

# Select and train model

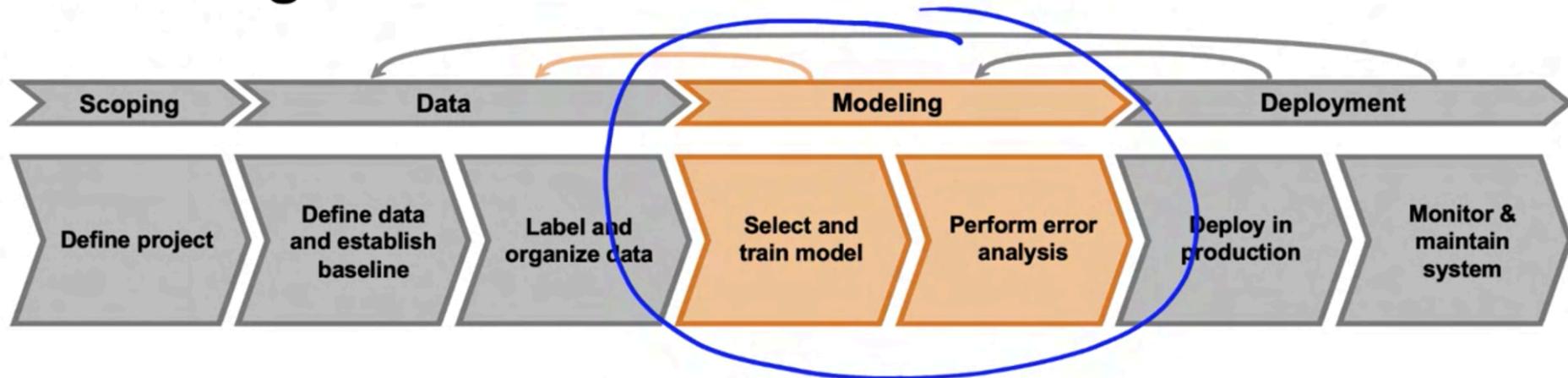
---

## Modeling overview



DeepLearning.AI

# Modeling



Model-centric AI  
development

Data-centric AI  
development

# Select and train model

---

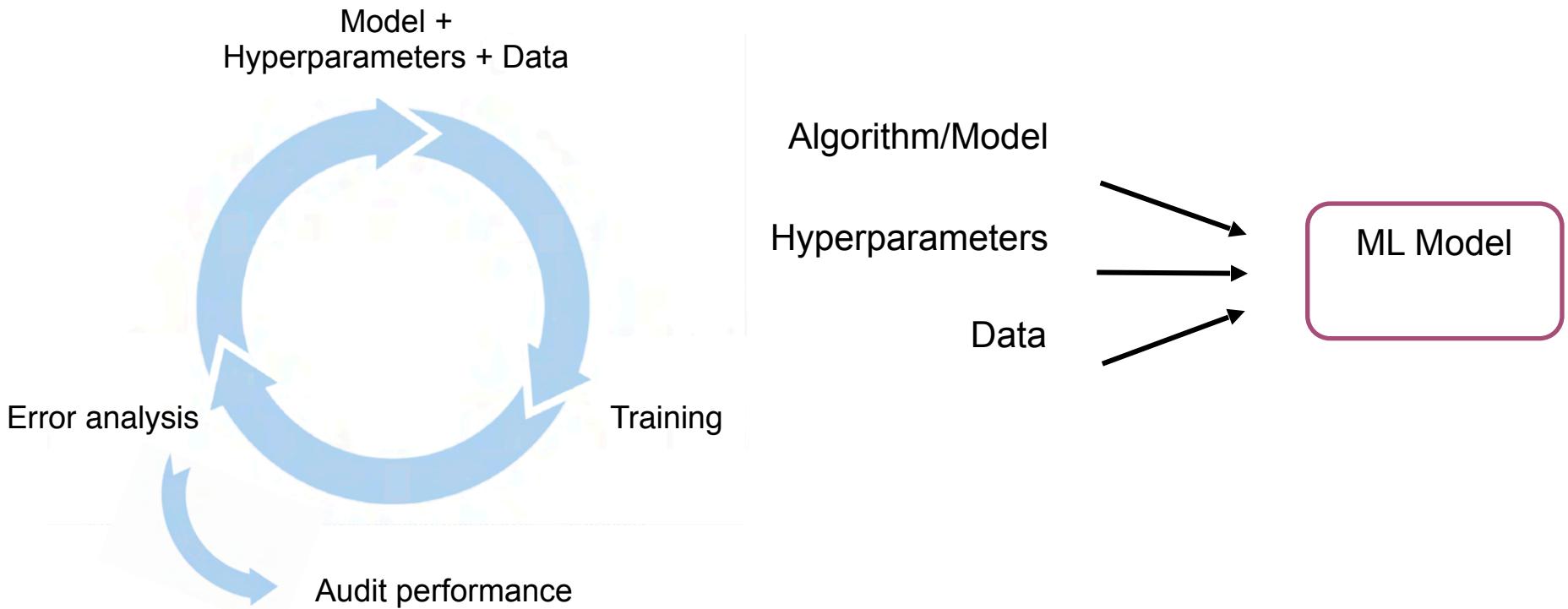


DeepLearning.AI

## Key challenges

**AI system = Code + Data**  
(algorithm/model)

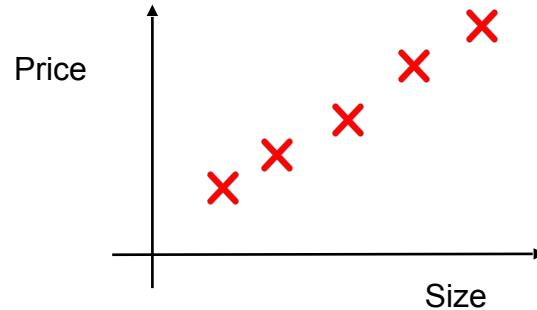
# Model development is an iterative process



# 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.

# Select and train model

---



DeepLearning.AI

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"

"Stanford"

"Reddit"

"Youtube"



**Informational and  
Transactional queries**



**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

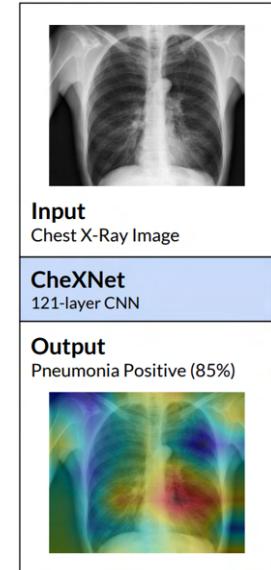
99% negative

1% positive

print("0") ←

Accuracy in rare classes

Condition	Performance
Effusion	0.901 ←
Edema	0.924
Mass	0.909
Hernia	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

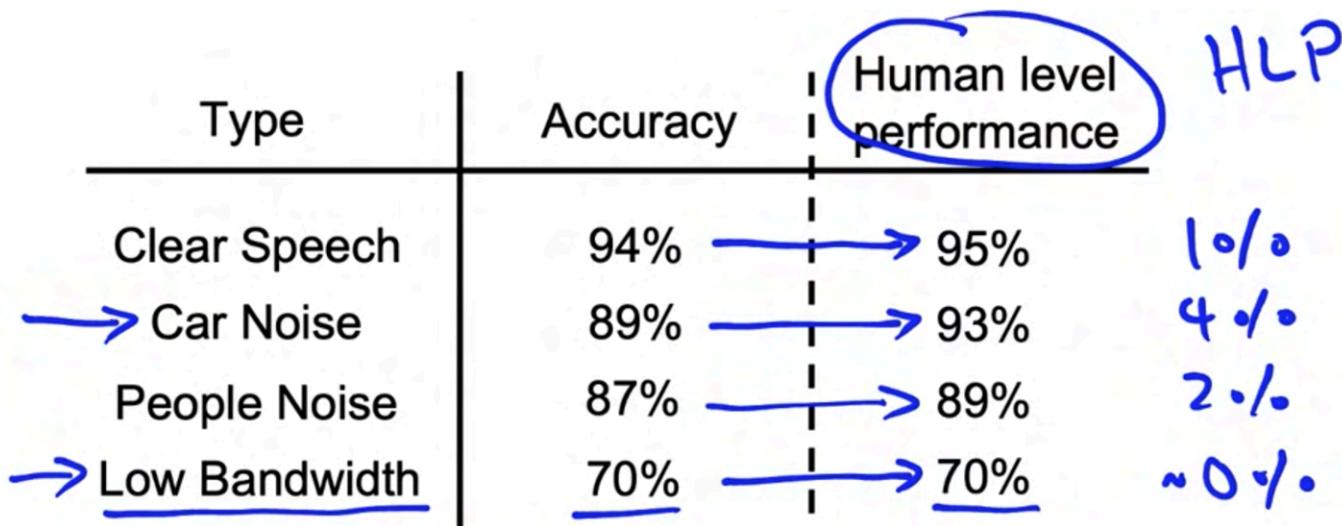
---

## Establish a baseline

# Establishing a baseline level of performance



Speech recognition example:



# Structured and unstructured data

Unstructured data	Structured Data												
Image 	<table><thead><tr><th>User Id</th><th>Purchase</th><th>Number</th><th>Price</th></tr></thead><tbody><tr><td>3421</td><td>Blue shirt</td><td>5</td><td>\$20</td></tr><tr><td>612</td><td>Brown shoes</td><td>1</td><td>\$35</td></tr></tbody></table>	User Id	Purchase	Number	Price	3421	Blue shirt	5	\$20	612	Brown shoes	1	\$35
User Id	Purchase	Number	Price										
3421	Blue shirt	5	\$20										
612	Brown shoes	1	\$35										
Audio 													
Text <div style="border: 1px solid black; padding: 10px;">This restaurant was great!</div>	<table><thead><tr><th>Price</th><th>Product</th></tr></thead><tbody><tr><td>3421</td><td>Red skirt</td></tr></tbody></table>	Price	Product	3421	Red skirt								
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.

# Select and train model

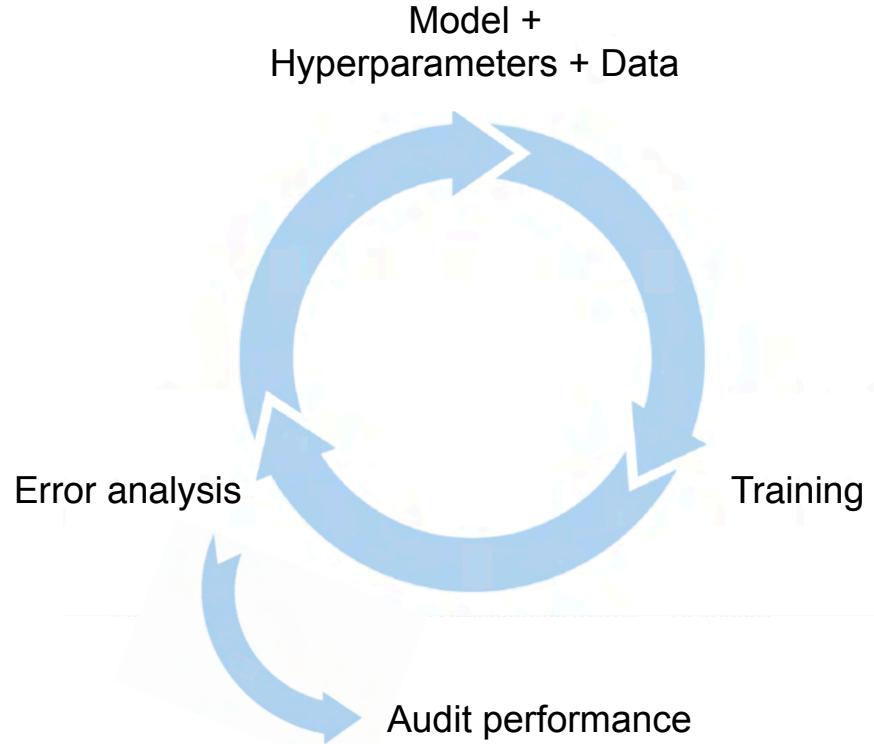
---



DeepLearning.AI

## 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



- Example #2: Image segmentation



- Example #3: Image classification

# Error analysis and performance auditing

---



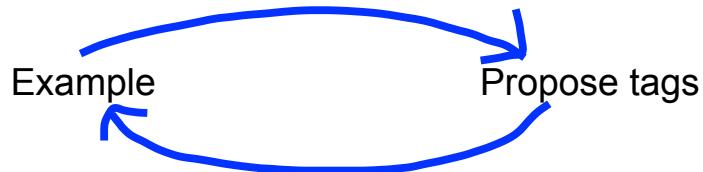
DeepLearning.AI

## 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?

# Error analysis and performance auditing

---



DeepLearning.AI

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:

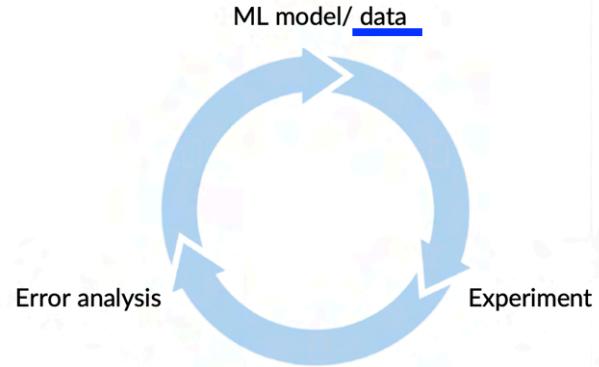
- How much room for improvement there is.
- How frequently that category appears.
- How easy is to improve accuracy in that category.
- How important it is to improve in that category.

# 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%



# Error analysis and performance auditing

---



DeepLearning.AI

Skewed  
datasets

# Examples of skewed datasets



## Manufacturing example

99.7% no defect

$$y=0$$

print("0")  
99.7%

0.3% defect

$$y=1$$



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

		Actual	
		$y=0$	$y=1$
Predicted	$y=0$	905 TN	18 FN
	$y=1$	9 FP	68 TP

$\hookrightarrow 914$        $\hookrightarrow 86$

$TN$ : True Negative

$TP$ : True Positive

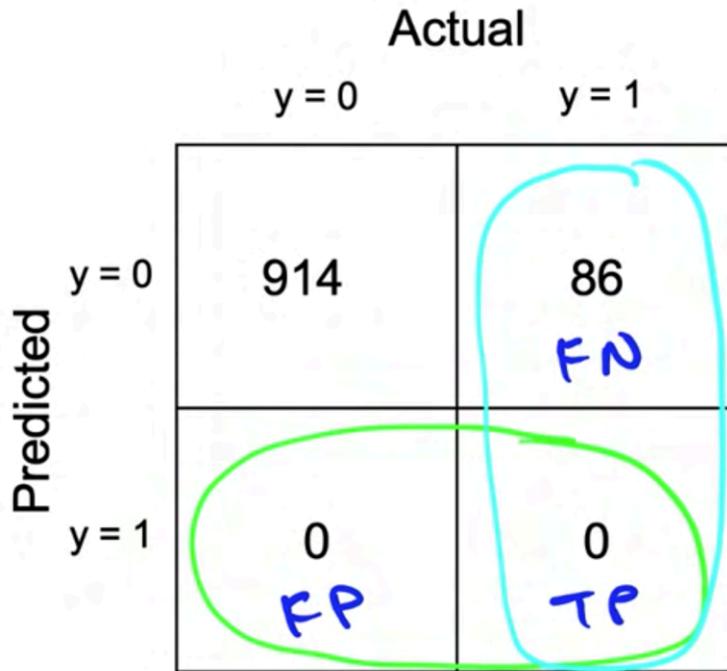
$FN$ : False Negative

$FP$ : 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%

# Error analysis and performance auditing

---



DeepLearning.AI

## 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.

# Data iteration



DeepLearning.AI

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.*

# Data iteration

---



DeepLearning.AI

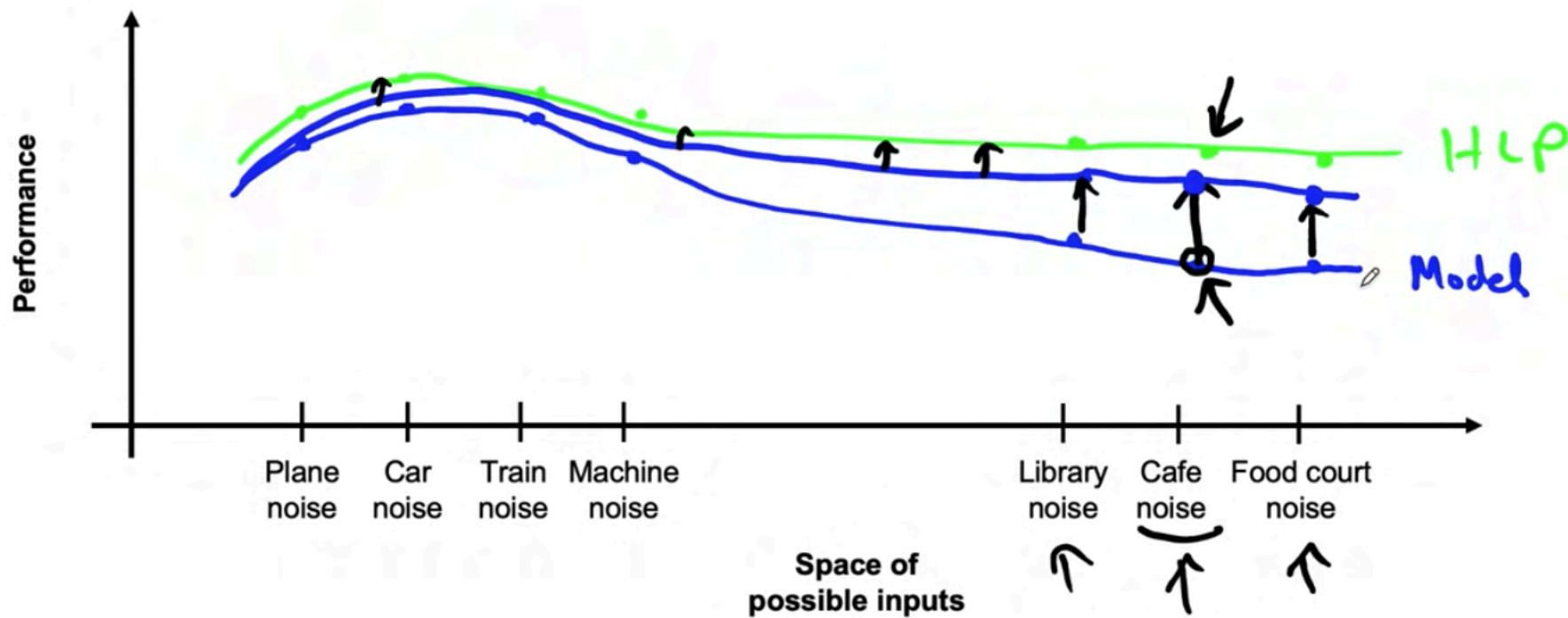
A useful picture of data  
augmentation

# 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



# Data iteration



DeepLearning.AI

# Data augmentation

# Data augmentation

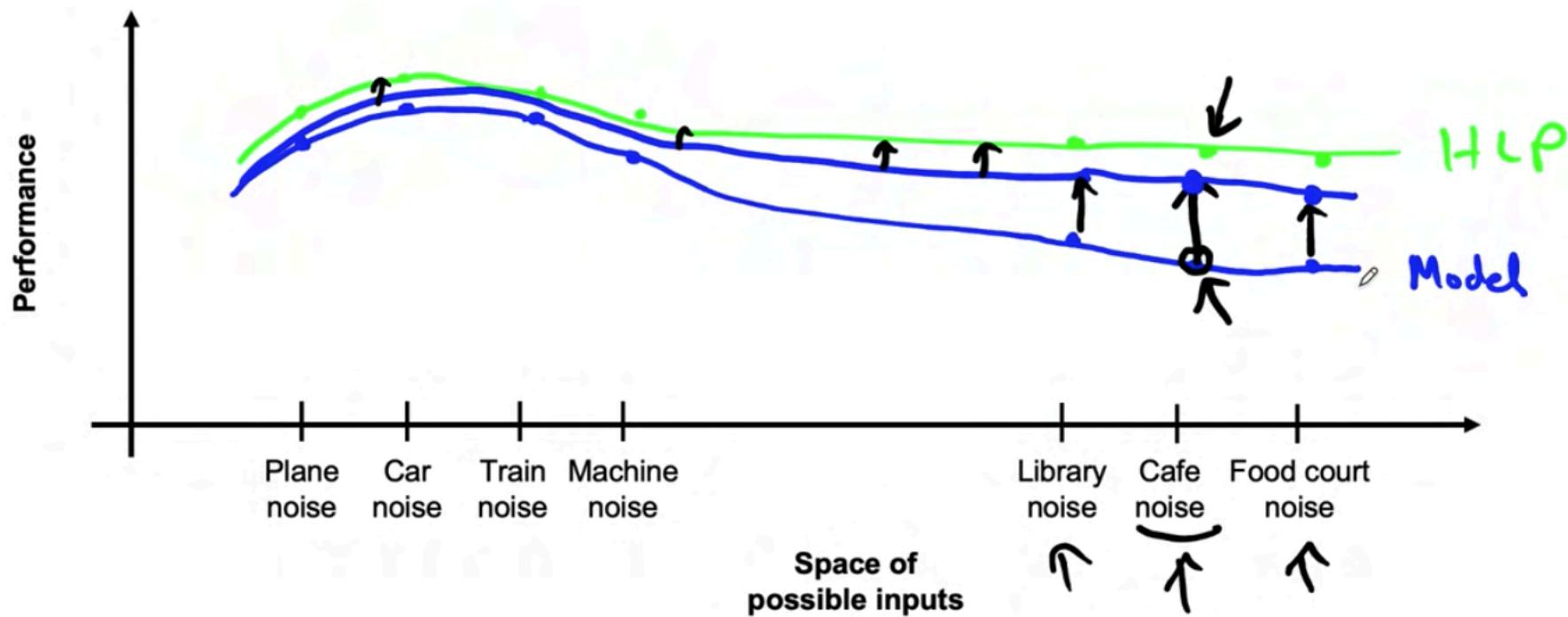
Goal:

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

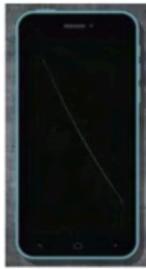
Checklist:

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

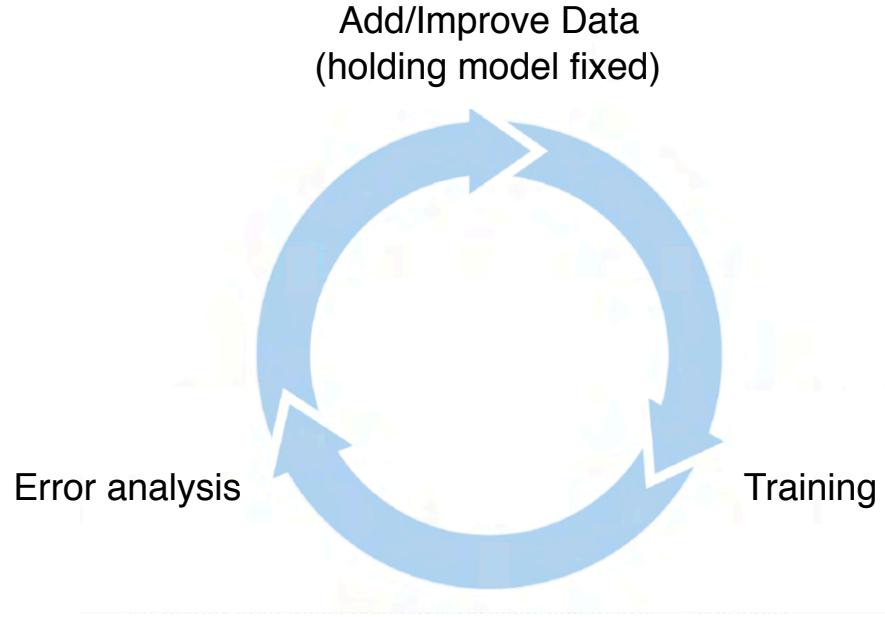
# The rubber sheet analogy



# 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  $X \rightarrow Y$  is clear (e.g., humans can make accurate predictions).

Then, **adding data rarely hurts accuracy.**

# Photo OCR counterexample



1

high accuracy



I

low accuracy

42I



↖

1? I?

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

# Data iteration

---



DeepLearning.AI

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

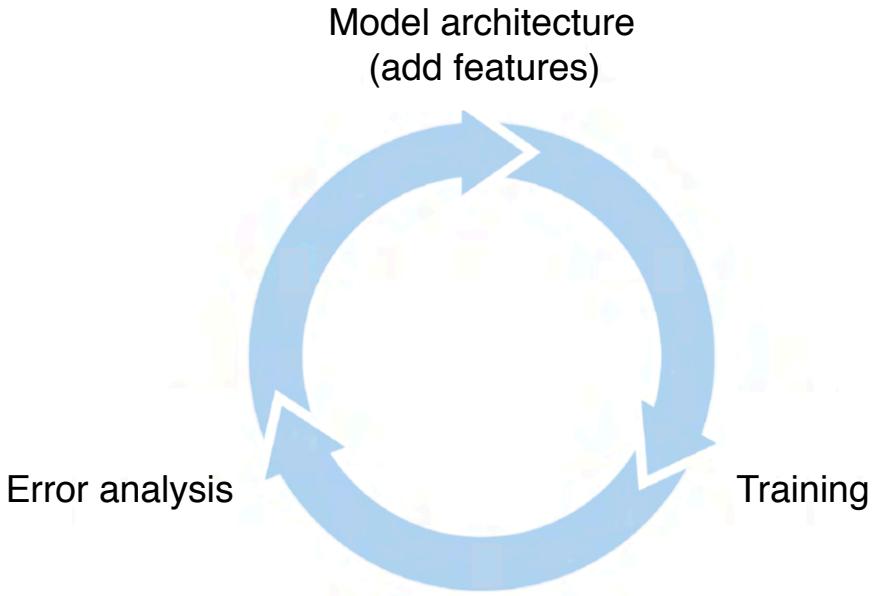
What are the added signals (features) that can help make a decision?

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.

# Data iteration



DeepLearning.AI

## 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

# Data iteration

---



DeepLearning.AI

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