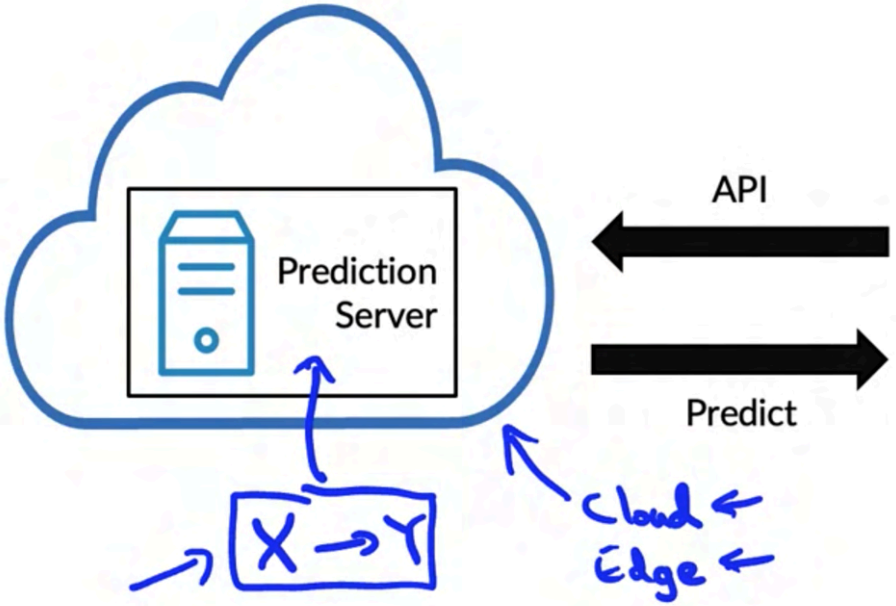
DeepLearning.AI

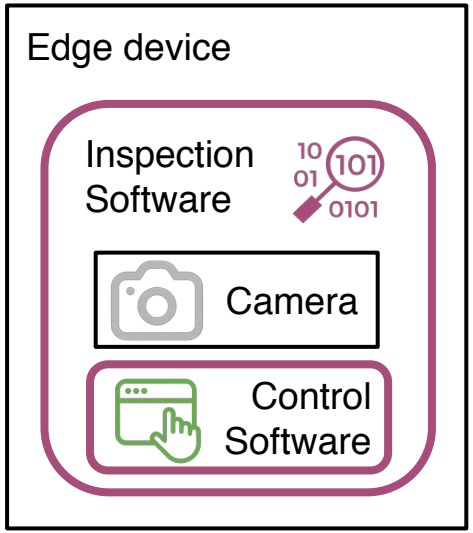
The Machine Learning Project Lifecycle

Welcome

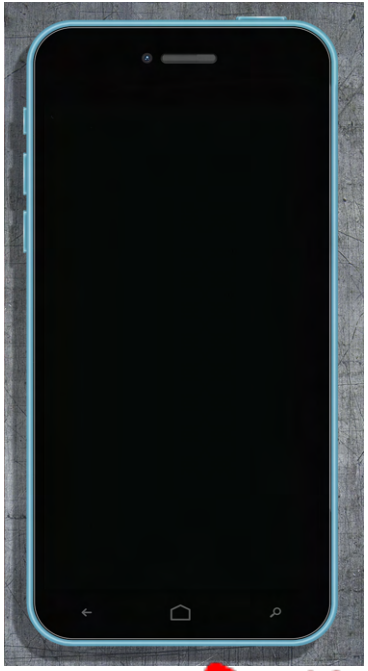
Deployment example

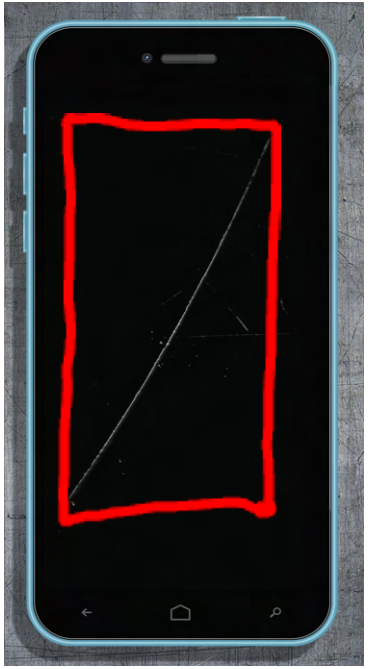
Photo from camera



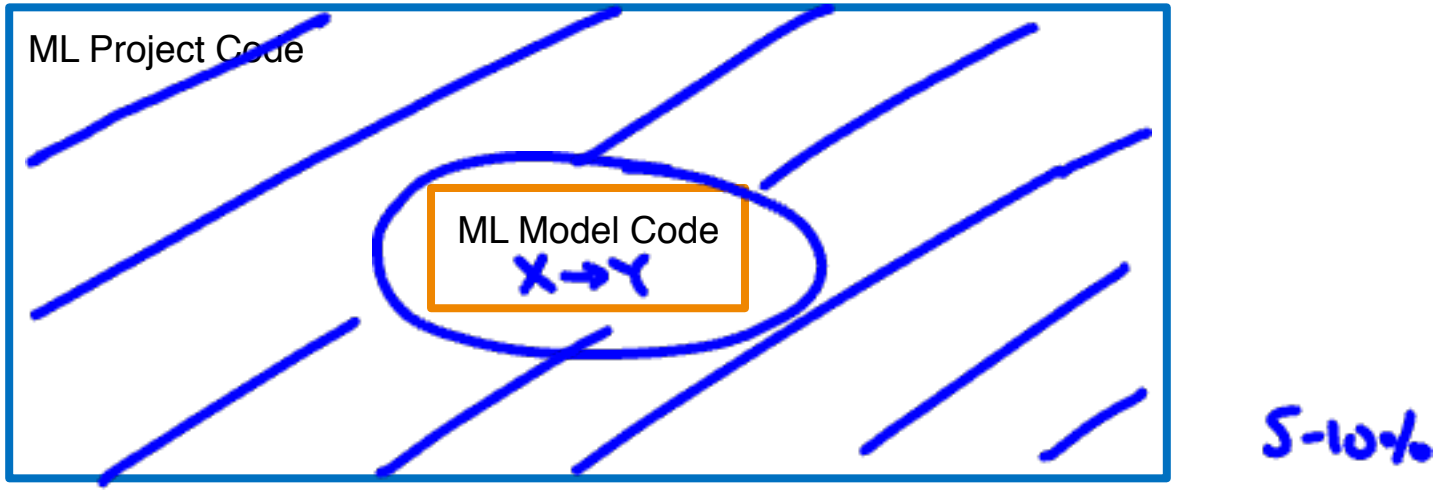


Visual inspection example



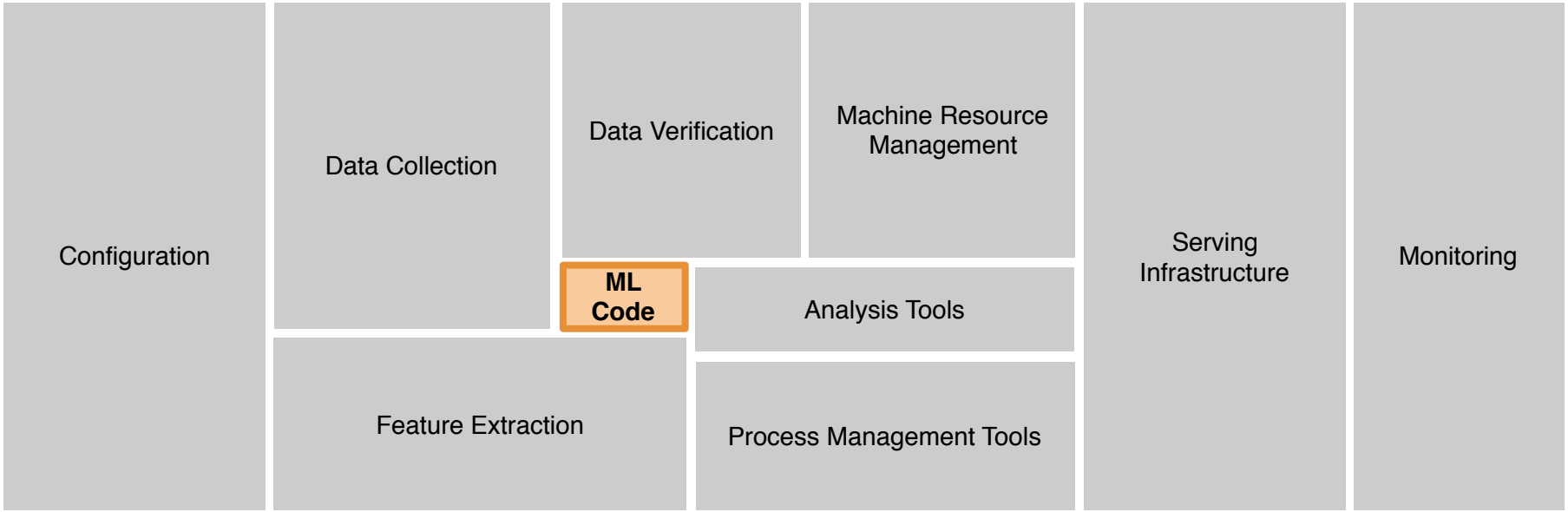


ML in production



"POC to Production Gap"

The requirements surrounding ML infrastructure



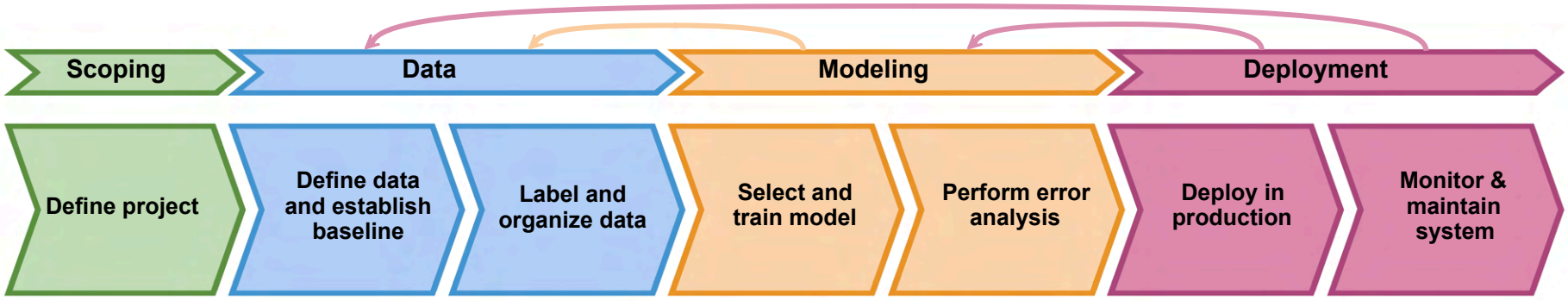
[D. Sculley et. al. NIPS 2015: Hidden Technical Debt in Machine Learning Systems]

DeepLearning.AI

The Machine Learning Project Lifecycle

Steps of an ML project

The ML project lifecycle

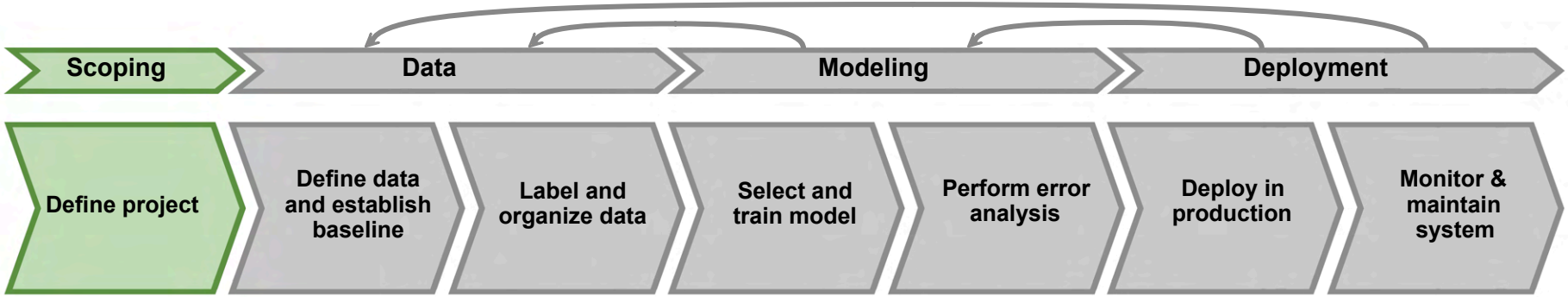


DeepLearning.AI

The Machine Learning Project Lifecycle

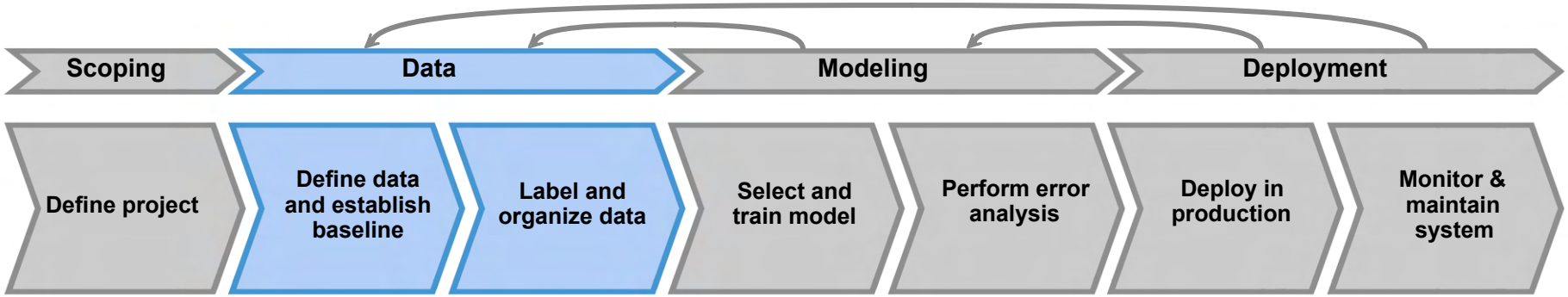
Case study:
speech recognition

Speech recognition: Scoping stage



- Decide to work on speech recognition for voice search.
- Decide on key metrics:
  - Accuracy, latency, throughput
- Estimate resources and timeline

Speech recognition: Data stage



Define data

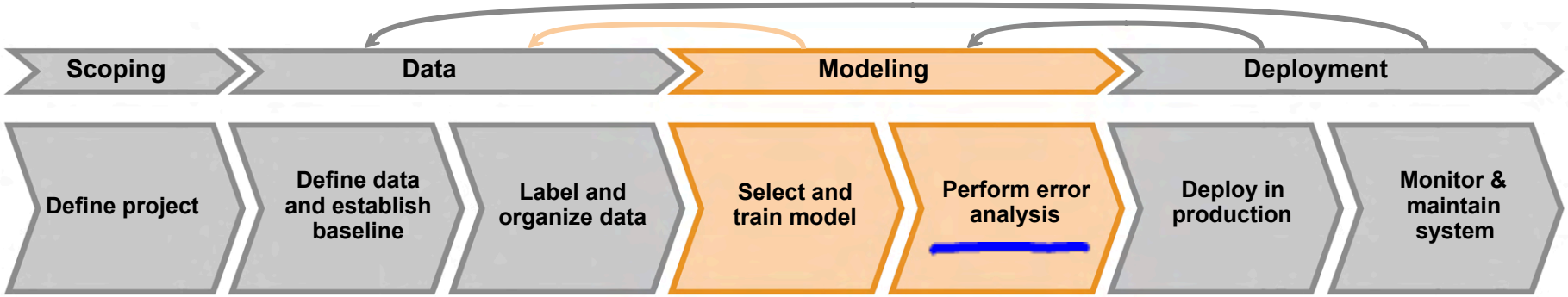
- Is the data labeled consistently?
- How much silence before/after each clip?
- How to perform volume normalization?

"Um, today's weather"

"Um… today's weather"

"Today's weather"

Speech recognition: Modeling stage



Research/Academia

Product Team

Code (algorithm/model)

Hyperparameters

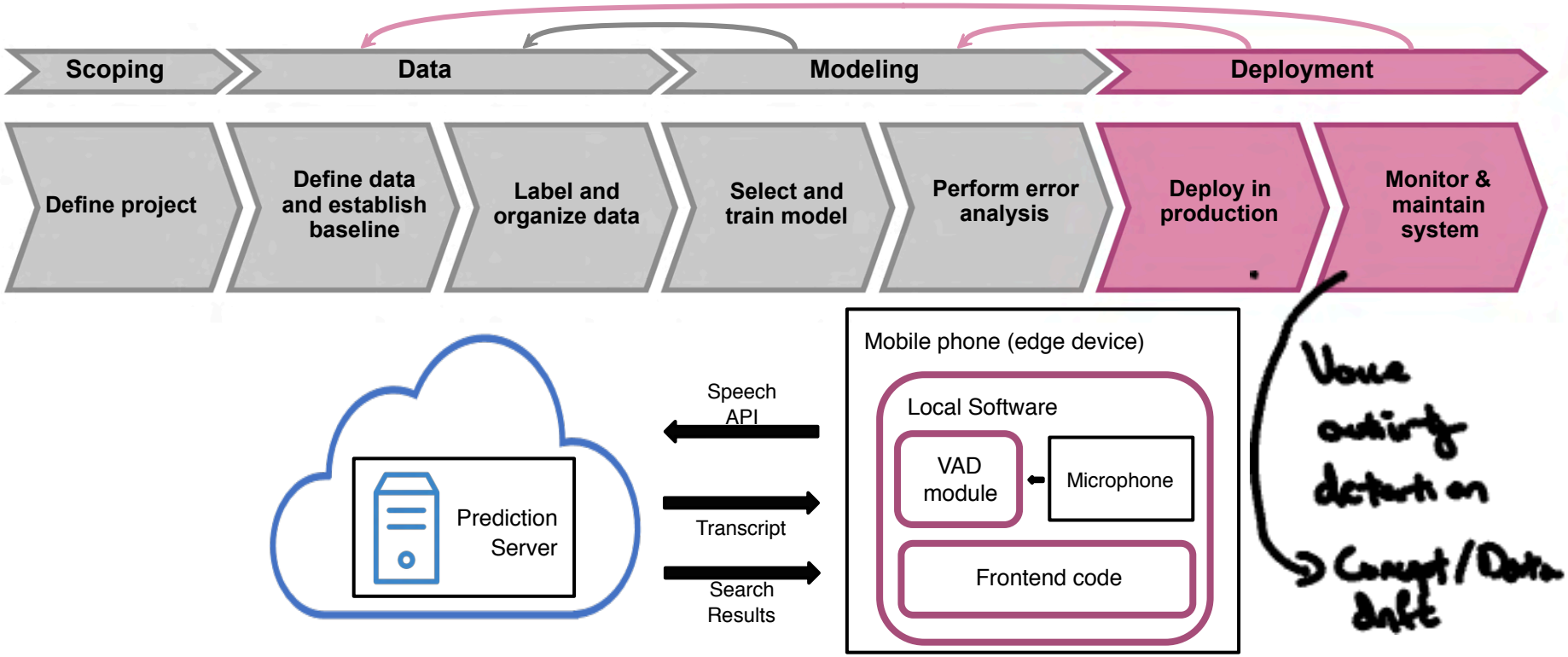
Data

ML Model

X → Y

ML System = Code + Data

Speech recognition: Deployment stage

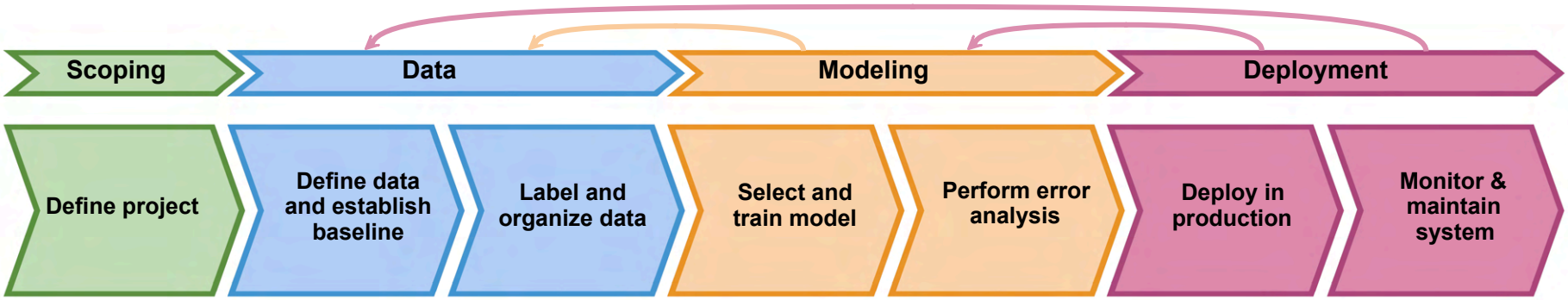


DeepLearning.AI

The Machine Learning Project Lifecycle

Course outline

Course outline



1. Deployment
2. Modeling
3. Data

Optional: Scoping

MLOps (Machine Learning Operations) is an emerging discipline, and comprises a set of tools and principles to support progress through the ML project lifecycle.

DeepLearning.AI

Deployment

Key challenges

Concept drift and Data drift

Speech recognition example

Training set:

- Purchased data, historical user data with transcripts

Test set:

- Data from a few months ago

Gradual change

Sudden shock

How has the data changed?

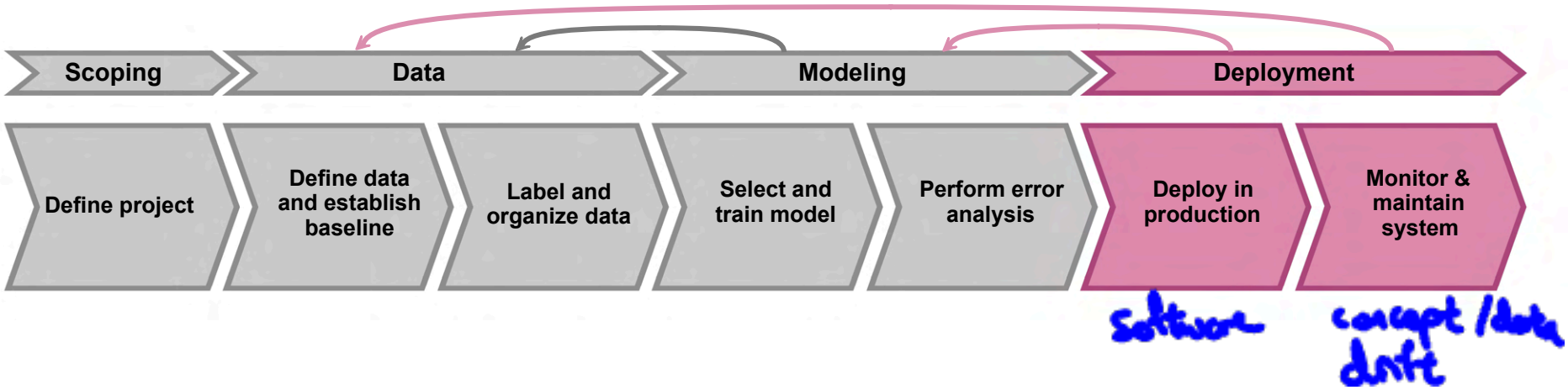
Software engineering issues

Checklist of questions

- Realtime or Batch
- Cloud vs. Edge/Browser
- Compute resources (CPU/GPU/memory)
- Latency, throughput (QPS)
- Logging
- Security and privacy

500ms, 1000 QPS

First deployment vs. maintenance



DeepLearning.AI

Deployment

Deployment patterns

Common deployment cases

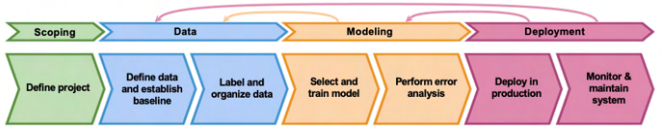
1. New product/capability
2. Automate/assist with manual task
3. Replace previous ML system

Key ideas:

- Gradual ramp up with monitoring
- Rollback

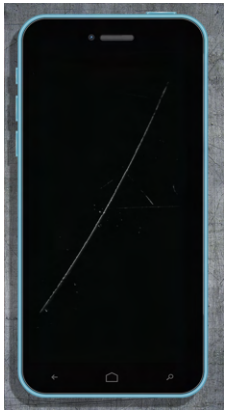
Visual inspection example

shadow mode



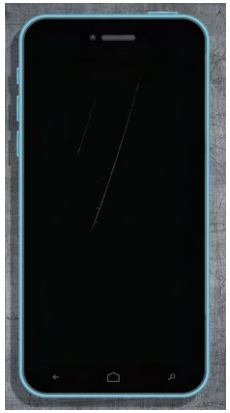
Human ✓

ML ✓



Human ✗

ML ✗



Human ✗

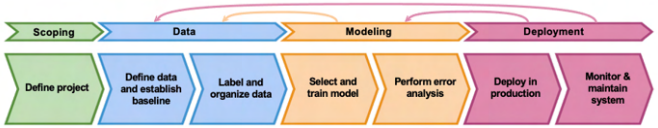
ML ✓

ML system shadows the human and runs in parallel.

ML system's output not used for any decisions during this phase.

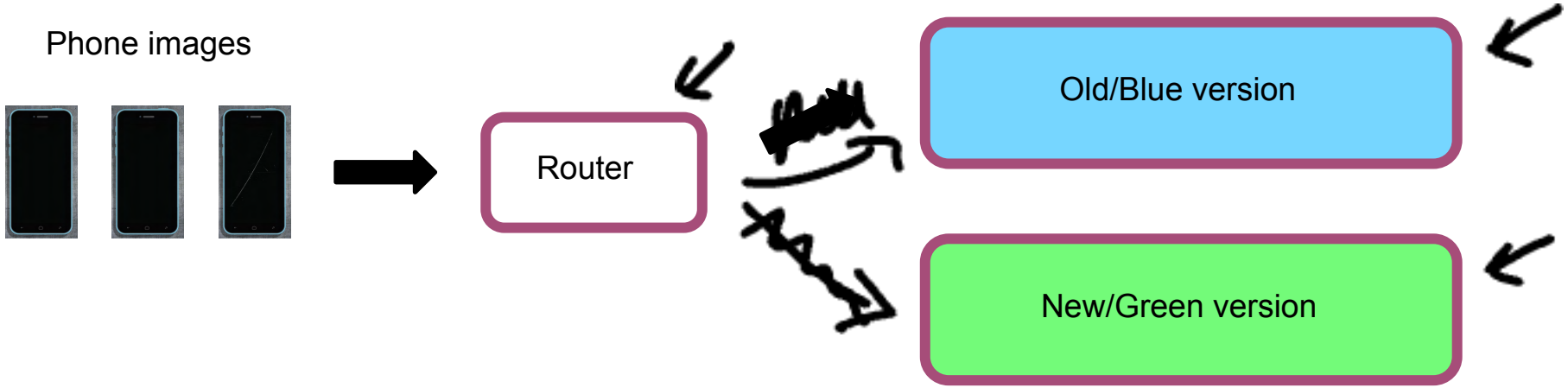
Sample outputs and verify predictions of ML system.

Canary deployment



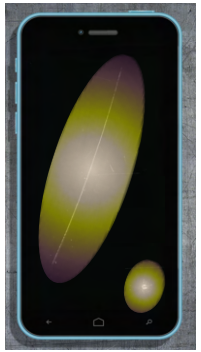
- Roll out to small fraction (say 5%) of traffic initially.
- Monitor system and ramp up traffic gradually.

Blue green deployment



Easy way to enable rollback

Degrees of automation



Human

AI

Human
only

Shadow
mode

AI assistance

Partial
automation

Full
automation

human in the loop

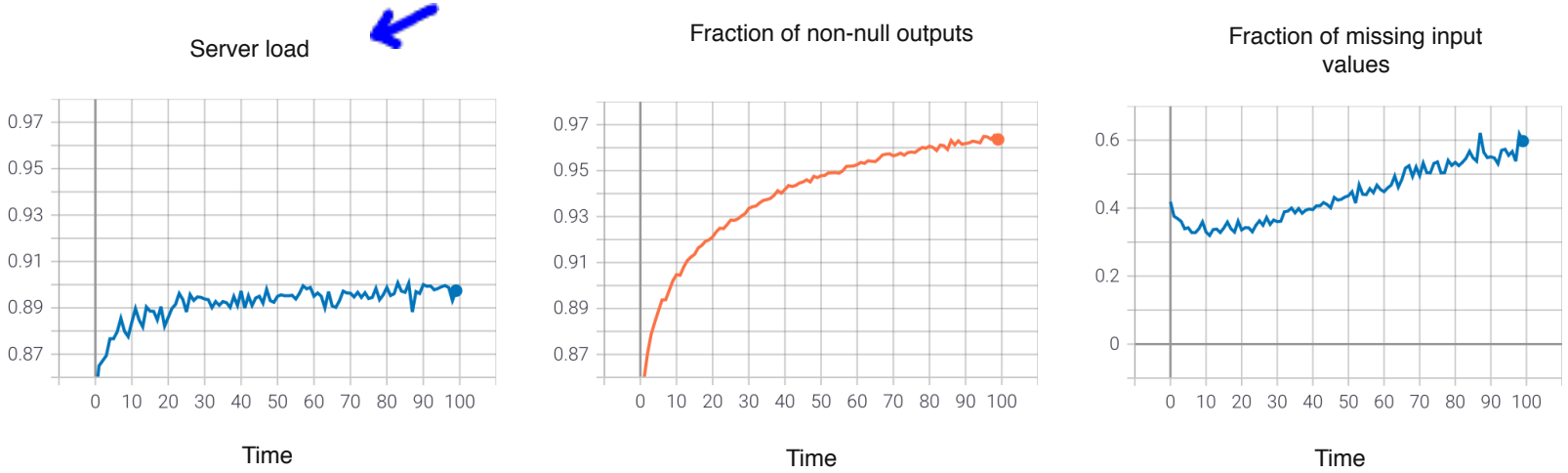
You can choose to stop before getting to full automation.

DeepLearning.AI

Deployment

Monitoring

Monitoring dashboard

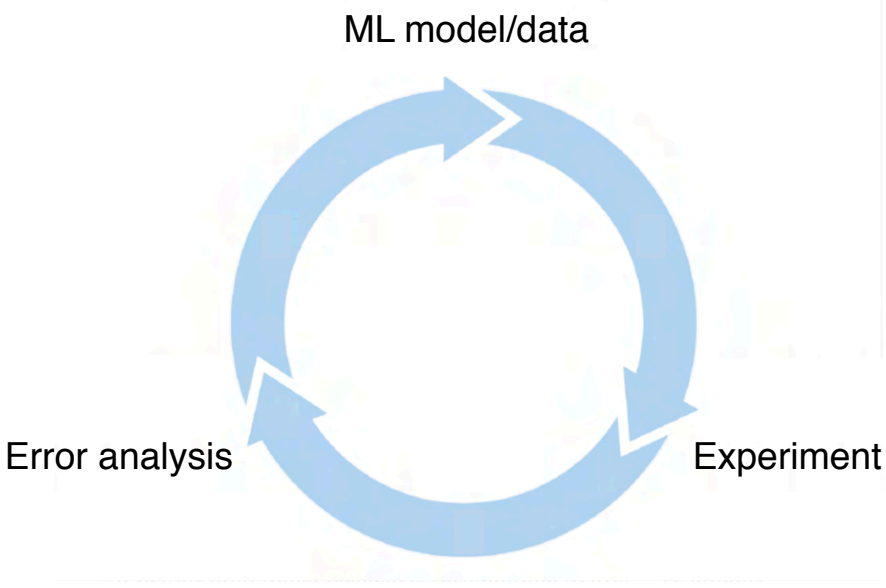


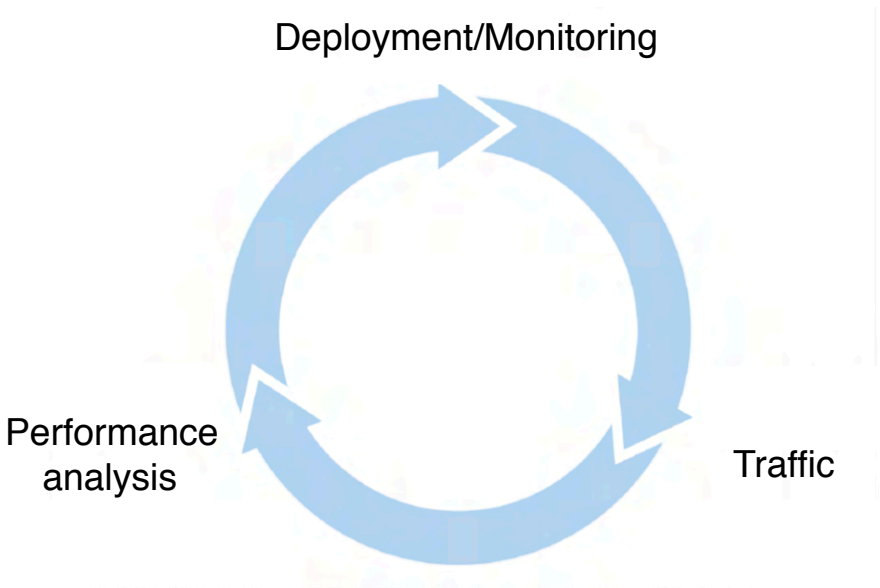
- Brainstorm the things that could go wrong.
- Brainstorm a few statistics/metrics that will detect the problem.
- It is ok to use many metrics initially and gradually remove the ones you find not useful.

Examples of metrics to track

Software metrics:	Memory, compute, latency, throughput, server load
Input metrics: 	Avg input length Avg input volume Num missing values Avg image brightness
Output metrics: 	# times return " " (null) # times user redoes search # times user switches to typing CTR

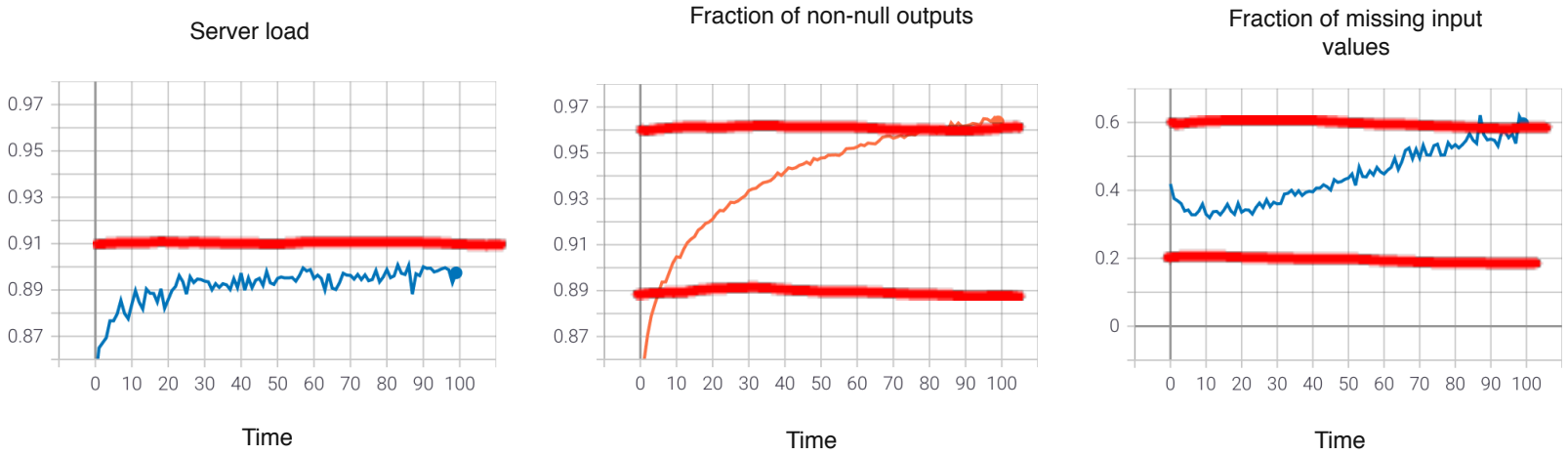
Just as ML modeling is iterative, so is deployment





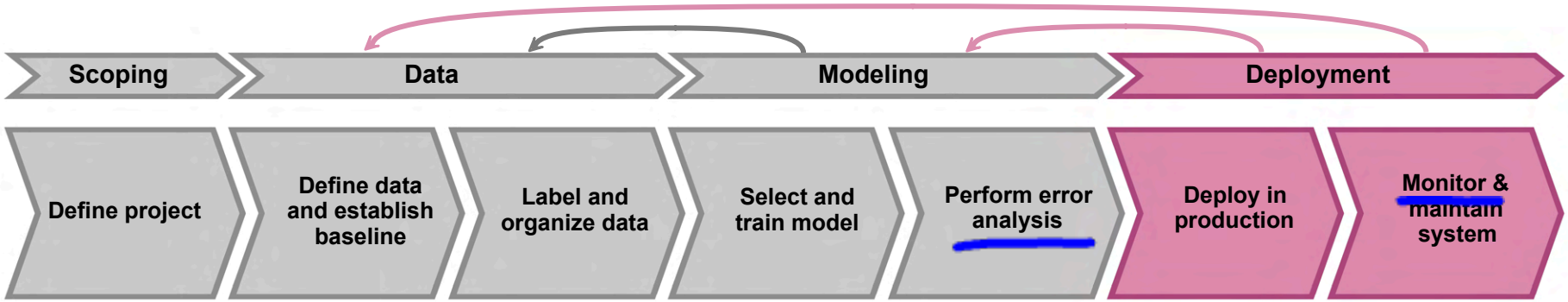
Iterative process to choose the right set of metrics to monitor.

Monitoring dashboard



- Set thresholds for alarms
- Adapt metrics and thresholds over time

Model maintenance



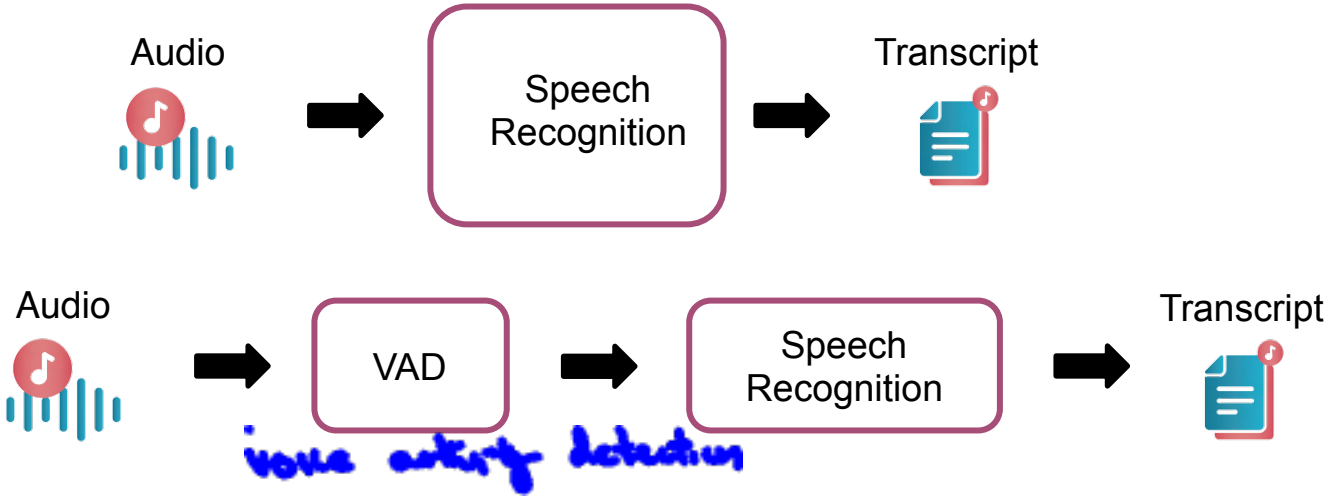
- Manual retraining
- Automatic retraining

DeepLearning.AI

Deployment

Pipeline monitoring

Speech recognition example



Some cellphones might have VAD clip audio differently, leading to degraded performance

User profile example

User Data

(e.g., clickstream)

User Profile

(e.g., own car?)

y
n
unknown

Recommender system

Product recommendations

Metrics to monitor

Monitor

- Software metrics
- Input metrics
- Output metrics

How quickly do they change?

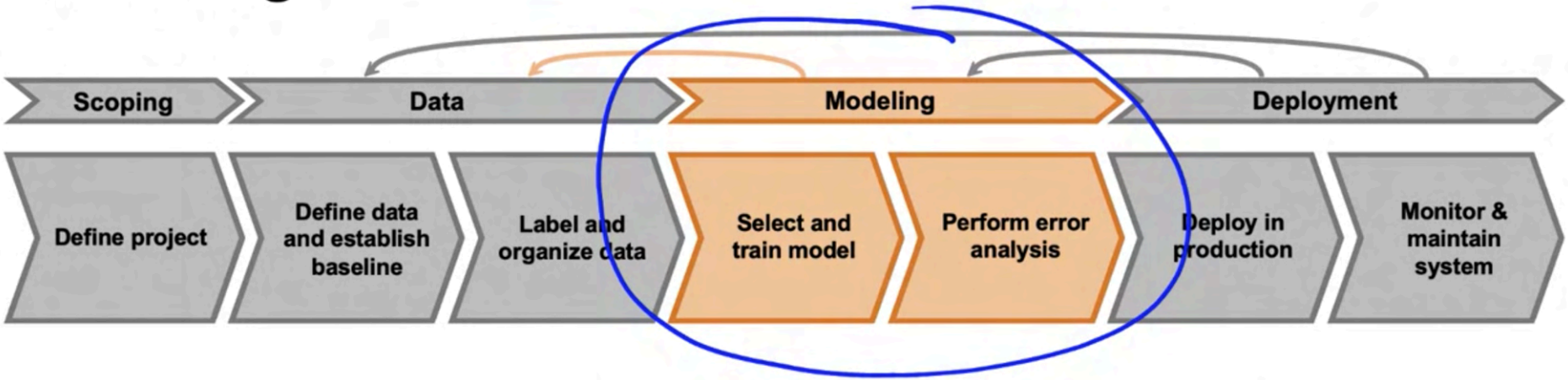
- User data generally has slower drift.
- Enterprise data (B2B applications) can shift fast.

DeepLearning.AI

Select and train model

Modeling overview

Modeling



Model-centric AI development

Data-centric AI development

DeepLearning.AI

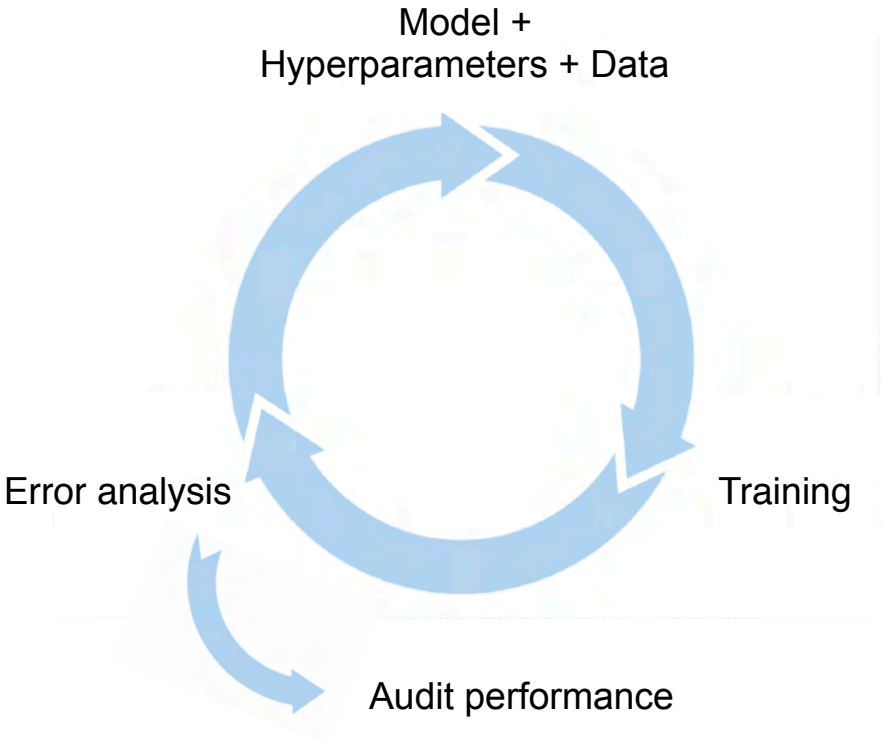
Select and train model

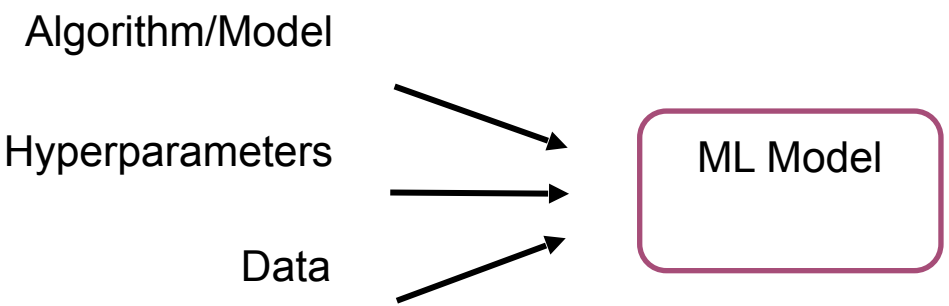
Key challenges

AI system = Code + Data

(algorithm/model)

Model development is an iterative process

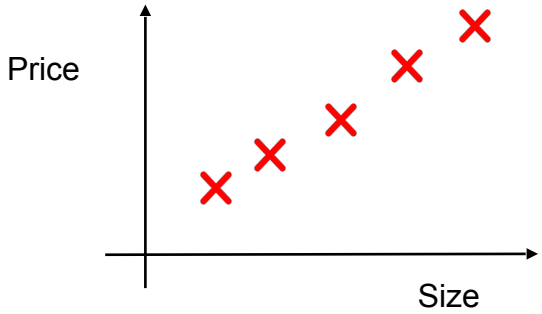




Challenges in model development

1. Doing well on training set (usually measured by average training error).

2. Doing well on dev/test sets.



3. Doing well on business metrics/project goals.

DeepLearning.AI

Select and train model

Why low average test error isn’t good enough

Performance on disproportionately important examples

Web Search example

"Apple pie recipe"

"Latest movies"

"Wireless data plan"

"Diwali festival"

Informational and Transactional queries

"Stanford"

"Reddit"

"Youtube"

Navigational queries

Performance on key slices of the dataset

Example: ML for loan approval

Make sure not to discriminate by ethnicity, gender, location, language or other protected attributes.

Example: Product recommendations from retailers

Be careful to treat fairly all major user, retailer, and product categories.

Rare classes

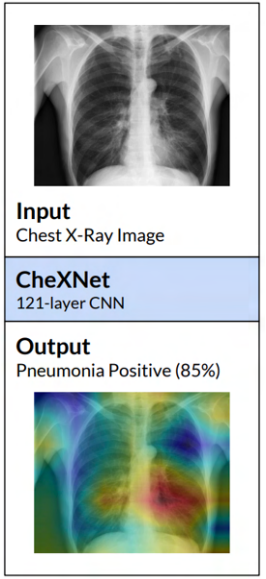
Skewed data distribution

99% negative 1% positive

print("0")

Accuracy in rare classes

Condition	Performance
Effusion (10,000)	0.901
Edema	0.924
Mass	0.909
Hernia (~100)	0.851



Unfortunate conversation in many companies

MLE: "I did well on the test set!"

Product Owner: "But this doesn't work for my application"

MLE: "But... I did well on the test set!"

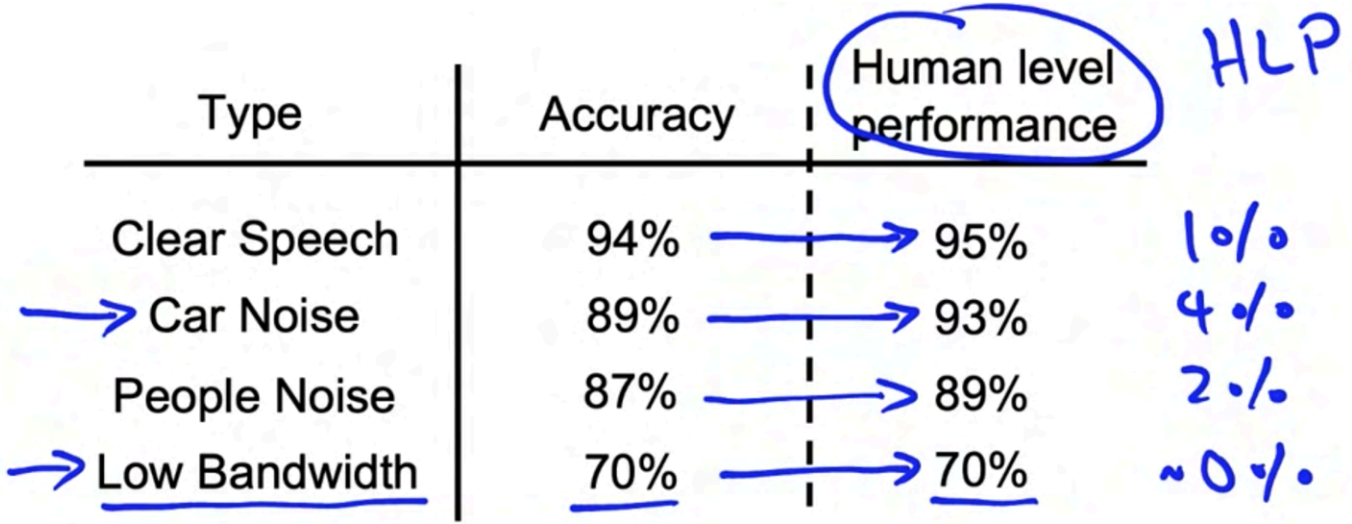
DeepLearning.AI

Select and train model

Establish a baseline

Establishing a baseline level of performance

Speech recognition example:



Structured and unstructured data

Unstructured data

Image

Audio

Text

This restaurant was great!

Structured Data

User Id	Purchase	Number	Price
3421	Blue shirt	5	$20
612	Brown shoes	1	$35

Price	Product
3421	Red skirt

Ways to establish a baseline

- Human level performance (HLP)
- Literature search for state-of-the-art/open source
- Older system

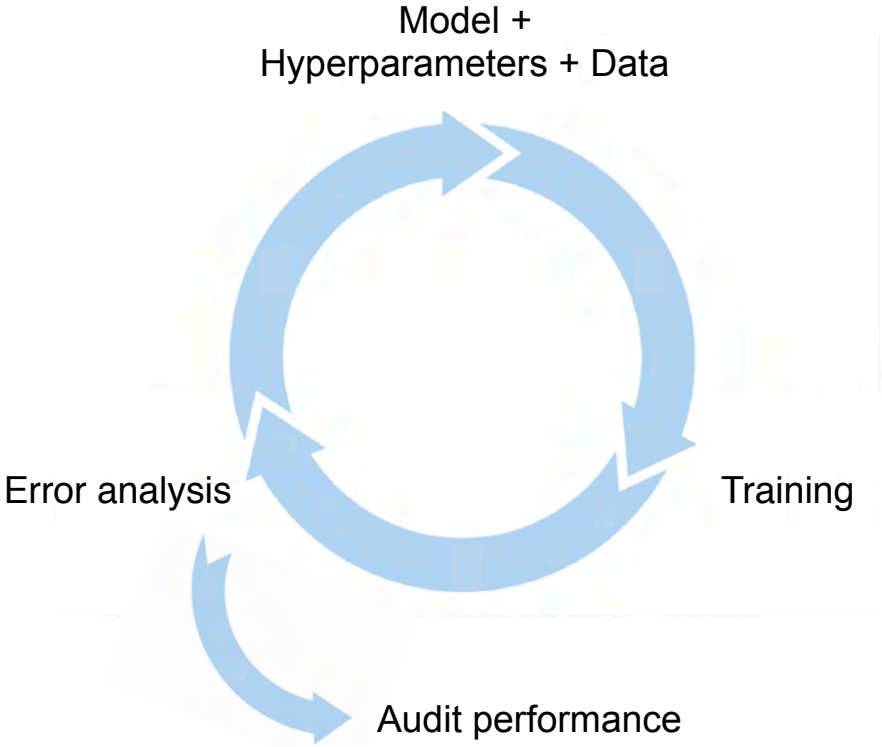
Baseline gives an estimate of the irreducible error / Bayes error and indicates what might be possible.

DeepLearning.AI

Select and train model

Tips for getting started

ML is an iterative process



Getting started on modeling

- Literature search to see what's possible.
- Find open-source implementations if available.
- A reasonable algorithm with good data will often outperform a great algorithm with not so good data.

Deployment constraints when picking a model

Should you take into account deployment constraints when picking a model?

Yes, if baseline is already established and goal is to build and deploy.

No, if purpose is to establish a baseline and determine what is possible and might be worth pursuing.

Sanity-check for code and algorithm

- Try to overfit a small training dataset before training on a large one.
- Example #1: Speech recognition

audio transcript

X → Y

␣ ␣ ␣ ␣ ␣ ␣

- Example #2: Image segmentation

- Example #3: Image classification

DeepLearning.AI

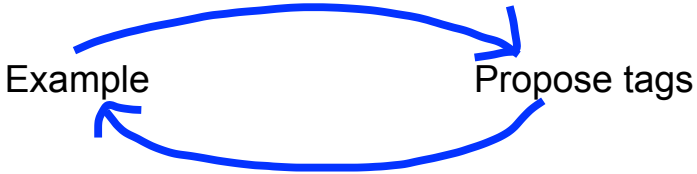
Error analysis and performance auditing

Error analysis example

Speech recognition example

Example	Label	Prediction	Car Noise	People Noise	Low Bandwidth
1	"Stir fried lettuce recipe"	"Stir fry lettuce recipe"	✓		
2	"Sweetened coffee"	"Swedish coffee"		✓	✓
3	"Sail away song"	"Sell away some"		✓	
4	"Let's catch up"	"Let's ketchup"	✓	✓	✓

Iterative process of error analysis



Visual inspection:

- Specific class labels (scratch, dent, etc.)
- Image properties (blurry, dark background, light background, reflection….)
- Other meta-data: phone model, factory

Product recommendations:

- User demographics
- Product features

Useful metrics for each tag

- What fraction of errors has that tag?
- Of all data with that tag, what fraction is misclassified?
- What fraction of all the data has that tag?
- How much room of improvement is there in that tag?

DeepLearning.AI

Error analysis and performance auditing

Prioritizing what to work on

Prioritizing what to work on

Type	Accuracy	Human level performance	Gap to HLP	% of data
Clean Speech	94%	95%	1%	60% → 0.6%
Car Noise	89%	93%	4%	4% → 0.16%
People Noise	87%	89%	2%	30% → 0.6%
Low Bandwidth	70%	70%	0%	6% → ~0%

Prioritizing what to work on

Decide on most important categories to work on based on:

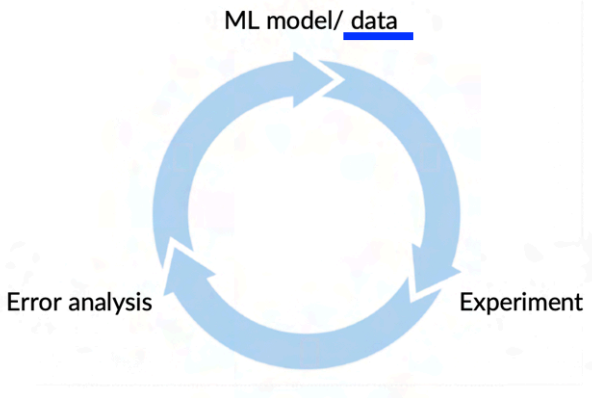
- How much room for improvement there is.
- How frequently that category appears.
- How easy is to improve accuracy in that category.
- How important it is to improve in that category.

Adding data

For categories you want to prioritize:

- Collect more data (or improve label accuracy)
- Use data augmentation to get more data

Type	Accuracy	Human level performance	Gap to HLP	% of data
Clean Speech	94%	95%	1%	60%
Car Noise	84%	93%	4%	40%
People Noise	87%	84%	2%	30%
Low Bandwidth	70%	70%	0%	6%



DeepLearning.AI

Error analysis and performance auditing

Skewed datasets

Examples of skewed datasets

Manufacturing example

99.7% no defect $y=0$

0.3% defect $y=1$

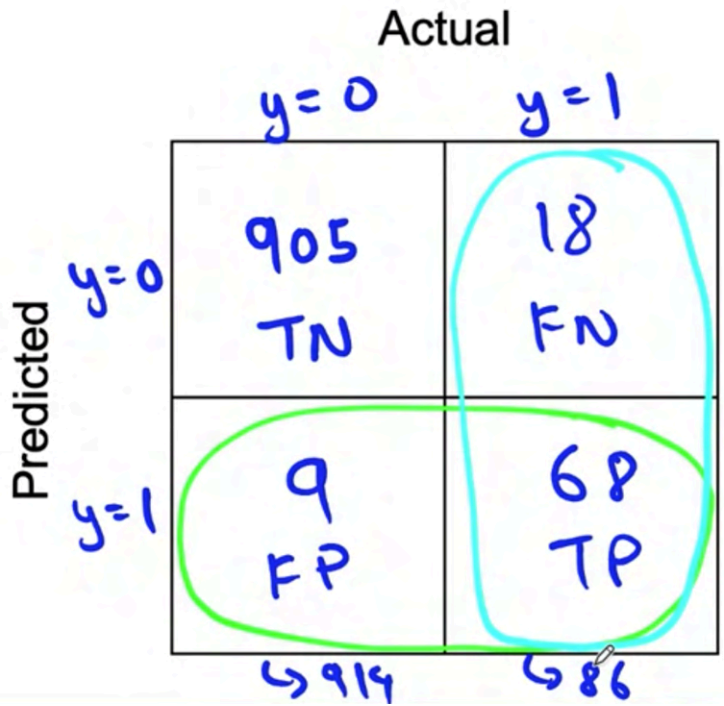
print("0")

99.7%

Medical Diagnosis example: 98% of patients don't have a disease

Speech Recognition example: In wake word detection, 96.7% of the time wake word doesn't occur

Confusion matrix: precision and recall



TN: True Negative

TP: True Positive

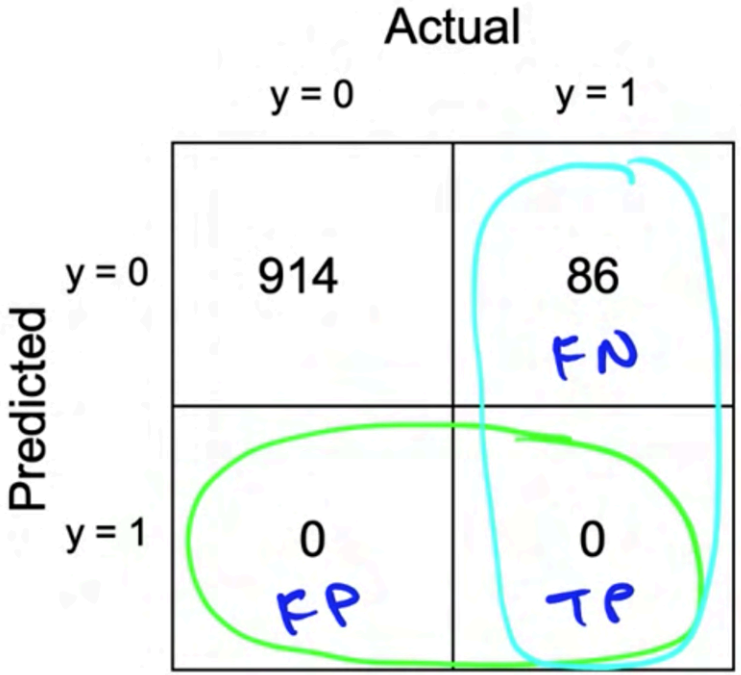
FN: False Negative

FP: False Positive

$$\text{Precision} = \frac{TP}{TP+FP} = \frac{68}{68+9} = 88.3\%$$

$$\text{Recall} = \frac{TP}{TP+FN} = \frac{68}{68+18} = 79.1\%$$

What happens with print("0")?



$$\text{Precision} = \frac{TP}{TP + FP} = \frac{0}{0+0}$$

$$\text{Recall} = \frac{TP}{TP + FN} = \frac{0}{0+86} = 0\%$$

Combining precision and recall – F_1 score

	Precision (P)	Recall (R)	F_1
Model 1	88.3	79.1	83.4 %
Model 2	97.0	7.3	13.6 %

$$F_1 = \frac{2}{\frac{1}{P} + \frac{1}{R}}$$

Multi-class metrics

Classes: Scratch, Dent, Pit mark, Discoloration

Defect Type	Precision	Recall	F_1
Scratch	82.1%	99.2%	89.8%
Dent	92.1%	99.5%	95.7%
Pit mark	85.3%	98.7%	91.5%
Discoloration	72.1%	97%	82.7%

DeepLearning.AI

Error analysis and performance auditing

Performance auditing

Auditing framework

Check for accuracy, fairness and bias.

1. Brainstorm the ways the system might go wrong.
    - Performance on subsets of data (e.g., ethnicity, gender).
    - Prevalence of specific errors/outputs (e.g., FP, FN).
    - Performance on rare classes.

2. Establish metrics to assess performance against these issues on appropriate slices of data.

3. Get business/product owner buy-in.

Speech recognition example

1. Brainstorm the ways the system might go wrong.
    - Accuracy on different genders and ethnicities.
    - Accuracy on different devices.
    - Prevalence of rude mistranscriptions.

2. Establish metrics to assess performance against these issues on appropriate slices of data.
    - Mean accuracy for different genders and major accents.
    - Mean accuracy on different devices.
    - Check for prevalence of offensive words in the output.

DeepLearning.AI

Data iteration

Data-centric AI development

Data-centric AI development

Model-centric view

Collect what data you can, and develop a model good enough to deal with the noise in the data.

Hold the data fixed and iteratively improve the code/ model.

Data-centric view

The consistency of the data is paramount. Use tools to improve the data quality; this will allow multiple models to do well.

Hold the code fixed and iteratively improve the data.

DeepLearning.AI

Data iteration

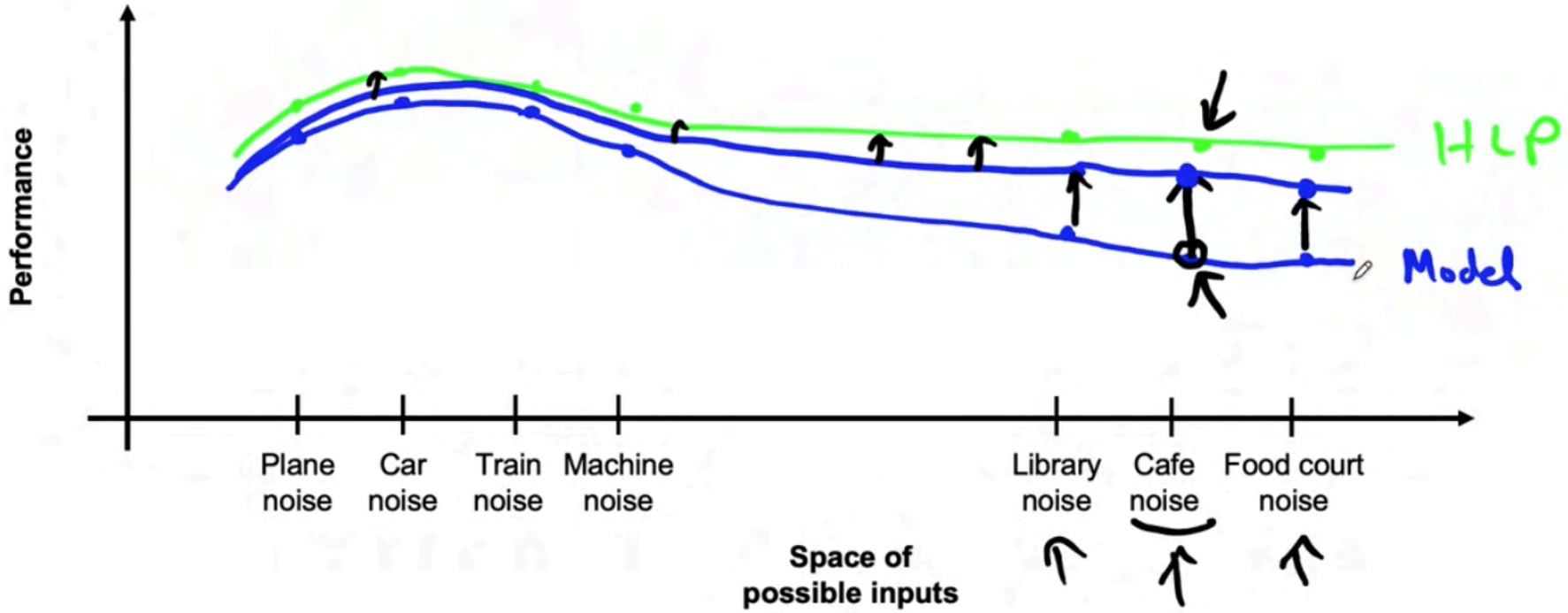
A useful picture of data augmentation

Speech recognition example

Different types of speech input:

- Car noise
- Plane noise
- Train noise
- Machine noise
- Cafe noise
- Library noise
- Food court noise

Speech recognition example



DeepLearning.AI

Data iteration

Data augmentation

Data augmentation

Goal:

Create realistic examples that (i) the algorithm does poorly on, but (ii) humans (or other baseline) do well on

Checklist:

- Does it sound realistic?
- Is the X → Y mapping clear? (e.g., can humans recognize speech?)
- Is the algorithm currently doing poorly on it?

The rubber sheet analogy

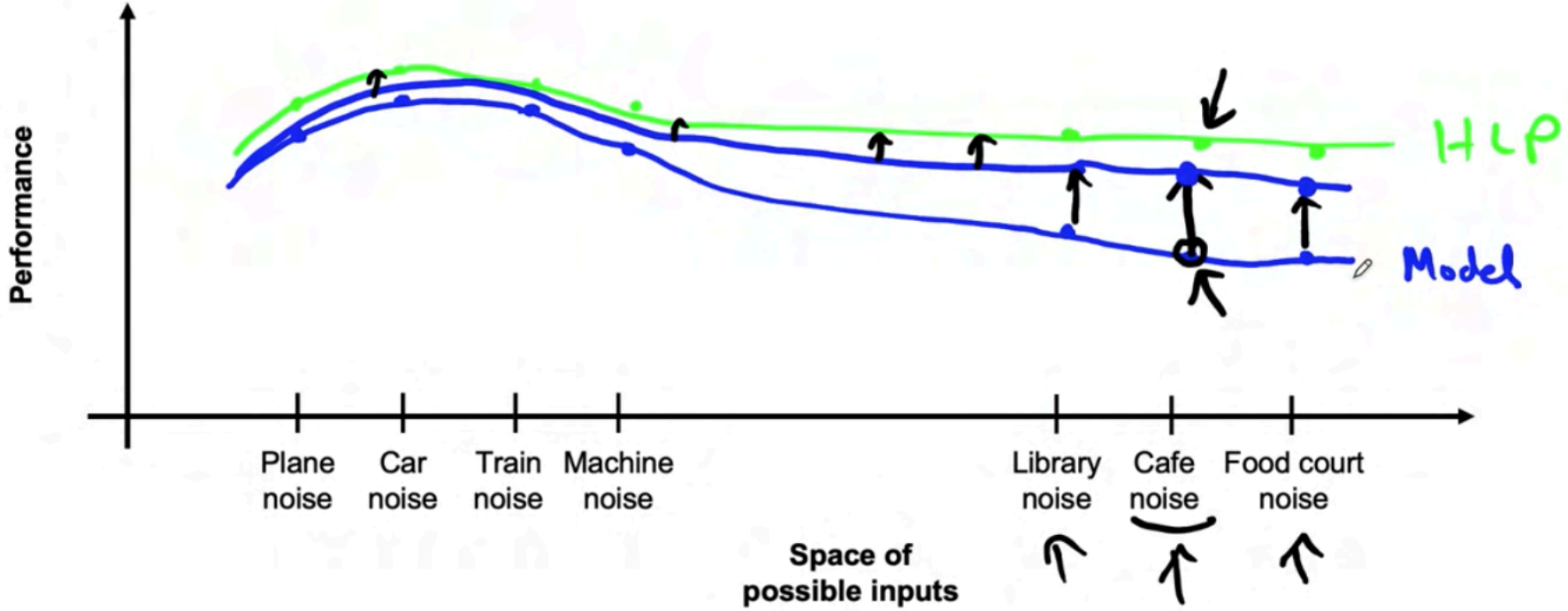
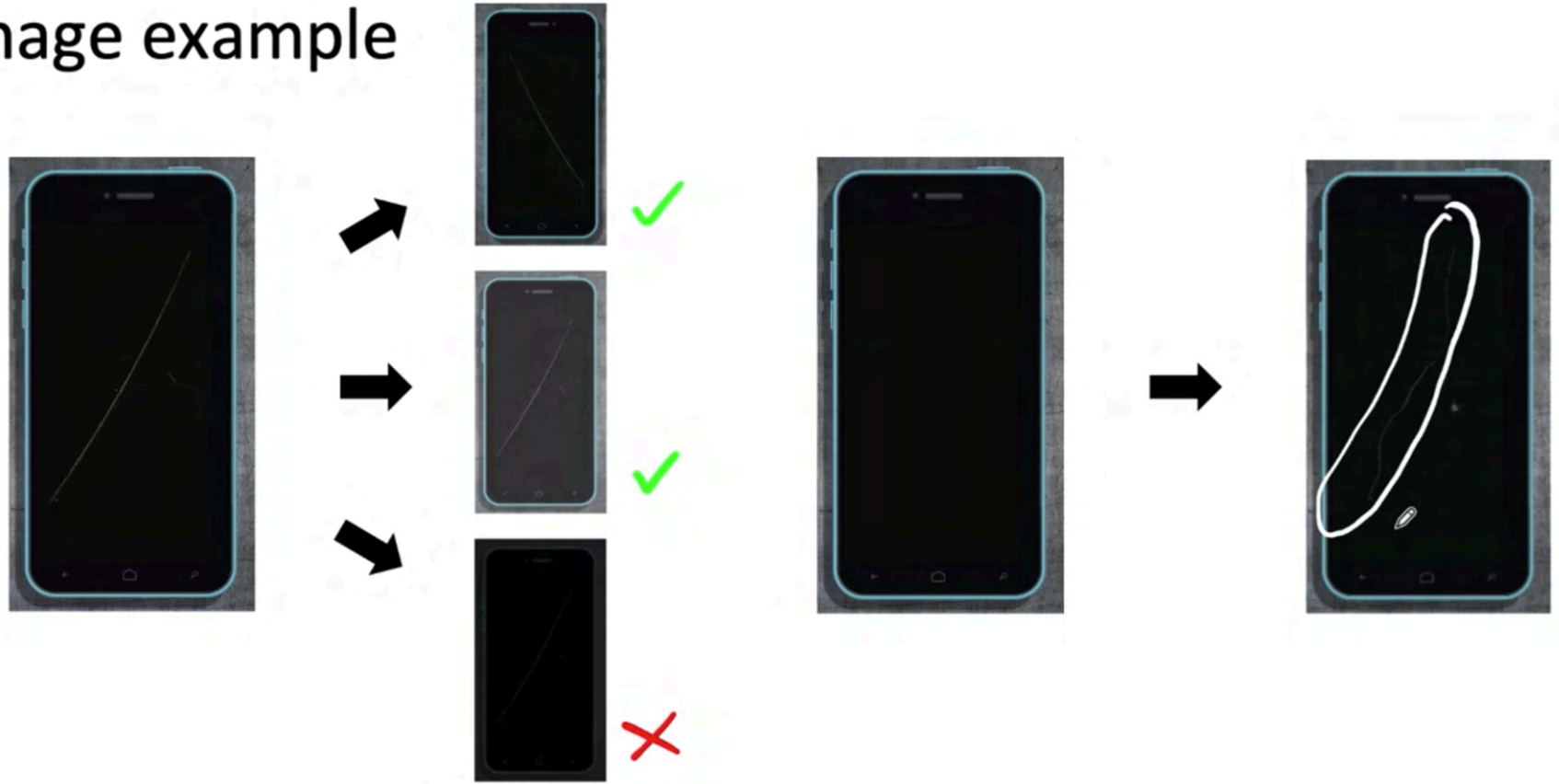
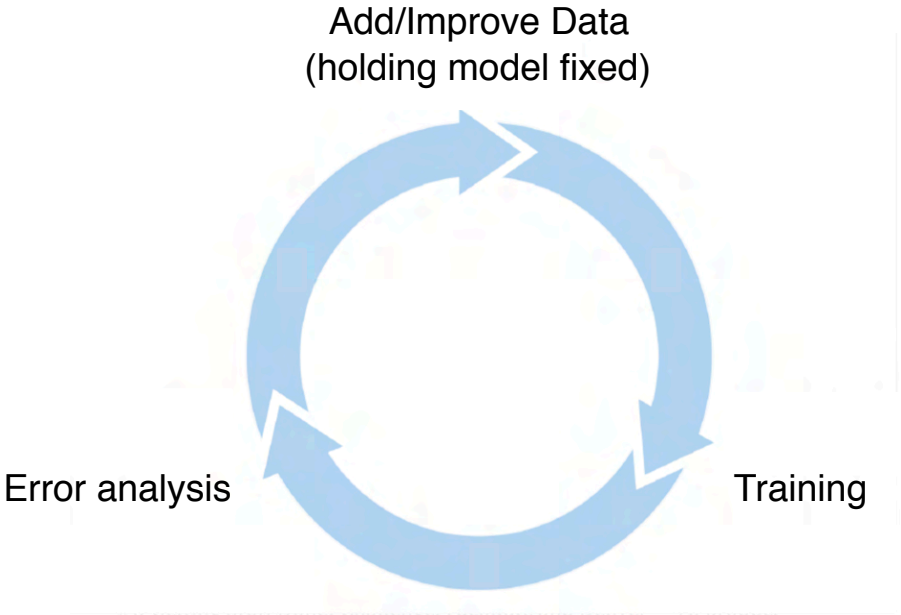


Image example



Data iteration loop



DeepLearning.AI

Data iteration

Can adding
data hurt?

Can adding data hurt performance?

For unstructured data problems, if:

- The model is large (low bias).
- The mapping X → Y is clear (e.g., humans can make accurate predictions).
  Then, **adding data rarely hurts accuracy.**

Photo OCR counterexample

1

high accuracy

I

42I

low accuracy

1? I?

Adding a lot of new "I"s may skew the dataset and hurt performance

DeepLearning.AI

Data iteration

Adding features

Structured data

Restaurant recommendation example

Vegetarians are frequently recommended restaurants with only meat options.

Possible features to add?

- Is person vegetarian (based on past orders)?
- Does restaurant have vegetarian options (based on menu)?

Other food delivery examples

- Only tea/coffee
- Only pizza

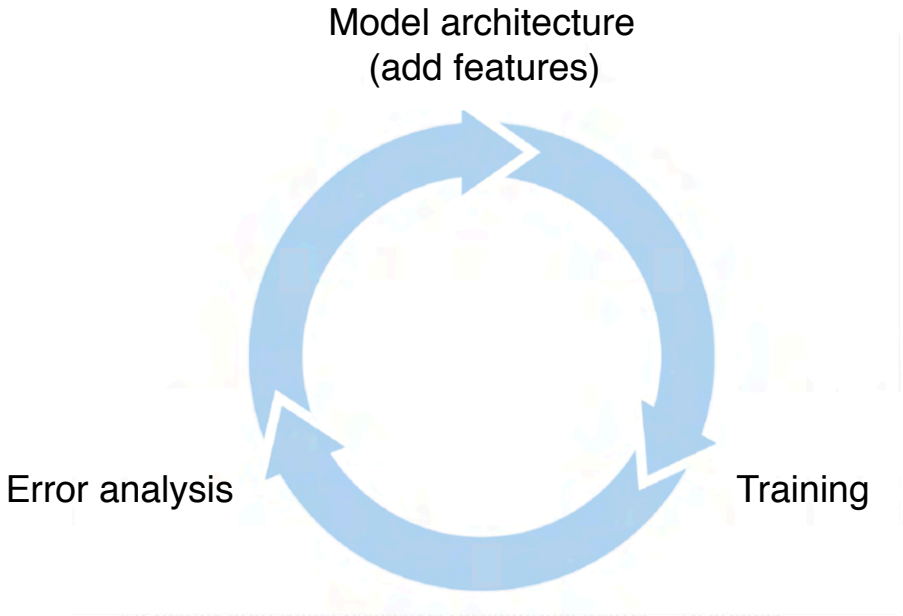
What are the added signals (features) that can help make a decision?

Product recommendation:

Collaborative filtering

Context based filtering

Data iteration



- Error analysis can be harder if there is no good baseline (such as HLP) to compare to.
- Error analysis, user feedback and benchmarking to competitors can all provide inspiration for features to add.

DeepLearning.AI

Data iteration

Experiment tracking

Experiment tracking

What to track?

- Algorithm/code versioning
- Dataset used
- Hyperparameters
- Results

Tracking tools

- Text files
- Spreadsheet
- Experiment tracking system

Desirable features

- Data needed to replicate results
- In-depth analysis of experiment results
- Perhaps also: Resource monitoring, visualization, model error analysis

DeepLearning.AI

Data iteration

From big data to good data

From Big Data to Good Data

Try to ensure consistently high-quality data in all phases of the ML project lifecycle.

Good data is:

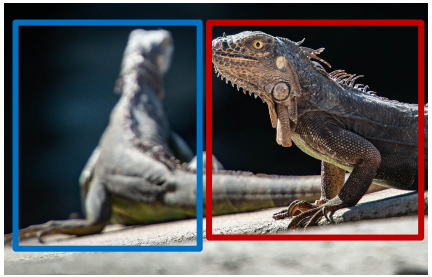
- Cover of important cases (good coverage of inputs x)
- Defined consistently (definition of labels y is unambiguous)
- Has timely feedback from production data (distribution covers data drift and concept drift)
- Sized appropriately

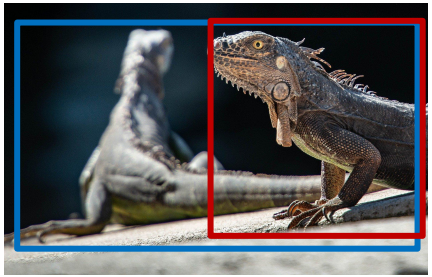
DeepLearning.AI

Define data and establish baseline

Why is data definition hard?

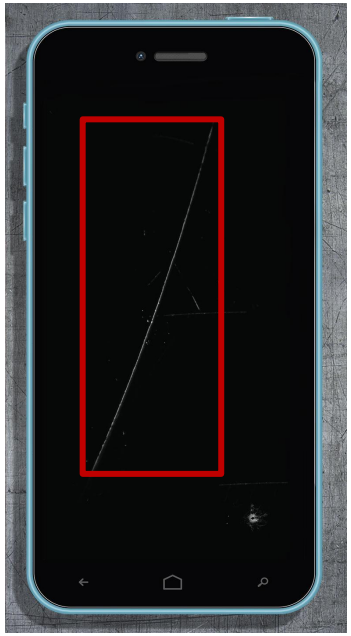
Iguana detection example

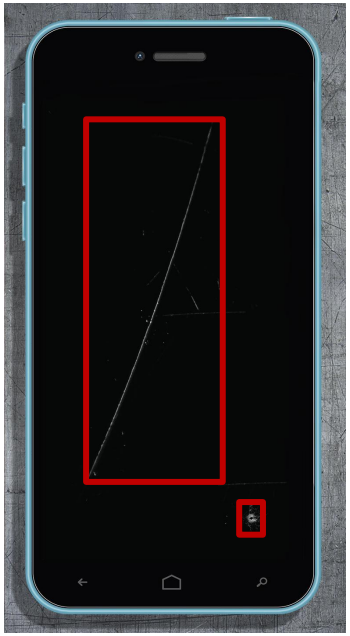




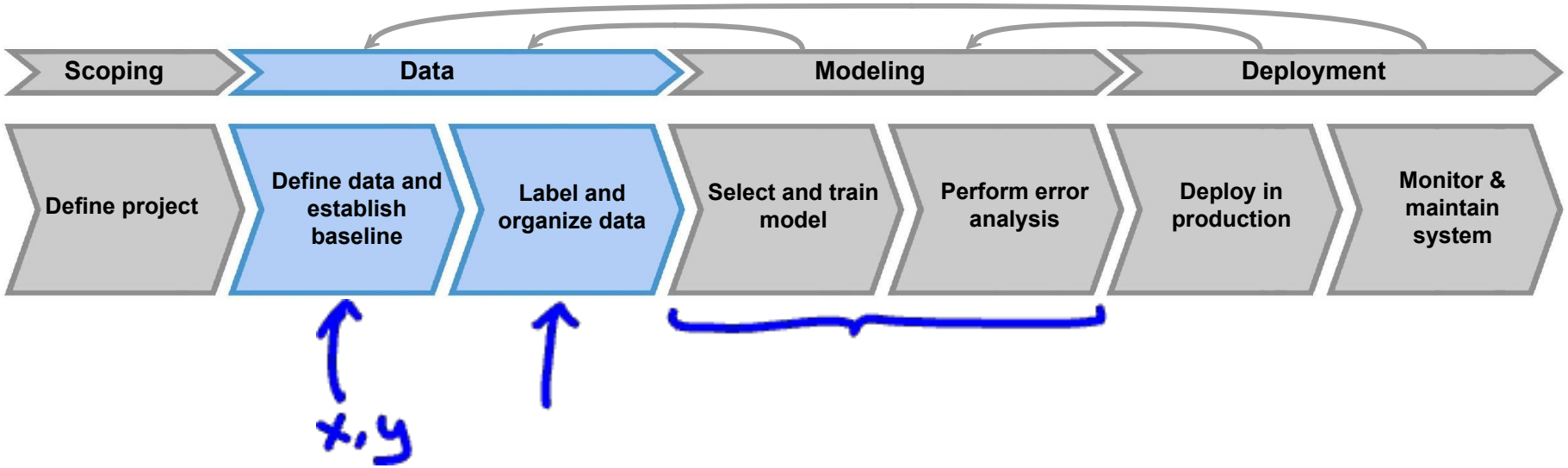
Labeling instructions: "Use bounding boxes to indicate the position of iguanas"

Phone defect detection





Data stage


Scoping
Data
Modeling
Deployment
Define project
Define data and establish baseline
Label and organize data
Select and train model
Perform error analysis
Deploy in production
Monitor & maintain system
x, y

DeepLearning.AI

Define data and establish baseline

More label ambiguity examples

Speech recognition example

"Um, nearest gas station"

"Umm, nearest gas station"

”Nearest gas station [unintelligible]"

User ID merge example

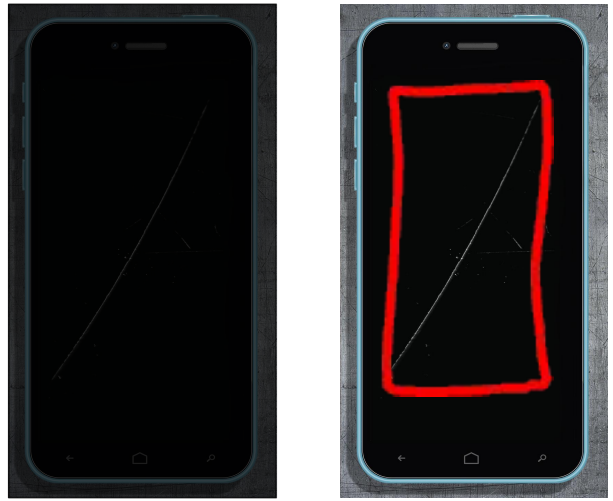
	Job Board (website)
Email	nova@deeplearning.ai
First Name	Nova
Last Name	Ng
Address	1234 Jane Way
State	CA
Zip	94304

- is it a bot/spam account?
- fraudulent transaction?
- looking for job?

1 if same

0 if different

Data definition questions



- What is the input x?
  - Lightning? Contrast? Resolution?
  - What features need to be included?

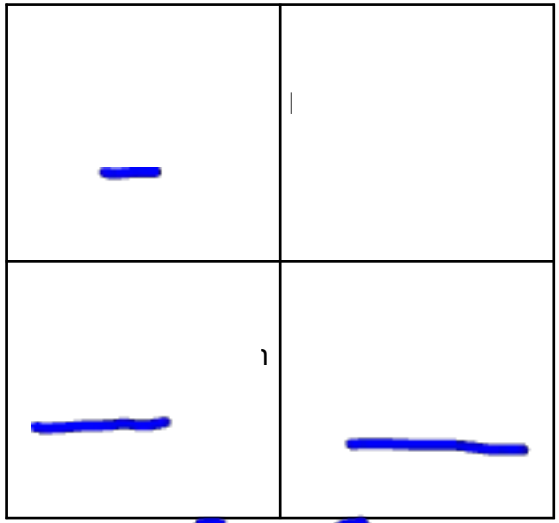
- What is the target label y?
  - How can we ensure labelers give consistent labels?

DeepLearning.AI

Define data and establish baseline

Major types of data problems

Major types of data problems



≤ 10,000

Clean labels are critical.

> 10,000

Emphasis on data process.

Humans can label data.

Data augmentation.

Harder to obtain more data.

Unstructured vs. structured data

Unstructured data

- May or may not have huge collection of unlabeled examples x.
- Humans can label more data.
- Data augmentation more likely to be helpful.

Structured data

- May be more difficult to obtain more data.
- Human labeling may not be possible (with some exceptions).

Small data vs. big data

Small data

- Clean labels are critical.
- Can manually look through dataset and fix labels.
- Can get all the labelers to talk to each other.

Big data

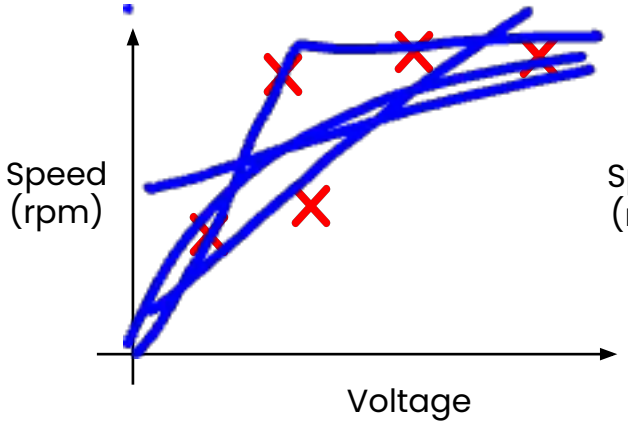
- Emphasis data process.

DeepLearning.AI

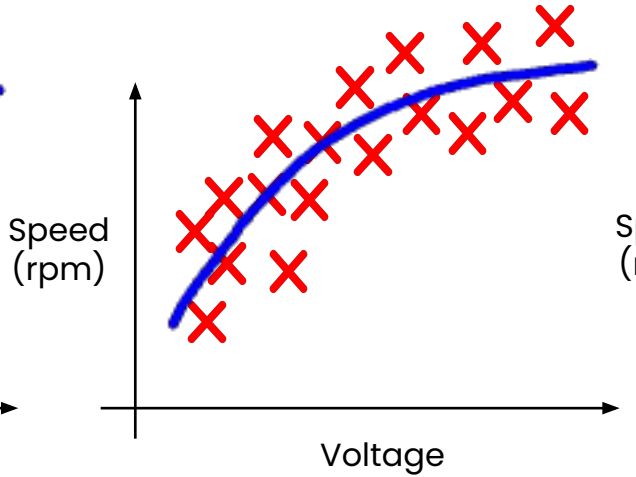
Define data and establish baseline

Small data and label consistency

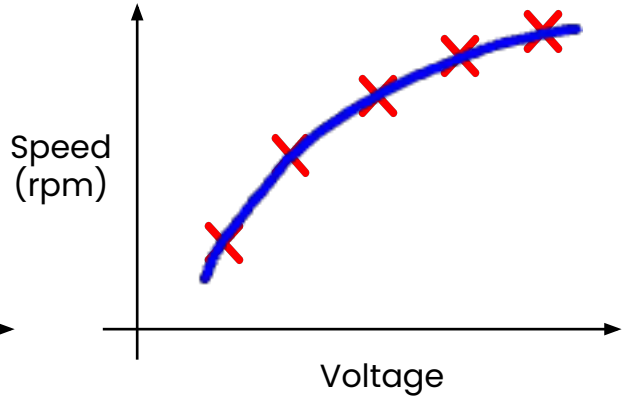
Why label consistency is important



- Small data
- Noisy labels

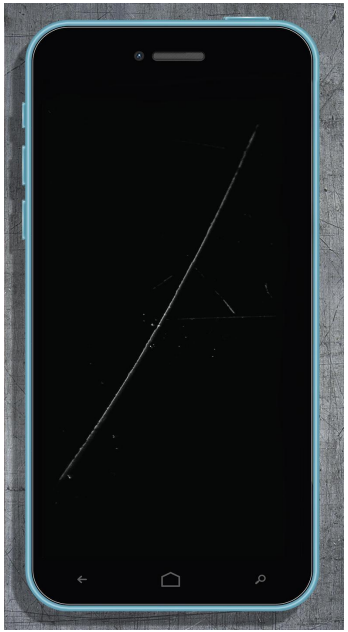


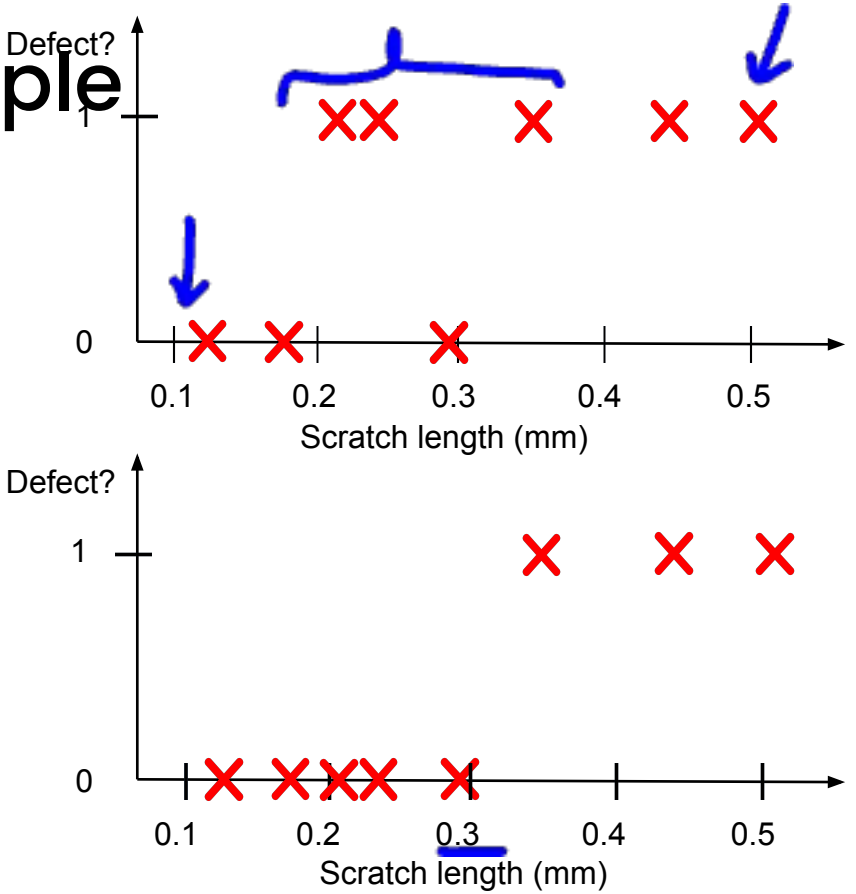
- Big data
- Noisy labels



- Small data
- Clean (consistent) labels

Phone defect example





Big data problems can have small data challenges too

Problems with a large dataset but where there's a long tail of rare events in the input will have small data challenges too.

- Web search
- Self-driving cars
- Product recommendation systems

DeepLearning.AI

Define data and establish baseline

Improving label consistency

Improving label consistency

- Have multiple labelers label same example.
- When there is disagreement, have MLE, subject matter expert (SME) and/or labelers discuss definition of y to reach agreement.
- If labelers believe that x doesn't contain enough information, consider changing x.
- Iterate until it is hard to significantly increase agreement.

Examples

- Standardize labels

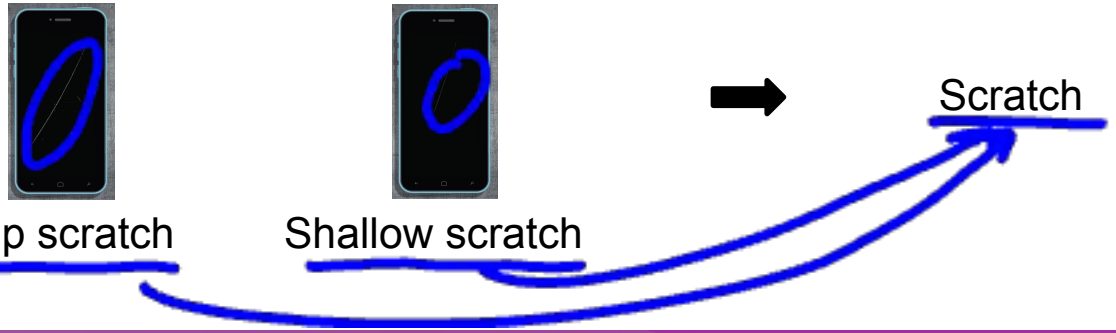
  "Um, nearest gas station"

  "Umm, nearest gas station"

  ”Nearest gas station [unintelligible]”

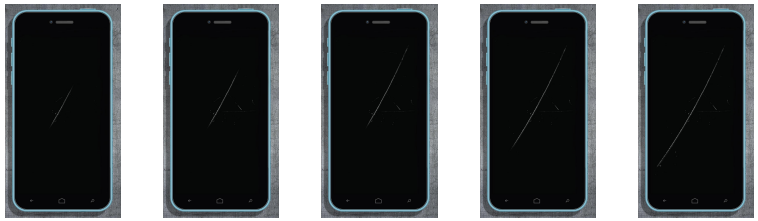
  ➡ "Um, nearest gas station"

- Merge classes



Have a class/label to capture uncertainty

- Defect: 0 or 1



Alternative: 0, Borderline, 1

- Unintelligible audio 

“nearest go”

“nearest grocery”

“nearest [unintelligible]”

Small data vs. big data (unstructured data)

Small data

- Usually small number of labelers.
- Can ask labelers to discuss specific labels.

Big data

- Get to consistent definition with a small group.
- Then send labeling instructions to labelers.
- Can consider having multiple labelers label every example and using voting or consensus labels to increase accuracy.

DeepLearning.AI

Define data and establish baseline

Human level performance (HLP)

Why measure HLP?

- Estimate Bayes error / irreducible error to help with error analysis and prioritization.

99%

Ground Truth Label	
1	✓
1	✗
1	✓
0	✓
0	✓
0	✗

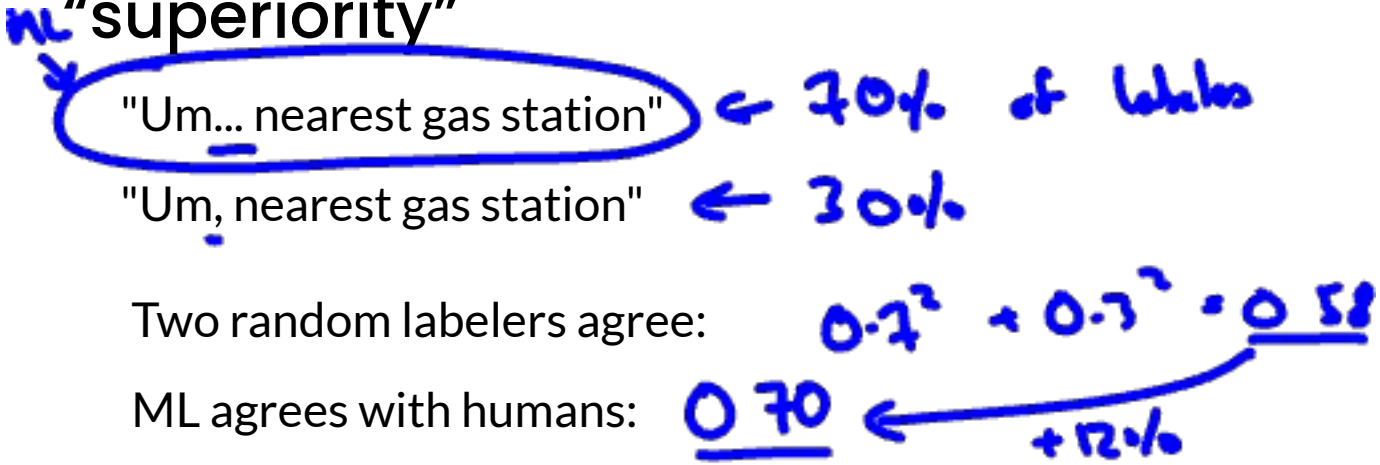
Human?

66.7% accuracy

Other uses of HLP

- In academia, establish and beat a respectable benchmark to support publication.
- Business or product owner asks for 99% accuracy. HLP helps establish a more reasonable target.
- “Prove” the ML system is superior to humans doing the job and thus the business or product owner should adopt it.

The problem with beating HLP as a “proof” of ML “superiority”



The 12% better performance is not important for anything! This can also mask more significant errors ML may be making.

DeepLearning.AI

Define data and establish baseline

Raising HLP

Raising HLP

When the ground truth label is externally defined, HLP gives an estimate for Bayes error / irreducible error.

But often ground truth is just another human label.

Ground Truth Label	Inspector
1	1
0	0
1	1
0	0
0	0
0	1

Raising HLP

- When the label *y* comes from a human label, HLP << 100% may indicate ambiguous labeling instructions.

Um, Um...

- Improving label consistency will raise HLP.

- This makes it harder for ML to beat HLP. But the more consistent labels will raise ML performance, which is ultimately likely to benefit the actual application performance.

HLP on structured data

Structured data problems are less likely to involve human labelers, thus HLP is less frequently used.

Some exceptions:

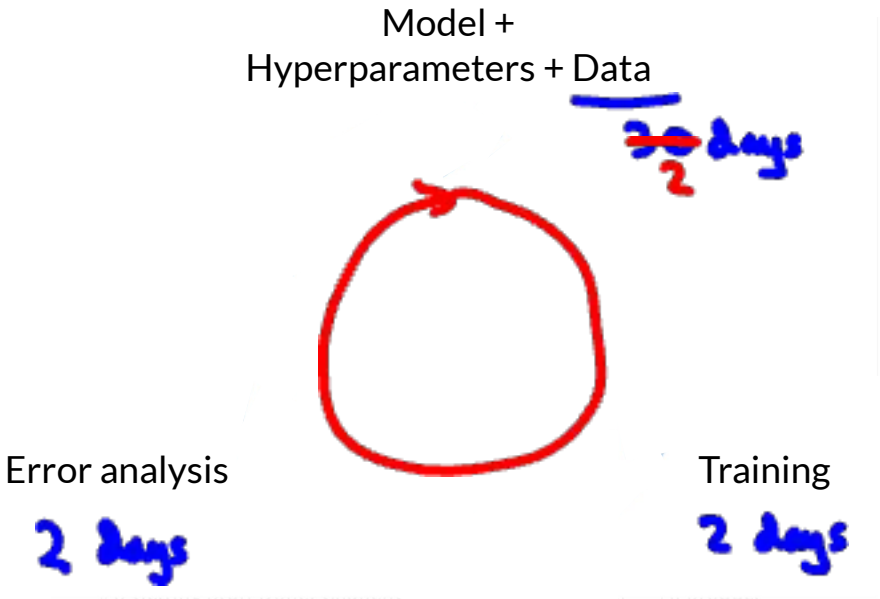
- User ID merging: Same person?
- Based on network traffic, is the computer hacked?
- Is the transaction fraudulent?
- Spam account? Bot?
- From GPS, what is the mode of transportation – on foot, bike, car, bus?

DeepLearning.AI

Label and organize data

Obtaining data

How long should you spend obtaining data?



- Get into this iteration loop as quickly possible.
- Instead of asking: How long it would take to obtain m examples?
  Ask: How much data can we obtain in k days.
- Exception: If you have worked on the problem before and from experience you know you need m examples.

Inventory data

Brainstorm list of data sources ( speech recognition)

Source	Amount	Cost	
Owned	100h	$0	✓
Crowdsourced – Reading	1000h	$10000	
Pay for labels	100h	$6000	
Purchase data	1000h	$10000	✓

Other factors: Data quality, privacy, regulatory constraints

Labeling data

- Options: In-house vs. outsourced vs. crowdsourced
- Having MLEs label data is expensive. But doing this for just a few days is usually fine.
- Who is qualified to label?

Speech recognition – any reasonably fluent speaker

Factory inspection, medical image diagnosis – SME (subject matter expert)

Recommender systems – maybe impossible to label well

- Don’t increase data by more than 10x at a time

DeepLearning.AI

Label and organize data

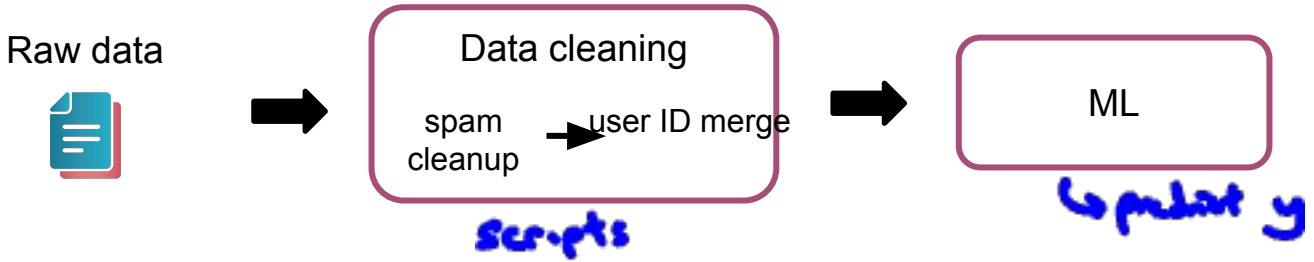
Data pipeline

Data pipeline example

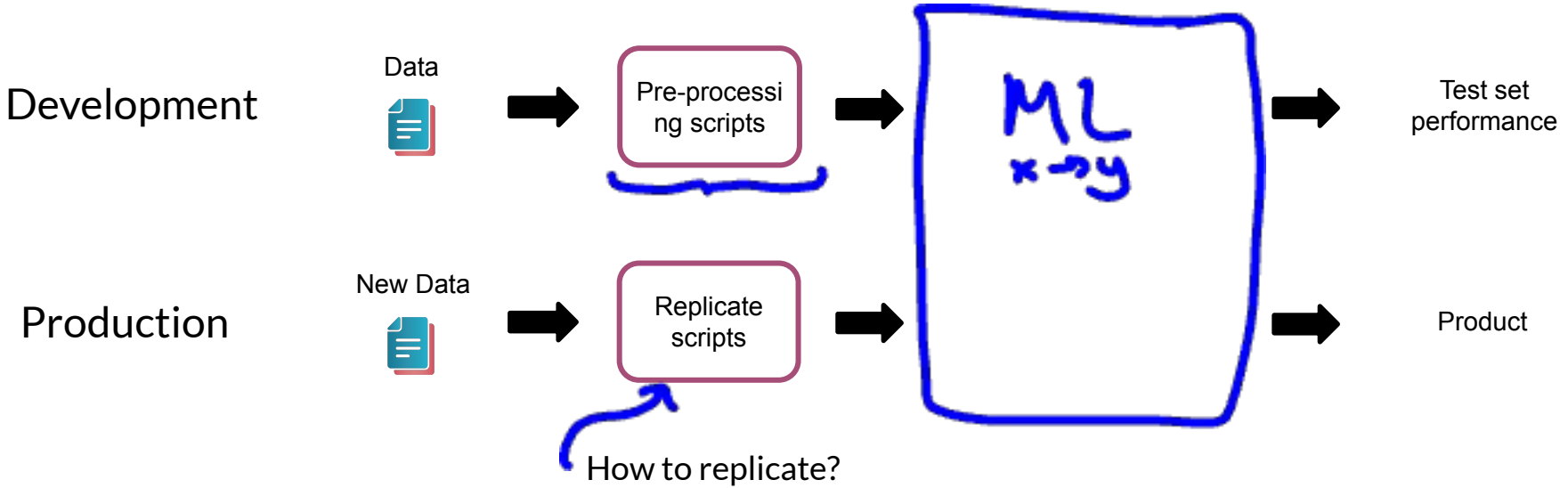
	Job Board (website)	Resume chat (app)
Email	nova@deeplearning.ai	nova@chatapp.com
First Name	Nova	Nova
Last Name	Ng	Ng
Address	1234 Jane Way	?
State	CA	?
Zip	94304	94304

x = user info

y = looking for job



Data pipeline example



POC and Production phases

POC (proof-of-concept):

- Goal is to decide if the application is workable and worth deploying.
- Focus on getting the prototype to work!
- It's ok if data pre-processing is manual. But take extensive notes/comments.

Production phase:

- After project utility is established, use more sophisticated tools to make sure the data pipeline is replicable.
- E.g., TensorFlow Transform, Apache Beam, Airflow,….

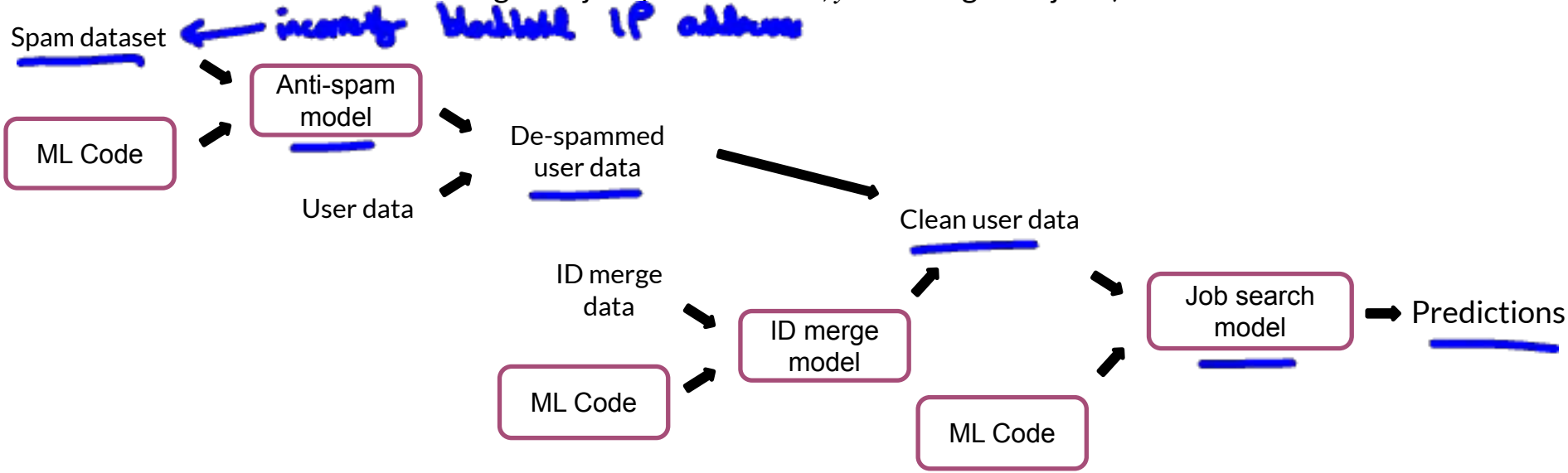
DeepLearning.AI

Label and organize data

Meta-data, data provenance and lineage

Data pipeline example

Task: Predict if someone is looking for a job. (x = user data, y = looking for a job?)



Keep track of data provenance and lineage

where it comes from

sequence of steps

Meta-data

Examples:

Manufacturing visual inspection: Time, factory, line #, camera settings, phone model, inspector ID,....

Speech recognition: Device type, labeler ID, VAD model ID,....

Useful for:

- Error analysis. Spotting unexpected effects.
- Keeping track of data provenance.

DeepLearning.AI

Label and organize data

Balanced train/dev/test splits

Balanced train/dev/test splits in small data problems

Visual inspection example: 100 examples, 30 positive (defective)

Train/dev/test: 60% / 20% / 20%

Random split: 21 / 2 / 7 positive example

3% 1% 3%

Want: 18 / 6 / 6 } balanced split

30% / 30% / 30%

No need to worry about this with large datasets – a random split will be representative.

DeepLearning.AI

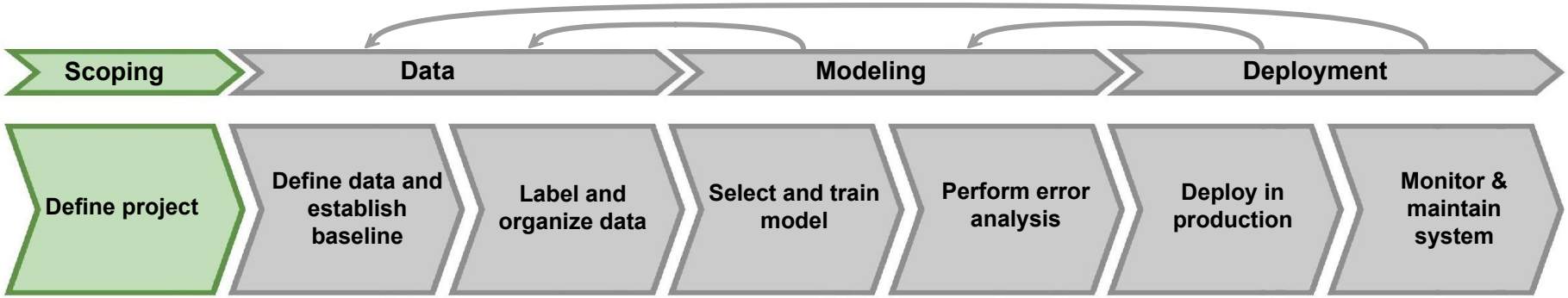
C1W3 Slides (Optional)

Scoping

DeepLearning.AI

Scoping (optional)

What is scoping?



Scoping example: Ecommerce retailer looking to increase sales

- Better recommender system
- Better search
- Improve catalog data
- Inventory management
- Price optimization

Questions:

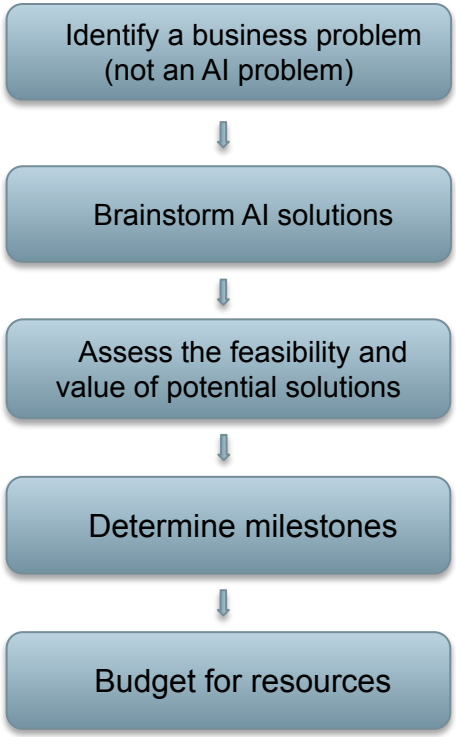
- What projects should we work on?
- What are the metrics for success?
- What are the resources (data, time, people) needed?

DeepLearning.AI

Scoping (optional)

Scoping process

Scoping process



What are the top 3 things you wish were working better?

- Increase conversion
- Reduce inventory
- Increase margin (profit per item)

Separating problem identification from solution

Problem	Solution
Increase conversion	Search, recommendations
Reduce inventory	Demand prediction, marketing
Increase margin (profit per item)	Optimizing what to sell (e.g., merchandising), recommend bundles
What to achieve	How to achieve

DeepLearning.AI

Scoping (optional)

Diligence on feasibility and value

Feasibility: Is this project technically feasible?

Use external benchmark (literature, other company, competitor)

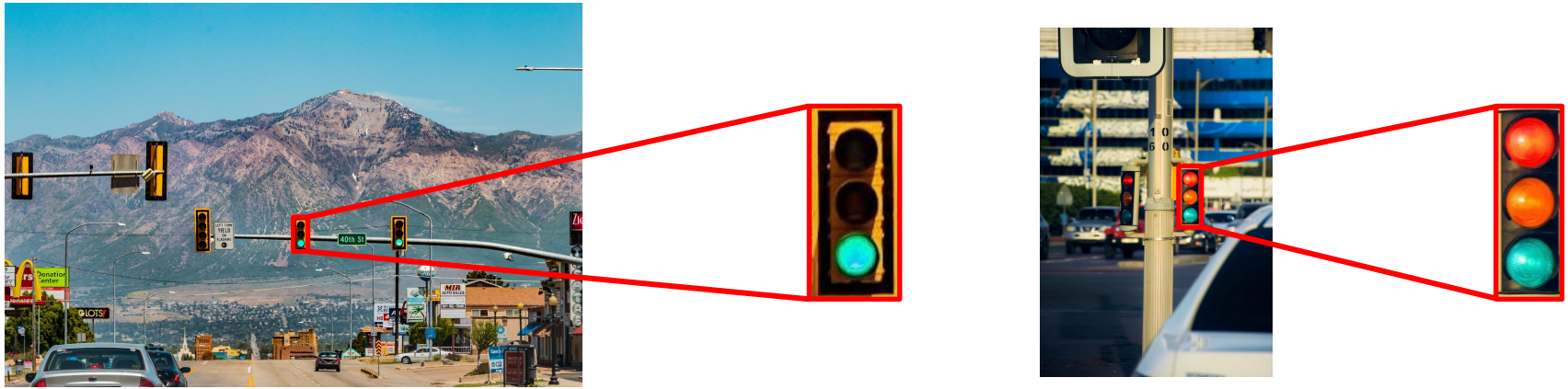
	Unstructured (e.g., speech, images)	Structured (e.g., transactions, records)
New	HLP	Predictive features available?
Existing	HLP History of project	New predictive features? History of project

HLP: Can a human, given the same data, perform the task?

Why use HLP to benchmark?

People are very good on unstructured data tasks

Criteria: Can a human, given the same data, perform the task?



Do we have features that are predictive?

x y

Given past purchases, predict future purchases ✓

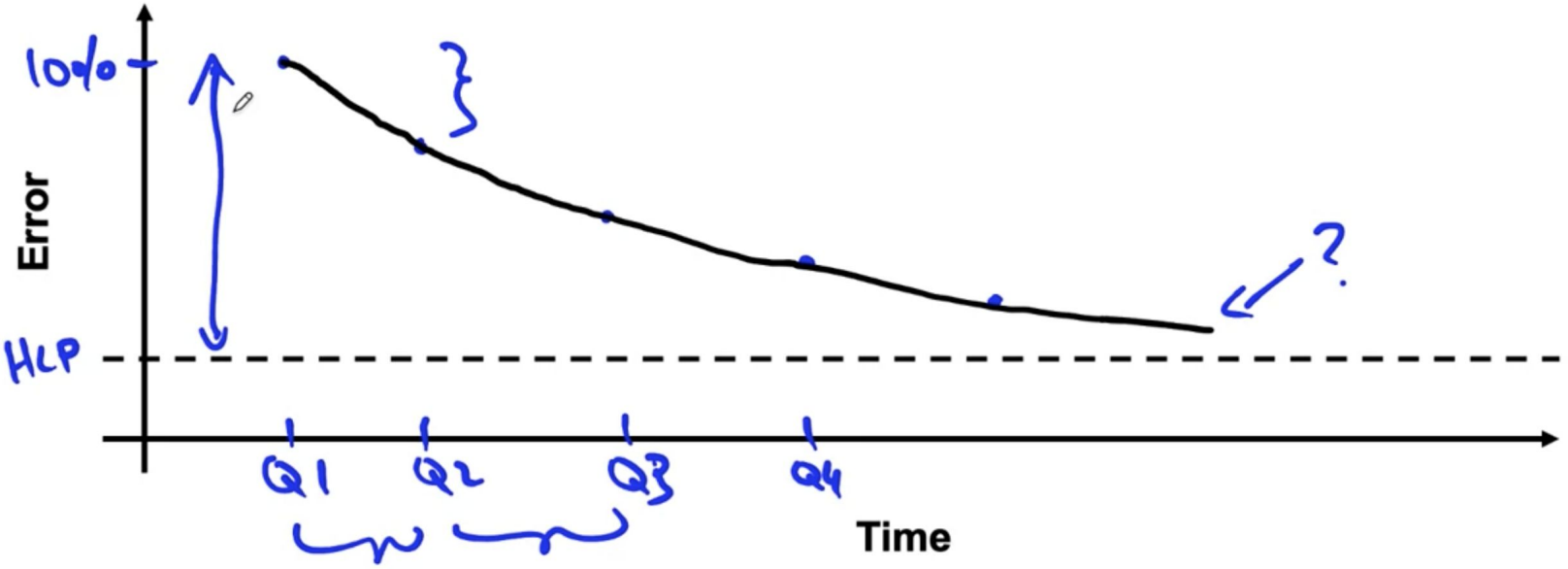
Given weather, predict shopping mall foot traffic ✓

Given DNA info, predict heart disease ?

Given social media chatter, predict demand for a clothing style ?

Given history of a stock's price, predict future price of that stock ✗

History of project



DeepLearning.AI

Scoping (optional)

Diligence on value

Diligence on value

Have technical and business teams try to agree on metrics that both are comfortable with.

Fermi estimates

Ethical considerations

- Is this project creating net positive societal value?
- Is this project reasonably fair and free from bias?
- Have any ethical concerns been openly aired and debated?

DeepLearning.AI

Scoping (optional)

Milestones and resourcing

Milestones

Key specifications:

- ML metrics (accuracy, precision/recall, etc.)
- Software metrics (latency, throughput, etc. given compute resources)
- Business metrics (revenue, etc.)
- Resources needed (data, personnel, help from other teams)

Timeline

If unsure, consider benchmarking to other projects, or building a POC (Proof of Concept) first.

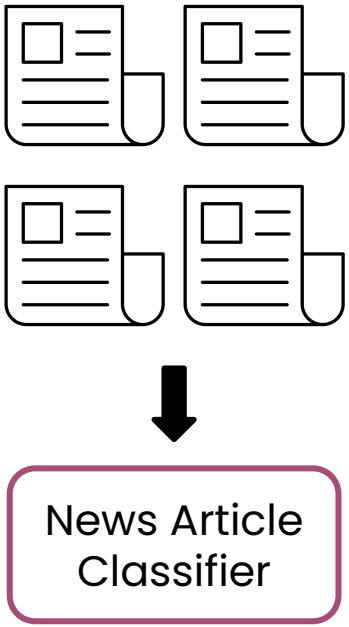
DeepLearning.AI

Final project

Final project overview

Project overview

- Classify news articles
- Start from an existing prototype
- Iteratively improve the performance of the system



Techniques

- Establish a baseline
- Balanced train/dev/test split
- Error analysis
- Track experiments
- Deploy using Tensorflow Serving

DeepLearning.AI Community
community.deeplearning.ai/c/ai-projects/