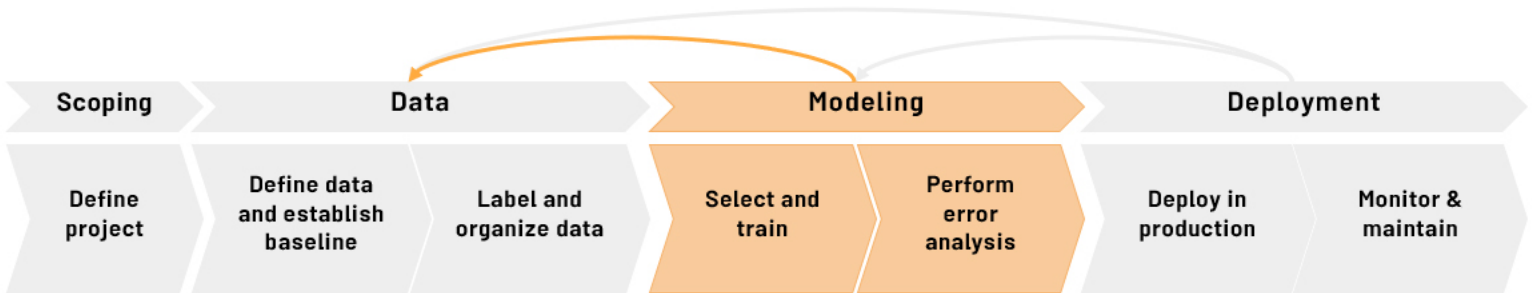
ML in Production

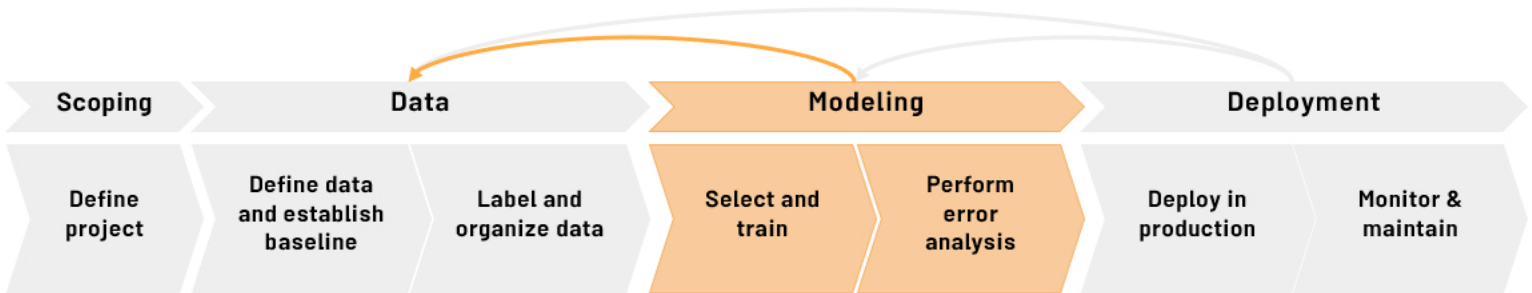
Part 2: Select and Train a Model

Ramin Toosi

Modeling

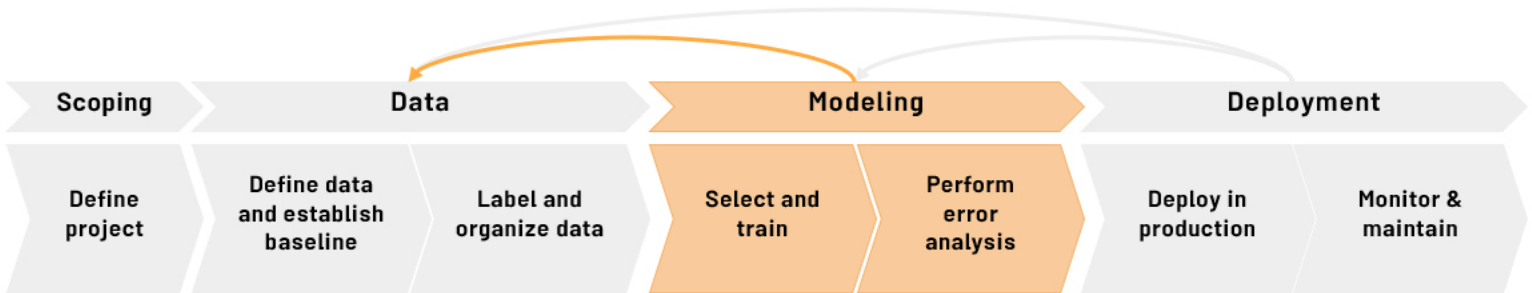

Scoping
Data
Modeling
Deployment
Define project
Define data and establish baseline
Label and organize data
Select and train
Perform error analysis
Deploy in production
Monitor & maintain

Modeling



Model centric AI development

Modeling



Model centric AI development

Data centric AI development

Key challenges

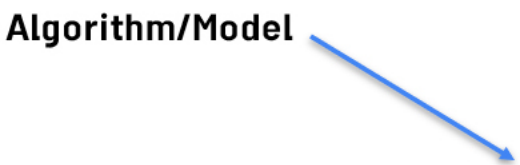
AI system = Code + Data

Key challenges

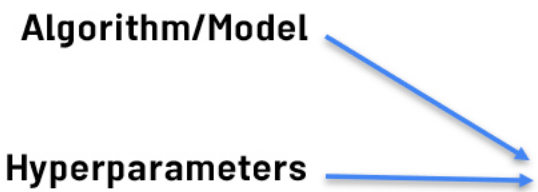
AI system = Code + Data

(algorithm/model)

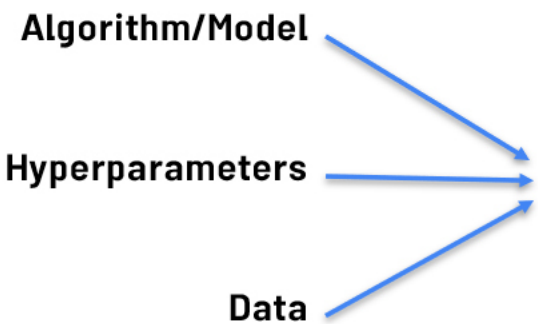
Model development is an iterative process



Model development is an iterative process



Model development is an iterative process

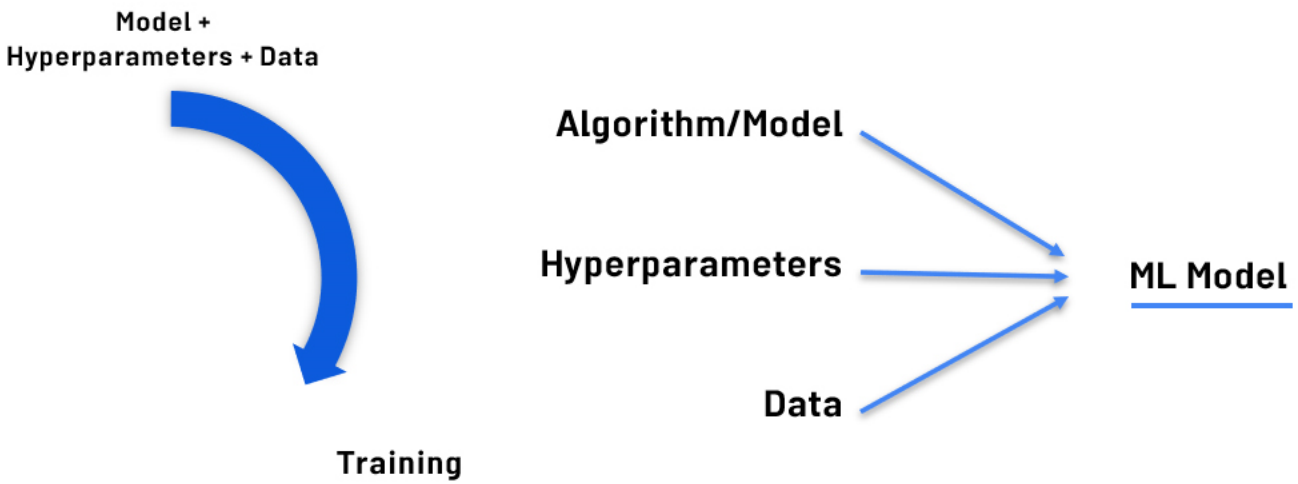


Model development is an iterative process

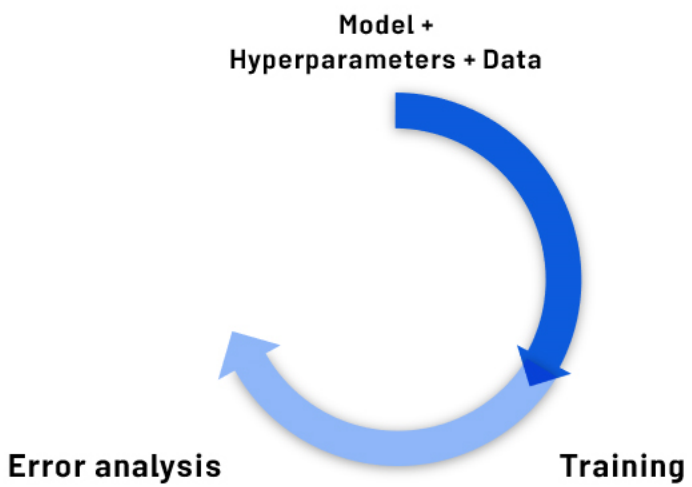
Model development is an iterative process

Model +
Hyperparameters + Data

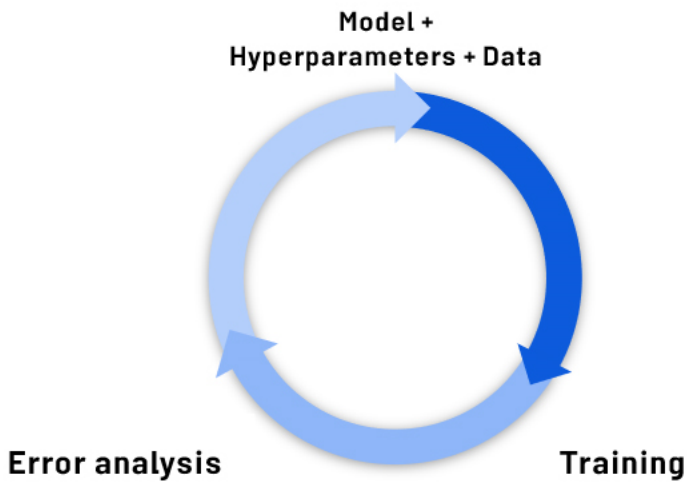
Model development is an iterative process



Model development is an iterative process

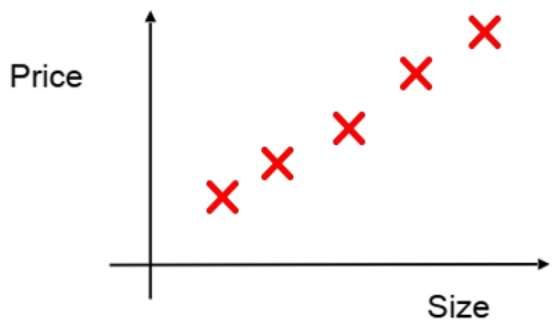


Model development is an iterative process



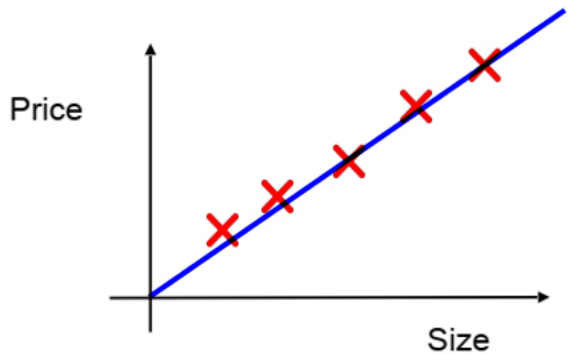
Challenges in model development

- Doing well on training set (usually measured by average training error).
- Doing well on dev/test sets.
- Doing well on business metrics/project goals.

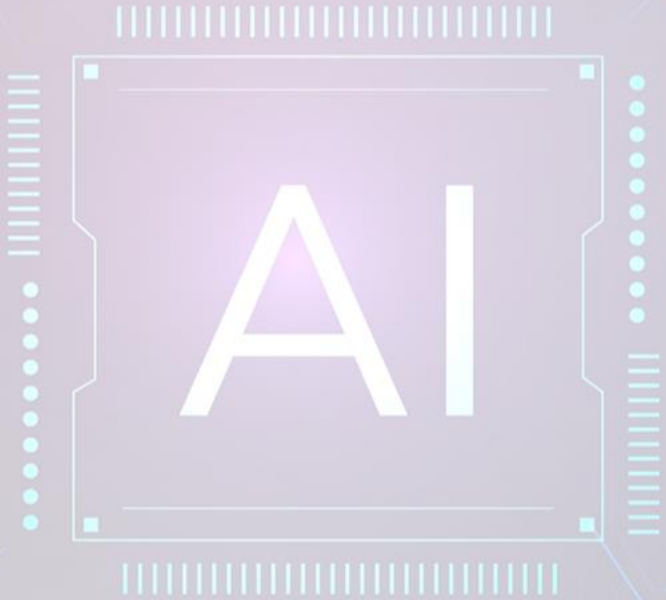


Challenges in model development

- Doing well on training set (usually measured by average training error).
- Doing well on dev/test sets.
- Doing well on business metrics/project goals.



Why low average test error isn't good enough



Performance on disproportionately important example

Web Search example

"Apple pie recipe"

"Diwali festival"

"Latest movies"

"Wireless data plan"

Performance on disproportionately important example

Web Search example

"Apple pie recipe"

"Diwali festival"

"Latest movies"

"Wireless data plan"

Informational and

Transactional queries

Performance on disproportionately important example

Web Search example

"Apple pie recipe"
"Diwali festival"
"Latest movies"
"Wireless data plan"

Informational and
Transactional queries

"Stanford" "Reddit" "Youtube"

Performance on disproportionately important example

Web Search example

"Apple pie recipe"
"Diwali festival"
"Latest movies"
"Wireless data plan"

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.

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”) !

Rare classes

- **Skewed data distribution**
  - 99% negative
  - 1% positive
  - print (“0”) !

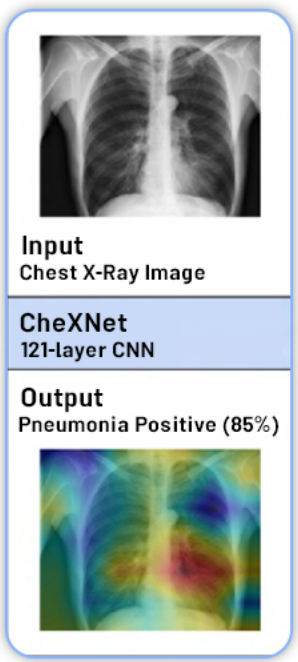
Accuracy in rare classes

Rare classes

- Skewed data distribution
  - 99% negative
  - 1% positive
  - print (“0”) !

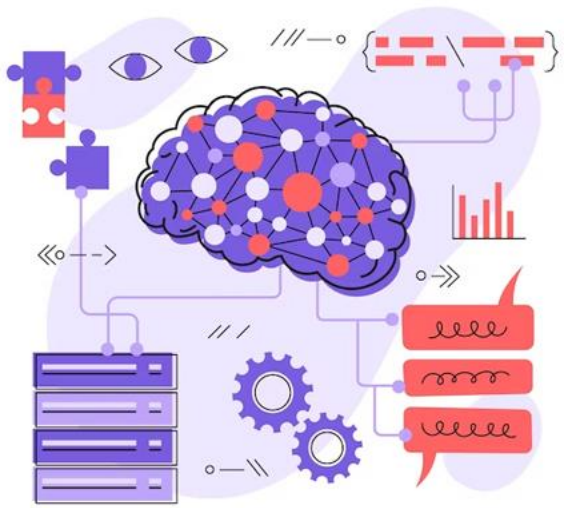
Accuracy in rare classes

Condition	Performance
Effusion	0.901
Edma	0924
Mass	0.909
Hernia	0.751



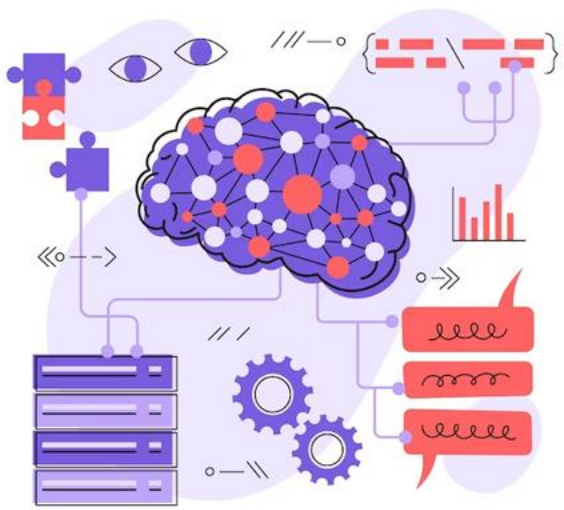
Unfortunate conversation in many companies

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



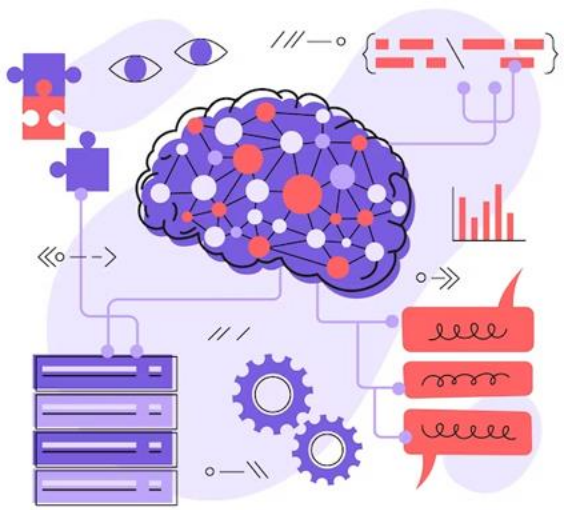
Unfortunate conversation in many companies

- MLE: "I did well on the test set!"
- Product Owner: "But this doesn't work for my application"

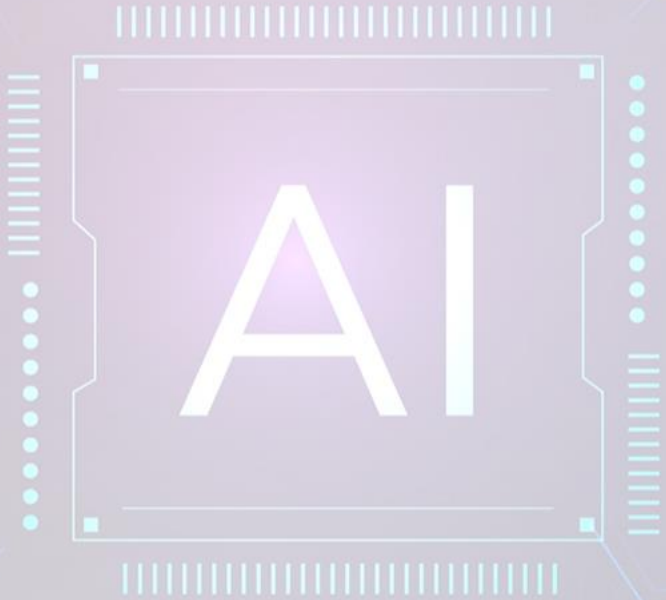


Unfortunate conversation in many companies

- 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!"



Establish a baseline



Establishing a baseline level of performance

Speech recognition example

Type	Accuracy
Clear Speech	94%
Car Noise	89%
People Noise	87%
Low Bandwidth	70%

Establishing a baseline level of performance

Speech recognition example

Type	Accuracy	Human Level Performance
Clear Speech	94%	95% (1%)
Car Noise	89%	93% (4%)
People Noise	87%	89% (2%)
Low Bandwidth	70%	70% (0%)

Structured and unstructured data

Unstructured data

Image

Audio

Text

This restaurant was great!

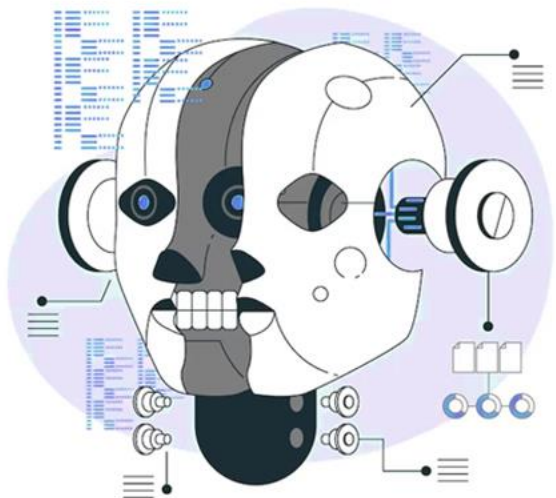
Structured Data

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

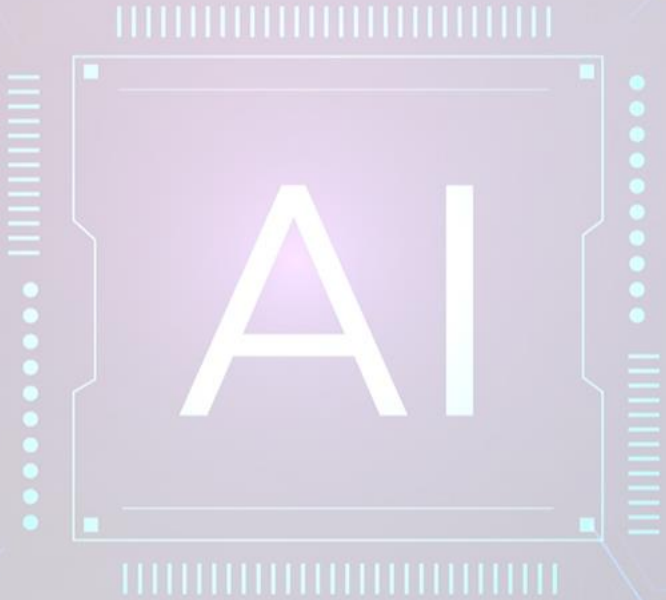
Ways to establish a baseline

- 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.

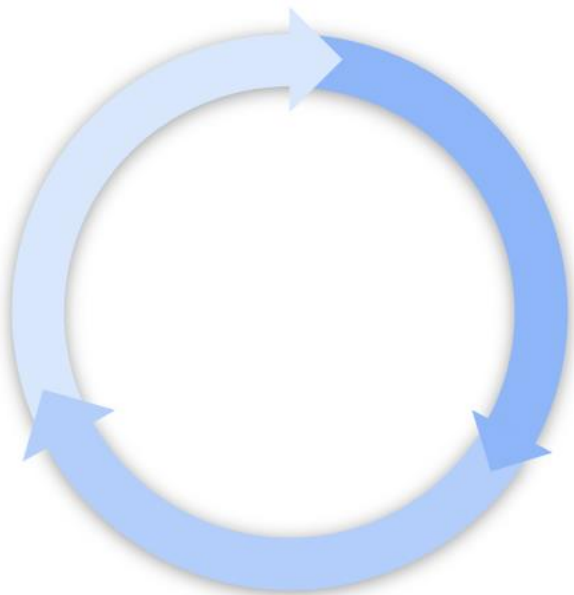


Tips for Getting Started a ML Project



ML is an iterative process

Model +
Hyperparameters + Data

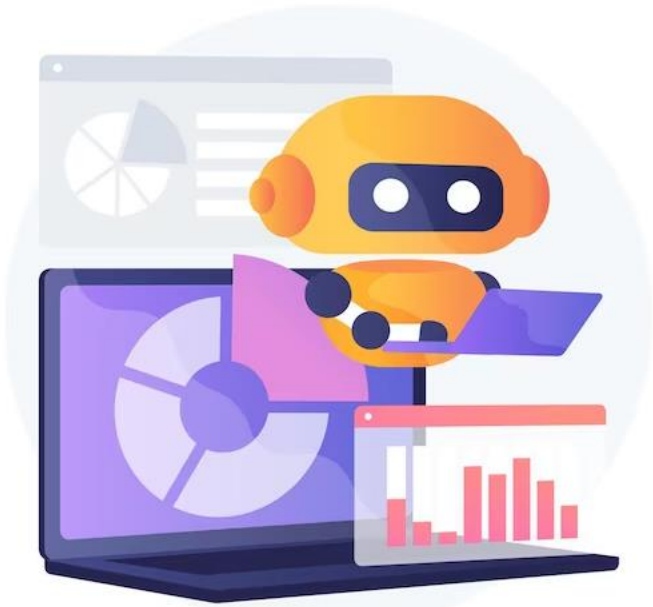


Training

Error analysis

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

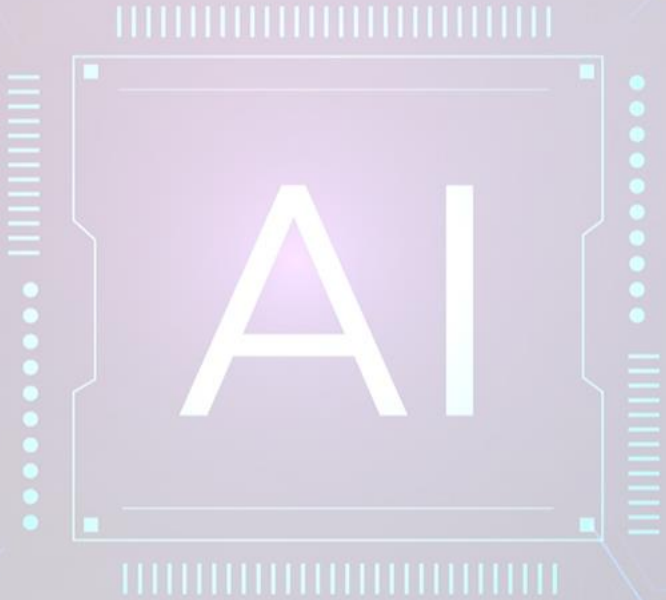
- Try to overfit a small training dataset before training on a large one.
- Example #1: Speech recognition
- Example #2: Image segmentation
- Example #3: Image classification

audio

transcript ␣ ␣ ␣ ␣ ␣ ␣

X→Y

Error analysis and performance auditing



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"			

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"			

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"			

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"			

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

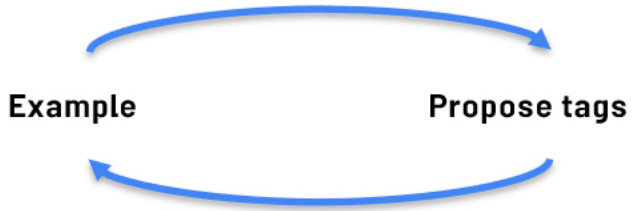
Example	Propose tags

Iterative process of error analysis

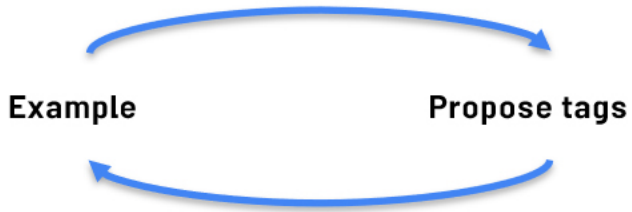
Example

Propose tags

Iterative process of error analysis

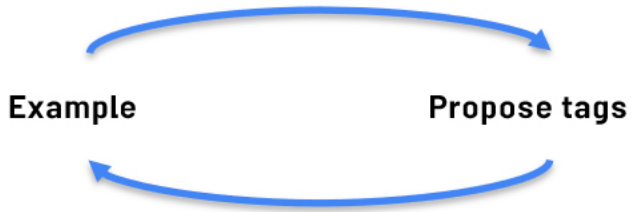


Iterative process of error analysis



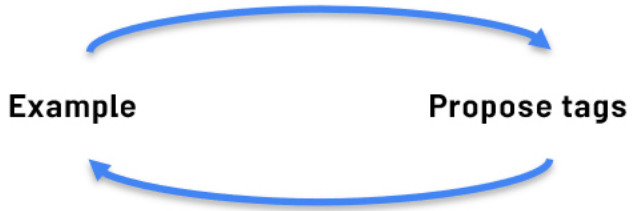
- **Visual inspection:**
  - Specific class labels (scratch, dent, etc.)
  - Image properties (blurry, dark background, light background, reflection….)
  - Other meta-data: phone model, factory

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

Iterative process of error analysis

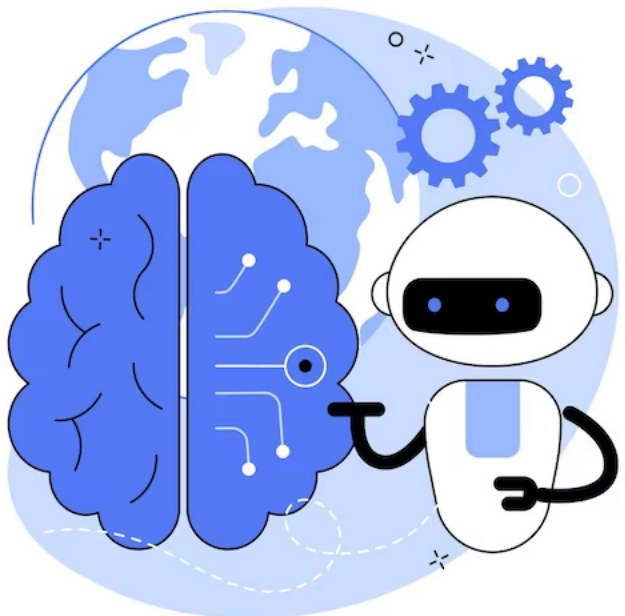


- **Visual inspection:**
  - Specific class labels (scratch, dent, etc.)
  - Image properties (blurry, dark
  - background, light background, reflection….)
  - Other meta-data: phone model, factory

- **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?



Prioritizing what to work on

Type	Accuracy	HLP
Clear Speech	94%	95% (1%)
Car Noise	89%	93% (4%)
People Noise	87%	89% (2%)
Low Bandwidth	70%	70% (0%)

Prioritizing what to work on

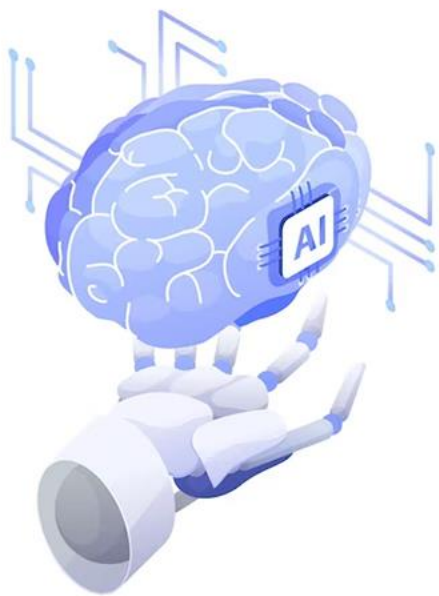
Type	Accuracy	HLP	% data
Clear Speech	94%	95% (1%)	60
Car Noise	89%	93% (4%)	4
People Noise	87%	89% (2%)	30
Low Bandwidth	70%	70% (0%)	6

Prioritizing what to work on

Type	Accuracy	HLP	% data	Improvement
Clear 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/improving data for specific categories

- **For categories you want to prioritize:**
  - Collect more data
  - Use data augmentation to get more data
  - Improve label accuracy / data quality

Skewed datasets

- **Manufacturing** example 
  - 99.7% no defect
  - 0.3% defect
- **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

Predicted \ Actual	Y = 0	Y = 1
Y = 1	905 TN	18 FN
Y = 0	9 FP	68 TP

Precision = TP / (TP + FP)

Recall = TP / (TP + FN)

What happens with print("0")?

Predicted \ Actual	Y = 0	Y = 1
Y = 1	914 TN	86 FN
Y = 0	0 FP	0 TP

Precision = TP / (TP + FP) = 0/0

Recall = TP / (TP + FN) = 0/86 = 0

Combining precision and recall – F_1 score

	Precision	Recall	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

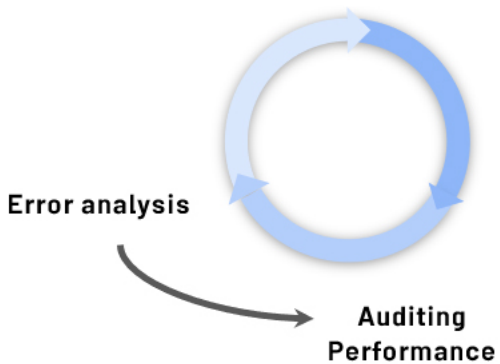
Defect Type	Precision	Recall	F_1
Scratch	82.1%	99.2%	89.8%
Dent	92.1%	99.5%	95.7%
Pit mark	85.3%	98.7%	91.5%
Discoloration	72.1%	97%	82.7%

Performance Auditing framework

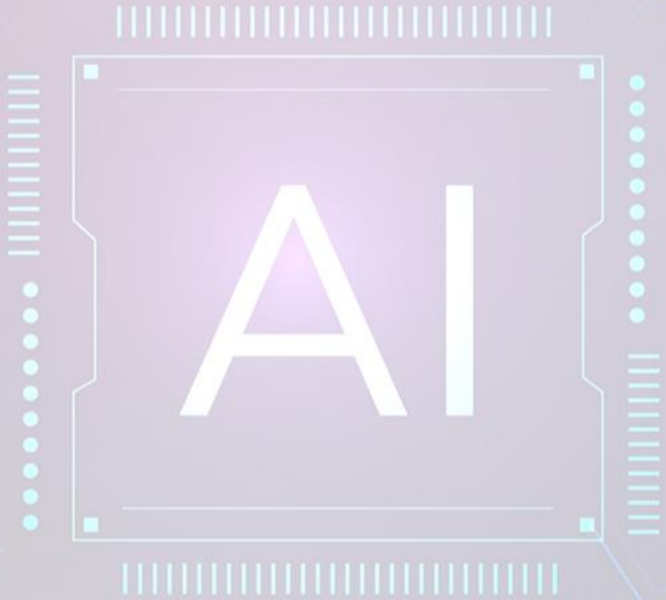
- Check for accuracy, fairness and bias.
- 1. Brainstorm the ways the system might go wrong.
  - Performance on subsets of data (e.g., ethnicity, gender).
  - Prevalence of specific errors/outputs (e.g., FP, FN).
  - Performance on rare classes.
- 2. Establish metrics to assess performance against these issues on appropriate slices of data.
- 3. Get business/product owner buy-in.

Speech recognition example

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



Data-centric AI development

- Model-centric view

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 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

Data-centric AI development

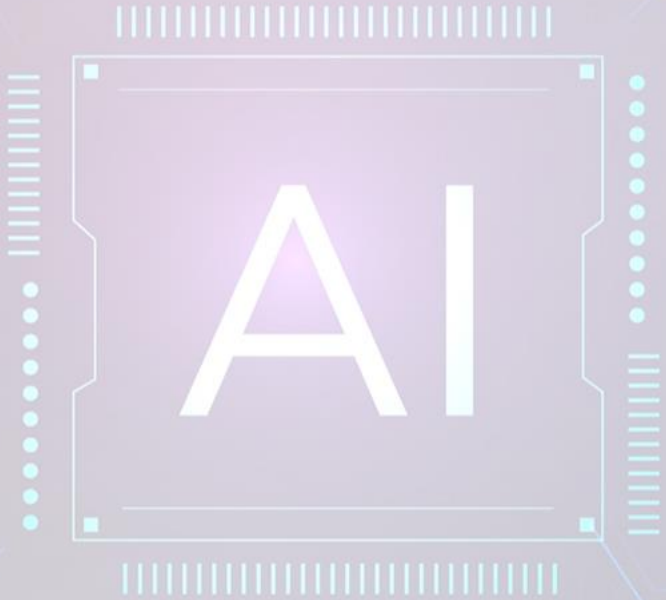
Model-centric view

- 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.

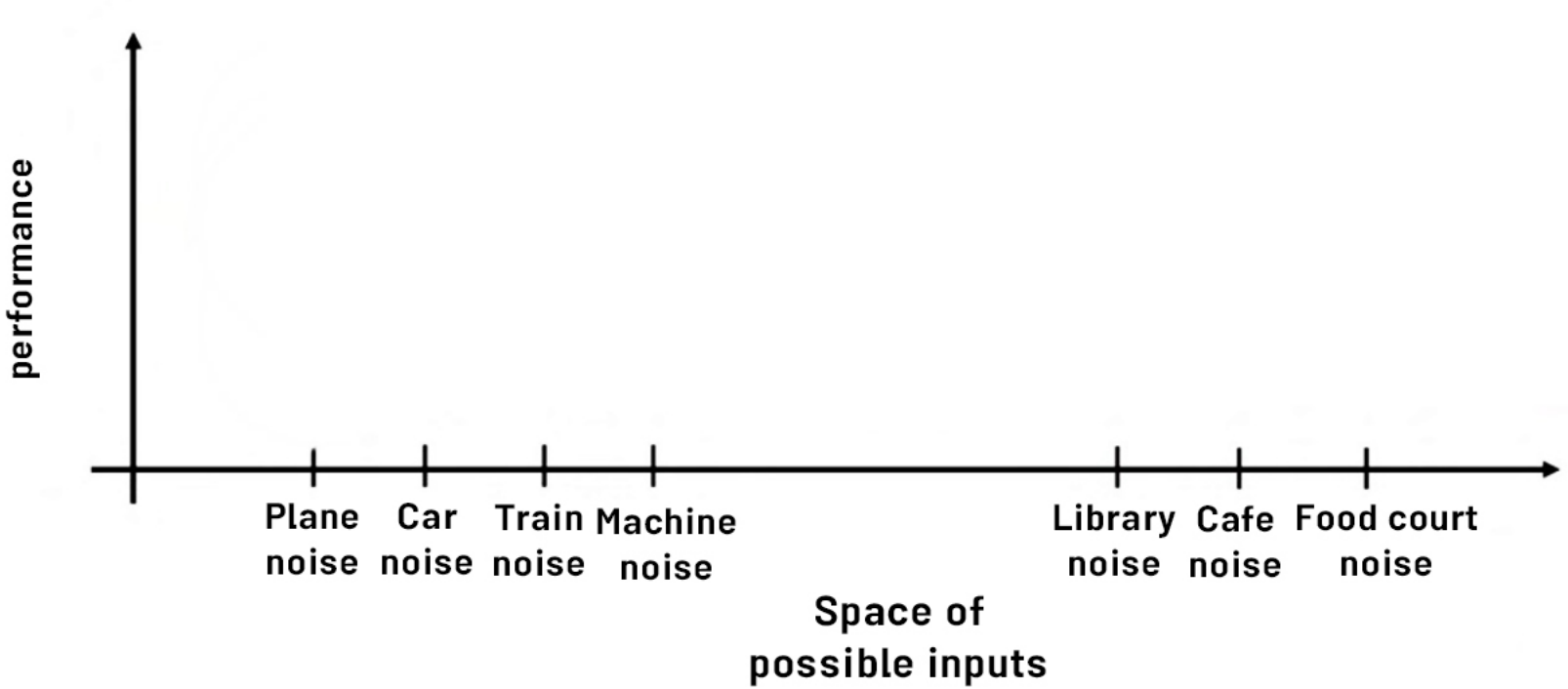
A useful picture of data augmentation



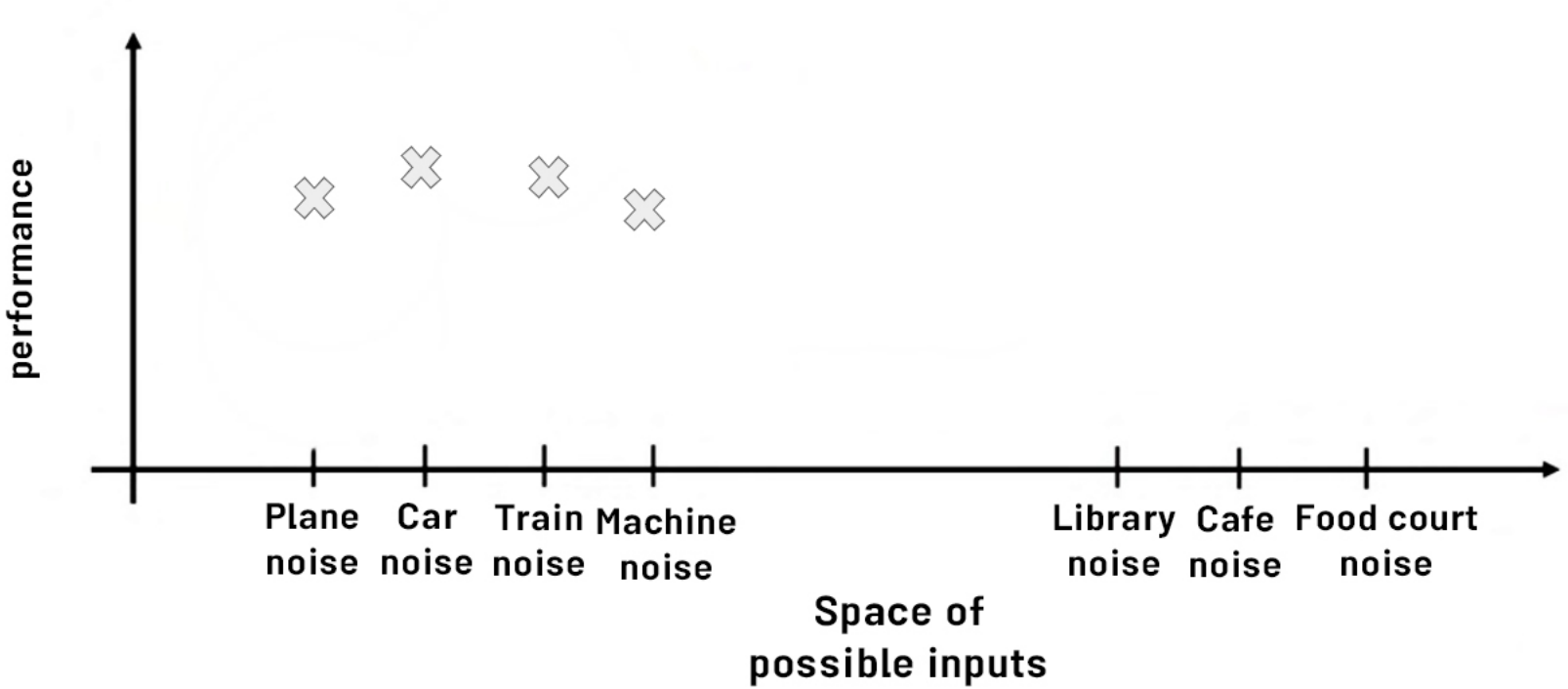
Speech Recognition example

- **Different types of speech input:**
  - Car noise
  - Plane noise
  - Train noise
  - Machine noise
  - Cafe noise
  - Library noise
  - Food court noise

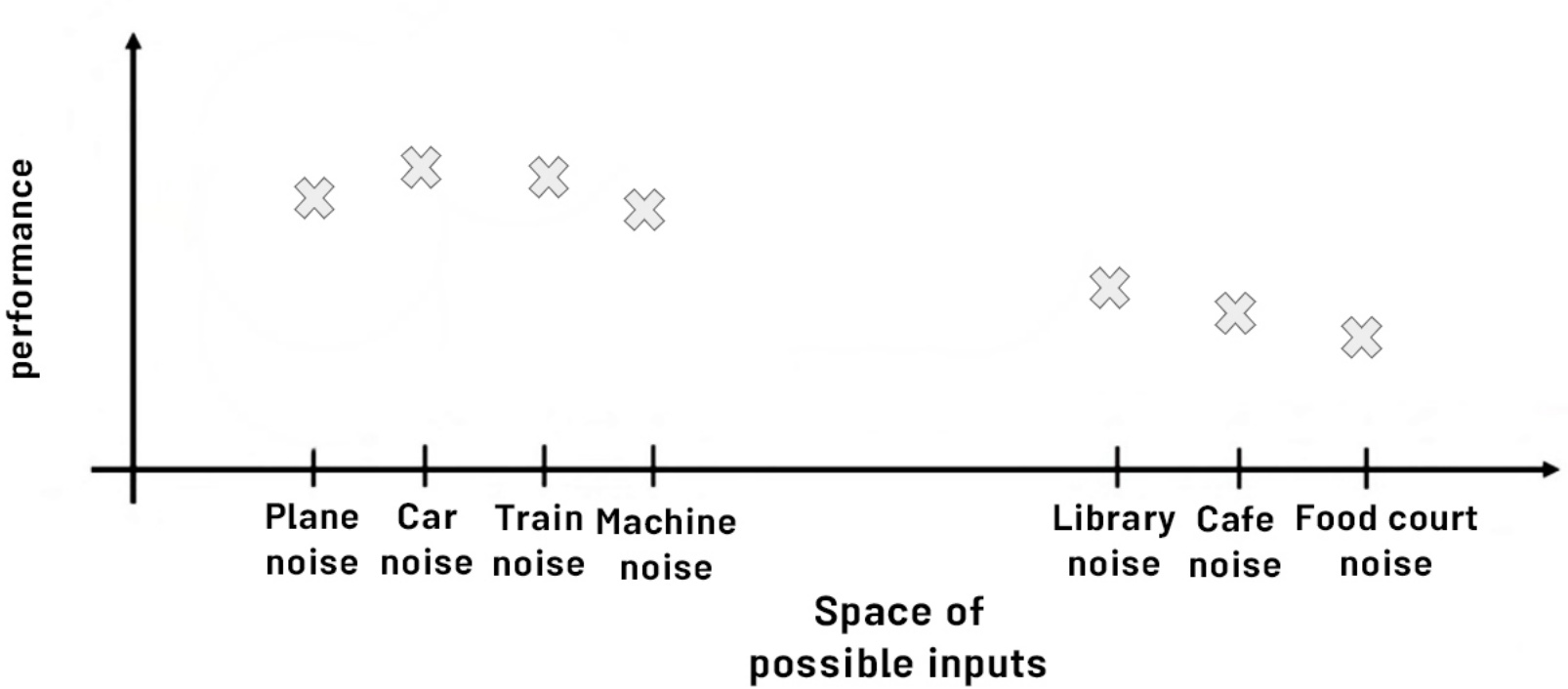
Speech Recognition example



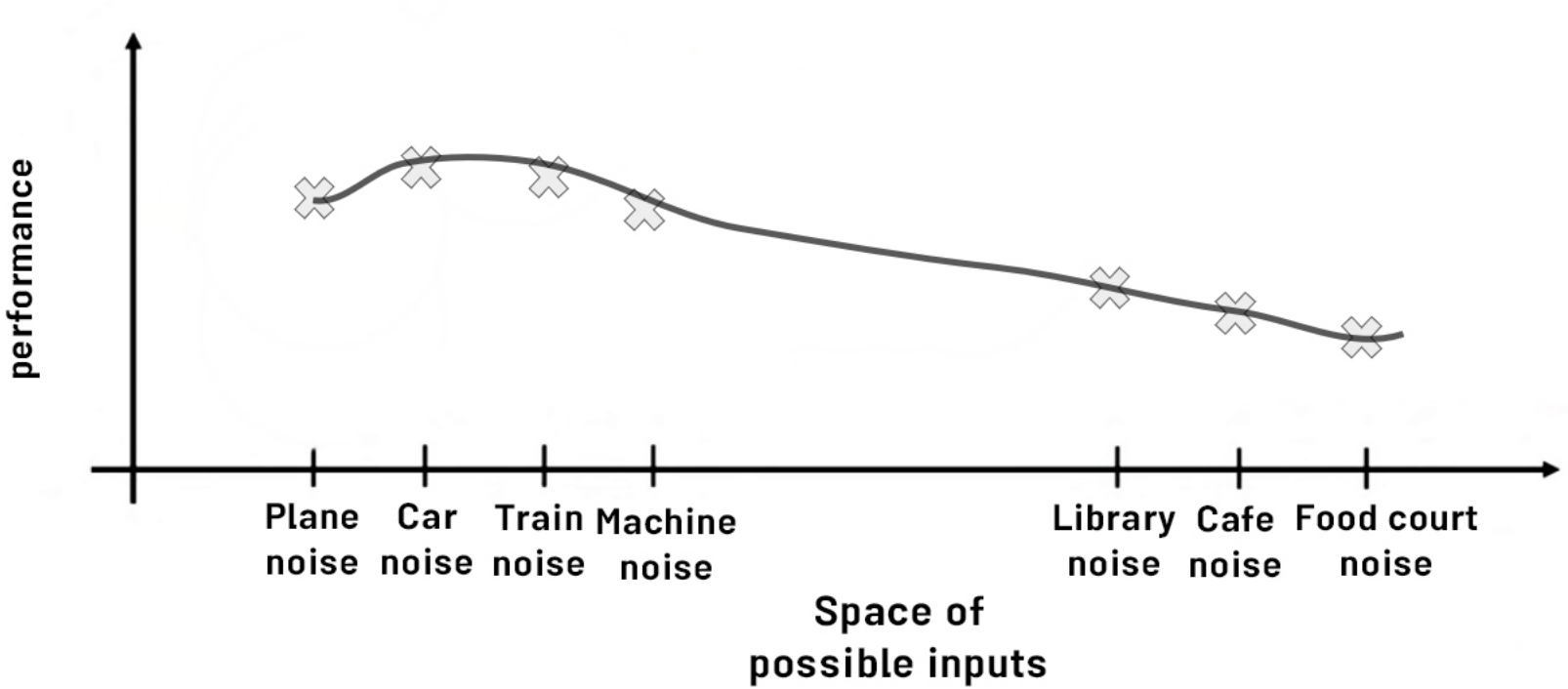
Speech Recognition example



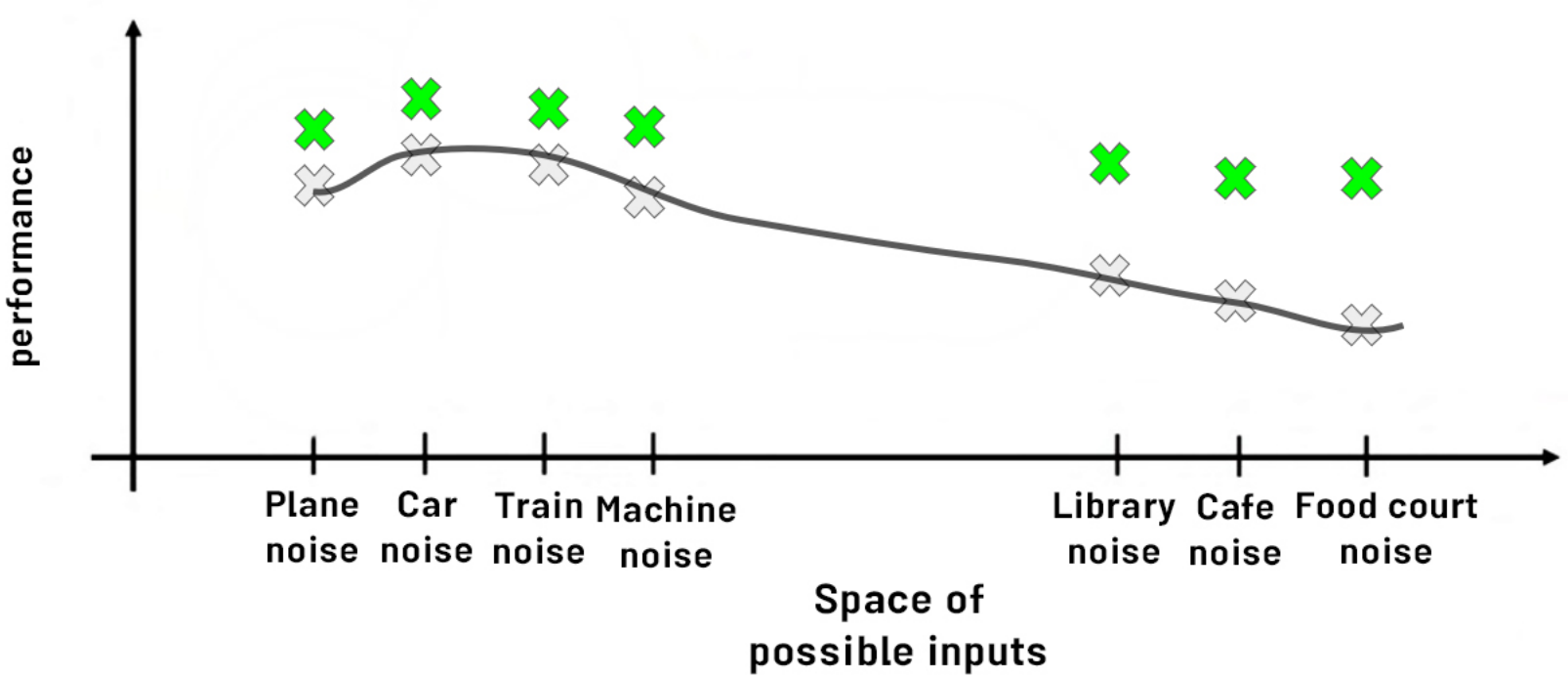
Speech Recognition example



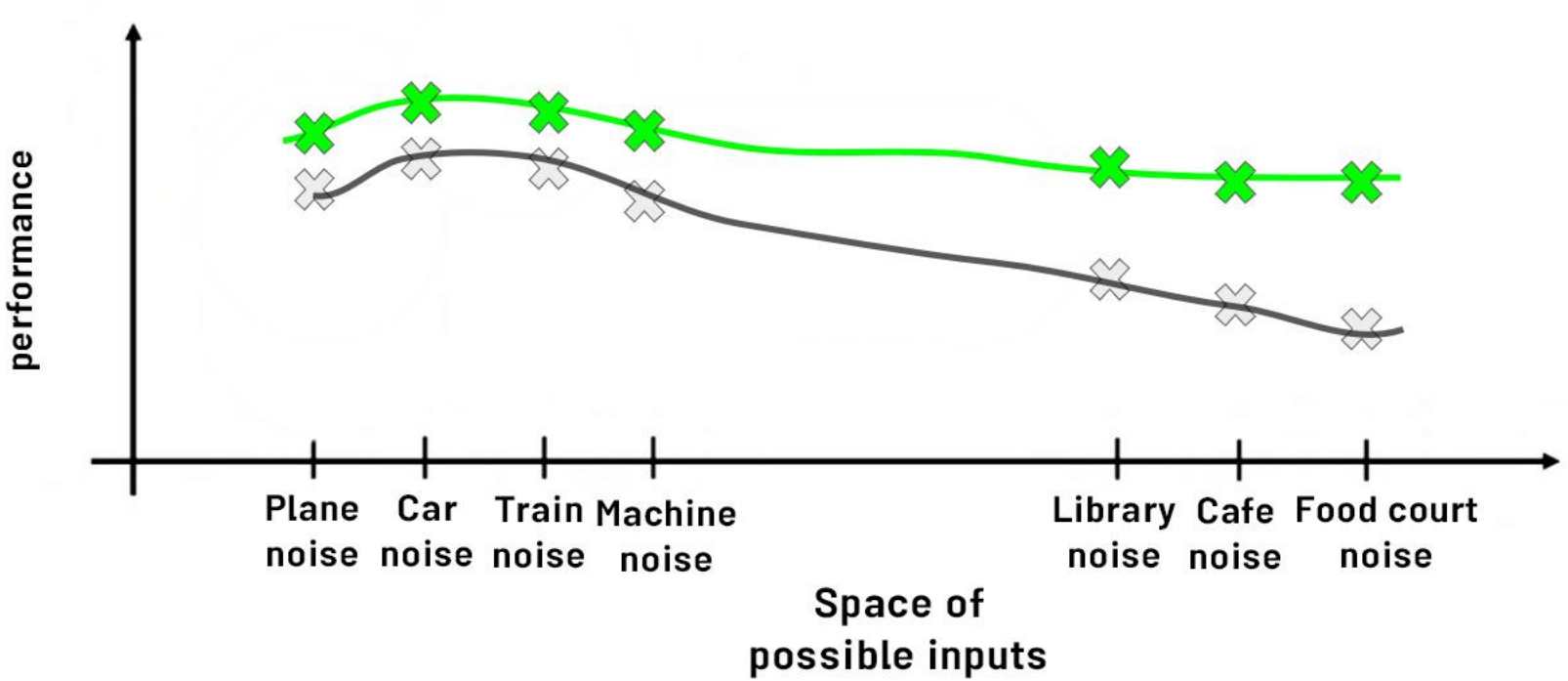
Speech Recognition example



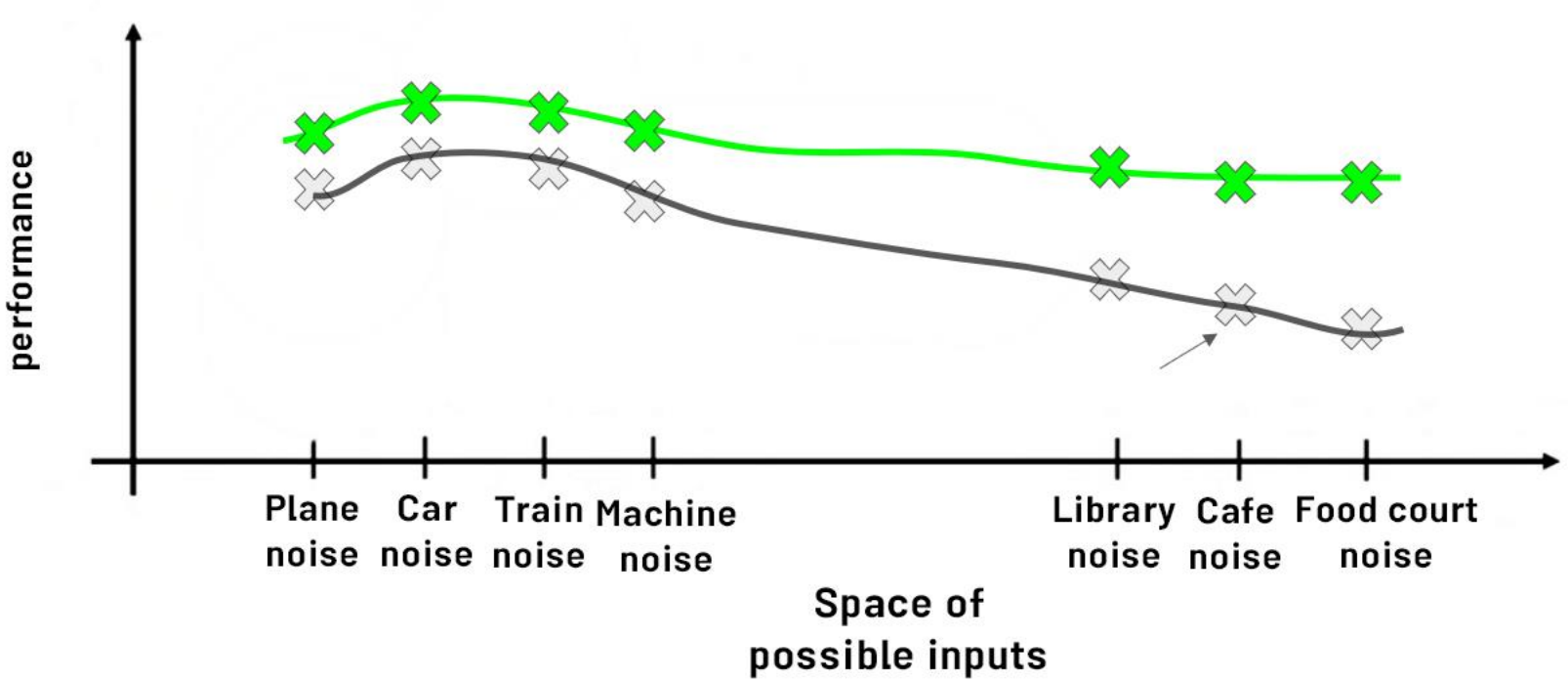
Speech Recognition example



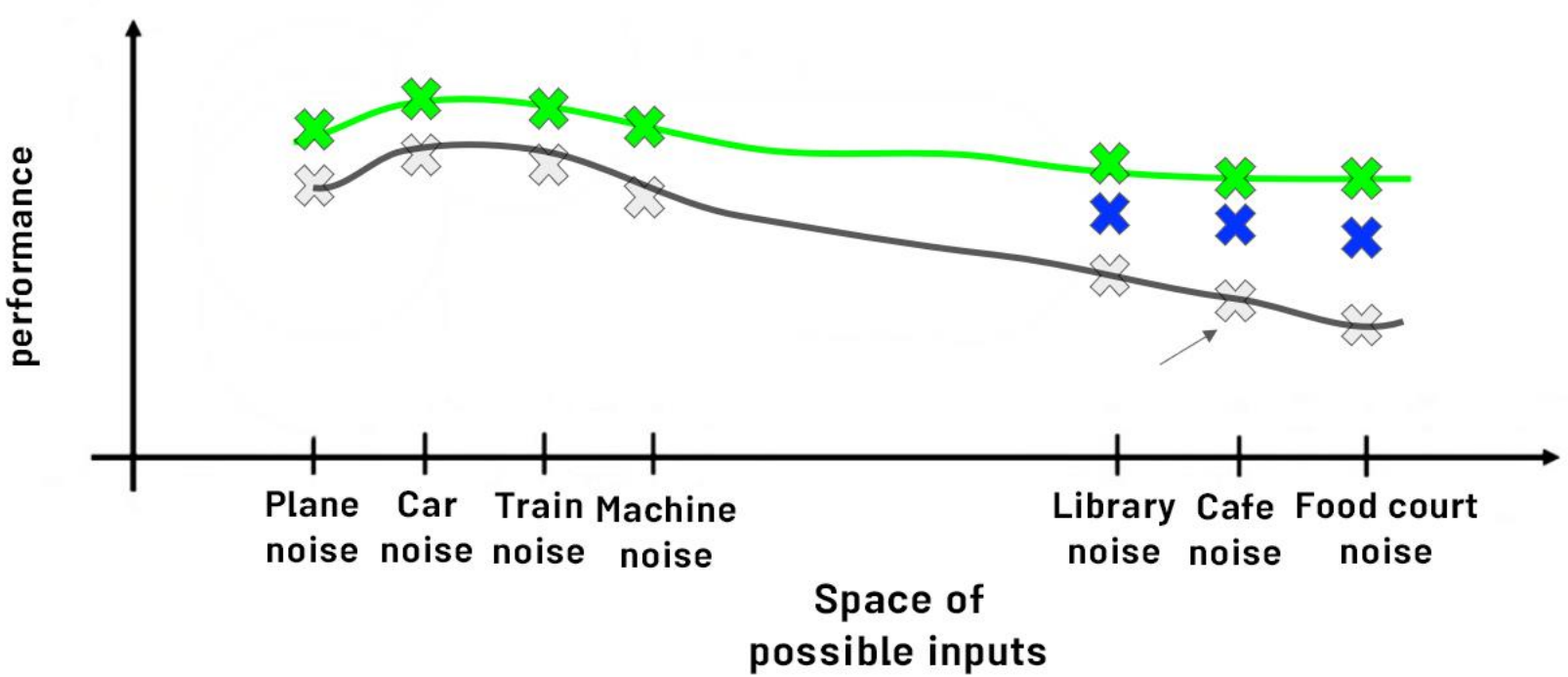
Speech Recognition example



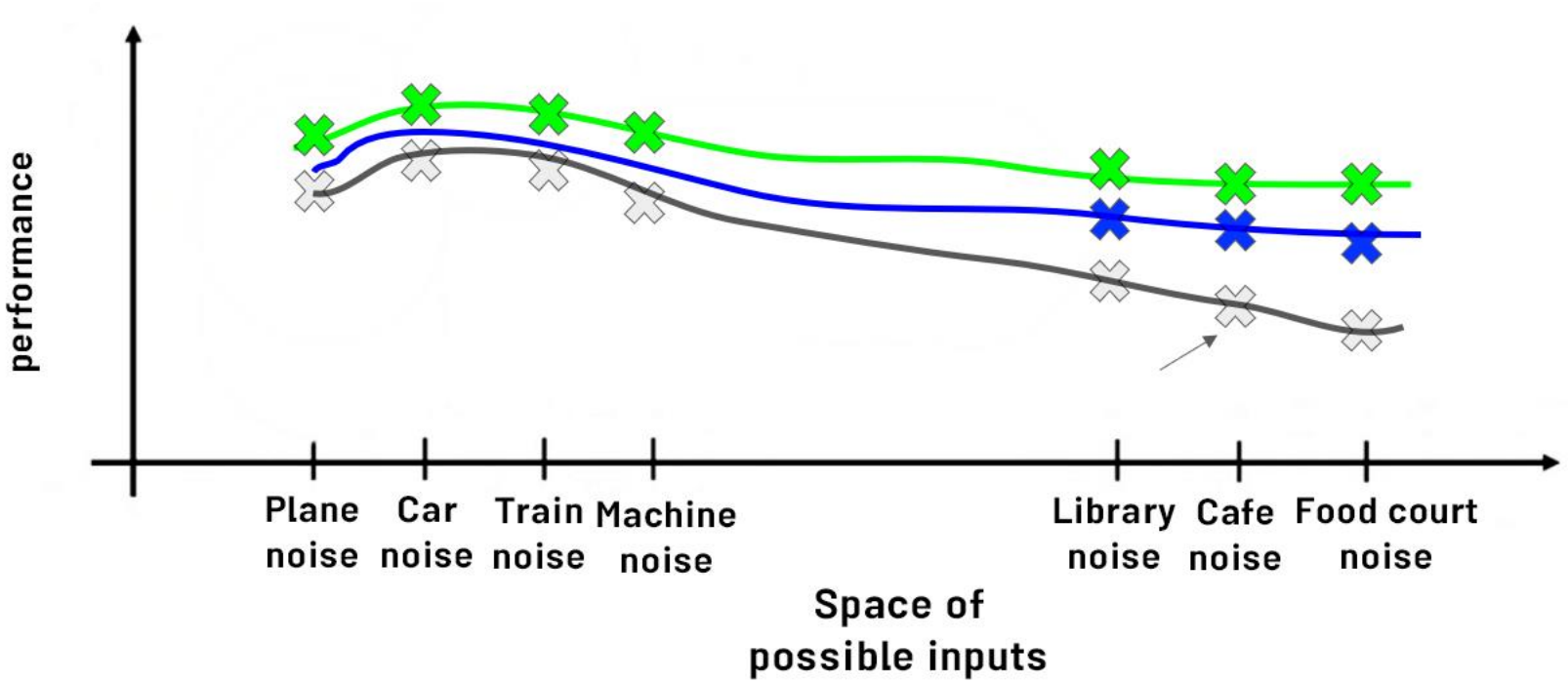
Speech Recognition example



Speech Recognition example



Speech Recognition example



Data augmentation

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?

Data augmentation

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?

Goal:

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

Image example

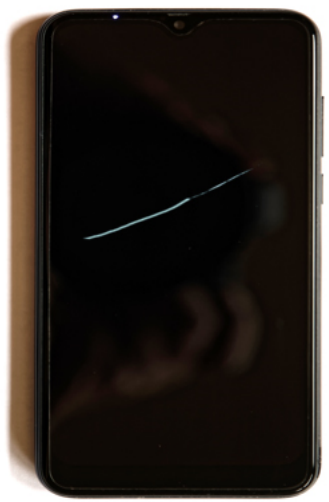


Image example

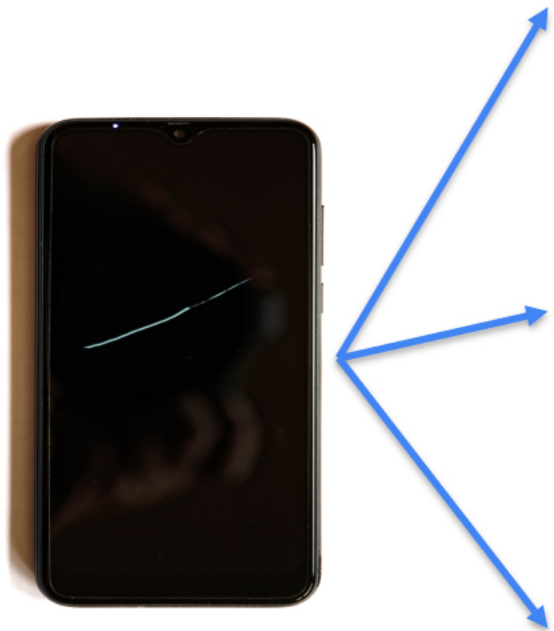


Image example

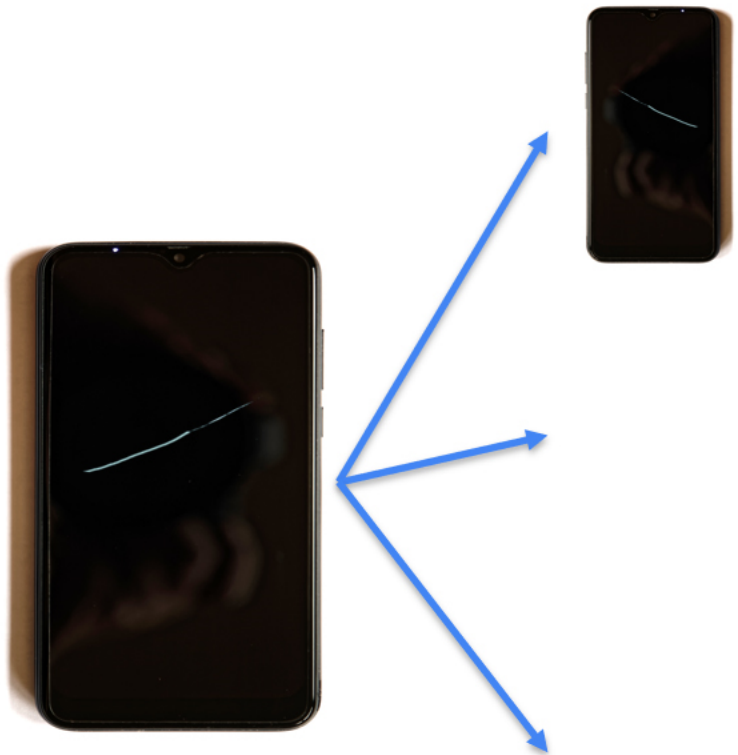


Image example

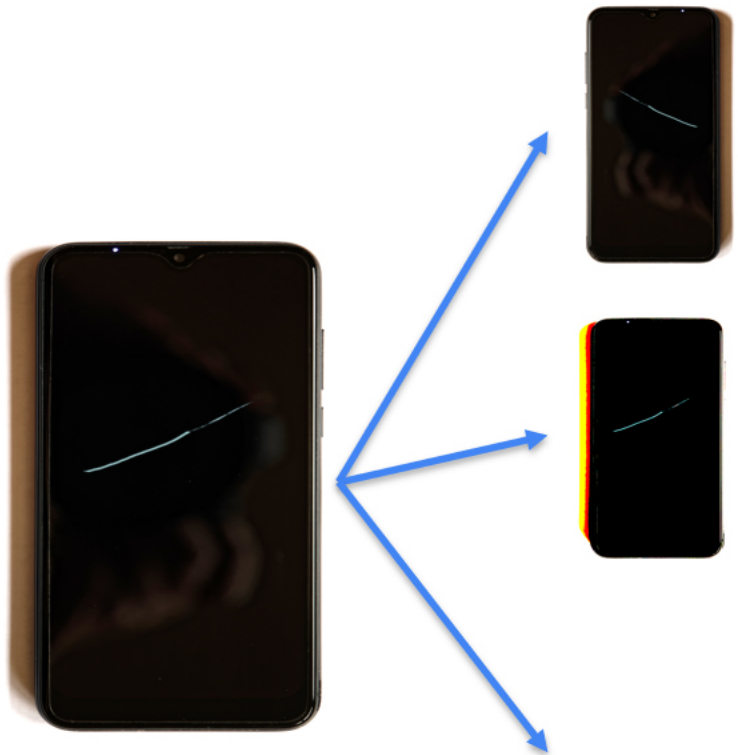


Image example

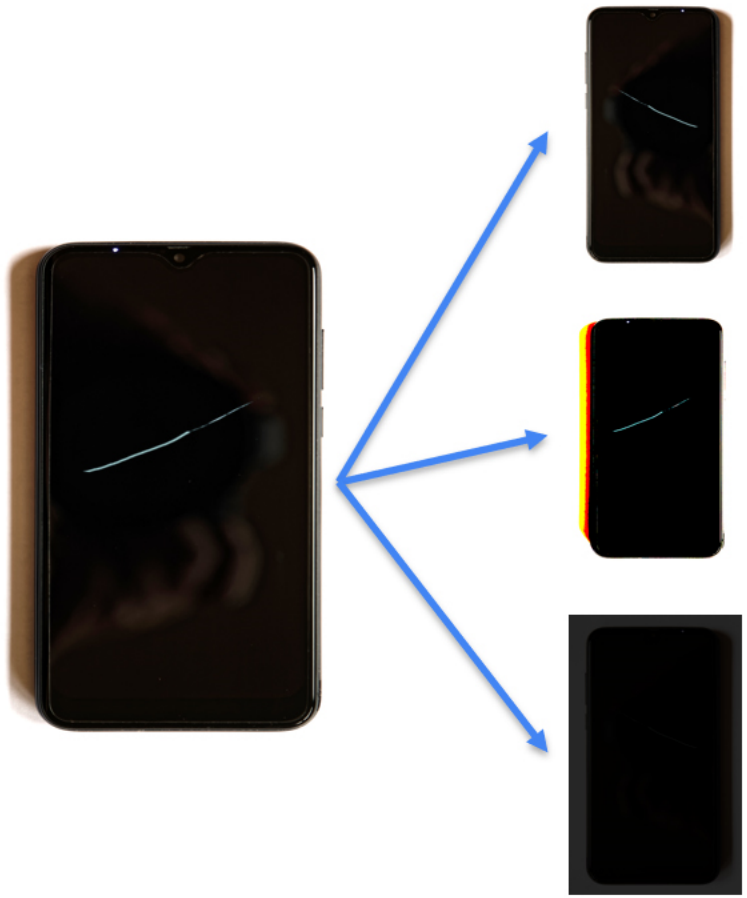
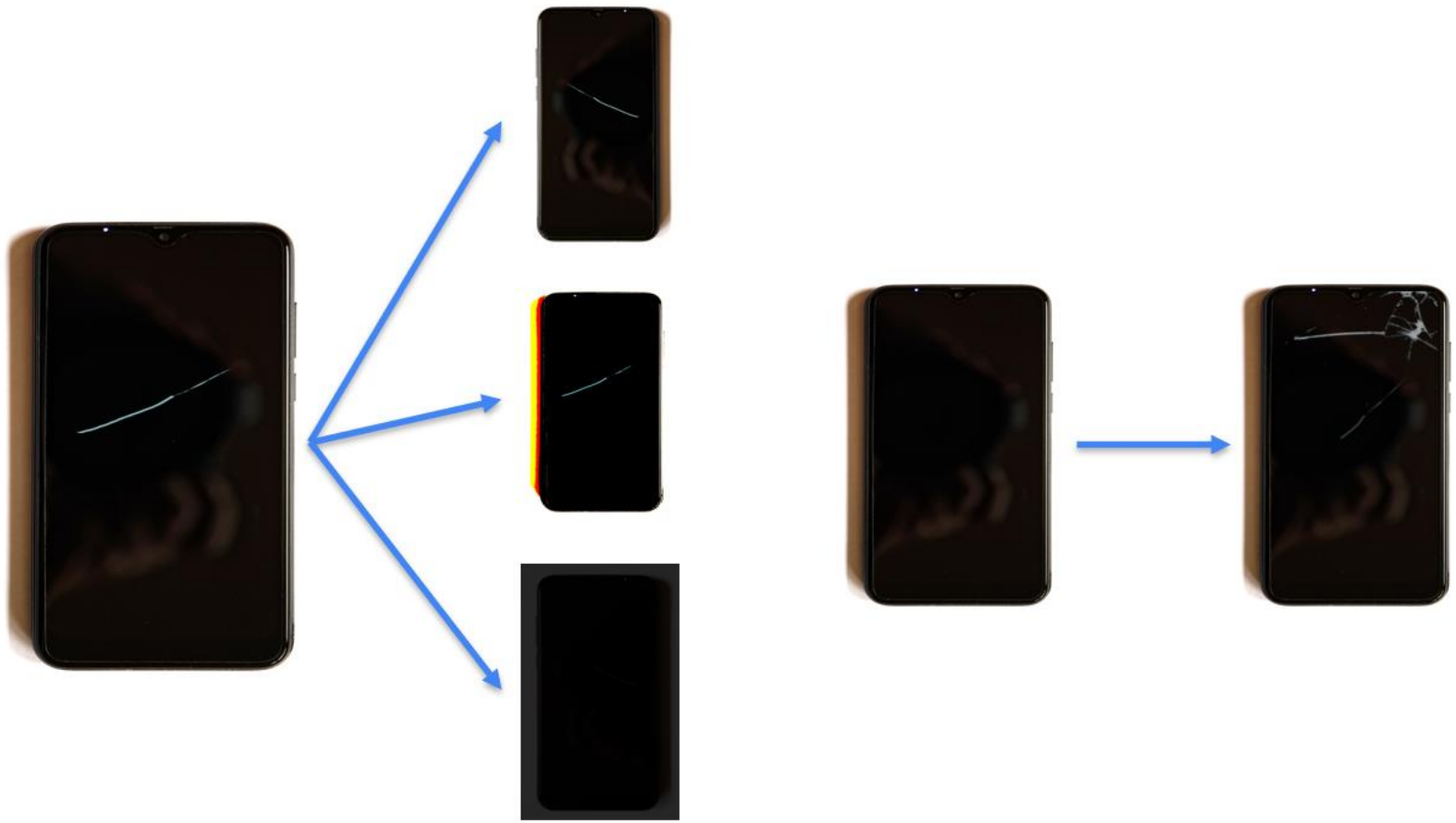


Image example



Data iteration loop

Add/improve Data

(holding model fixed)

Data iteration loop

Add/improve Data

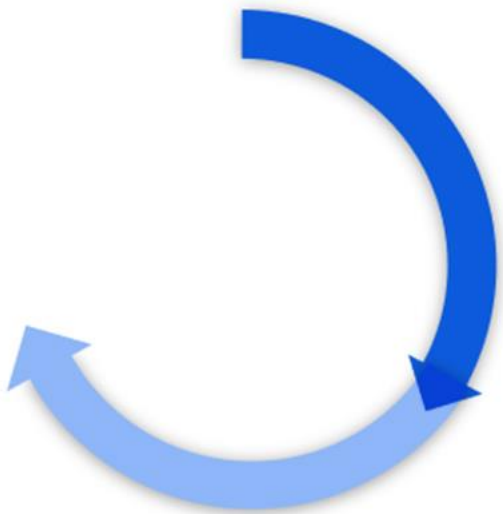
(holding model fixed)

Training

Data iteration loop

Add/improve Data

(holding model fixed)



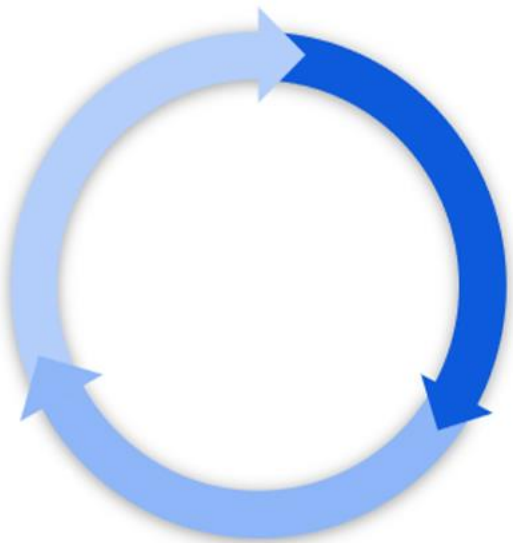
Error analysis

Training

Data iteration loop

Add/improve Data

(holding model fixed)



Error analysis

Training

Can adding data hurt performance?

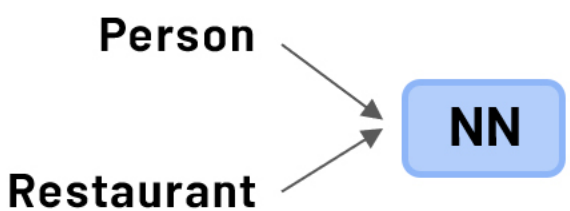
- For unstructured data problems, if:
  - The model is large (low bias).
  - The mapping X→Y is clear (e.g., humans can make accurate predictions).
- Then, adding data rarely hurts accuracy.

Corner case

?

Adding features for 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)?

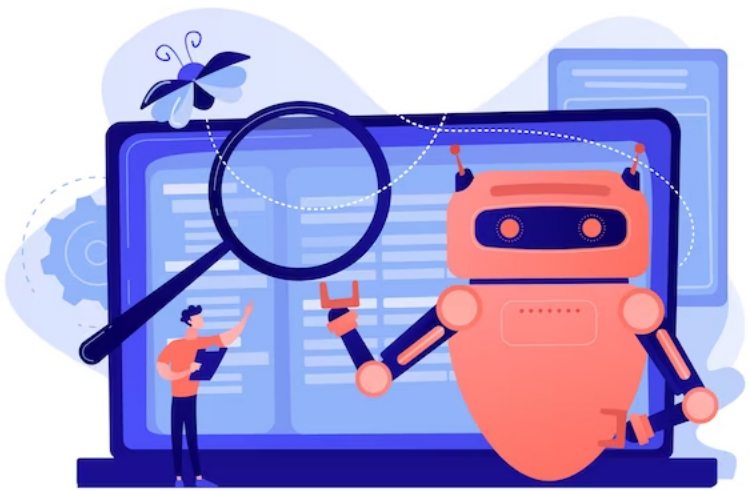


Other food delivery examples

- Only tea/coffee
- Only pizza
- What are the added signals (features) that can help make a decision?

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

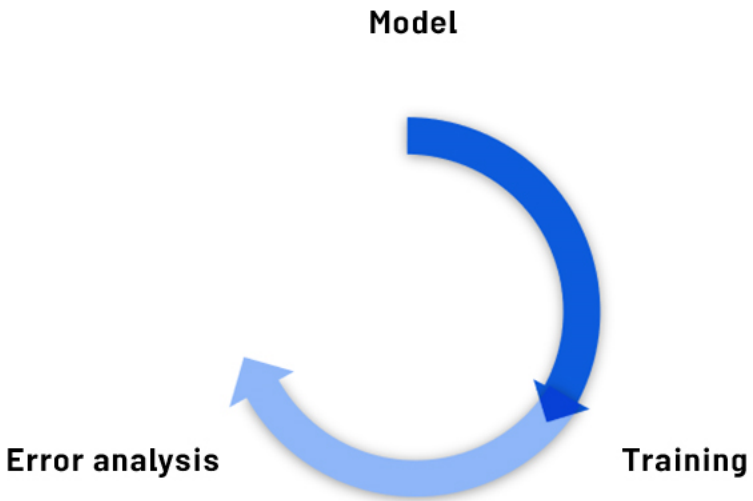
Model

Data iteration

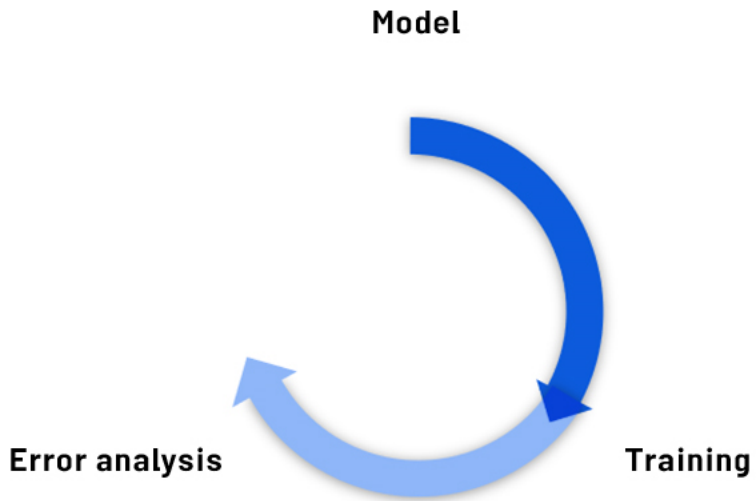
Model

Training

Data iteration

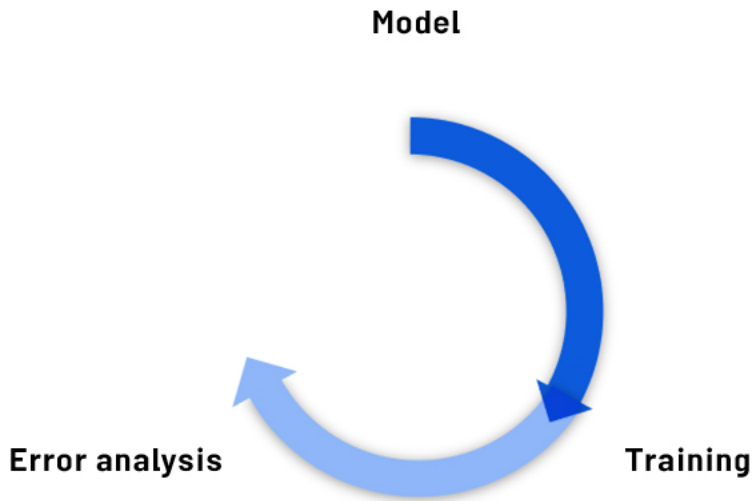


Data iteration



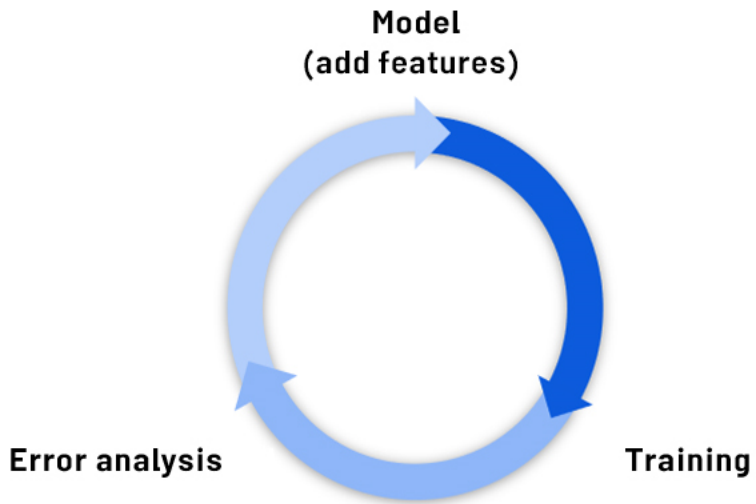
- Error analysis can be harder if there is no good baseline (such as HLP) to compare to.

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



- 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.

Experiment tracking

What to track?

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

Experiment tracking

What to track?

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

Tracking tools

- Text files
- Spreadsheet
- Experiment tracking system

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

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