DeepLearning.AI

Labeling Data

Case Study: Degraded Model Performance

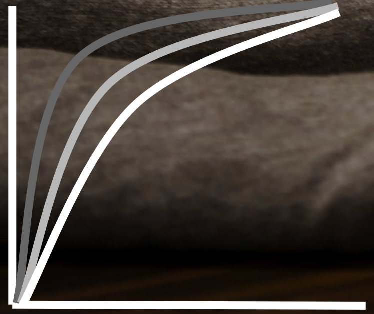
You're an Online Retailer Selling Shoes ...

Your model predicts **click-through rates (CTR)**, helping you decide how much inventory to order

DeepLearning.AI

When suddenly

Your AUC and prediction accuracy have dropped on men's dress shoes!




Why?

How do we know that we have a problem?

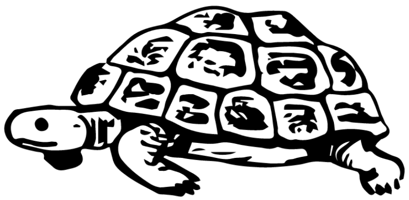
Case study: taking action

- How to detect problems early on?
- What are the possible causes?
- What can be done to solve these?

What causes problems?

Kinds of problems:

- Slow - example: drift
- Fast - example: bad sensor, bad software update



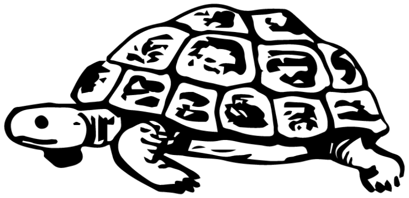
Gradual problems

Data changes

- Trend and seasonality
- Distribution of features changes
- Relative importance of features changes

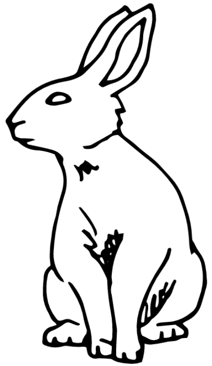
World changes

- Styles change
- Scope and processes change
- Competitors change
- Business expands to other geos



Sudden problems

Data collection problem	Systems problem
• Bad sensor/camera • Bad log data • Moved or disabled sensors/cameras	• Bad software update • Loss of network connectivity • System down • Bad credentials



Why “Understand” the model?

- Mispredictions do not have uniform **cost** to your business
- The **data you have** is rarely the data you wish you had
- Model objective is nearly always a **proxy** for your business objectives
- Some percentage of your customers may have a **bad experience**

The real world does not stand still!

DeepLearning.AI

Labeling Data

Data and Concept Change in Production ML

Outline

- Detecting problems with deployed models
    - Data and concept change
- Changing ground truth
    - Easy problems
    - Harder problems
    - Really hard problems

Detecting problems with deployed models

- Data and scope changes
- Monitor models and validate data to find problems early
- Changing ground truth: **label** new training data

Easy problems

- Ground truth changes slowly (months, years)
- Model retraining driven by:
    - Model improvements, better data
    - Changes in software and/or systems
- Labeling
    - Curated datasets
    - Crowd-based

Harder problems

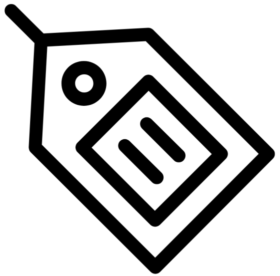
- Ground truth changes faster (weeks)
- Model retraining driven by:
    - **Declining model performance**
    - Model improvements, better data
    - Changes in software and/or system
- Labeling
    - Direct feedback
    - Crowd-based

Really hard problems

- Ground truth changes very fast (days, hours, min)
- Model retraining driven by:
    - **Declining model performance**
    - Model improvements, better data
    - Changes in software and/or system
- Labeling
    - Direct feedback
    - Weak supervision

Key points

- Model performance decays over time
    - Data and Concept Drift
- Model retraining helps to improve performance
    - Data labeling for changing ground truth and scarce labels



DeepLearning.AI

Labeling Data

Process Feedback and Human Labeling

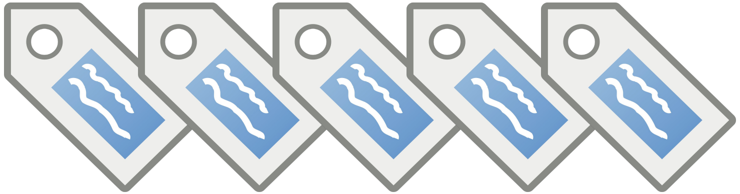
Data labeling

Variety of Methods

- Process Feedback (Direct Labeling)
- Human Labeling
- ~~Semi-Supervised Labeling~~
- ~~Active Learning~~
- ~~Weak Supervision~~

➡ Practice later as advanced labeling methods

Data labeling

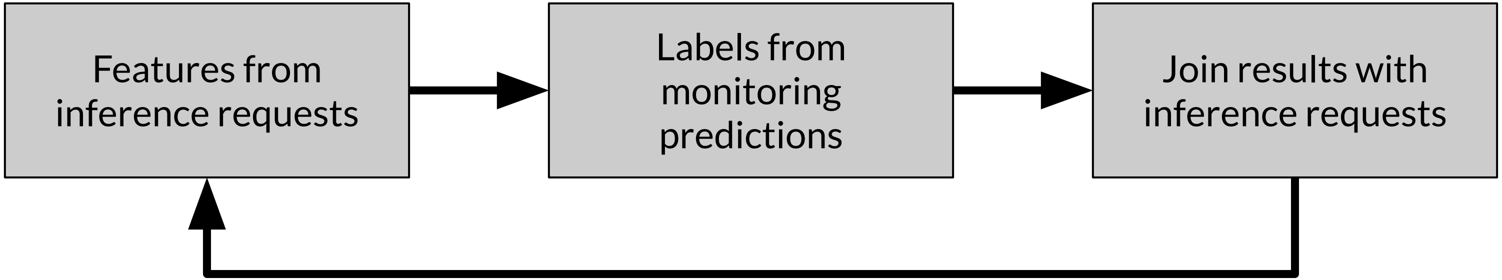


- **Process Feedback** — Example: Actual vs predicted click-through
- **Human Labeling** — Example: Cardiologists labeling MRI images

Why is labeling important in production ML?

- Using business/organisation available data
- Frequent model retraining
- Labeling ongoing and critical process
- Creating a training datasets requires labels

Direct labeling: continuous creation of training dataset



Similar to reinforcement learning rewards

Process feedback - advantages

- Training dataset continuous creation
- Labels evolve quickly
- Captures strong label signals

Process feedback - disadvantages

- Hindered by inherent nature of the problem
- Failure to capture ground truth
- Largely bespoke design

Process feedback - Open-Source log analysis tools

Logstash

Free and open source data processing pipeline

- Ingests data from a multitude of sources
- Transforms it
- Sends it to your favorite "stash."

Fluentd

Open source data collector

Unify the data collection and consumption

Process feedback - Cloud log analytics

Cloud Log Analysis

Google Cloud Logging

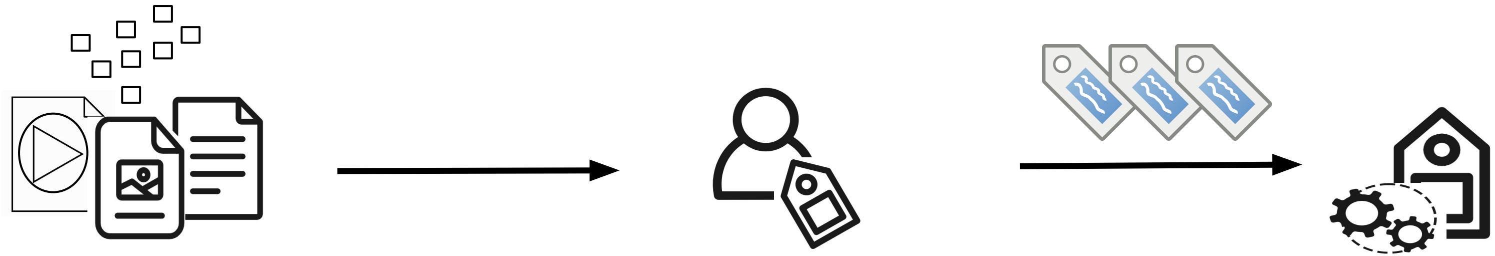
- Data and events from Google Cloud and AWS
- BindPlane. Logging: application components, on-premise and hybrid cloud systems
- Sends it to your favorite "stash"

AWS ElasticSearch

Azure Monitor

Human labeling

People (“raters”) to examine data and assign labels manually



Raw data

Unlabeled and ambiguous data is sent to raters for annotation

A training data set is ready for use

DeepLearning.AI

Human labeling - Methodology

Unlabeled data is collected

Human “raters” are recruited

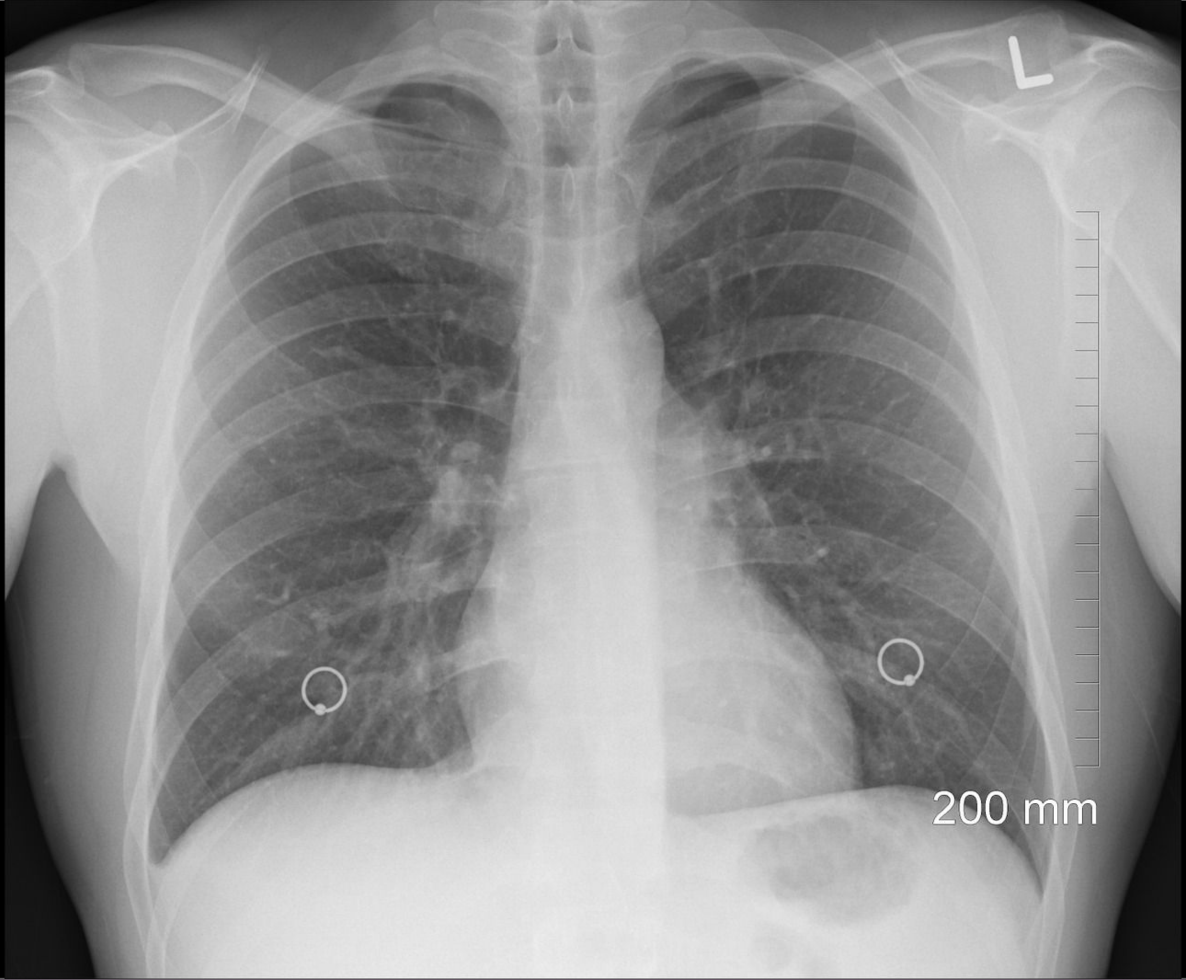
Instructions to guide raters are created

Data is divided and assigned to raters

Labels are collected and conflicts resolved

Human labeling - advantages

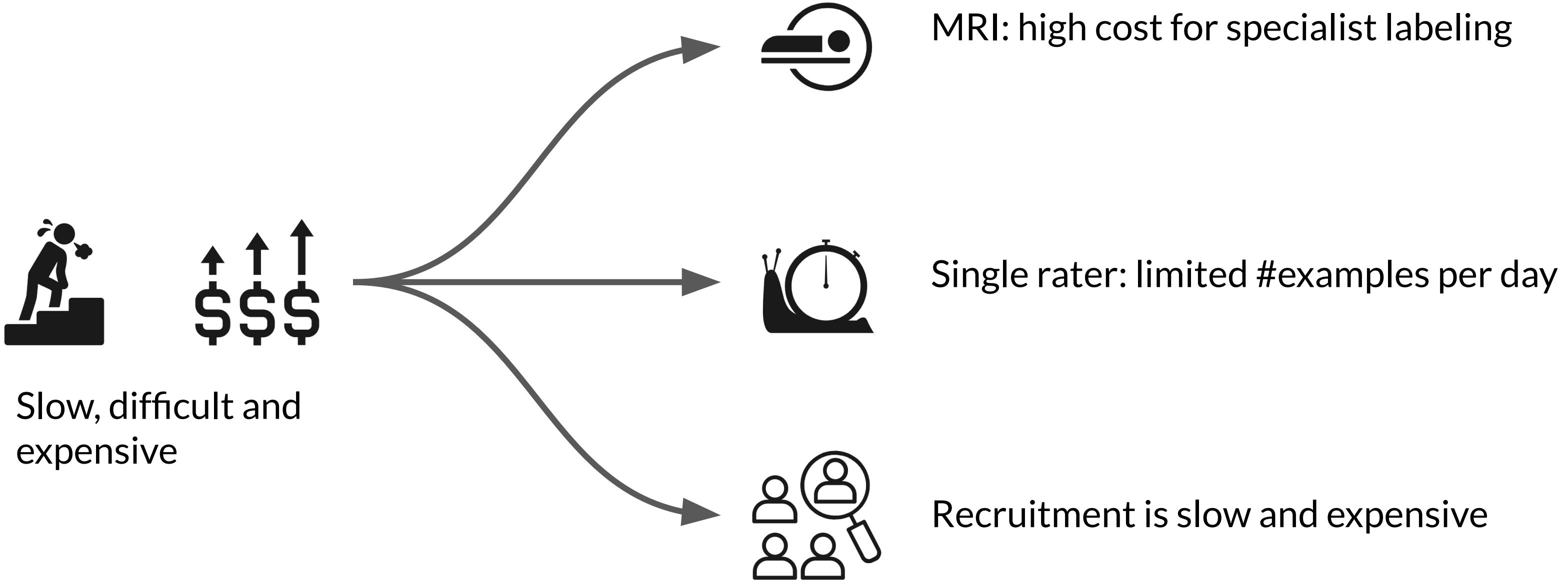
- More labels
- Pure supervised learning


L
200 mm

Human labeling - Disadvantages

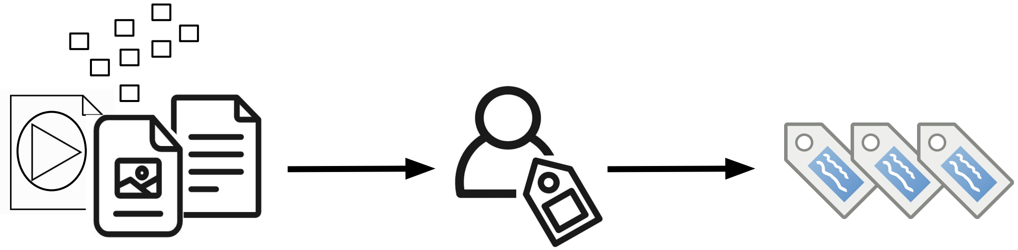
- Quality consistency: Many datasets difficult for human labeling
- Slow
- Expensive
- Small dataset curation

Why is human labeling a problem?



Key points

- Various methods of data labeling
  - Process feedback
  - Human labeling



- Advantages and disadvantages of both

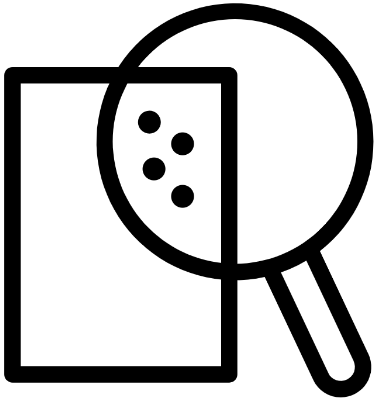
DeepLearning.AI

Validating Data

Detecting Data Issues

Outline

- Data issues
  - Drift and skew
    - Data and concept Drift
    - Schema Skew
    - Distribution Skew
- Detecting data issues



Drift and skew

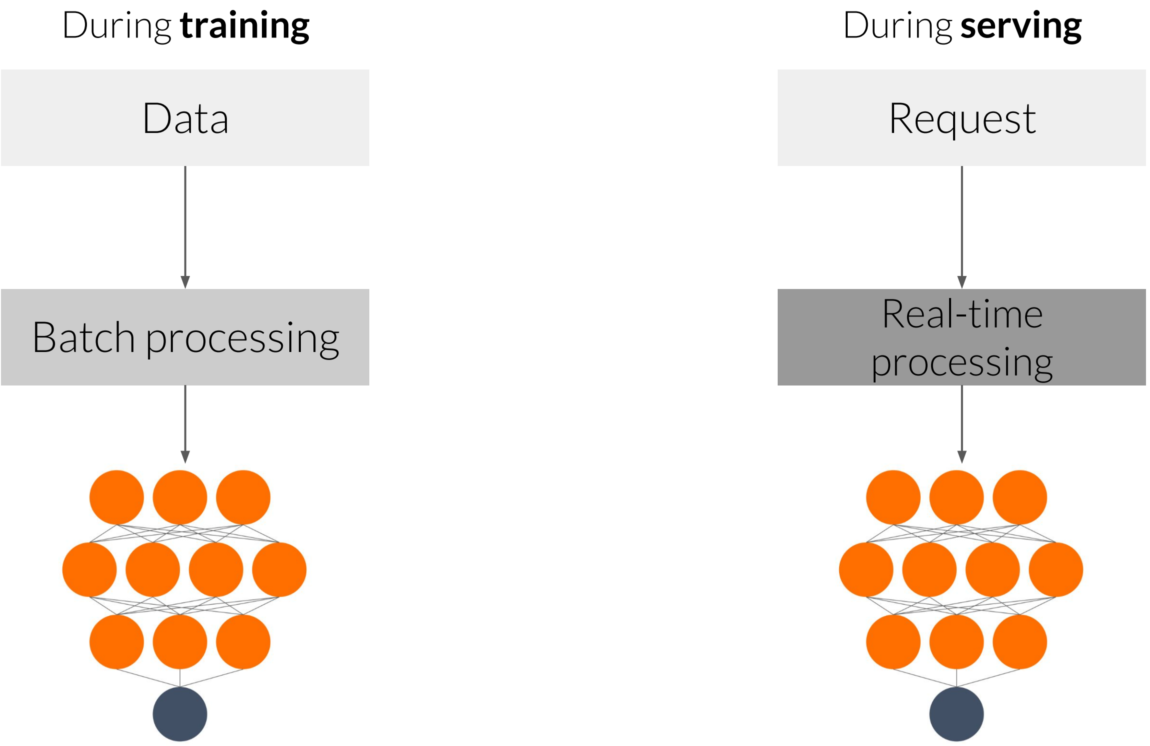
Drift

Changes in data over time, such as data collected once a day

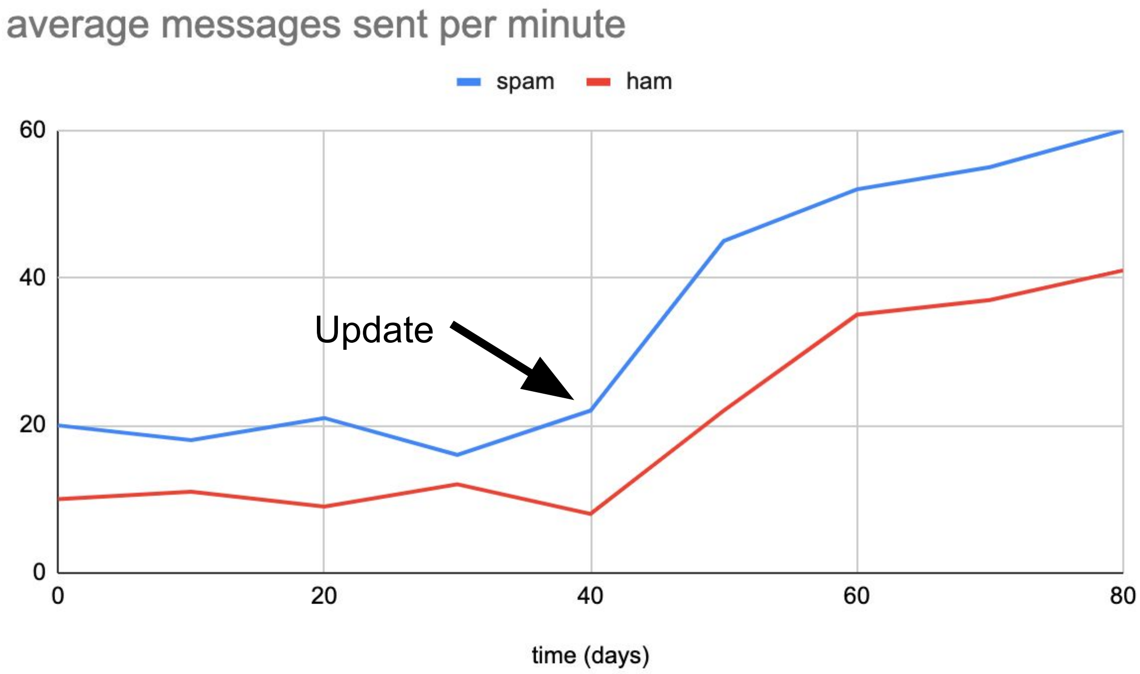
Skew

Difference between two static versions, or different sources, such as training set and serving set

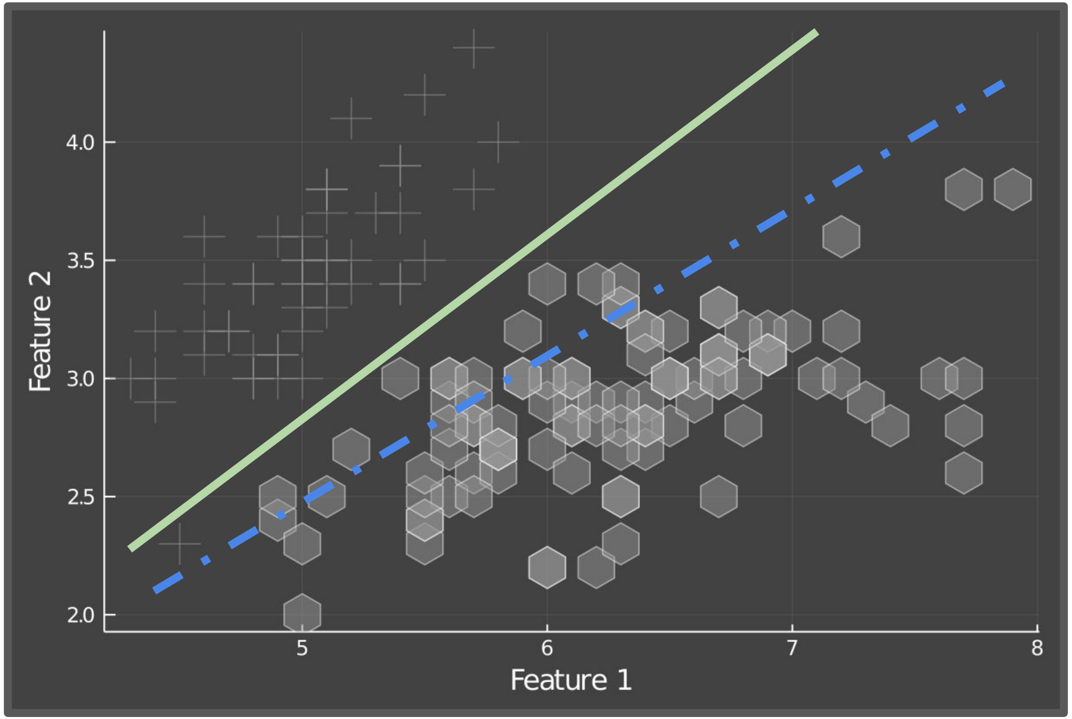
Typical ML pipeline



Model Decay : Data drift



Performance decay : Concept drift



Training

Serving

Detecting data issues

- Detecting schema skew
    - Training and serving data do not conform to the same schema
- Detecting distribution skew
    - Dataset shift → covariate or concept shift
- Requires continuous evaluation

Detecting distribution skew

	Training	Serving
Joint	$P_{\text{train}}(y, x)$	$P_{\text{serve}}(y, x)$
Conditional	$P_{\text{train}}(y\|x)$	$P_{\text{serve}}(y\|x)$
Marginal	$P_{\text{train}}(x)$	$P_{\text{serve}}(x)$

Dataset shift $P_{\text{train}}(y, x) \neq P_{\text{serve}}(y, x)$

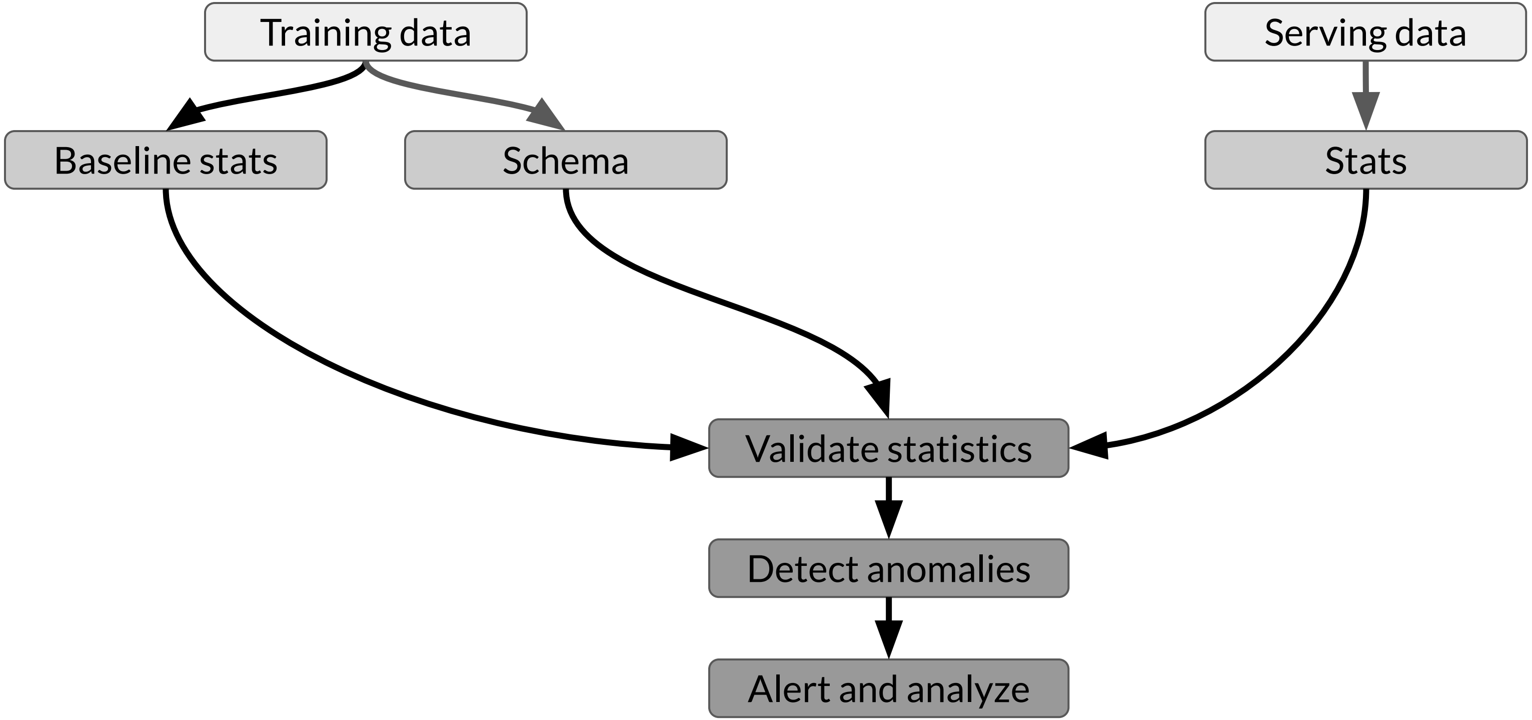
Covariate shift $P_{\text{train}}(y|x) = P_{\text{serve}}(y|x)$

$P_{\text{train}}(x) \neq P_{\text{serve}}(x)$

Concept shift $P_{\text{train}}(y|x) \neq P_{\text{serve}}(y|x)$

$P_{\text{train}}(x) = P_{\text{serve}}(x)$

Skew detection workflow



Validating Data

TensorFlow Data Validation

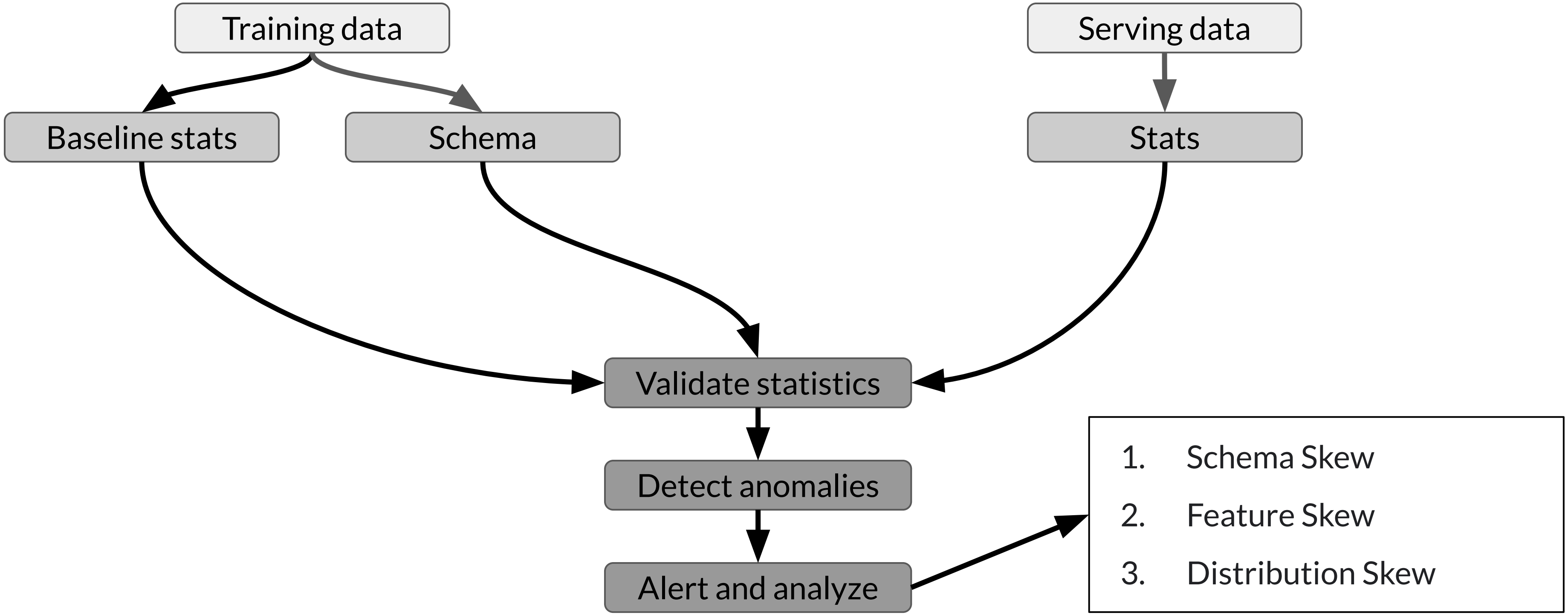
TensorFlow Data Validation (TFDV)

- Understand, validate, and monitor ML data at scale
- Used to analyze and validate petabytes of data at Google every day
- Proven track record in helping TFX users maintain the health of their ML pipelines

TFDV capabilities

- Generates data statistics and browser visualizations
- Infers the data schema
- Performs validity checks against schema
- Detects training/serving skew

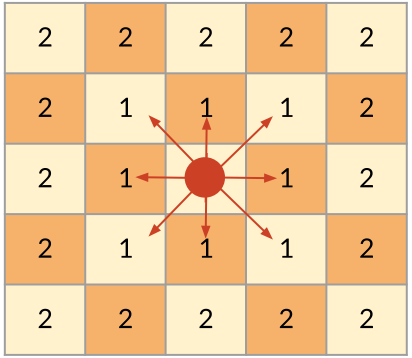
Skew detection - TFDV



Skew - TFDV

- Supported for categorical features
- Expressed in terms of L-infinity distance (Chebyshev Distance):

$$D_{\text{Chebyshev}}(x, y) = \max_i(|x_i - y_i|)$$



- Set a threshold to receive warnings

Schema skew

Serving and training data don't conform to same schema:

- For example, `int != float`

Feature skew

Training **feature values** are different than the serving **feature values:**

- Feature values are modified between training and serving time
- Transformation applied only in one of the two instances

Distribution skew

Distribution of serving and training dataset is significantly different:

- Faulty sampling method during training
- Different data sources for training and serving data
- Trend, seasonality, changes in data over time

Key points

- TFDV: Descriptive statistics at scale with the embedded facets visualizations
- It provides insight into:
    - What are the underlying statistics of your data
    - How does your training, evaluation, and serving dataset statistics compare
    - How can you detect and fix data anomalies

Wrap up

- Differences between ML modeling and a production ML system
- Responsible data collection for building a fair production ML system
- Process feedback and human labeling
- Detecting data issues

Practice data validation with TFDV in this week's exercise notebook

Test your skills with the programming assignment