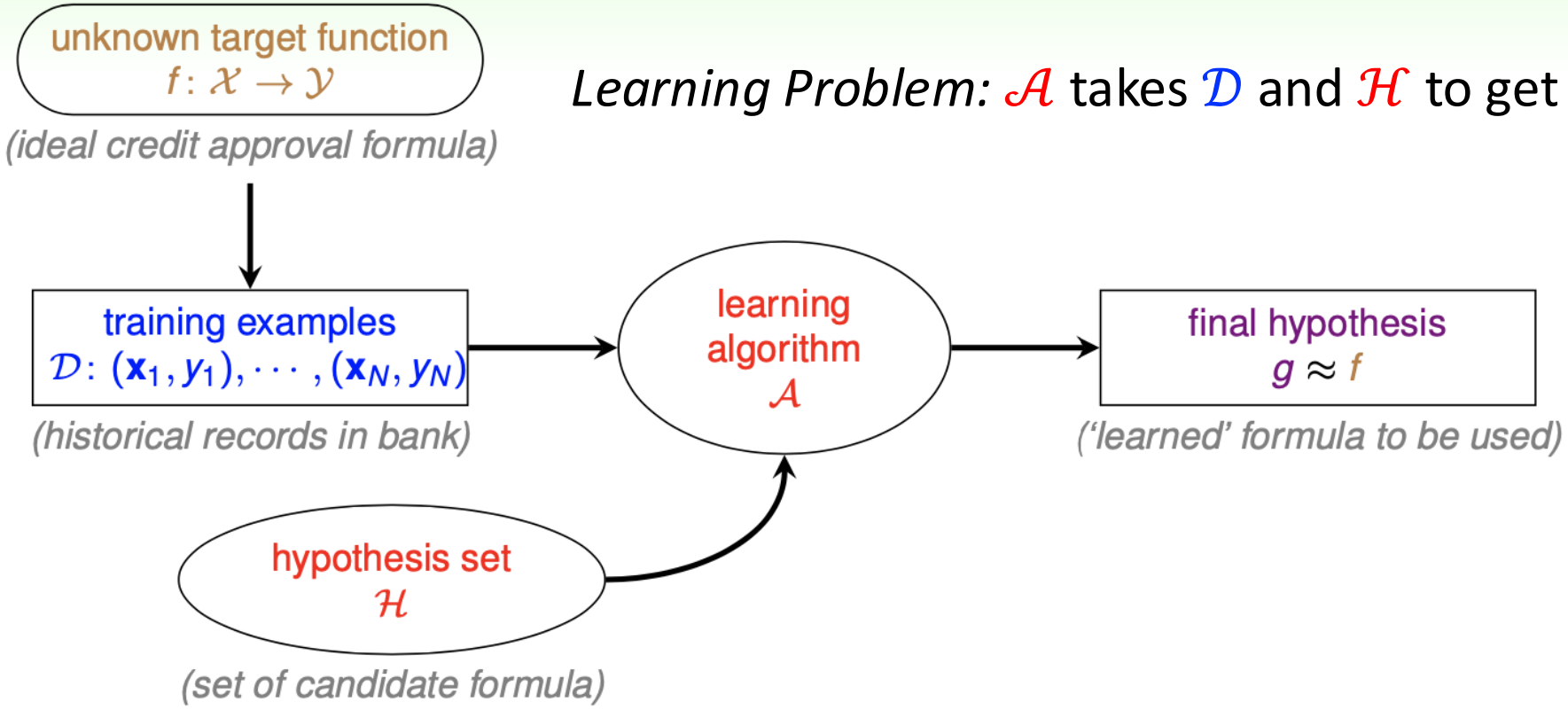
Fundamentals of Machine Learning (II)

- Recap: Learning? What do we mean?
- Is learning feasible?
- Learn the framing of supervised learning
- Know the modern machine learning landscape
- Learn the basics of Scikit-Learn

https://www.sli.do/
#ML101

Practical Definition of Machine Learning

Review



Learning Problem: $\mathcal{A}$ takes $\mathcal{D}$ and $\mathcal{H}$ to get g

machine learning:
use data to compute hypothesis g
that approximates target f

Source: Hsuan-Tien Lin, **Machine Learning Foundations**
https://www.csie.ntu.edu.tw/~htlin/mooc/

Review

Summary

Is learning feasible?

- Learning is only feasible in a *probabilistic* way (PAC: probably approximately correct) and we can **predict** something useful outside the training set $\mathcal{D}$ using only $\mathcal{D}$.
- We don't insist on using any particular probability distribution, or even on knowing what distribution is used. However, whatever distribution we use for generating the samples, we must also use when we evaluate how well g approximates the *unknown* target function f.
- The hypothesis g is not fixed ahead of time before generating the data, because which hypothesis is selected to be g depends on the data.

用 30 字解釋 PAC Learning Framework

GPT-4o

PAC（Probably Approximately Correct）學習框架是衡量演算法學習效率的理論，目標是在有限數據下達到高機率的接近正確預測。

用 30 字解釋 PAC Learning Framework

GPT-4o

PAC（Probably Approximately Correct）學習框架評估演算法在有限數據下，以高機率學習近似正確的概念，保證效率與泛化能力。

- 答案每次都不大一樣
- 但是每次的答案都可能大概對（**Probably Approximately Correct**）

Boolean Learning Example: Take Two

Let us consider a Boolean target function (i.e., $\mathcal{Y} = \{0, 1\}$) over a four-bit vector representation of input space $\{0000, 0001, \ldots, 0111, 1000, 1001, \ldots, 1111\}$.

Fun Time: For this example, what is the dimension of the input space $\mathcal{X}$? how big is the entire input space $\mathcal{X}$? how big is the entire Boolean hypothesis set $\mathcal{H}$?

(1) 4, 16, 16
(2) 4, 16, 65536
(3) 16, 16, 16
(4) 16, 16, 65536
(5) None of the above

slido

Please download and install the Slido app on all computers you use

Fun Time: For this example, what is the dimension of the input space $\mathcal{X}$? how big is the entire input space $\mathcal{X}$? how big is the entire Boolean hypothesis set $\mathcal{H}$?

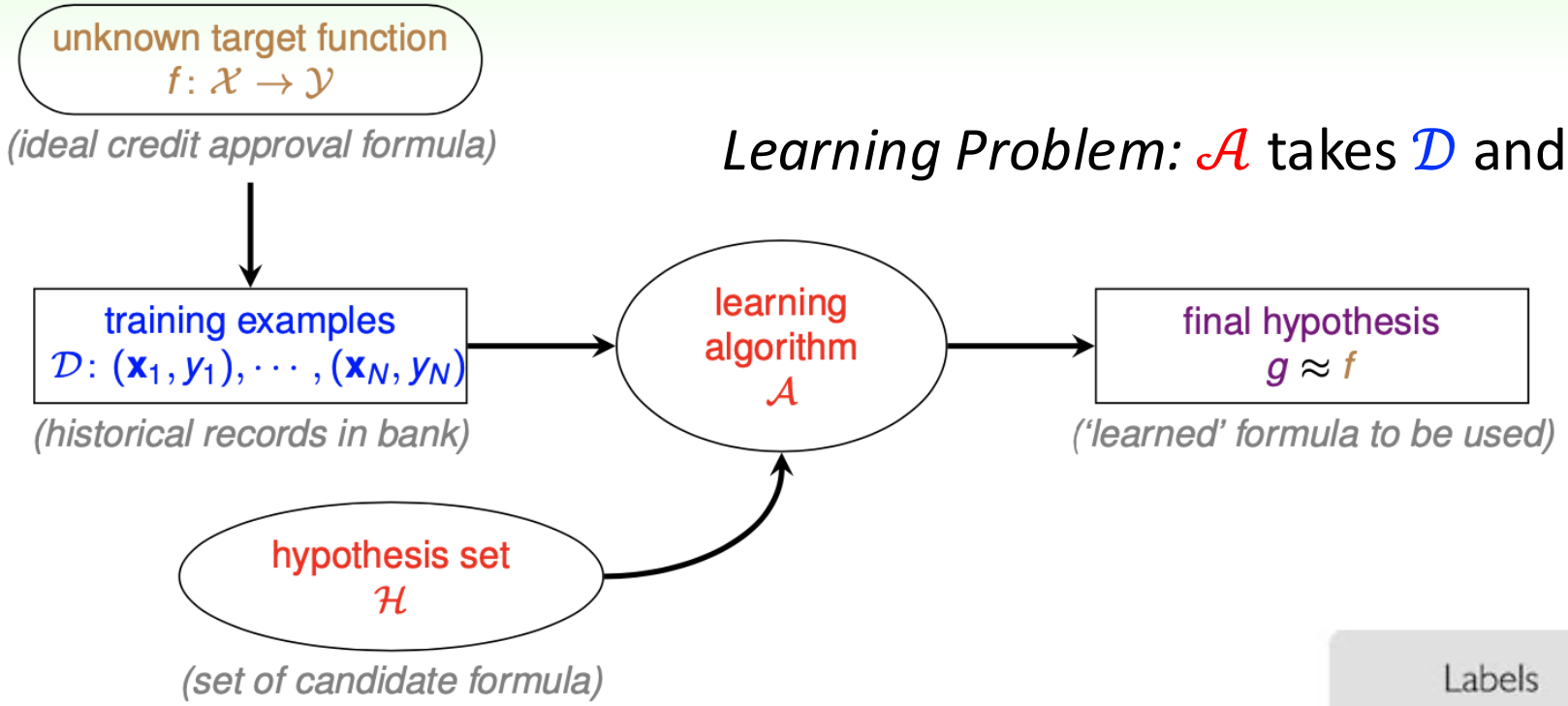
ⓘ Start presenting to display the poll results on this slide.

Boolean Learning Example: Take Two

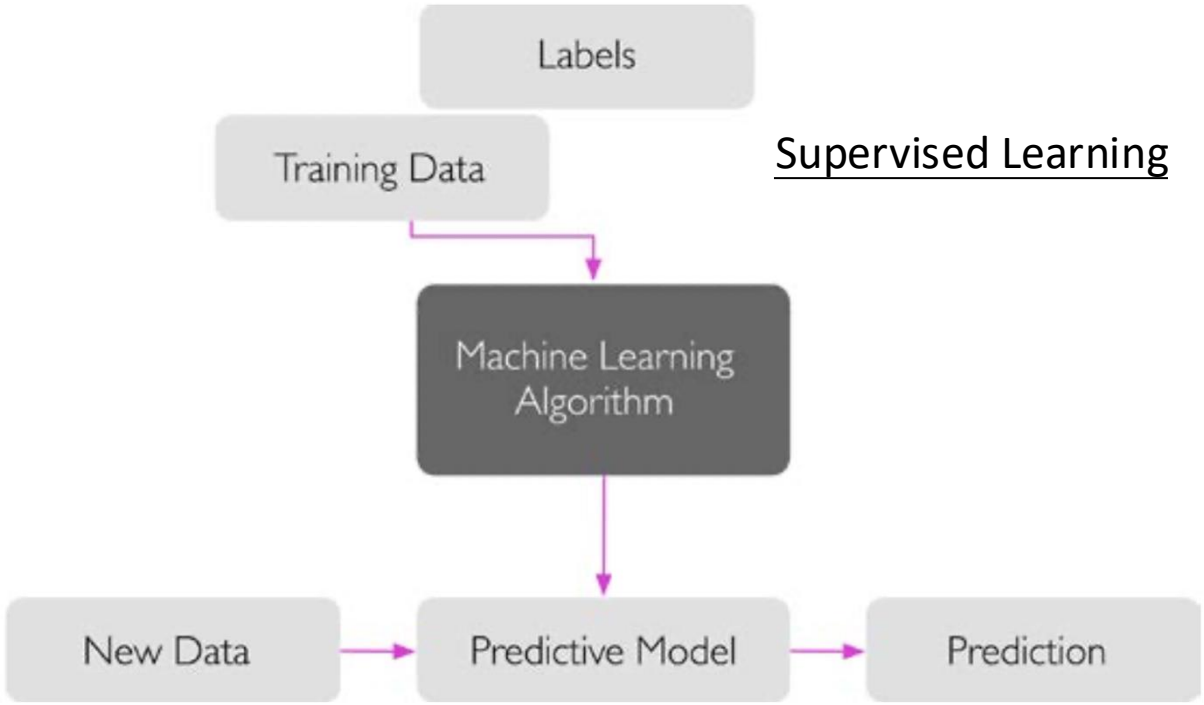
Let us consider a Boolean target function (i.e., $\mathcal{Y} = \{0, 1\}$) over a four-bit vector representation of input space $\{0000, 0001, \ldots, 0111, 1000, 1001, \ldots, 1111\}$.

See

- Boolean_Learning_Example.pdf
- Boolean_Learning_Example.ipynb



Learning Problem: $\mathcal{A}$ takes $\mathcal{D}$ and $\mathcal{H}$ to get g



Supervised Learning

Source: Hsuan-Tien Lin, **Machine Learning Foundations**
https://www.csie.ntu.edu.tw/~htlin/mooc/

Summary

Learning? What do we mean?

Is learning feasible?

- Machine learning: use data to compute hypothesis g that approximate unknown target f.
- In practice, learning algorithm $\mathcal{A}$ takes training examples $\mathcal{D}$ and hypothesis set $\mathcal{H}$ to get final hypothesis g.
- Learning is only feasible in a *probabilistic* way and we can predict something useful outside the training set $\mathcal{D}$ using only $\mathcal{D}$.

framing of supervised learning

Supervised Machine Learning

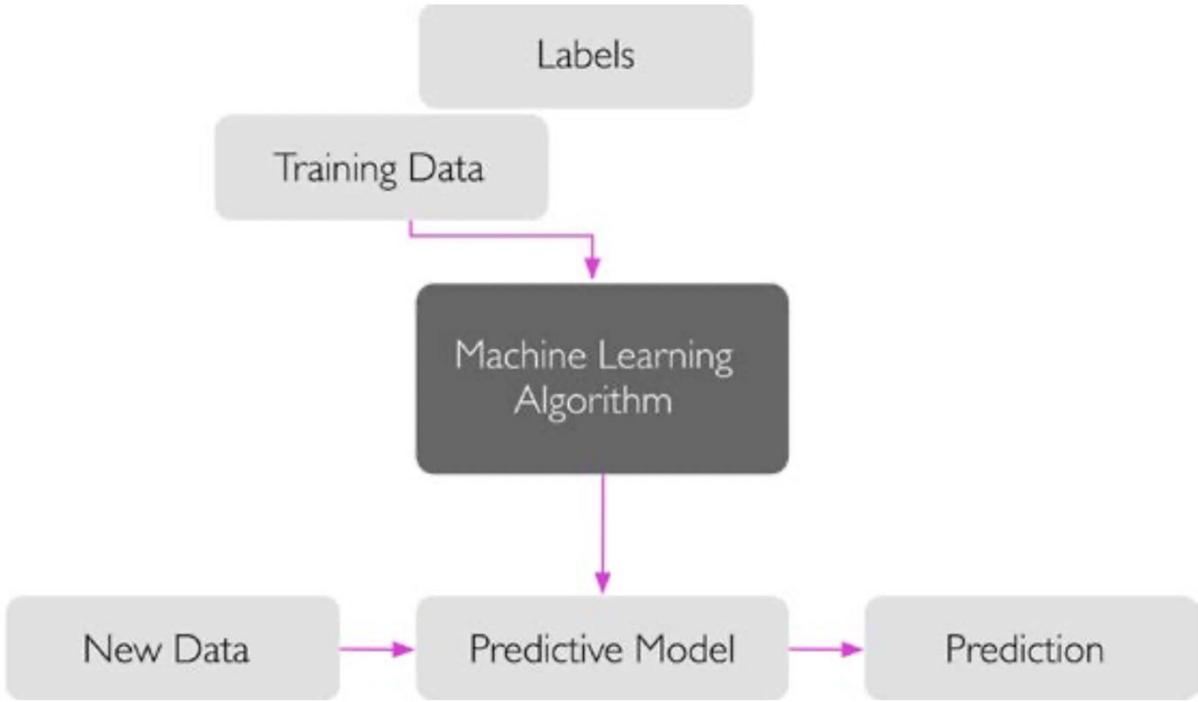
credit: Google, machine learning crash course,
https://developers.google.com/machine-learning/crash-course

ML models learn

how to combine input

to produce useful predictions

on never-before-seen data



Supervised Machine Learning

credit: Google, machine learning crash course,
https://developers.google.com/machine-learning/crash-course

ML models learn

how to combine input

to produce useful predictions

on never-before-seen data

(The deep neural) networks learn from the most complicated and diverse scenarios in the world, iteratively sourced from our fleet of nearly 1M vehicles in real time. A full build of Autopilot neural networks involves 48 networks that take 70,000 GPU hours to train 🔥. Together, they **output 1,000 distinct tensors (predictions)** at each timestep.

Supervised Machine Learning Terminology

- **Label** is the variable we're predicting
  - Typically represented by the variable y
- **Features** are input variables describing our data
  - Typically represented by the variables $\{x_1, x_2, \ldots, x_D\}$
- **Example** is a particular instance of data, $\boldsymbol{x}$ (**bold** indicates a vector)
- **Labeled example** has {features, label}: $\{\boldsymbol{x}, y\}$
  - Used to train the model
- **Unlabeled example** has {features, ?}
  - Used to making prediction on new data
- **Model** maps unlabeled examples to predicted labels: y'
  - Defined by (training) parameters, which are learned.

credit: Google, machine learning crash course,
https://developers.google.com/machine-learning/crash-course

Supervised Machine Learning (Credit Approval)

Labeled examples

age (feature)	gender (feature)	annual salary (feature)	year in residence (feature)	year in job (feature)	current debt (feature)	approval (label)
23	female	1,000,000	1	0.5	200,000	Yes
45	male	500,000	1	0.5	250,000	No
75	male	0	20	0	0	Yes

Unlabeled examples

age (feature)	gender (feature)	annual salary (feature)	year in residence (feature)	year in job (feature)	current debt (feature)
45	female	1,500,000	10	5	500,000

Fun time: Supervised Machine Learning: Suppose you want to develop a supervised machine learning model to predict whether a given email is “spam” or “not spam.” Which of the following statements are true? (multiple answers, 多選題)

- Emails not marked as "spam" or "not spam" are unlabeled examples.
- The labels applied to some examples might be unreliable.
- We'll use unlabeled examples to train the model.
- Words in the subject header will make good labels.

slido

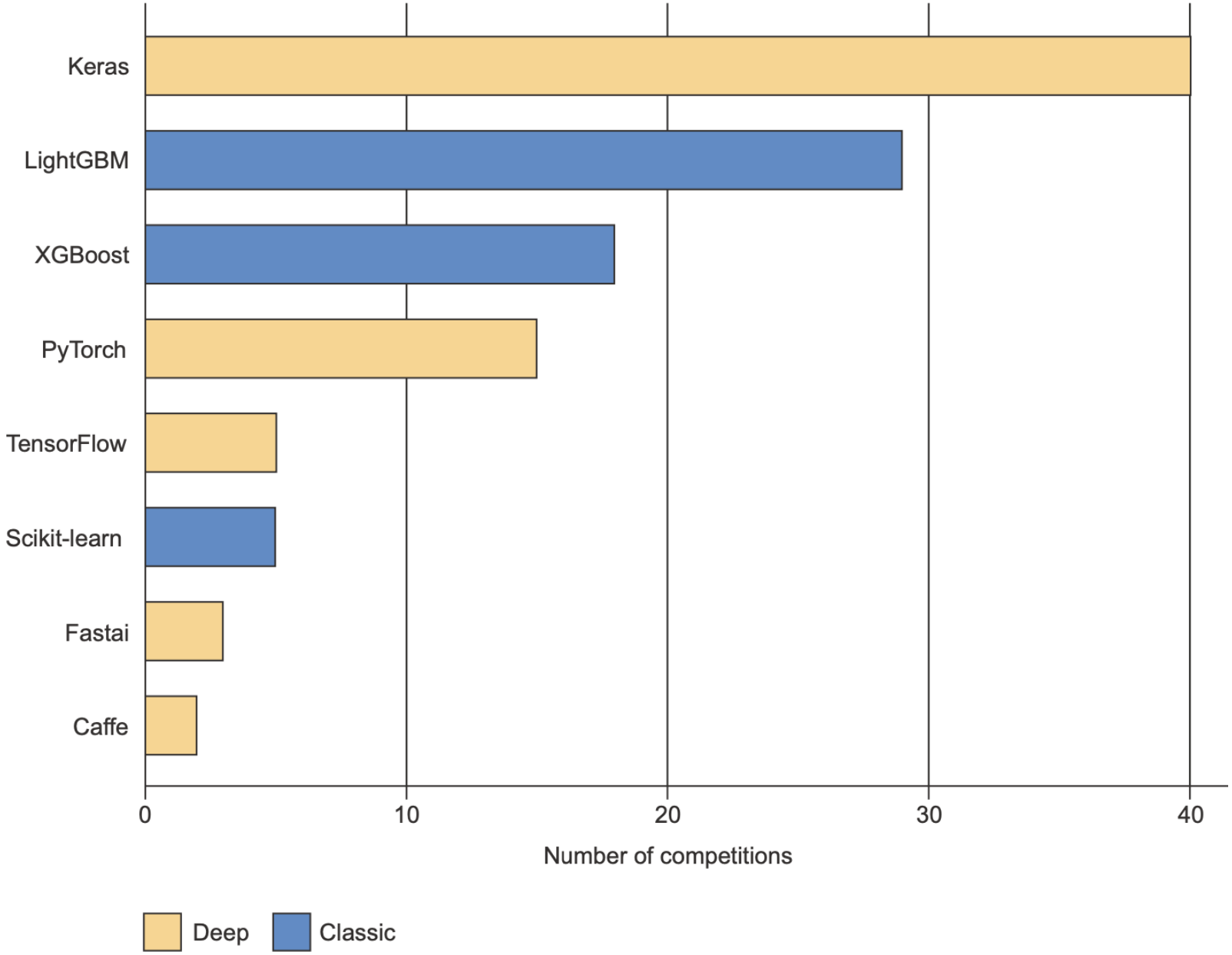
slido

Please download and install the Slido app on all computers you use

Fun time: Supervised Machine Learning: Suppose you want to develop a supervised machine learning model to predict whether a given email is “spam” or “not spam.” Which of the following statements are true? (multiple answers, 多選題)

ⓘ Start presenting to display the poll results on this slide.

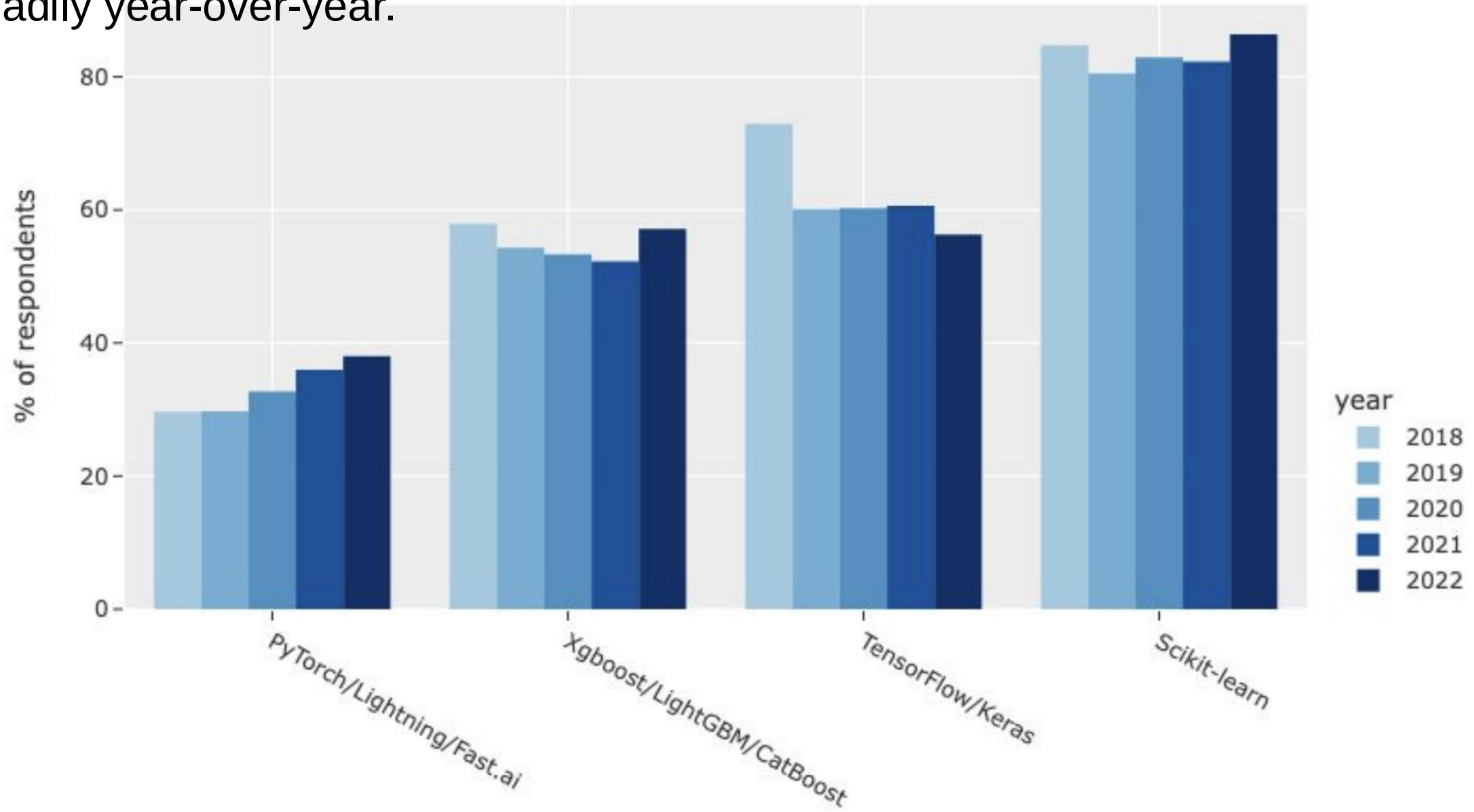
Know the modern machine learning landscape



In early **2019**, Kaggle ran a survey asking teams that ended in the top five of any competition since 2017 which primary software tool they had used in the competition (see figure 1.12). Top teams use **deep learning methods** (mainly via the Keras library) or **gradient-boosted trees** (often via the LightGBM or XGBoost libraries).

Francois Chollet (2021), Deep Learning with Python, Second Edition

Scikit-learn is the most popular ML framework while PyTorch has been growing steadily year-over-year.



Download the full survey results at:

kaggle.com/kaggle-survey-2022

Or

See Kaggle State of Machine Learning and Data Science Report 2022.pdf

Summary

Know the modern ML landscape

- Scikit-Learn and Keras (now part of TensorFlow) are mostly widely used ML software frameworks by ML professionals.
- From 2016 to 2020, the entire machine learning and data science industry has been dominated by these two approaches: deep learning and gradient boosted trees. Specifically, gradient boosted trees is used for problems where structured data is available, whereas deep learning is used for perceptual problems such as image classification.
- Users of gradient boosted trees tend to use Scikit-Learn, XGBoost or LightGBM. Meanwhile, most practitioners of deep learning use Keras, often in combination with its parent framework TensorFlow. PyTorch is getting momentum lately.
- The common point of these tools is they're all Python libraries: Python has is by far the most widely-used language for machine learning and data science.

Learn the basics of Scikit-Learn

C-S David Chen, Department of Civil Engineering, National Taiwan University

Extensions to **SciPy** (Scientific Python) are called **SciKits**. **SciKit-Learn** provides machine learning algorithms.

- **Algorithms for supervised & unsupervised learning**
- **Built on SciPy and Numpy**
- **Standard Python API interface**
- **Probably the best general ML framework out there.**

credit: Introduction to Machine Learning with Scikit-Learn, District Data Lab

Machine Learning with Scikit-Learn

Primary Features

- Generalized Linear Models
- SVMs, kNN, Bayes, Decision Trees, Ensembles
- Clustering and Density algorithms
- Cross Validation
- Grid Search
- Pipelining
- Model Evaluations
- Dataset Transformation
- Dataset Loading

credit: Introduction to Machine Learning with Scikit-Learn, District Data Lab

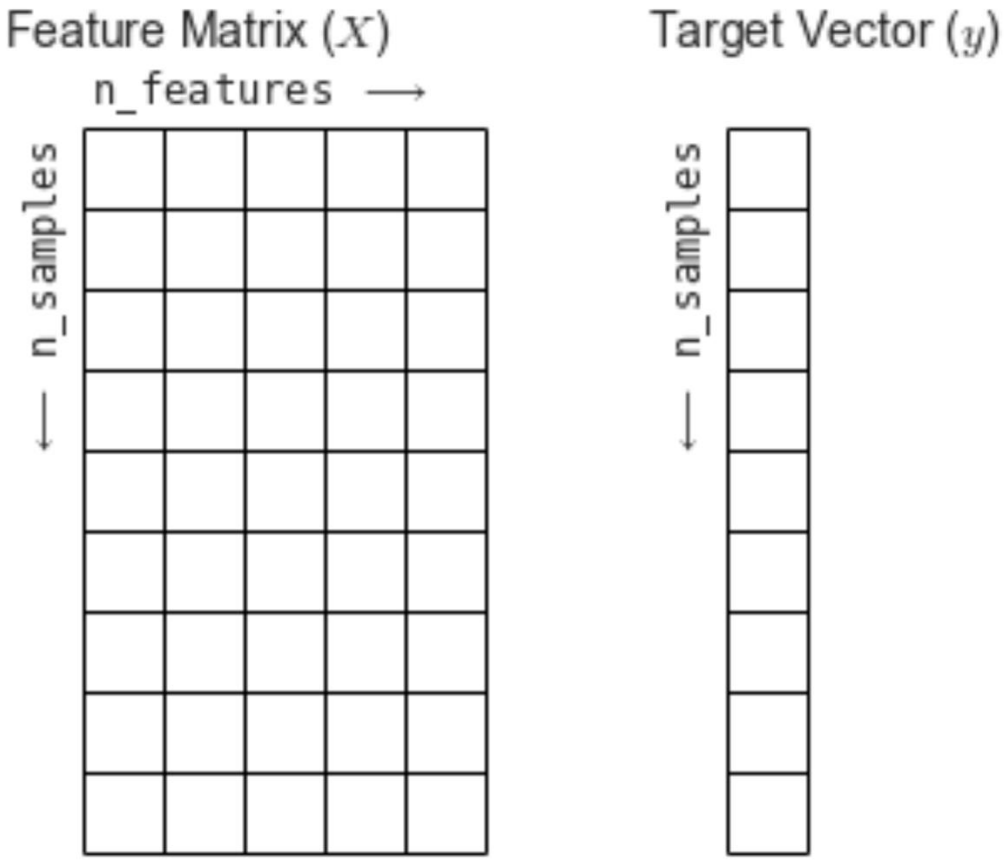
Object-oriented interface centered around the concept of an Estimator:

"**An estimator is any object that learns from data**; it may be a classification, regression or clustering algorithm or a transformer that extracts/filters useful features from raw data."

credit: Introduction to Machine Learning with Scikit-Learn, District Data Lab

Estimators

- **fit(X,y) sets the state of the estimator.**
  - **X is usually a 2D numpy array of shape (num_samples, num_features)**
  - **y is a 1D array with shape (n_samples,)**
- **predict(X) returns the class or value**



See Introducing_Scikit-Learn.pdf

credit: Introduction to Machine Learning with Scikit-Learn, District Data Lab