INTRODUCTION TO MACHINE LEARNING

Thomas Tiam-Lee, PhD

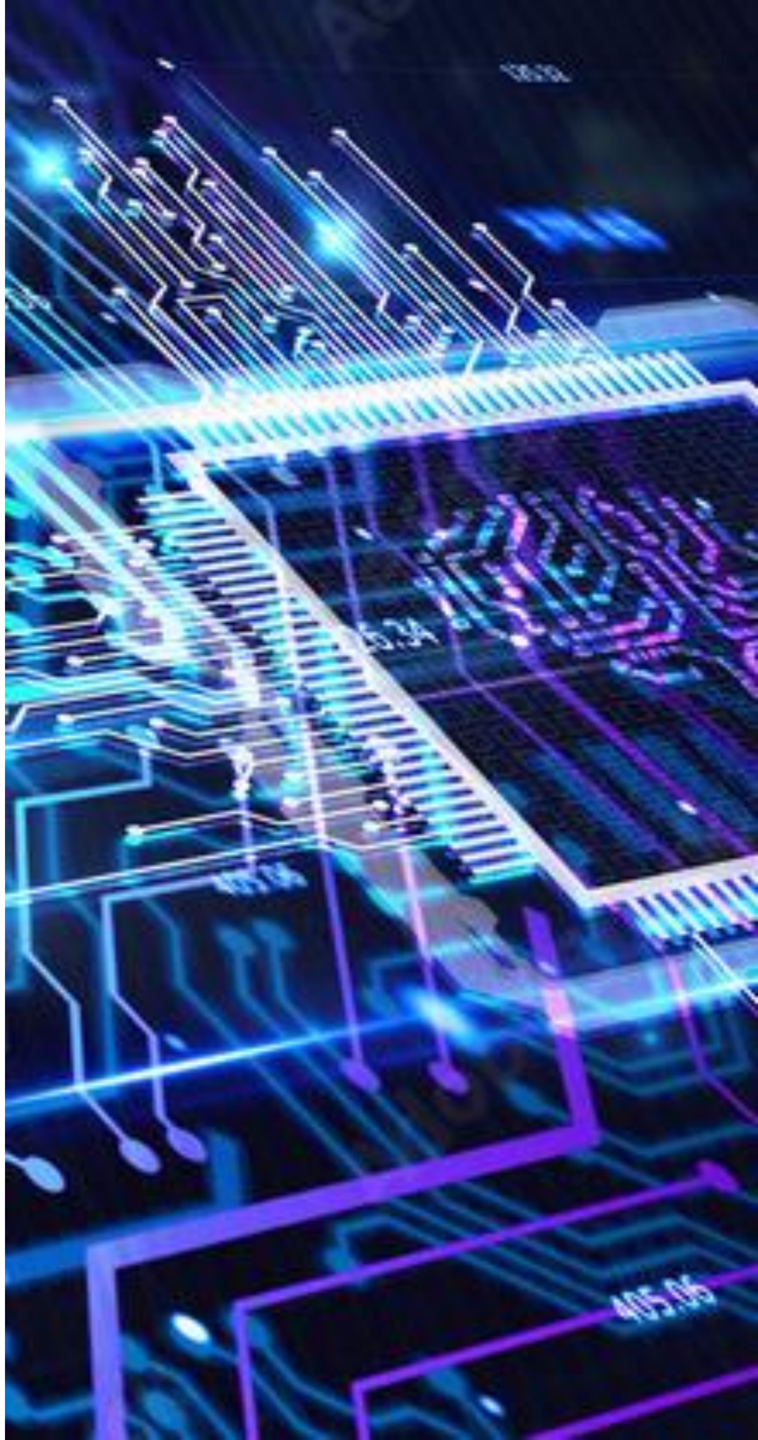
De La Salle University Manila

College of Computer Studies

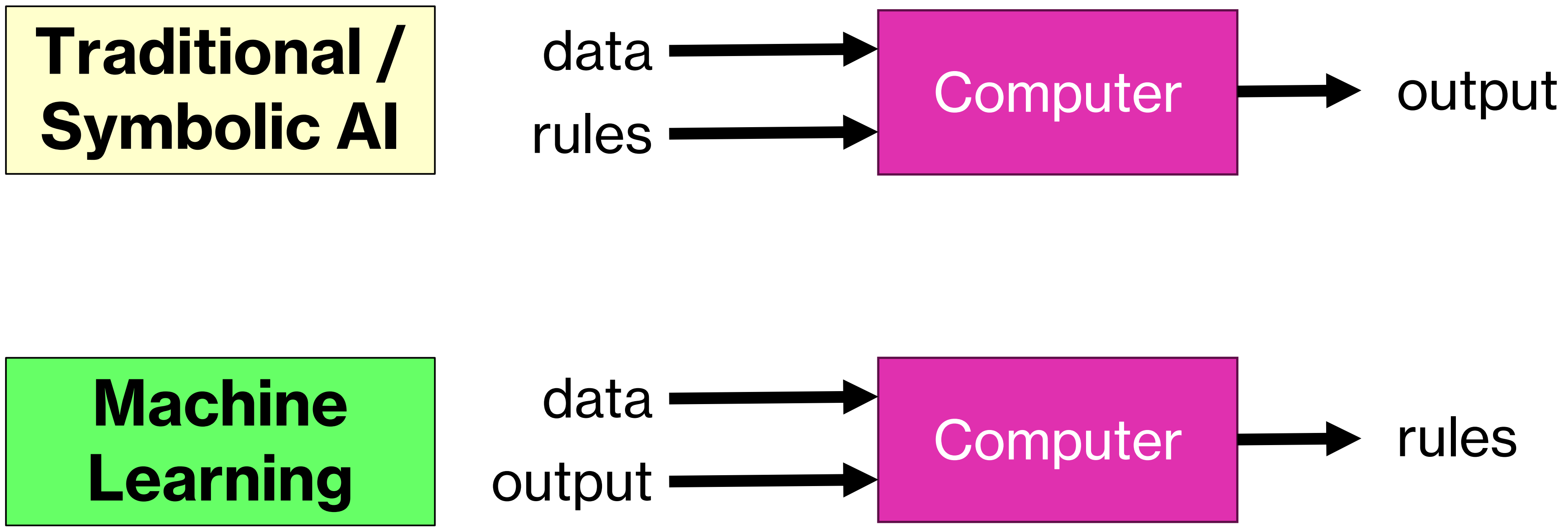
Machine Learning

- Part of "**neural AI**" school of thought
  - Bottom-up approach

- Top-down approach
  - Model the rules of the problem, then solve it.
- **Bottom-up approach**
  - **Show many examples (data), let the model learn the rules of the problem from it**



Traditional AI vs. Machine Learning



Traditional AI Mindset

- **Task:** Classify if an animal is quadruped.
  - Identify the rules.
  - Implement the rules.

Name	Class	Legs	Size
Horse	Mammal	4	Medium
Ram	Mammal	4	Small
Man	Mammal	2	Medium
Chicken	Bird	2	Small

+

if number of legs
is equal to 4,
quadruped

Quadruped?
Yes
Yes
No
No

Machine Learning Mindset

- **Task:** Classify if an animal is quadruped.
  - Feed many examples of quadruped vs. non-quadruped animals
  - Infer the rules on when an animal is quadruped

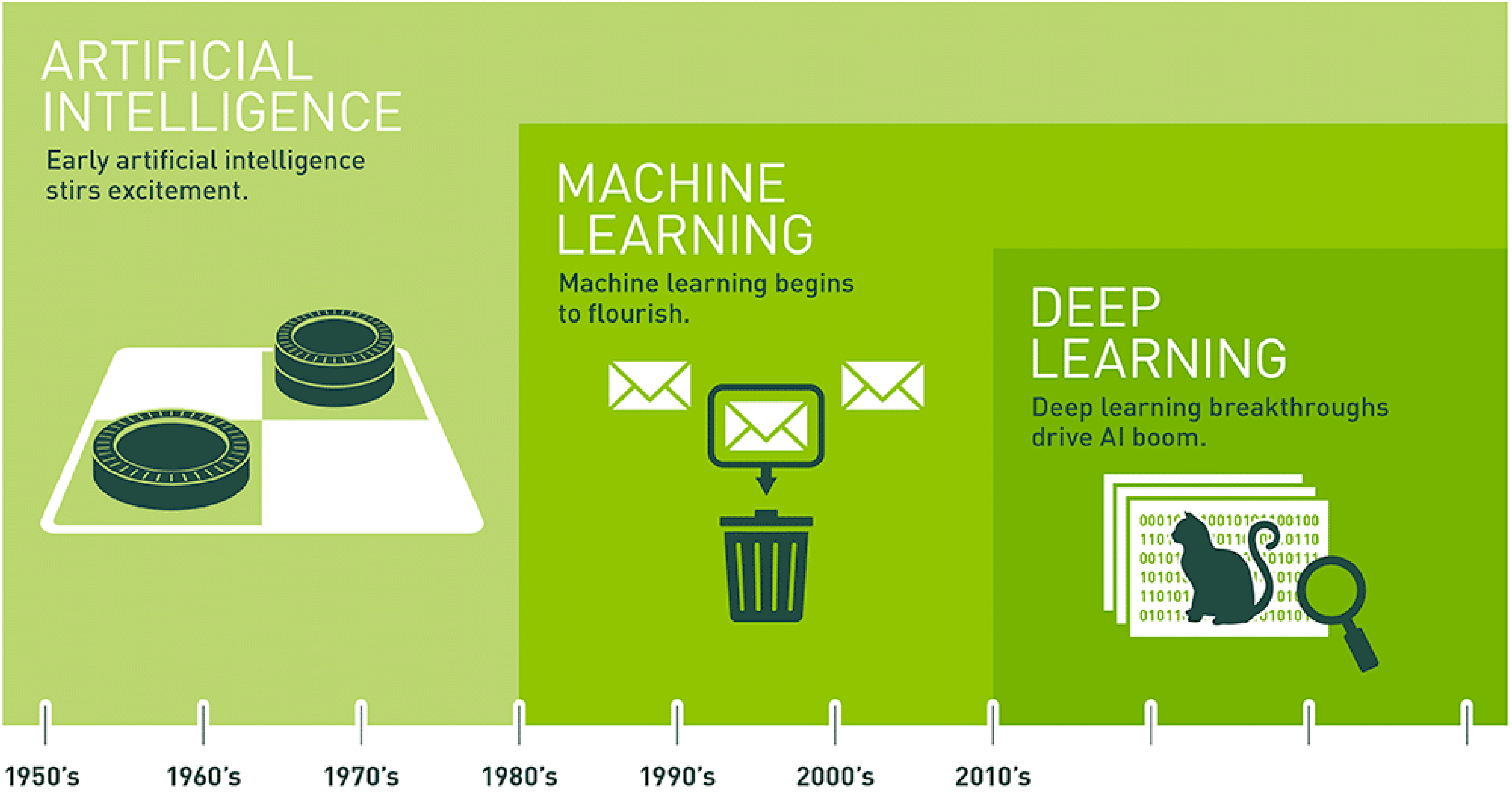
Name	Class	Legs	Size
Horse	Mammal	4	Medium
Ram	Mammal	4	Small
Man	Mammal	2	Medium
Chicken	Bird	2	Small

+

Quadruped?
Yes
Yes
No
No

→

if number of legs is equal to 4, **quadruped**

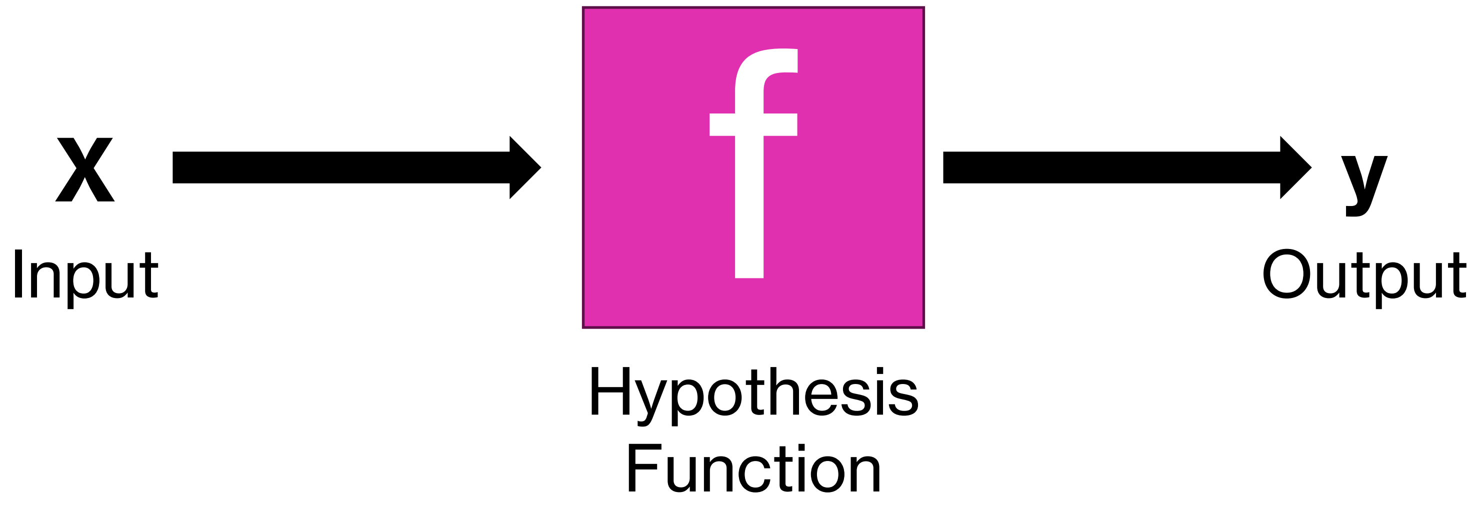


Since an early flush of optimism in the 1950s, smaller subsets of artificial intelligence – first machine learning, then deep learning, a subset of machine learning – have created ever larger disruptions.

Source: https://sitn.hms.harvard.edu/flash/2017/history-artificial-intelligence/

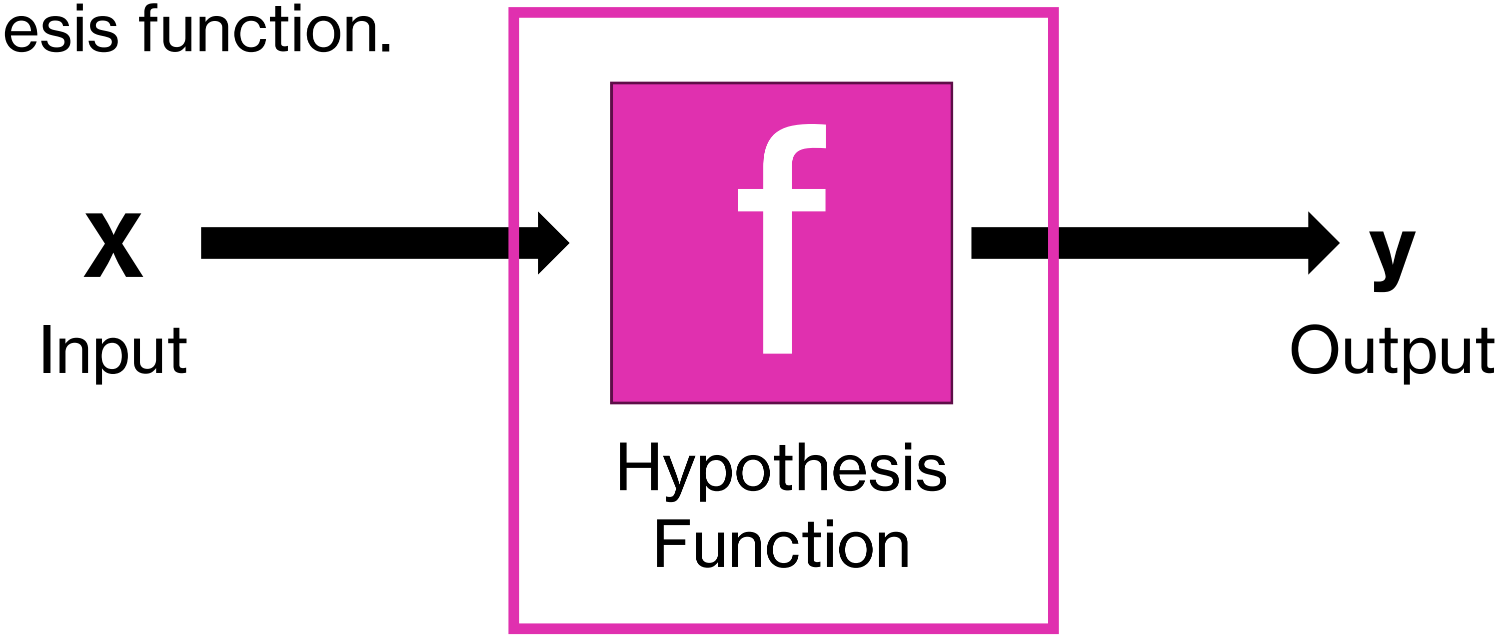
Machine Learning

- Machine learning models are **reflex-based models**.
- They take an input, process it one time, and produce an output.

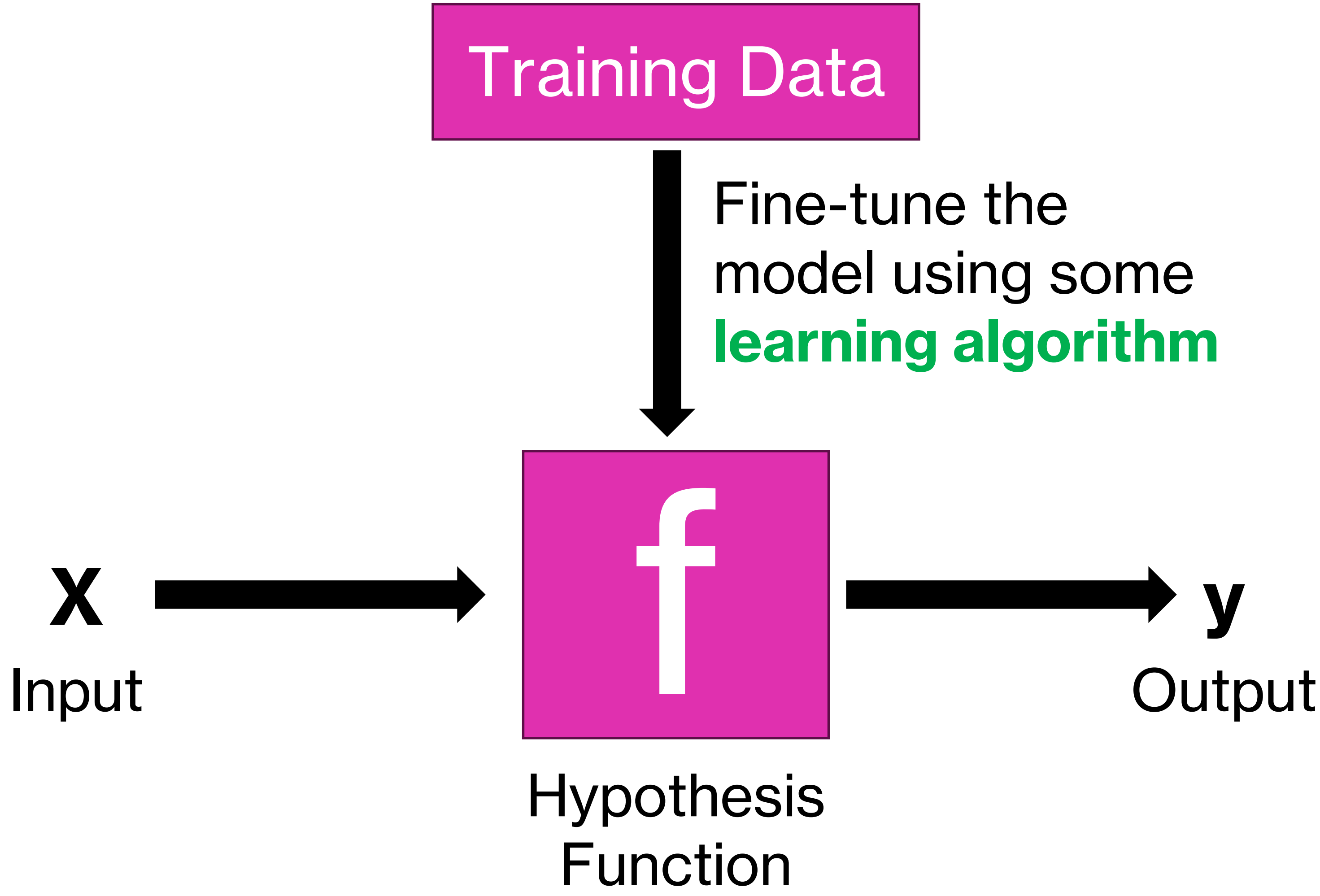


Hypothesis Function

- Where the “magic” happens.
- Transforms the input X to the output y.
- Many different types of ML models that can serve as the hypothesis function.



Machine Learning Framework



Terms

- **Dataset:** collection of data instances where the model will “learn” from.

	index	sepallength	sepalwidth	petallength	petalwidth	class
0	0	5.0	3.2	1.2	0.2	Iris-setosa
1	1	6.7	3.1	4.4	1.4	Iris-versicolor
2	2	5.7	2.8	4.5	1.3	Iris-versicolor
3	3	7.7	3.0	6.1	2.3	Iris-virginica
4	4	6.7	3.1	5.6	2.4	Iris-virginica
5	5	4.7	3.2	1.3	0.2	Iris-setosa
6	6	6.4	2.7	5.3	1.9	Iris-virginica

Terms

- **Instance:** A single object / row in the dataset.

	index	sepallength	sepalwidth	petallength	petalwidth	class
0	0	5.0	3.2	1.2	0.2	Iris-setosa
1	1	6.7	3.1	4.4	1.4	Iris-versicolor
2	2	5.7	2.8	4.5	1.3	Iris-versicolor
3	3	7.7	3.0	6.1	2.3	Iris-virginica
4	4	6.7	3.1	5.6	2.4	Iris-virginica
5	5	4.7	3.2	1.3	0.2	Iris-setosa
6	6	6.4	2.7	5.3	1.9	Iris-virginica

Terms

- **Label:** The target variable that is being predicted, i.e., the model is "learning" how to predict this variable.

	index	sepallength	sepalwidth	petallength	petalwidth	class
0	0	5.0	3.2	1.2	0.2	Iris-setosa
1	1	6.7	3.1	4.4	1.4	Iris-versicolor
2	2	5.7	2.8	4.5	1.3	Iris-versicolor
3	3	7.7	3.0	6.1	2.3	Iris-virginica
4	4	6.7	3.1	5.6	2.4	Iris-virginica
5	5	4.7	3.2	1.3	0.2	Iris-setosa
6	6	6.4	2.7	5.3	1.9	Iris-virginica

Terms

- **Classes:** The list of possible values for the label.
  - In this case: **{Iris-setosa, Iris-versicolor, Iris-virginica}**

	index	sepallength	sepalwidth	petallength	petalwidth	class
0	0	5.0	3.2	1.2	0.2	Iris-setosa
1	1	6.7	3.1	4.4	1.4	Iris-versicolor
2	2	5.7	2.8	4.5	1.3	Iris-versicolor
3	3	7.7	3.0	6.1	2.3	Iris-virginica
4	4	6.7	3.1	5.6	2.4	Iris-virginica
5	5	4.7	3.2	1.3	0.2	Iris-setosa
6	6	6.4	2.7	5.3	1.9	Iris-virginica
7	7	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]

Terms

- **Features:** The variables that will be considered when “learning” the rules / making the prediction.

	index	sepallength	sepalwidth	petallength	petalwidth	class
0	0	5.0	3.2	1.2	0.2	Iris-setosa
1	1	6.7	3.1	4.4	1.4	Iris-versicolor
2	2	5.7	2.8	4.5	1.3	Iris-versicolor
3	3	7.7	3.0	6.1	2.3	Iris-virginica
4	4	6.7	3.1	5.6	2.4	Iris-virginica
5	5	4.7	3.2	1.3	0.2	Iris-setosa
6	6	6.4	2.7	5.3	1.9	Iris-virginica

Classification of ML Algorithms

1. Supervised Machine Learning Algorithms
2. Unsupervised Machine Learning Algorithms
3. Reinforcement Learning Algorithms

Supervised Machine Learning

- There is a **label**.
  - The label is the “correct answer” for that instance.
  - The goal is to train the model to predict the target label.
- **Examples:**
  - *Given the dimensions of the different parts of a flower, classify it as either **iris-setosa**, **iris-virginica**, and **iris-versicolor**.*
  - *Given the lot area and the number of bedrooms of a house, predict its **price**.*
  - *Given an image of a handwritten digit, predict what **digit** it is.*

Unsupervised Machine Learning

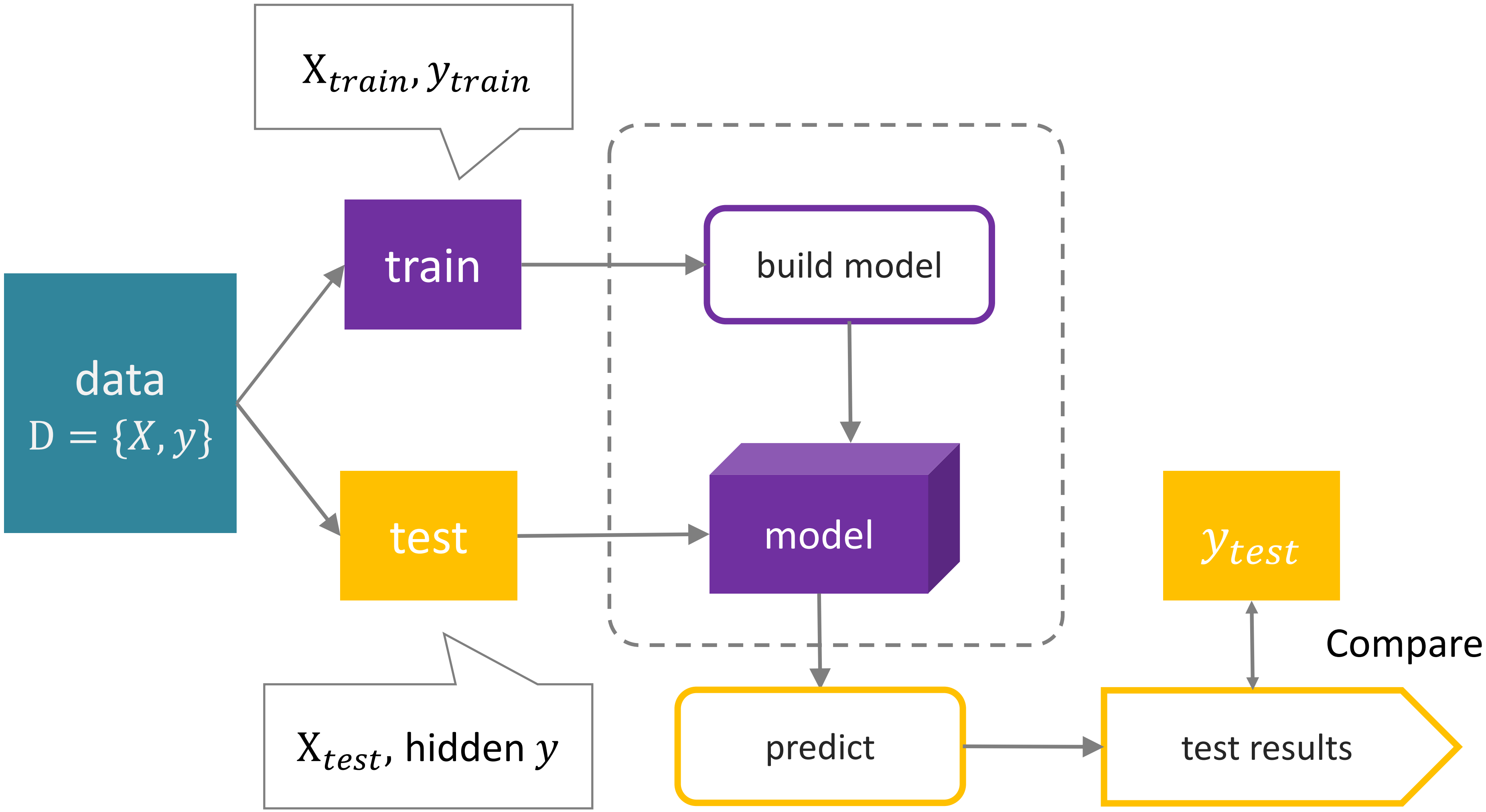
- There is no **label**.
  - You just want to feed the data, and gain some insights about those data without any particular target variable in mind.
- **Examples:**
  - *Given customer purchase histories in a website, create a clustering of similar customers*
    - This type of clustering has applications in recommender systems
  - *Detecting anomalies in server usage logs over time*

Types of Supervised Learning

- **Regression**: prediction of a numerical variable
  - *Predict the **lifespan** of a person given data about their lifestyle*
- **Classification**: prediction of a categorical variable
  - **Binary Classification:** only “yes” or “no”
    - *Given an email, predict if it is **spam** or **not spam***
  - **Multi-class Classification:** more than 2 possible classes
    - *Given a photo of a person, predict if he is **Filipino**, **Indonesian**, or **Malaysian**.*
  - **Structured Prediction:** complex output
    - *Given an image, output a **segmentation** of the image*

Basic Supervised Machine Learning Pipeline

- Collect data.
- Preprocess data (exploratory data analysis, cleaning, etc.)
- Identify features and label.
- Split data into training set and test set.
- Build and fine-tune model from the training set.
- Run the test set on the model to measure its performance.
- Iterate as needed.



X_{train}, y_{train}
data
$D = \{X, y\}$
train
test
build model
model
y_{test}
Compare with y_{test}
X_{test}, hidden y
predict
test results

Training and Test Sets

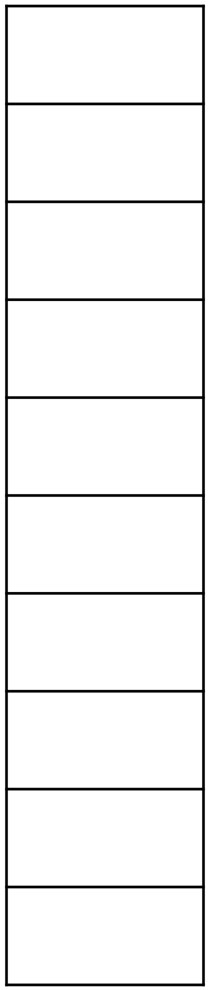
- Test set must be separated to **eliminate bias** in evaluating the performance of the model
- **Key idea:** the model must perform well even on data that it has not yet seen before.

- Amount to set aside for the test set depends on a case-to-case basis. Common splits are 10%, 20%, 30%.

K-Fold Cross Validation

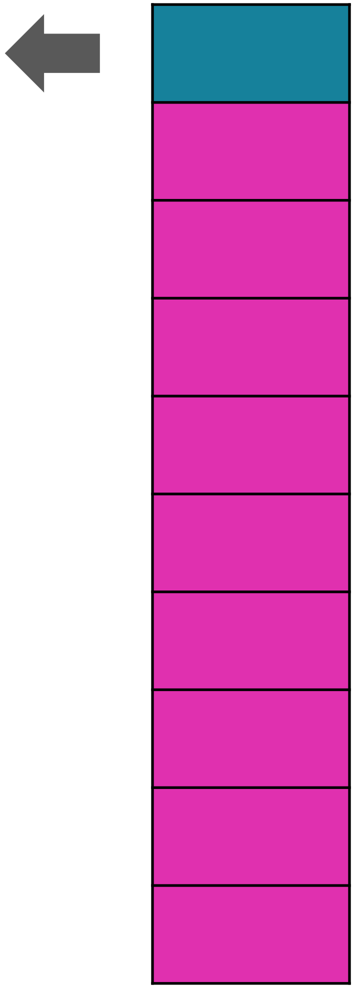
K-Fold Cross Validation

- Given $k = 10$:
- Divide the dataset into 10 parts.



K-Fold Cross Validation

Split 1 performance



- Given $k = 10$:
- Divide the dataset into 10 parts.

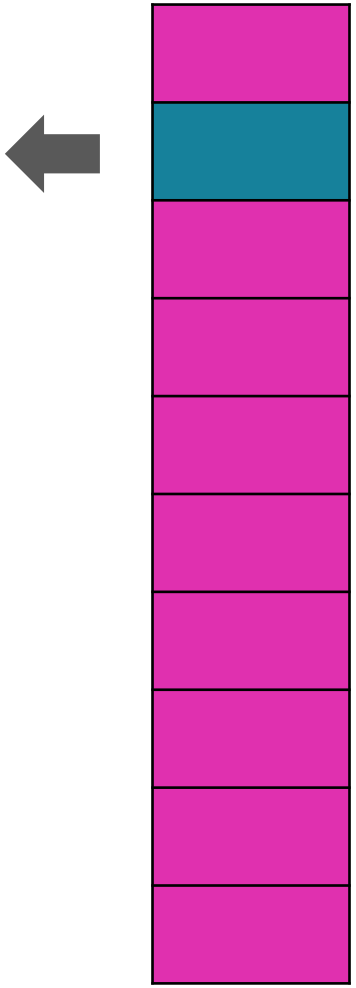
■ Training Set
■ Test Set

K-Fold Cross Validation

- Given $k = 10$:
- Divide the dataset into 10 parts.

Split 1 performance

Split 2 performance



■ Training Set
■ Test Set

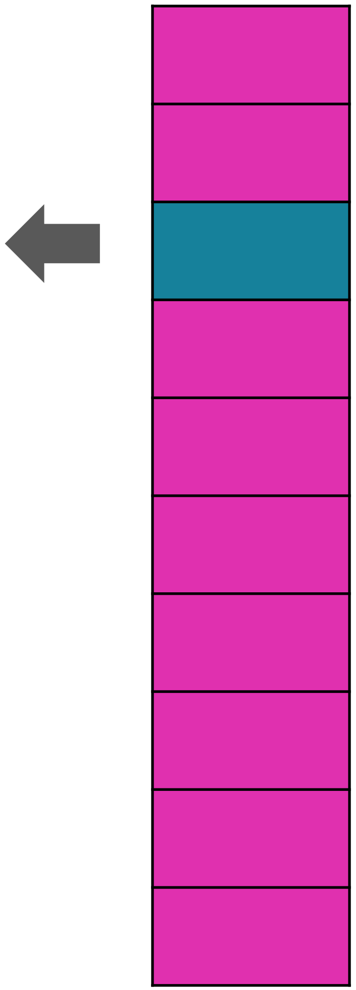
K-Fold Cross Validation

- Given $k = 10$:
- Divide the dataset into 10 parts.

Split 1 performance

Split 2 performance

Split 3 performance



■ Training Set
■ Test Set

K-Fold Cross Validation

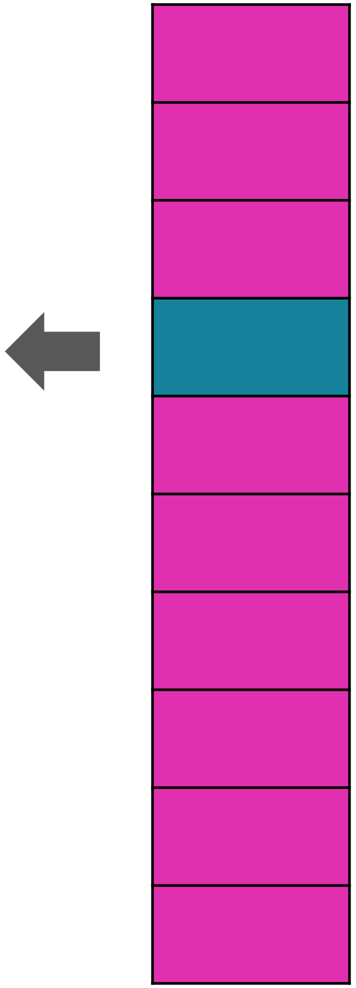
- Given $k = 10$:
- Divide the dataset into 10 parts.

Split 1 performance

Split 2 performance

Split 3 performance

Split 4 performance



■ Training Set
■ Test Set

K-Fold Cross Validation

- Given $k = 10$:
- Divide the dataset into 10 parts.

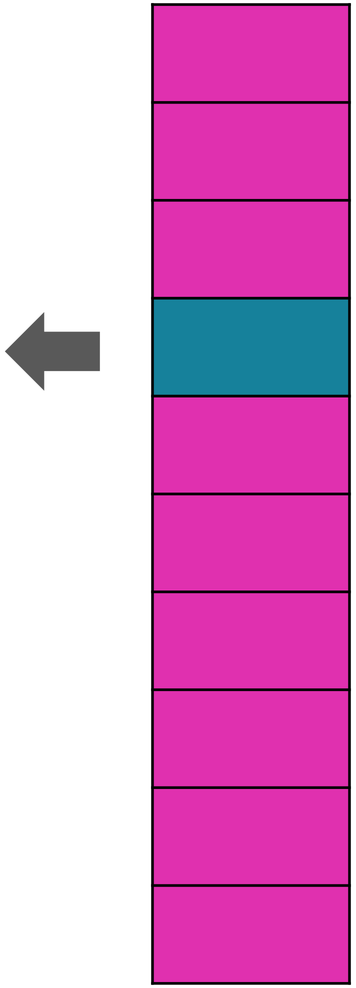
Split 1 performance

Split 2 performance

Split 3 performance

Split 4 performance

and so on until...

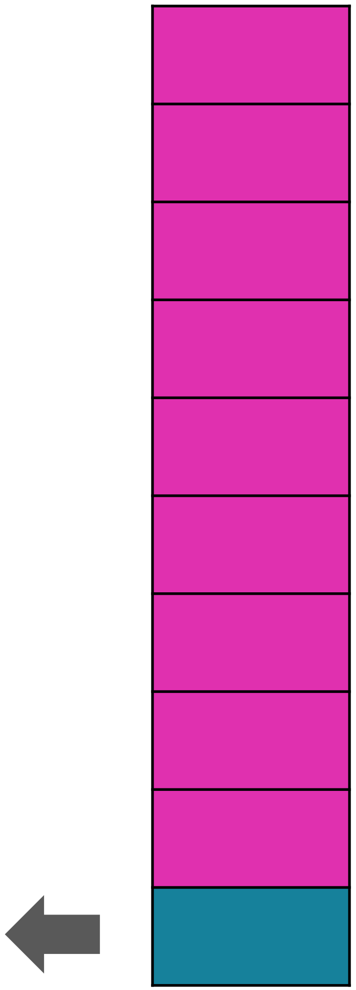


■ Training Set
■ Test Set

K-Fold Cross Validation

- Given $k = 10$:
- Divide the dataset into 10 parts.

Split 1 performance
Split 2 performance
Split 3 performance
Split 4 performance
Split 5 performance
Split 6 performance
Split 7 performance
Split 8 performance
Split 9 performance
Split 10 performance



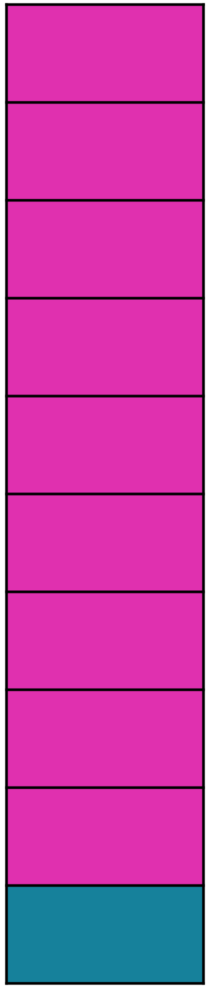
■ Training Set
■ Test Set

K-Fold Cross Validation

- Given $k = 10$:
- Divide the dataset into 10 parts.

Aggregate / get the average

Split 1 performance
Split 2 performance
Split 3 performance
Split 4 performance
Split 5 performance
Split 6 performance
Split 7 performance
Split 8 performance
Split 9 performance
Split 10 performance



■ Training Set
■ Test Set

ML Models

- The machine learning model determines **what it can predict** and **how it makes the prediction**.
- **Lingering questions:**
  - What does a machine learning model look like?
  - How can a model “learn” or be fined-tuned from training data? (learning algorithm)

Acknowledgments

- Stanford University CS221 Autumn 2021 course. Available online at: https://stanford-cs221.github.io/autumn2021
- Previous STINTSY slides by the following instructors:
  - Courtney Ngo
  - Arren Antioquia