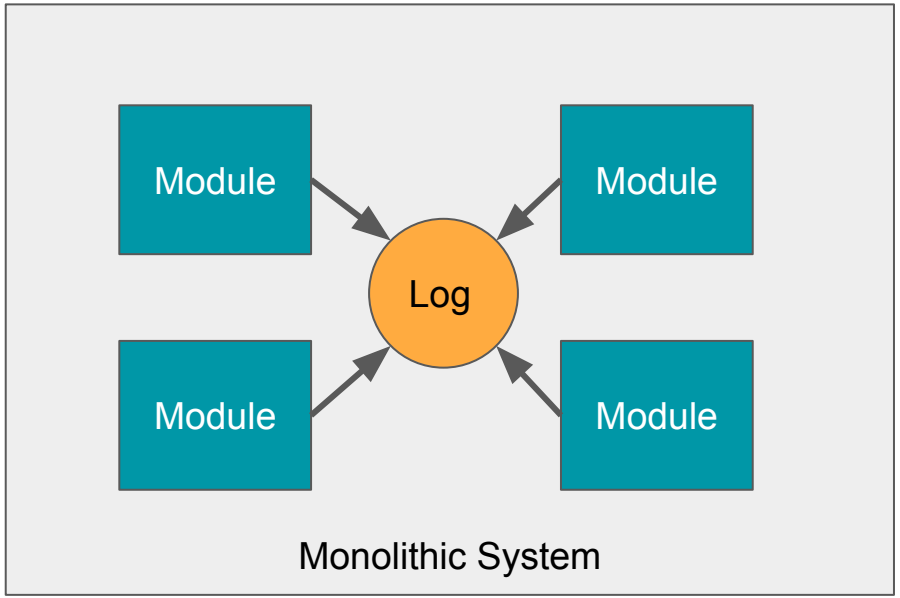
- Prediction requests form new training datasets
- For supervised learning, labels are required
  - Direct labeling
  - Manual labeling
  - Active learning
  - Weak supervision

DeepLearning.AI

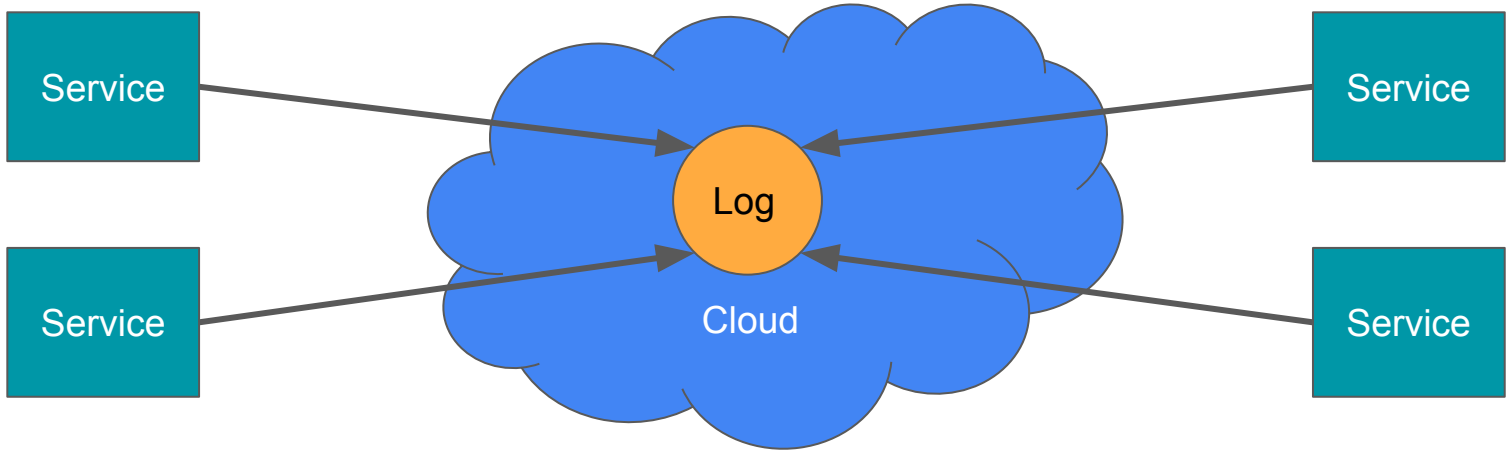
Model Monitoring

Tracing for ML Systems

Distributed Tracing



Distributed Tracing

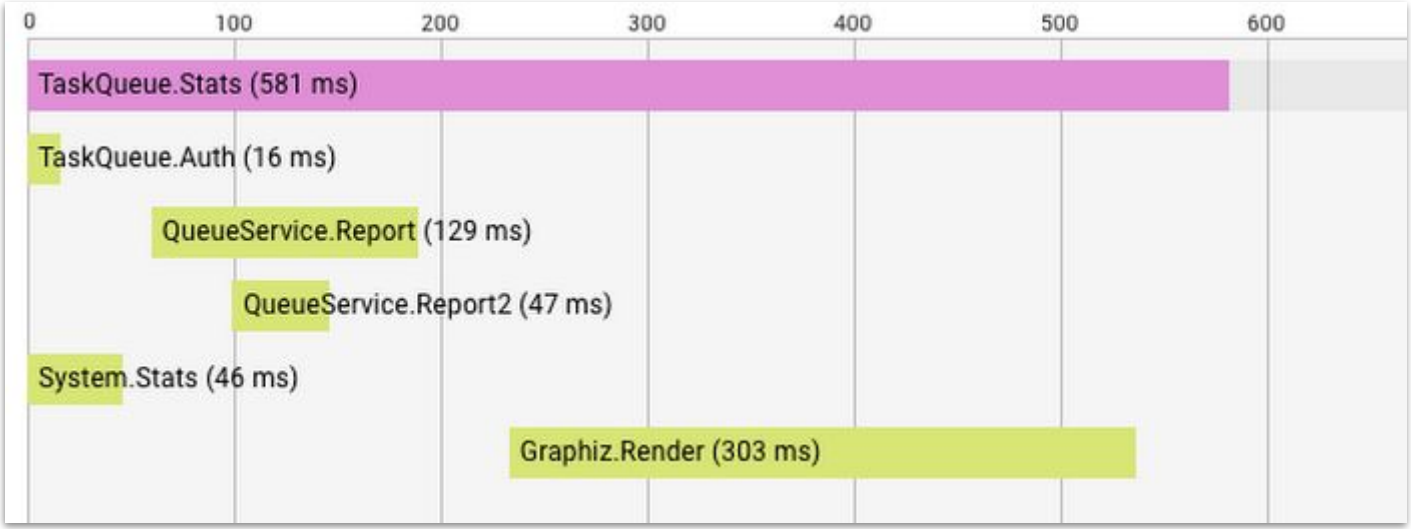


Tools for building observability

- Sequencing and parallelism of service requests
- Distributed tracing
  - Dapper
  - Zipkin
  - Jaeger

Dapper-Style Tracing

- Propagate trace between services
- A trace is a call tree, made up of one or more spans



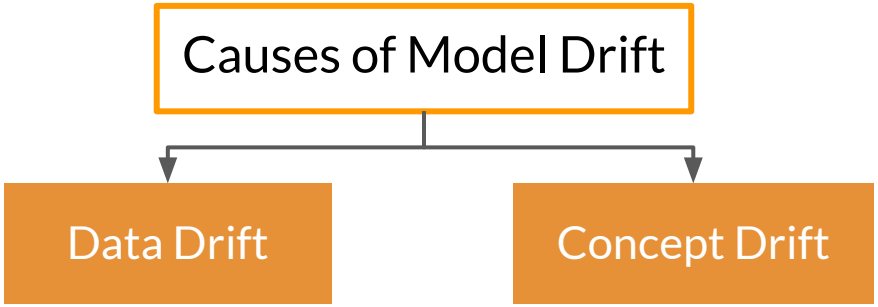
DeepLearning.AI

Model Monitoring

What is Model Decay?

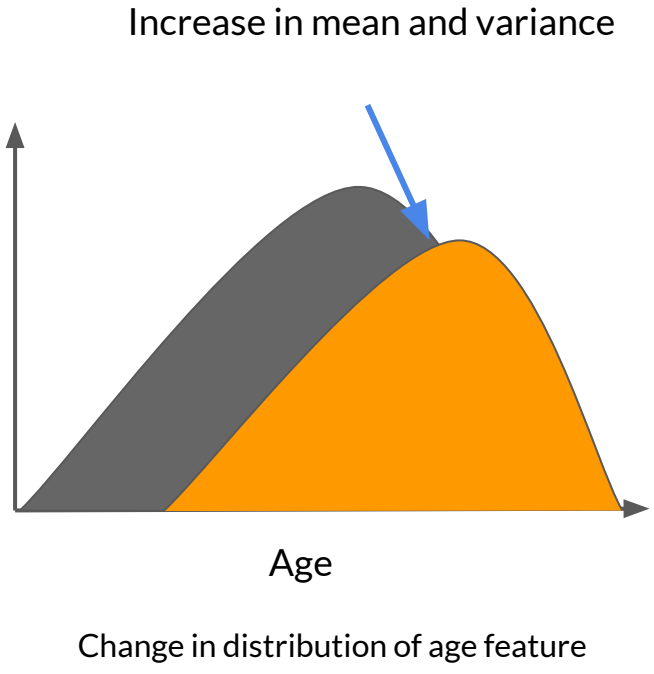
Model Decay

- Production ML models often operate in a dynamic environments
- The ground truth in dynamic environments changes
- If the model is static and does not change, then it gradually moves farther and farther away from the ground truth



Data Drift (aka Feature Drift)

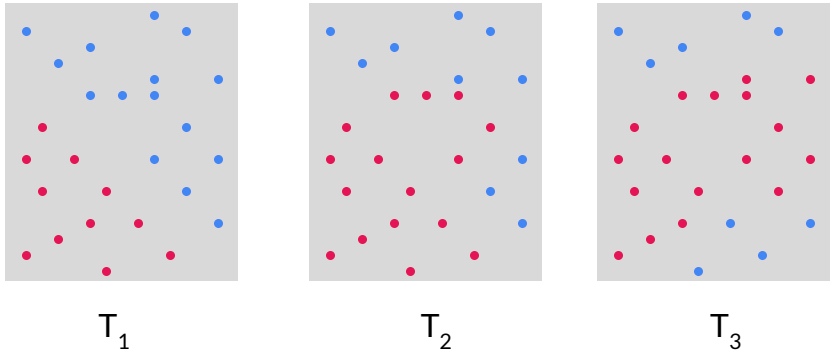
- Statistical properties of input changes
- Trained model is not relevant for changed data
- For eg., distribution of demographic data like age might change over time



Change in distribution of age feature

Concept Drift

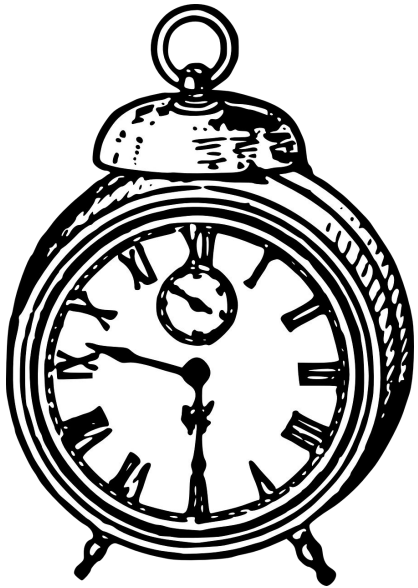
- Relationship between features and labels changes
- The very meaning of what you are trying to predict changes
- Prediction drift and label drift are similar



Change in relationship between the features and the labels

Detecting Drift on Time

- Drift creeps into the system slowly with time
- If it goes undetected, model accuracy suffers
- Important to monitor and detect drift early



DeepLearning.AI

Model Monitoring

Model Decay Detection

Detecting Concept and Data Drift

Log Predictions (Full Requests and Responses)

- Incoming prediction requests and generated prediction should be logged
- If possible log the ground truth that should have been predicted
    - Can be used as labels for new training data
- At a minimum log data in prediction request
    - This data is analysed to detect data drift that will cause model decay

Detecting Drift

- Detected by observing the statistical properties of logged data, model predictions, and possibly ground truth
- Deploy dashboards that plot statistical properties to observe how they change over time
- Use specialized libraries for detecting drift
    - TensorFlow Data Validation (TFDV)
    - Scikit-multiflow library

Continuous Evaluation and Labelling in Vertex Prediction

- **Vertex Prediction** offers continuous evaluation
- **Vertex Labelling Service** can be used to assign ground truth labels to prediction input for retraining
- Azure, AWS, and other cloud providers offer similar services

DeepLearning.AI

Model Monitoring

Ways to Mitigate Model Decay

Mitigating Model Decay

When you've detected model decay:

- At the minimum operational and business stakeholders should be notified of the decay
- Take steps to bring model back to acceptable performance

Steps in Mitigating Model Decay

- What if Drift is Detected?
  - If possible, determine the portion of your training set that is still correct
  - Keep the good data, discard the bad, and add new data - OR -
  - Discard data collected before a certain date and add new data - OR -
  - Create an entirely new training dataset from new data

Fine Tune, or Start Over?

- You can either continue training your model, fine tuning from the last checkpoint using new data - OR -
- Start over, reinitialize your model, and completely retrain it
- Either approach is valid, so it really depends on results
  - How much new labelled data do you have?
  - How far has it drifted?
  - Try both and compare

Model Re-Training Policy

On-Demand	• Manually re-train the model
On a Schedule	• New labelled data is available at a daily, weekly or monthly basis
Availability of New Training Data	• New data available on ad-hoc basis, when it is collected and available in source database

Automating Model Retraining

Model Performance Degradation	• Manually retrain the model
Data Drift	• When you notice significant changes in the data

Redesign Data Processing Steps and Model Architecture

- When model performance decays beyond an acceptable threshold you might have to consider redesigning your entire pipeline
- Re-think feature engineering, feature selection
- You may have to train your model from scratch
- Investigate on alternative architectures

DeepLearning.AI

Model Monitoring

Responsible AI

Responsible AI Practices

- Development of AI **creates new opportunities** to improve the lives of people around the world
    - Business, healthcare, education, etc.
- But it also **raises new questions** about implementing responsible practices
    - Fairness, interpretability, privacy, and security
    - Far from solved, active areas of research and development

Human-Centered Design

Actual users' experience is essential

- Design your features with appropriate disclosures built-in
- Consider augmentation and assistance
    - Offering multiple suggestions instead of one right answer
- Model potential adverse feedback early in the design process
- Engage with a diverse set of users and use-case scenarios

Identify Multiple Metrics

- Using several metrics helps you understand the tradeoffs
  - Feedback from user surveys
  - Quantities that track overall system performance
  - False positive and false negative sliced across subgroups
- Metrics must be appropriate for the context and goals of your system

Analyze your raw data carefully

- For sensitive raw data, respect privacy
  - Compute aggregate, anonymized summaries
- Does your data reflect your users?
  - Example: will be used for all ages, but all data from senior citizens
- Imperfect proxy labels?
  - Relationship between the labels and actual targets
  - Using label X as a proxy for target Y - any problematic gaps?

DeepLearning.AI

Model Monitoring

Legal Requirements for Secure & Private AI

Legal Implications of Data Security and Privacy

Companies must comply with data privacy protection laws in regions where they operate

General Data Protection Regulation (GDPR)

California Consumer Privacy Act (CCPA)

General Data Protection Regulation (GDPR)

- Regulation in EU law on data protection and privacy in the European Union (EU) and the European Economic Area (EEA)
- Give control to individuals over their data
- Companies should protect the data of employees and consumers
- When the data processing is based on consent, the data subject has the right to revoke their consent at any time

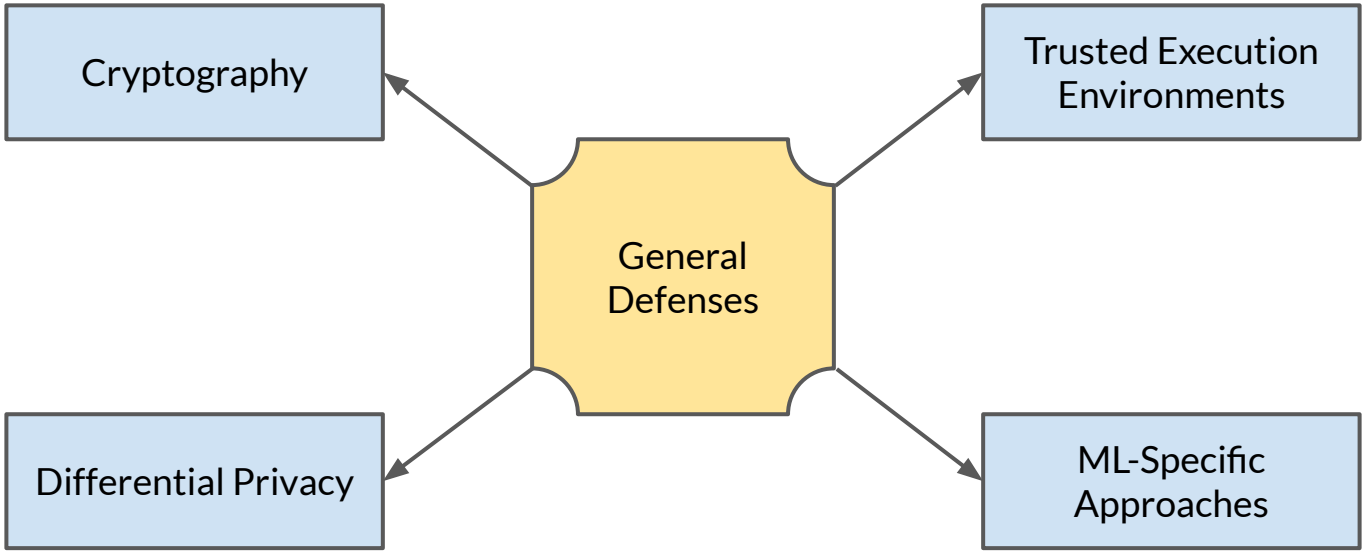
California Consumer Privacy Act (CCPA)

- Similar to GDPR
- Intended to enhance privacy rights and consumer protection for residents of California
- User has the right to know what personal data is being collected about them, whether the personal data is sold or disclosed, and to whom
- User can access the personal data, block the sale of their data, and request a business to delete their data

Security and Privacy Harms from ML Models

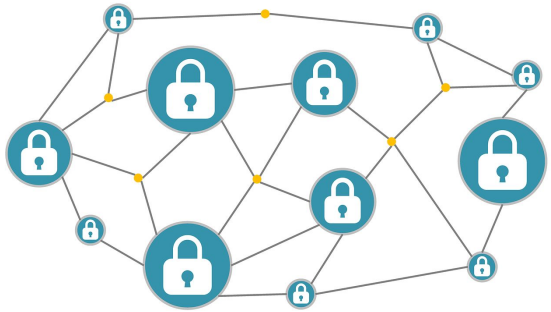
Informational Harms	Behavioural Harms
Relate to unintended or unanticipated leakage of information	Relate to manipulating the behavior of the model itself, impacting the predictions or outcomes of the model

Defenses


Cryptography
Trusted Execution Environments
General Defenses
Differential Privacy
ML-Specific Approaches

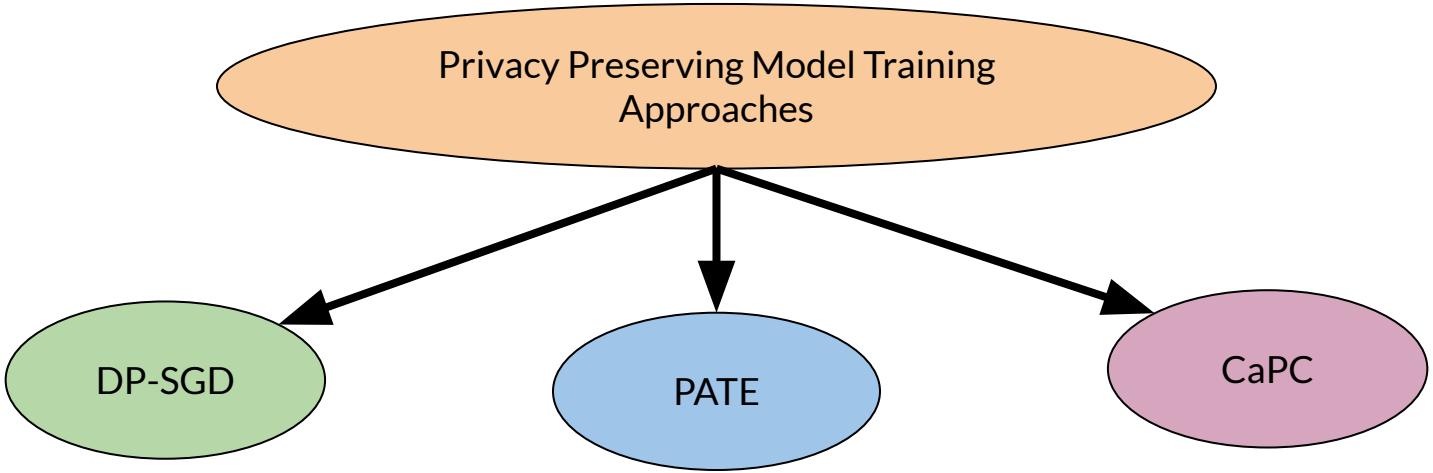
Cryptography

- Privacy-enhancing tools (like SMPC and FHE) should be considered to securely train supervised machine learning models
- Users can send encrypted prediction requests while preserving the confidentiality of the model
- Protects confidentiality of the training data



Differential Privacy

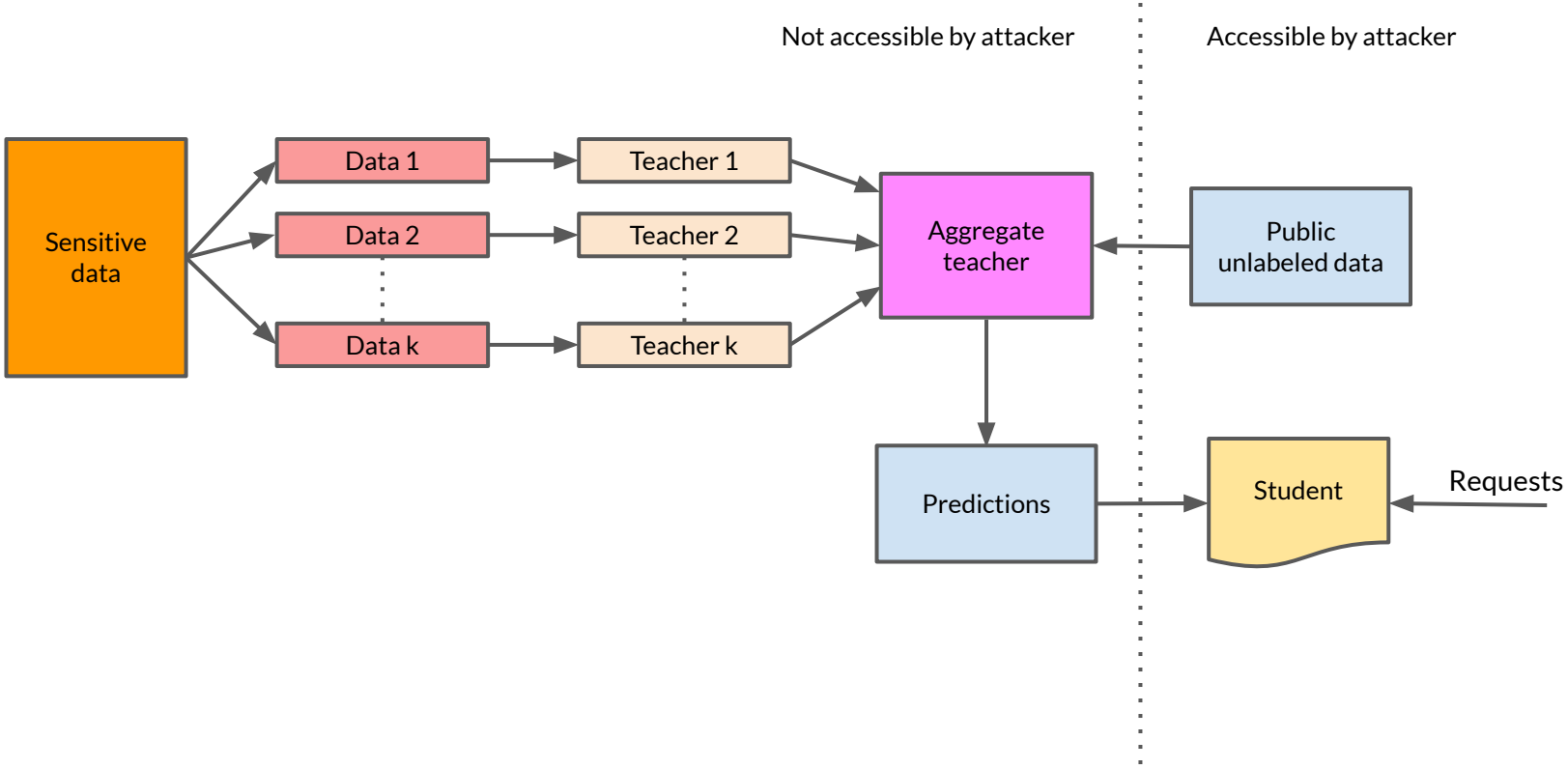
System for publicly sharing information about a dataset by describing the patterns of groups within the dataset while withholding information about individuals in the dataset



Differentially-Private Stochastic Gradient Descent (DP-SGD)

- Applies differential privacy during model training
- Modifies the minibatch stochastic optimization process
- Trained model retains differential privacy because of the post-processing immunity property of differential privacy

Private Aggregation of Teacher Ensembles (PATE)



Confidential and Private Collaborative Learning (CaPC)

- Enables models to collaborate while preserving the privacy of the underlying data
- Integrates building blocks from cryptography and differential privacy to provide confidential and private collaborative learning
- Encrypts prediction requests using Homomorphic Encryption (HE)
- Uses PATE to add noise to predictions for voting

DeepLearning.AI

Model Monitoring

Anonymization & Pseudonymisation

Data Anonymization in GDPR

- GDPR includes many regulations to preserve privacy of user data
- Since introduction of GDPR, two terms have been discussed widely

Anonymization

Pseudonymisation

Data Anonymization

Recital 26 of GDPR defines Data Anonymization

True data anonymization is:

- Irreversible
- Done in such a way that it is impossible to identify the person
- Impossible to derive insights or discrete information, even by the party responsible for anonymization

GDPR does not apply to data that has been anonymized

Pseudonymisation

- GDPR Article 4(5) defines pseudonymisation as:

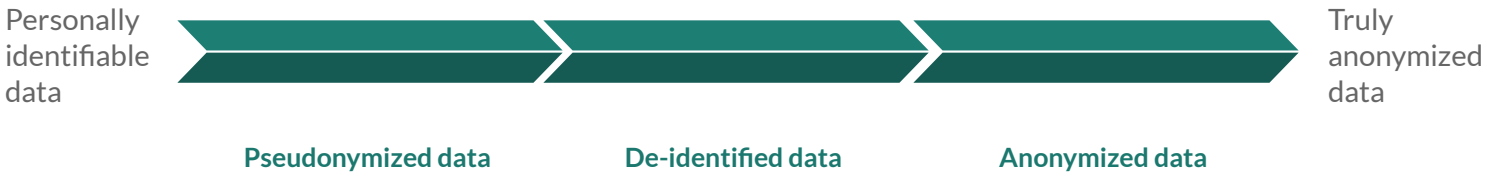
  "... the processing of personal data in such a way that the data can no longer be attributed to a specific data subject without the use of additional information"

- The data is anonymized by switching the identifiers (like email or name) with an alias or pseudonym

Pseudonymisation v Anonymization

Information	Pseudonymized	Anonymized
Chelsea	Puryfrn	*******
Kumar	Xhzne	*******
Zaed	Mnrq	*******
John	Wbua	*******
Doe	Qbr	*******
Alex	Nyrk	*******

Spectrum of Privacy Preservation



What Data Should be Anonymized?

- Any data that reveals the identity of a person, referred to as identifiers
- Identifiers applies to any natural or legal person, living or dead, including their dependents, ascendants, and descendants
- Included are other related persons, direct or through interaction
- For example: Family names, patronyms, first names, maiden names, aliases, address, phone, bank account details, credit cards, IDs like SSN

DeepLearning.AI

Model Monitoring

Right to Be Forgotten

What is the Right to Be Forgotten?

“The data subject shall have the right to obtain from the controller the erasure of personal data concerning him or her without undue delay and the controller shall have the obligation to erase personal data without undue delay”

- Recitals 65 and 66 and in Article 17 of the GDPR

What is the Right to Be Forgotten?

“The data subject shall have the right to obtain from the controller the erasure of personal data concerning him or her without undue delay and the controller shall have the obligation to erase personal data without undue delay”

- Recitals 65 and 66 and in Article 17 of the GDPR

Right to Rectification

“The data subject shall have the right to obtain from the controller without undue delay the rectification of inaccurate personal data concerning him or her. Taking into account the purposes of the processing, the data subject shall have the right to have incomplete personal data completed, including by means of providing a supplementary statement.”

- Chapter 3, Art. 16 GDPR

Other Rights of the Data Subject

Chapter 3 defines a number of other rights of the data subject, including:

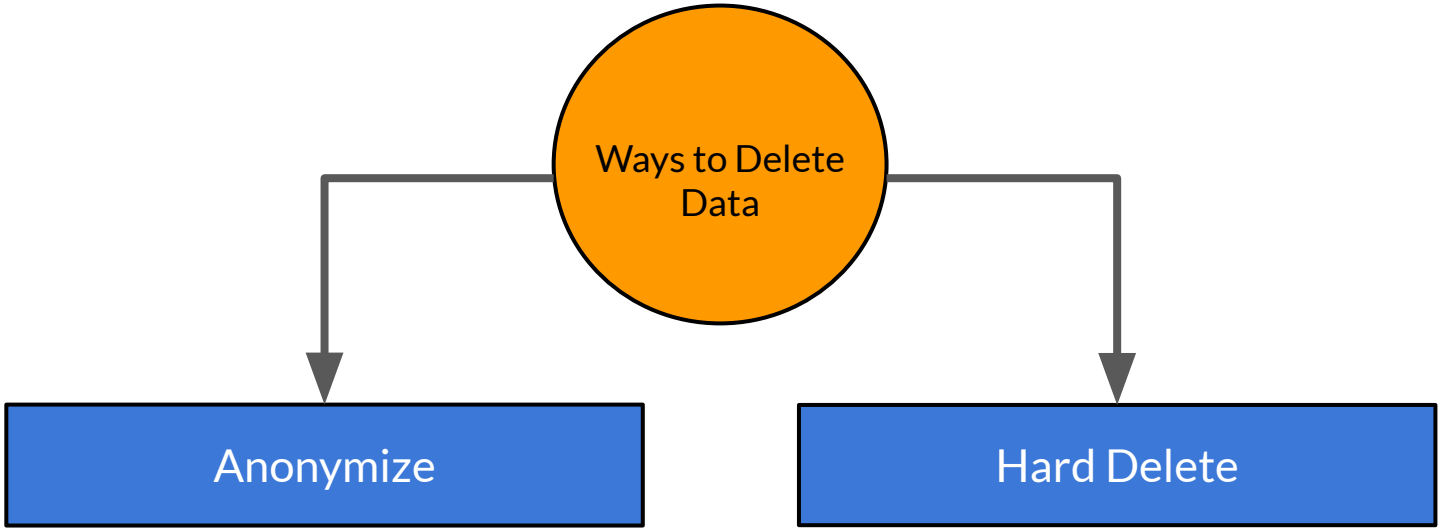
- Art. 15 GDPR – Right of access by the data subject
- Art. 18 GDPR – Right to restriction of processing
- Art. 20 GDPR – Right to data portability
- Art. 21 GDPR – Right to object

Implementing Right To Be Forgotten: Tracking Data

For a valid erasure claim

- Company needs to identify all of the information related to the content requested to be removed
- All of the associated metadata must also be erased
    - Eg., Derived data, logs etc.

Forgetting Digital Memories



Issues with Hard Delete

- Deleting records from a database can cause havoc
- User data is often referenced in multiple tables
- Deletion breaks the connections, which can be difficult in large, complex databases
- Can break foreign keys
- Anonymization keeps the records, and only anonymizes the fields containing PII

Challenges in Implementing Right to Be Forgotten