Lecture 01

What is Machine Learning? An Overview.

STAT 479: Machine Learning, Fall 2018

Sebastian Raschka

http://stat.wisc.edu/~sraschka/teaching/stat479-fs2018/

About this Course

When

- Tue 4:00-5:15 pm
- Thu 4:00-5:15 pm

Where

- VAN HISE 114

Office Hours

- **Sebastian Raschka** (Instructor):
  Tue 3:00-4:00, Room MSC 1171
- **Zheng Liu** (Teaching Assistant):
  Wed 3:00-4:00 pm, Room MSC 1275A

For details -> http://stat.wisc.edu/~sraschka/teaching/stat479-fs2019/

What is Machine Learning?

Some inspirational quotations ...

"Machine learning is the hot new thing"

— John L. Hennessy, President of Stanford (2000–2016)

"A breakthrough in machine learning would be worth ten Microsofts"

— Bill Gates, Microsoft Co-Founder

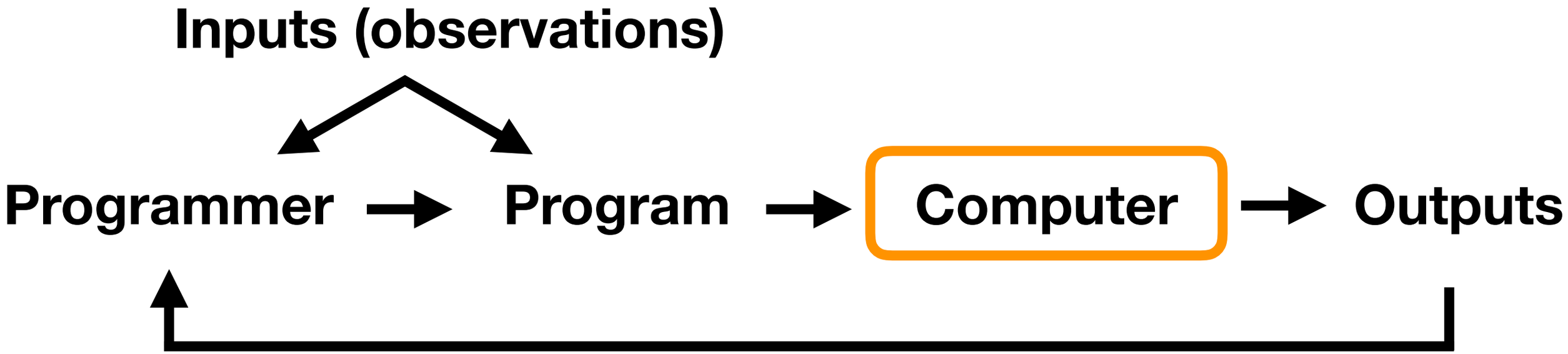
"Machine learning is the field of study that gives computers the ability to learn without being explicitly programmed"

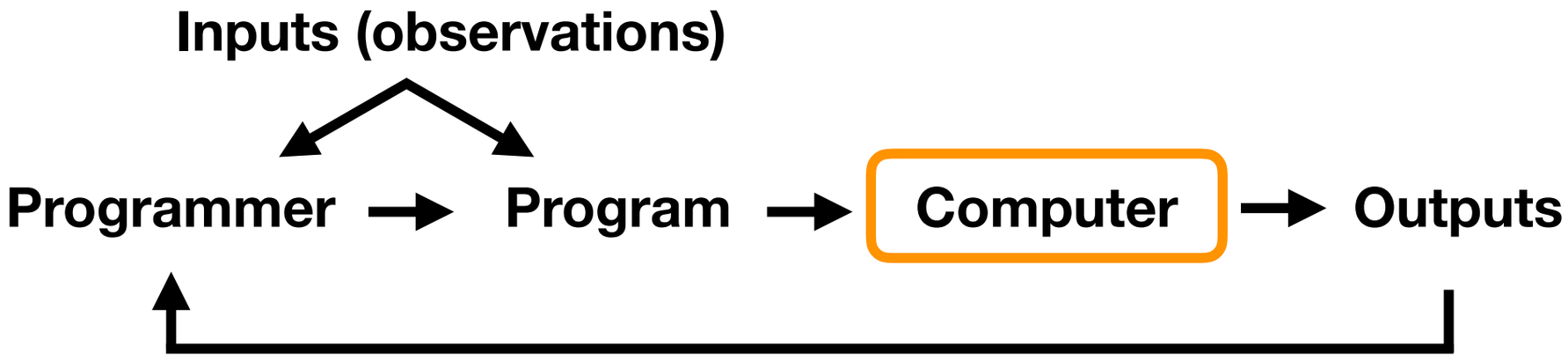
— Arthur L. Samuel, AI pioneer, 1959

(This is likely not an original quote but a paraphrased version of Samuel's sentence "Programming computers to learn from experience should eventually eliminate the need for much of this detailed programming effort.")

Arthur L Samuel. "Some studies in machine learning using the game of checkers". In: *IBM Journal of research and development* 3.3 (1959), pp. 210–229.

The Traditional Programming Paradigm





Machine learning is the field of study that gives computers the ability to learn without being explicitly programmed

— *Arthur Samuel (1959)*

"If software ate the world, models will run it"

— Steven A. Cohen and Matthew W. Granade, The Wallstreet Journal, 2018

"A computer program is said to **learn** from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E."

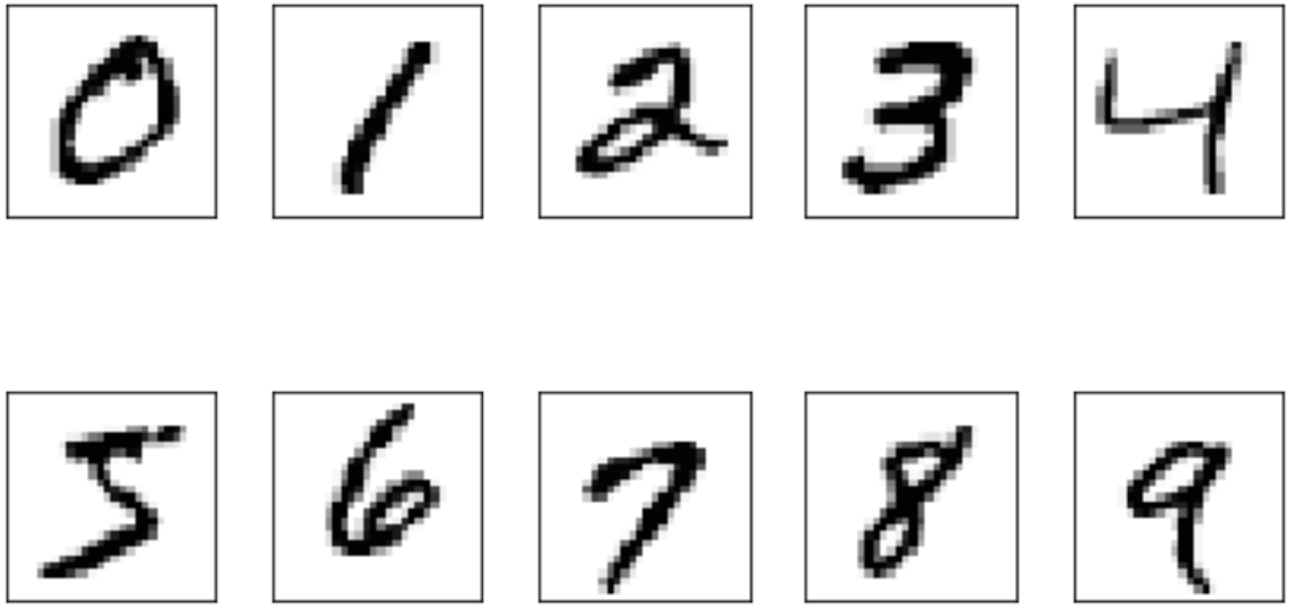
— Tom Mitchell, Professor at Carnegie Mellon University

Tom M Mitchell et al. "Machine learning. 1997". In: *Burr Ridge, IL: McGraw Hill* 45.37 (1997), pp. 870–877.

"A computer program is said to **learn** from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E."

— Tom Mitchell, Professor at Carnegie Mellon University

Handwriting Recognition Example:



- Task T: ?
- Performance measure P : ?
- Training experience E: ?

Some Applications of Machine Learning (1):

-
-
-
-
-

Some Applications of Machine Learning (2):

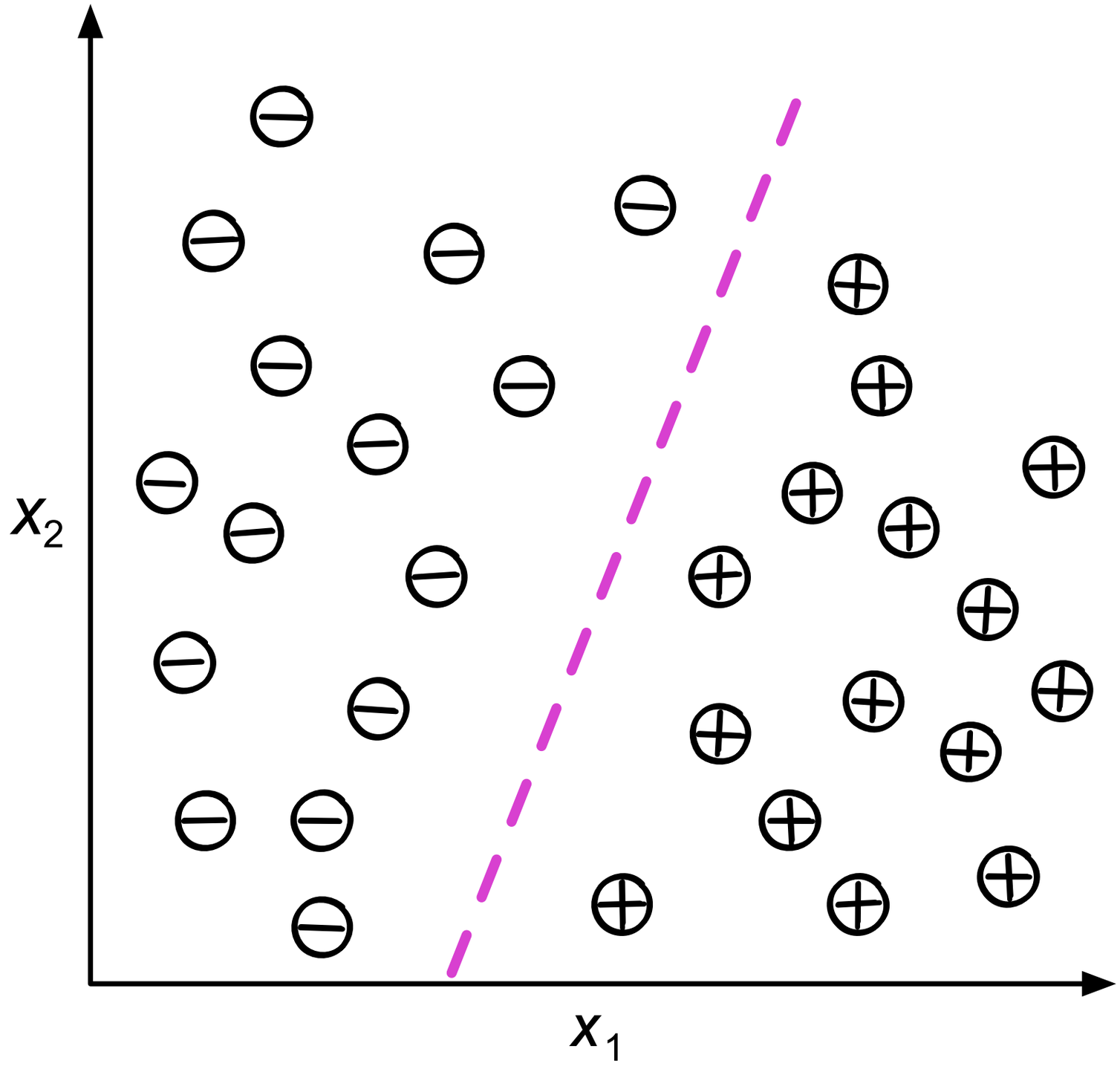
-
-
-
-
-

Categories of Machine Learning

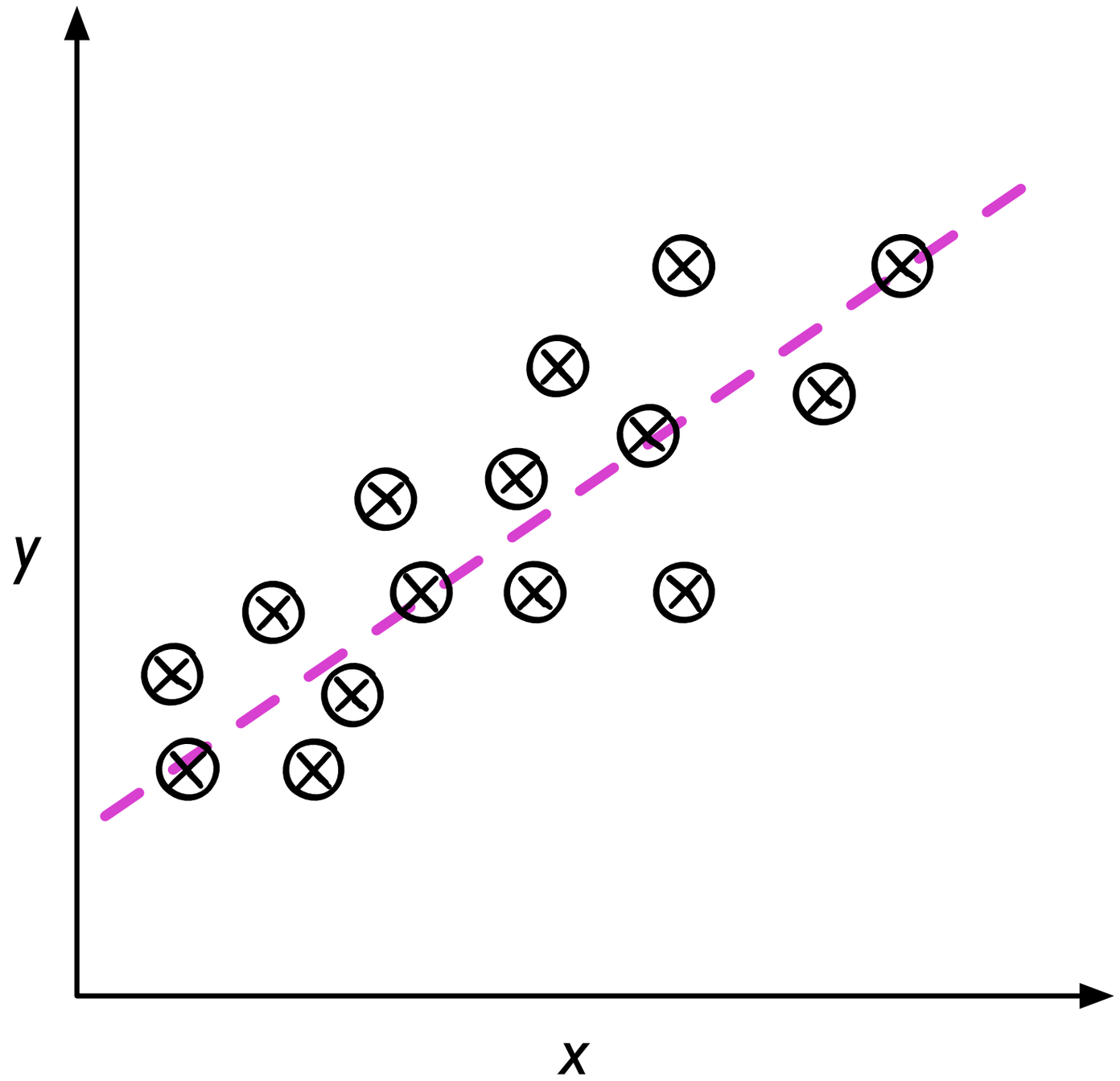
Supervised Learning

- Labeled data
- Direct feedback
- Predict outcome/future

Supervised Learning: Classification



Supervised Learning: Regression



Categories of Machine Learning

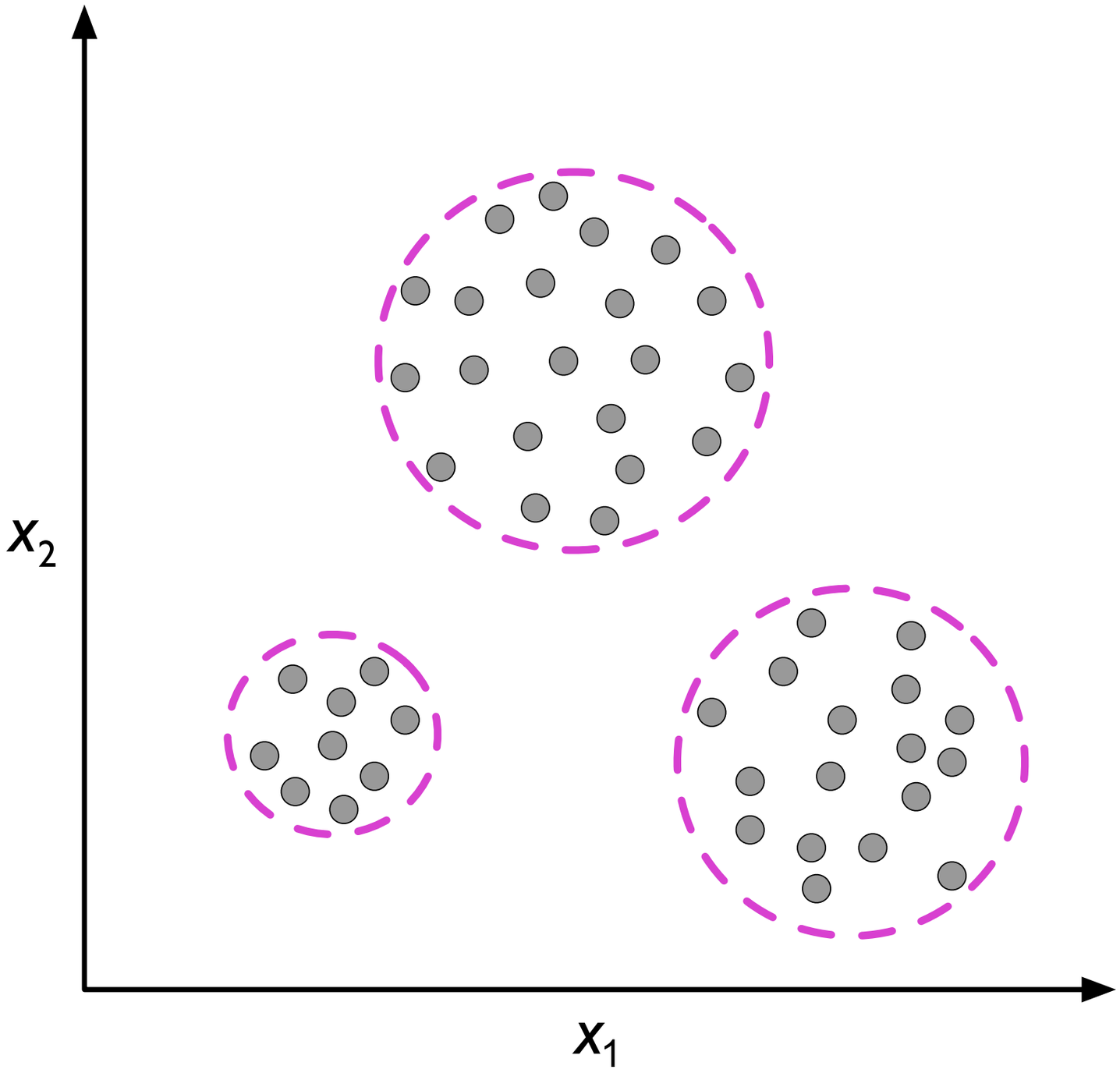
Supervised Learning

- Labeled data
- Direct feedback
- Predict outcome/future

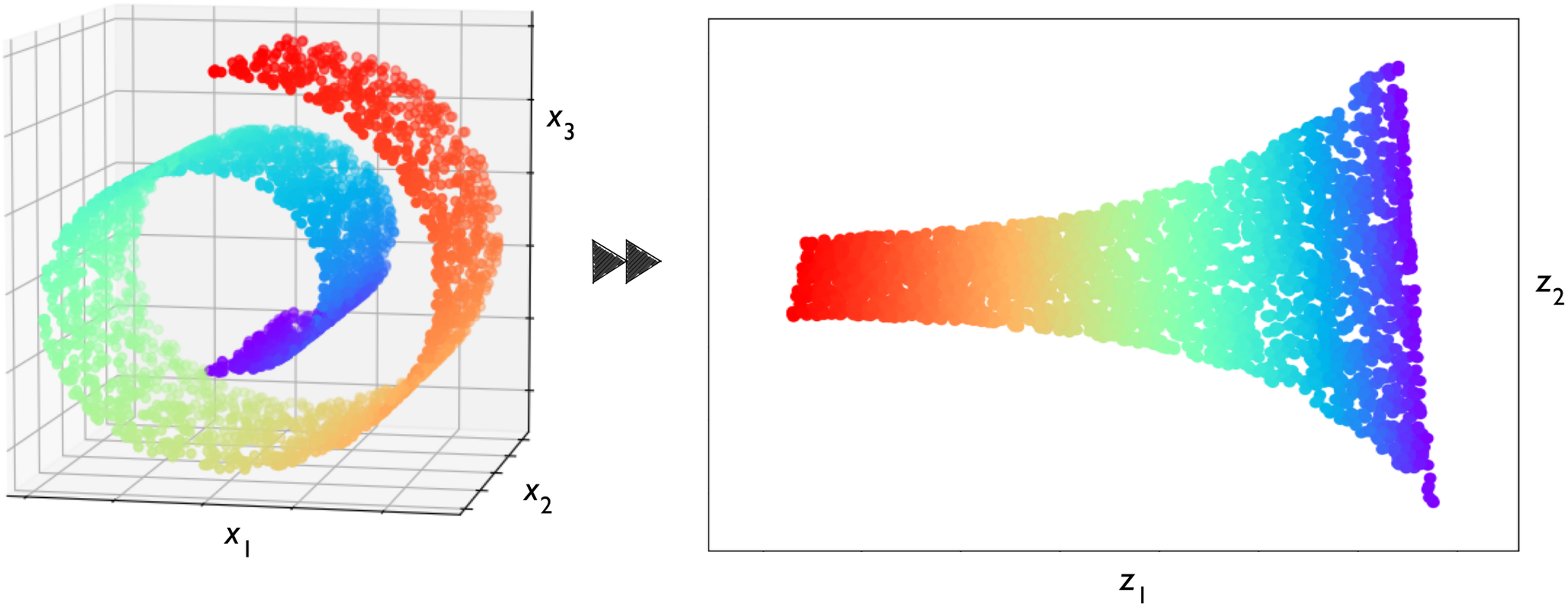
Unsupervised Learning

- No labels/targets
- No feedback
- Find hidden structure in data

Unsupervised Learning -- Clustering



Unsupervised Learning
-- Dimensionality Reduction



Categories of Machine Learning

Supervised Learning

- Labeled data
- Direct feedback
- Predict outcome/future

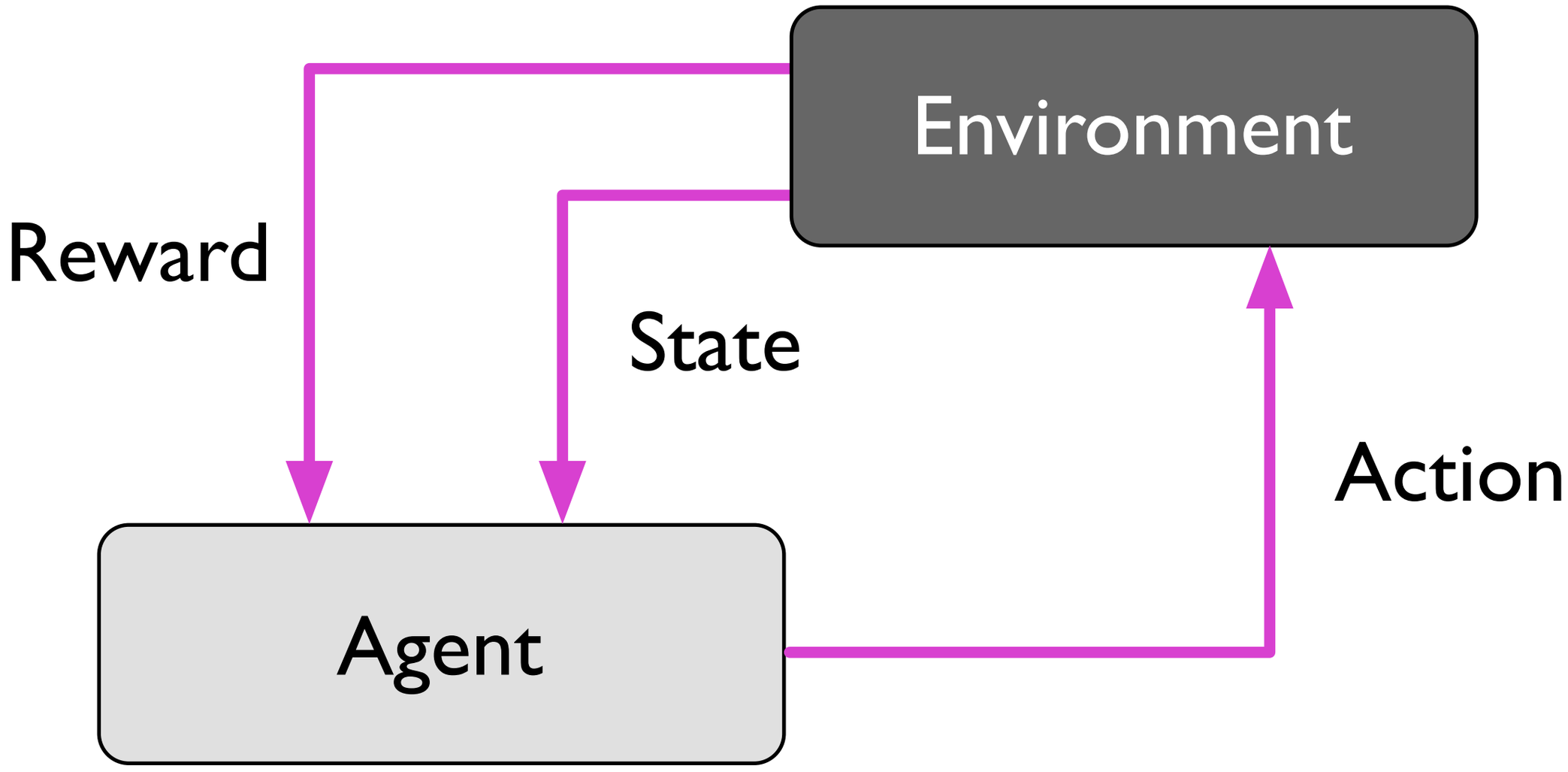
Unsupervised Learning

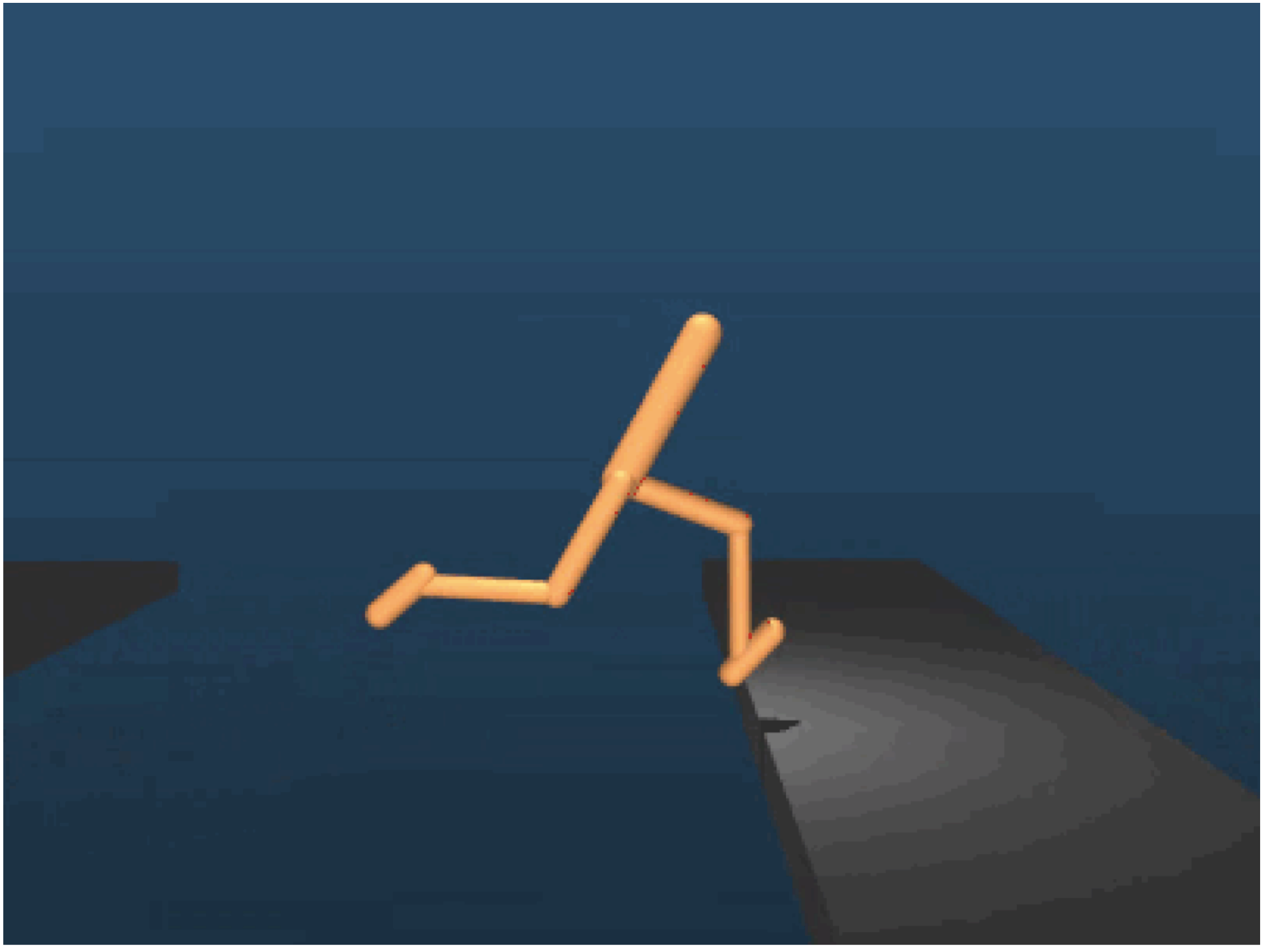
- No labels/targets
- No feedback
- Find hidden structure in data

Reinforcement Learning

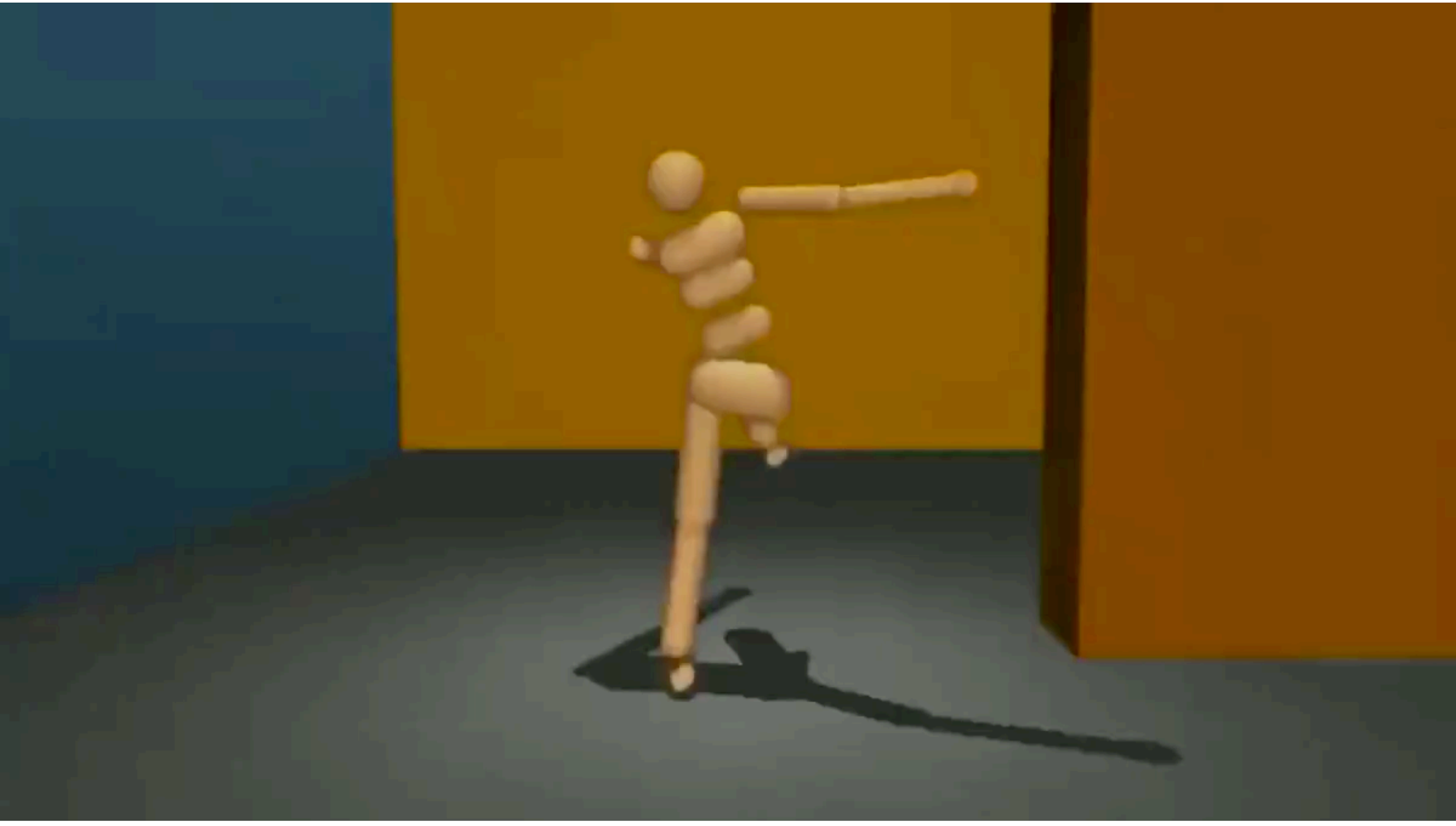
- Decision process
- Reward system
- Learn series of actions

Reinforcement Learning





https://www.theverge.com/tldr/2017/7/10/15946542/deepmind-parkour-agent-reinforcement-learning



https://video.twimg.com/ext_tw_video/1111683489890332672/pu/vid/1200x674/WqUJEhUETw0M0gCl.mp4?tag=8

Semi-Supervised Learning

Supervised Learning (Formal Notation)

Training set: $\mathscr{D} = \{\langle \mathbf{x}^{[i]}, y^{[i]} \rangle, i = 1, \ldots, n\},$

Unknown function: $f(\mathbf{x}) = y$

Hypothesis: $h(\mathbf{x}) = \hat{y}$

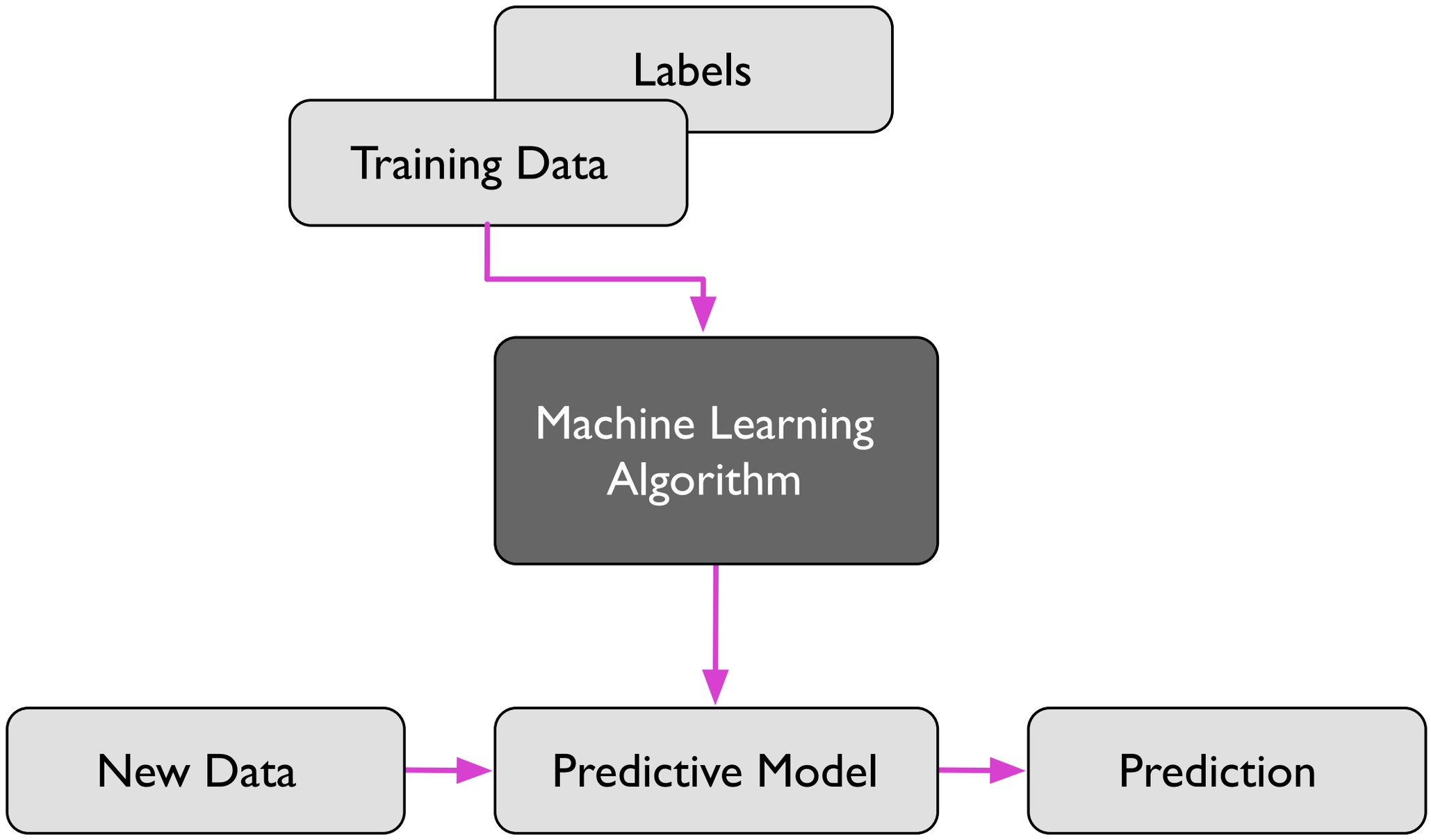
Classification

$h : \mathbb{R}^m \rightarrow \_\_$

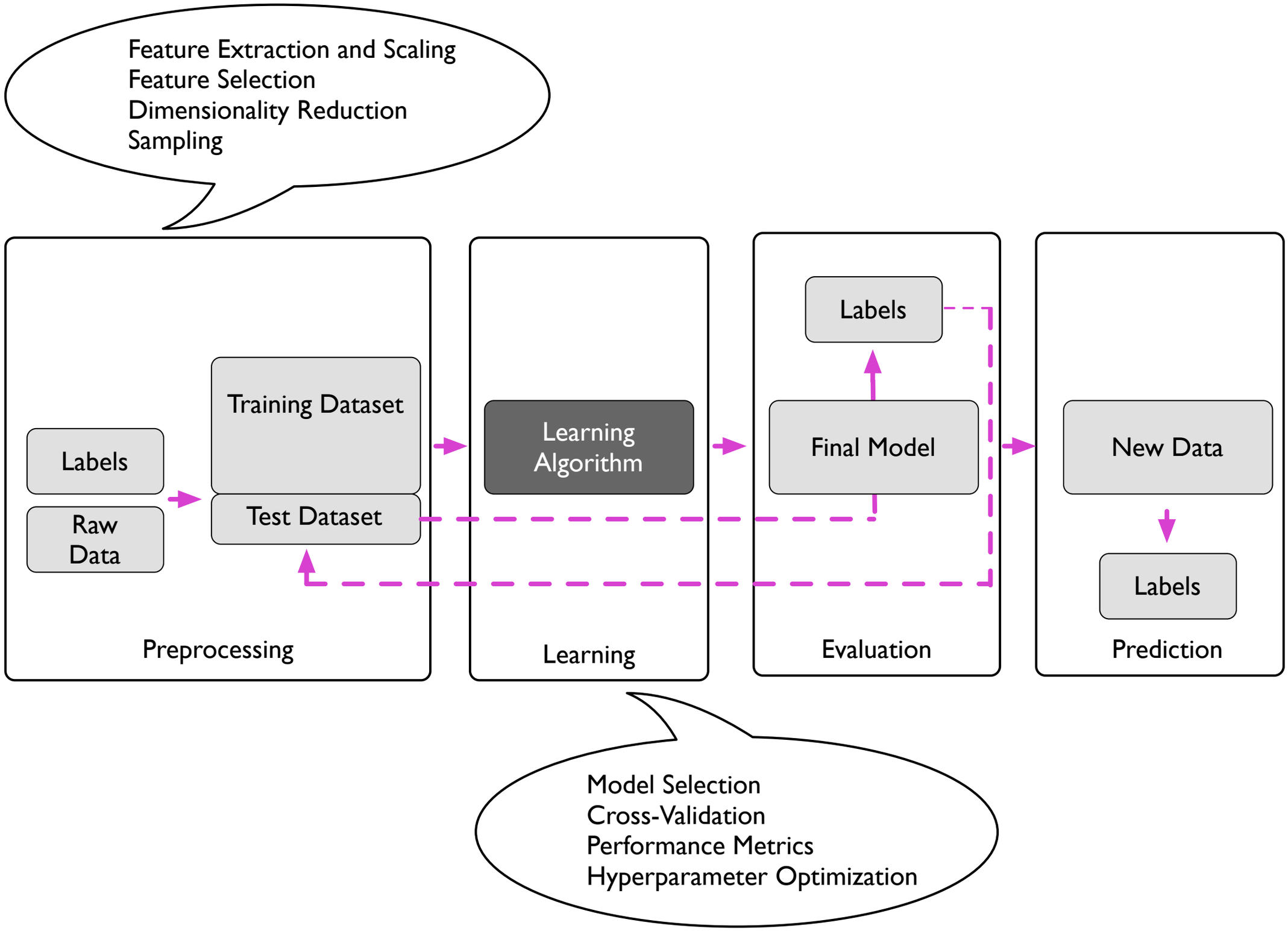
Regression

$h : \mathbb{R}^m \rightarrow \_\_$

Supervised Learning Workflow -- Overview



Supervised Learning Workflow
-- More Detailed Overview



Data Representation

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_m \end{bmatrix}$$

Feature vector

Data Representation

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_m \end{bmatrix}$$

$$\mathbf{X} = \begin{bmatrix} \mathbf{x}_1^T \\ \mathbf{x}_2^T \\ \vdots \\ \mathbf{x}_n^T \end{bmatrix}$$

Feature vector D___n m_____

Data Representation

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_m \end{bmatrix}$$

$$\mathbf{X} = \begin{bmatrix} \mathbf{x}_1^T \\ \mathbf{x}_2^T \\ \vdots \\ \mathbf{x}_n^T \end{bmatrix}$$

$$\mathbf{X} = \begin{bmatrix} x_1^{[1]} & x_2^{[1]} & \cdots & x_m^{[1]} \\ x_1^{[2]} & x_2^{[2]} & \cdots & x_m^{[2]} \\ \vdots & \vdots & \ddots & \vdots \\ x_1^{[n]} & x_2^{[n]} & \cdots & x_m^{[n]} \end{bmatrix}$$

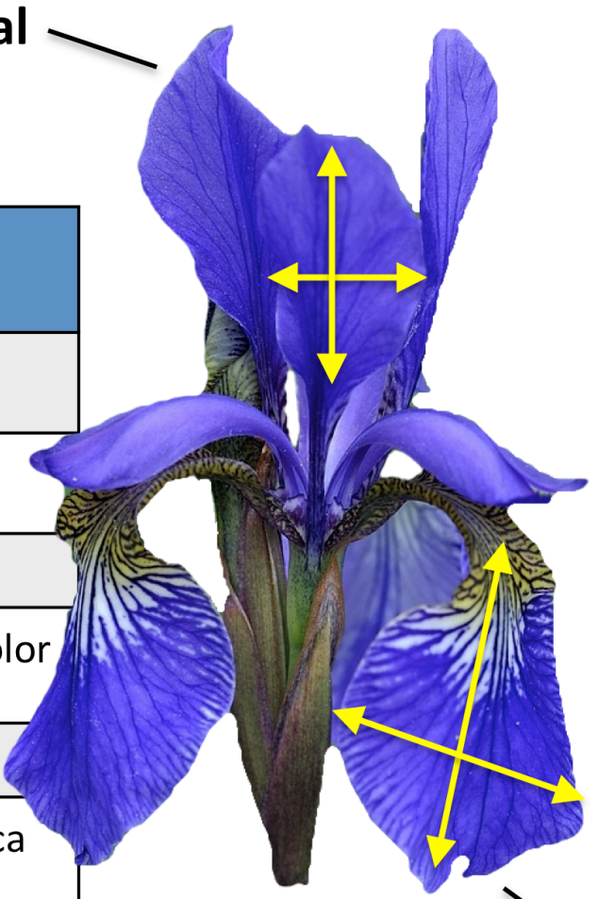
Feature vector

Data Representation

$m =$ ____

$n =$ ____

	Sepal length	Sepal width	Petal length	Petal width	
1	5.1	3.5	1.4	0.2	Setosa
2	4.9	3.0	1.4	0.2	Setosa
...					
50	6.4	3.5	4.5	1.2	Versicolor
...					
150	5.9	3.0	5.0	1.8	Virginica



Data Representation

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_m \end{bmatrix} \qquad \mathbf{y} = \begin{bmatrix} y^{[1]} \\ y^{[2]} \\ \vdots \\ y^{[n]} \end{bmatrix}$$

Input features

Diverse Terminology (Part 1)

- **Training example**: A row in the table representing the dataset. Synonymous to an observation, training record, training instance, training sample (in some contexts, sample refers to a collection of training examples)

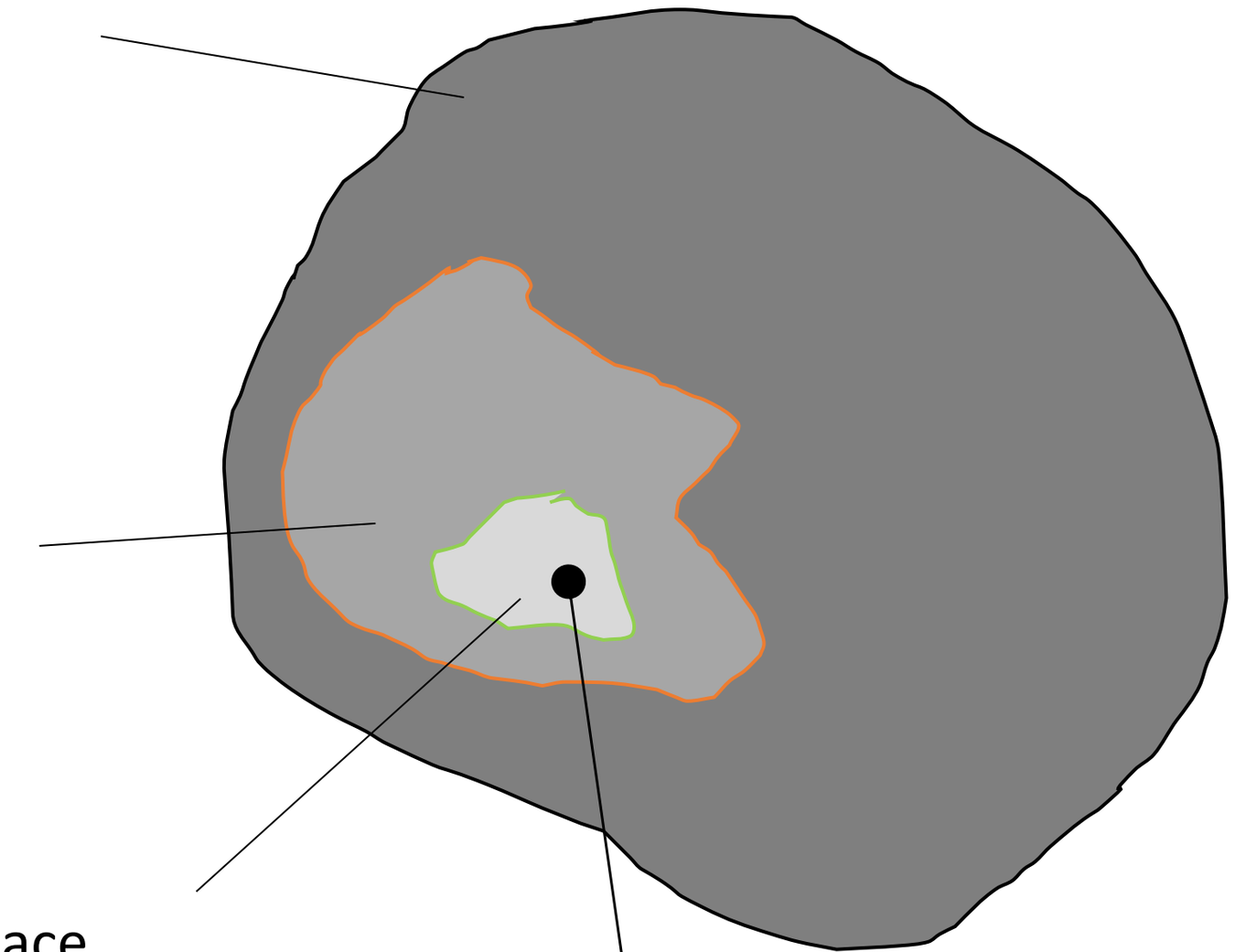
- **Feature**: a column in the table representing the dataset. Synonymous to predictor, variable, input, attribute, covariate.

- **Targets**: What we want to predict. Synonymous to outcome, output, ground truth, response variable, dependent variable, (class) label (in classification).

- **Output / prediction**: use this to distinguish from targets; here, means output from the model.

Hypothesis Space

Entire hypothesis space



Hypothesis space
a particular learning
algorithm category
has access to

Hypothesis space
a particular learning
algorithm can sample

Particular hypothesis
(i.e., a model/classifier)

Classes of Machine Learning Algorithms

- Generalized linear models (e.g.,
- Support vector machines (e.g.,
- Artificial neural networks (e.g.,
- Tree- or rule-based models (e.g.,
- Graphical models (e.g.,
- Ensembles (e.g.,
- Instance-based learners (e.g.,

5 Steps for Approaching a Machine Learning Application

1. Define the problem to be solved.

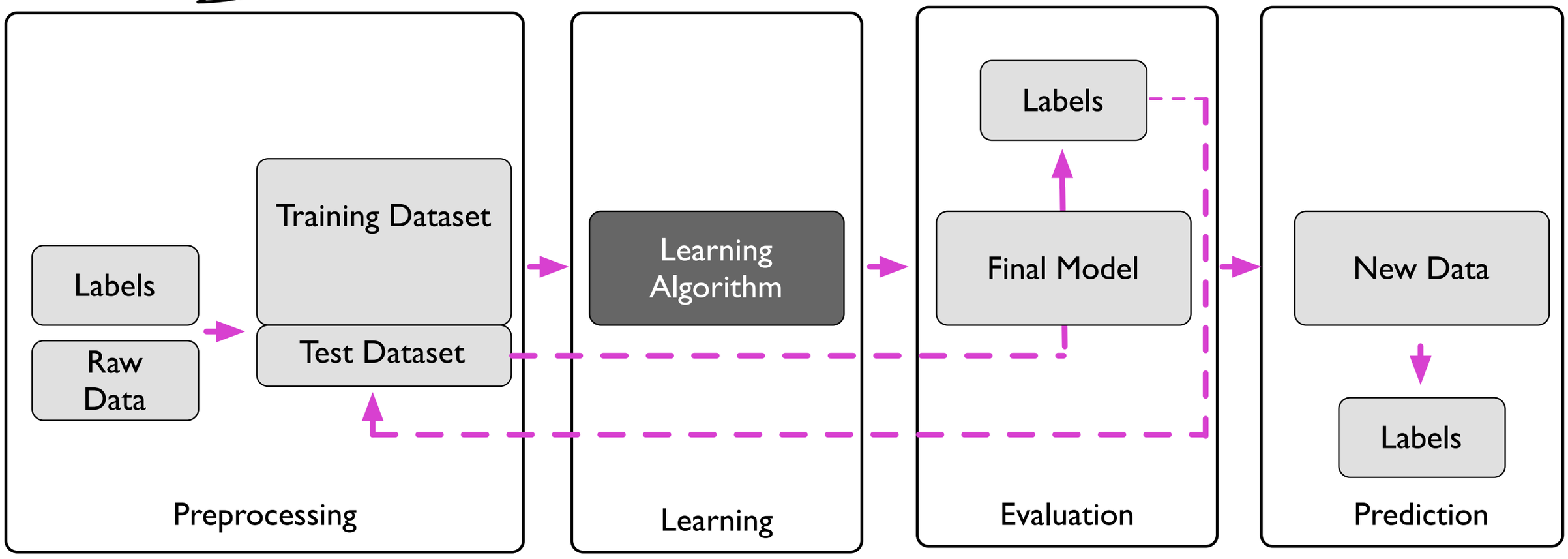
2. Collect (labeled) data.

3. Choose an algorithm class.

4. Choose an optimization metric for learning the model.

5. Choose a metric for evaluating the model.

Feature Extraction and Scaling
Feature Selection
Dimensionality Reduction
Sampling



Model Selection
Cross-Validation
Performance Metrics
Hyperparameter Optimization

Objective Functions

- Maximize the posterior probabilities (e.g., naive Bayes)
- Maximize a fitness function (genetic programming)
- Maximize the total reward/value function (reinforcement learning)
- Maximize information gain/minimize child node impurities (CART decision tree classification)
- Minimize a mean squared error cost (or loss) function (CART, decision tree regression, linear regression, adaptive linear neurons, ...)
- Maximize log-likelihood or minimize cross-entropy loss (or cost) function
- Minimize hinge loss (support vector machine)
-

Optimization Methods for Different Learning Algorithms

- Combinatorial search, greedy search (e.g., decision trees)
- Unconstrained convex optimization (e.g.,
- Constrained convex optimization (e.g.,
- Nonconvex optimization, here: using backpropagation, chain rule, reverse autodiff. (e.g.,
- Constrained nonconvex optimization (e.g.,

Diverse Terminology (Part 2)

- **Loss function:** Often used synonymously with cost function; sometimes also called error function. In some contexts the loss for a single data point, whereas the cost function refers to the overall (average or summed) loss over the entire dataset. Sometimes also called empirical risk.

Evaluation -- Misclassification Error

$$L(\hat{y}, y) = \begin{cases} 0 & \textbf{if } \hat{y} = y \\ 1 & \textbf{if } \hat{y} \neq y \end{cases}$$

$$ERR_{\mathcal{D}}\textbf{test} = \frac{1}{n} \sum_{i=1}^{n} L\left(\hat{y}^{[i]}, y^{[i]}\right)$$

Other Metrics in Future Lectures

- Accuracy (1-Error)
- ROC AUC
- Precision
- Recall
- (Cross) Entropy
- Likelihood
- Squared Error/MSE
- L-norms
- Utility
- Fitness
- ...

But more on other metrics in future lectures.

Categorizing Machine Learning Algorithms

- eager vs lazy;

Categorizing Machine Learning Algorithms

- eager vs lazy;
- batch vs online;

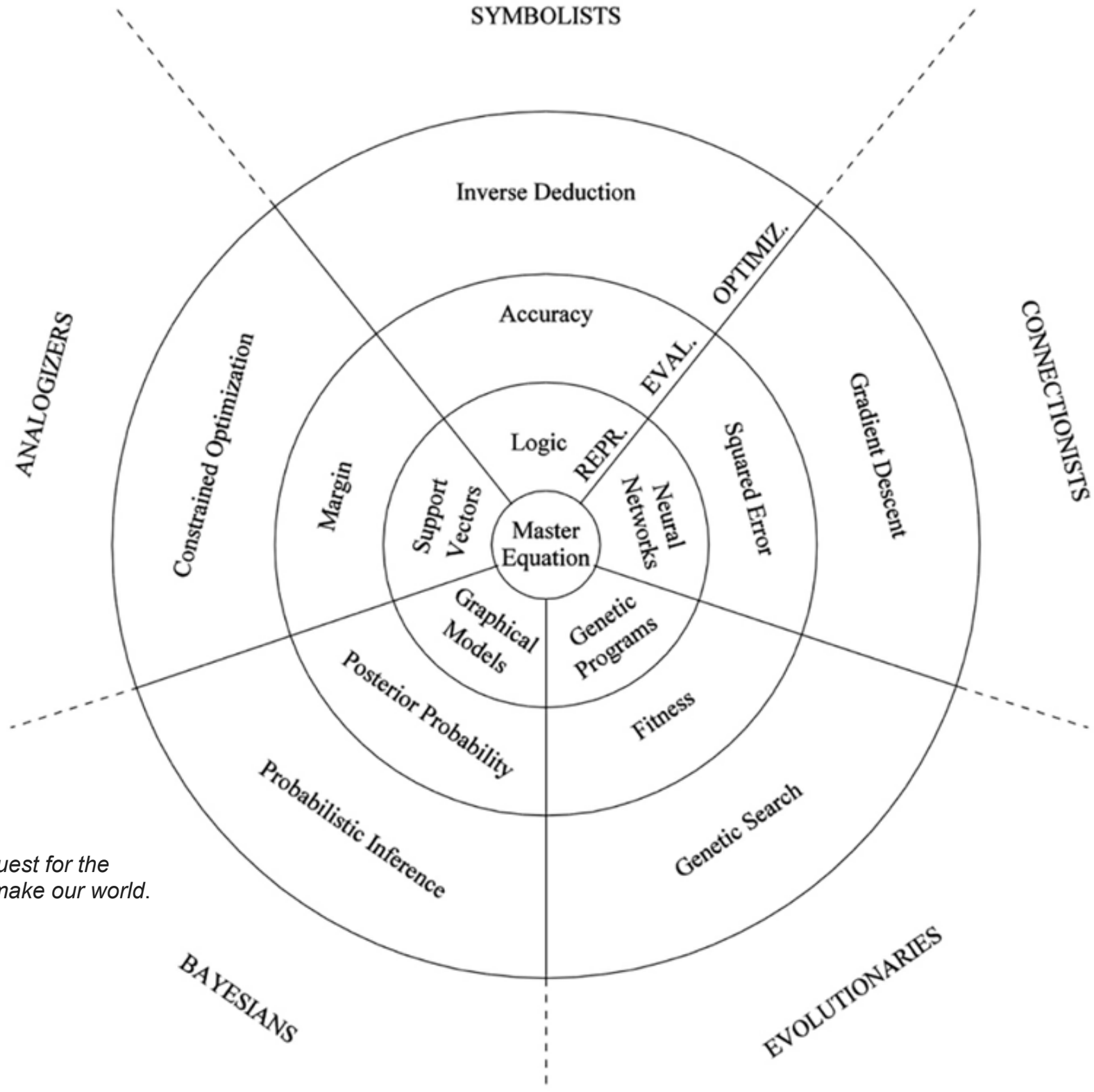
Categorizing Machine Learning Algorithms

- eager vs lazy;
- batch vs online;
- parametric vs nonparametric;

Categorizing Machine Learning Algorithms

- eager vs lazy;
- batch vs online;
- parametric vs nonparametric;
- discriminative vs generative.

Pedro Domingos's 5 Tribes of Machine Learning



Source: Domingos, Pedro.
The master algorithm: How the quest for the ultimate learning machine will remake our world.
Basic Books, 2015.

Breiman, Leo. "Statistical modeling: The two cultures (with comments and a rejoinder by the author). " *Statistical science* 16.3 (2001): 199-231.

A

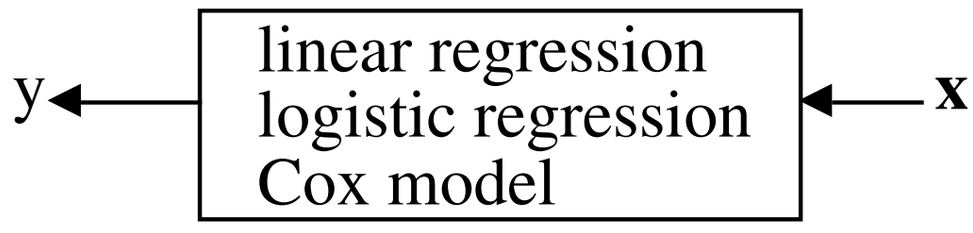
There are two goals in analyzing the data:

Prediction. To be able to predict what the responses are going to be to future input variables;
Information. To extract some information about how nature is associating the response variables to the input variables.

Breiman, Leo. "Statistical modeling: The two cultures (with comments and a rejoinder by the author). " *Statistical science* 16.3 (2001): 199-231.

B

The values of the parameters are estimated from the data and the model then used for information and/or prediction. Thus the black box is filled in like this:

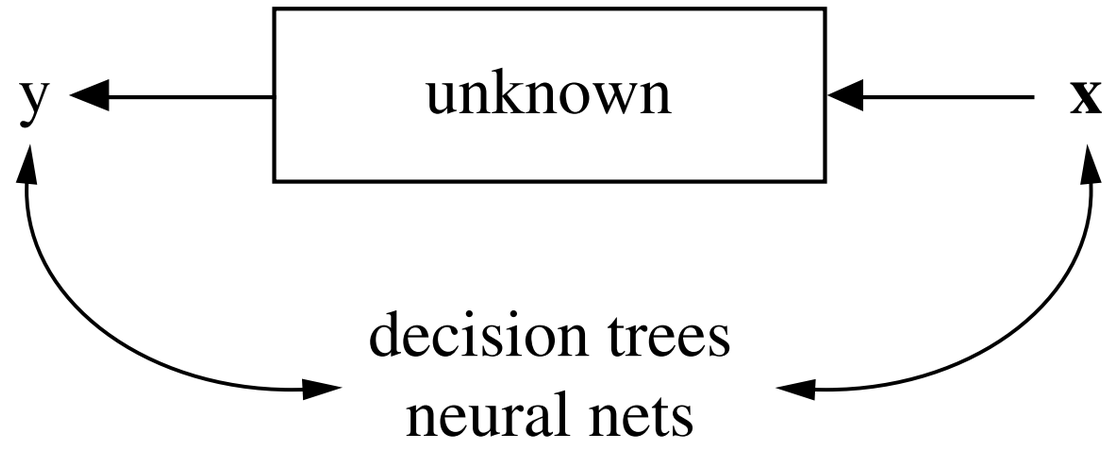


Model validation. Yes–no using goodness-of-fit tests and residual examination.

Breiman, Leo. "Statistical modeling: The two cultures (with comments and a rejoinder by the author). " *Statistical science* 16.3 (2001): 199-231.

C

The analysis in this culture considers the inside of the box complex and unknown. Their approach is to find a function $f(\mathbf{x})$—an algorithm that operates on $\mathbf{x}$ to predict the responses $\mathbf{y}$. Their black box looks like this:



Model validation. Measured by predictive accuracy.

Diverse Terminology (Part 3)

- **Hypothesis:** A hypothesis is a certain function that we believe (or hope) is similar to the true function, the target function that we want to model.

- **Model:** In the machine learning field, the terms hypothesis and model are often used interchangeably. In other sciences, they can have different meanings.

- **Learning algorithm:** Again, our goal is to find or approximate the target function, and the learning algorithm is a set of instructions that tries to model the target function using our training dataset. A learning algorithm comes with a hypothesis space, the set of possible hypotheses it explores to model the unknown target function by formulating the final hypothesis.

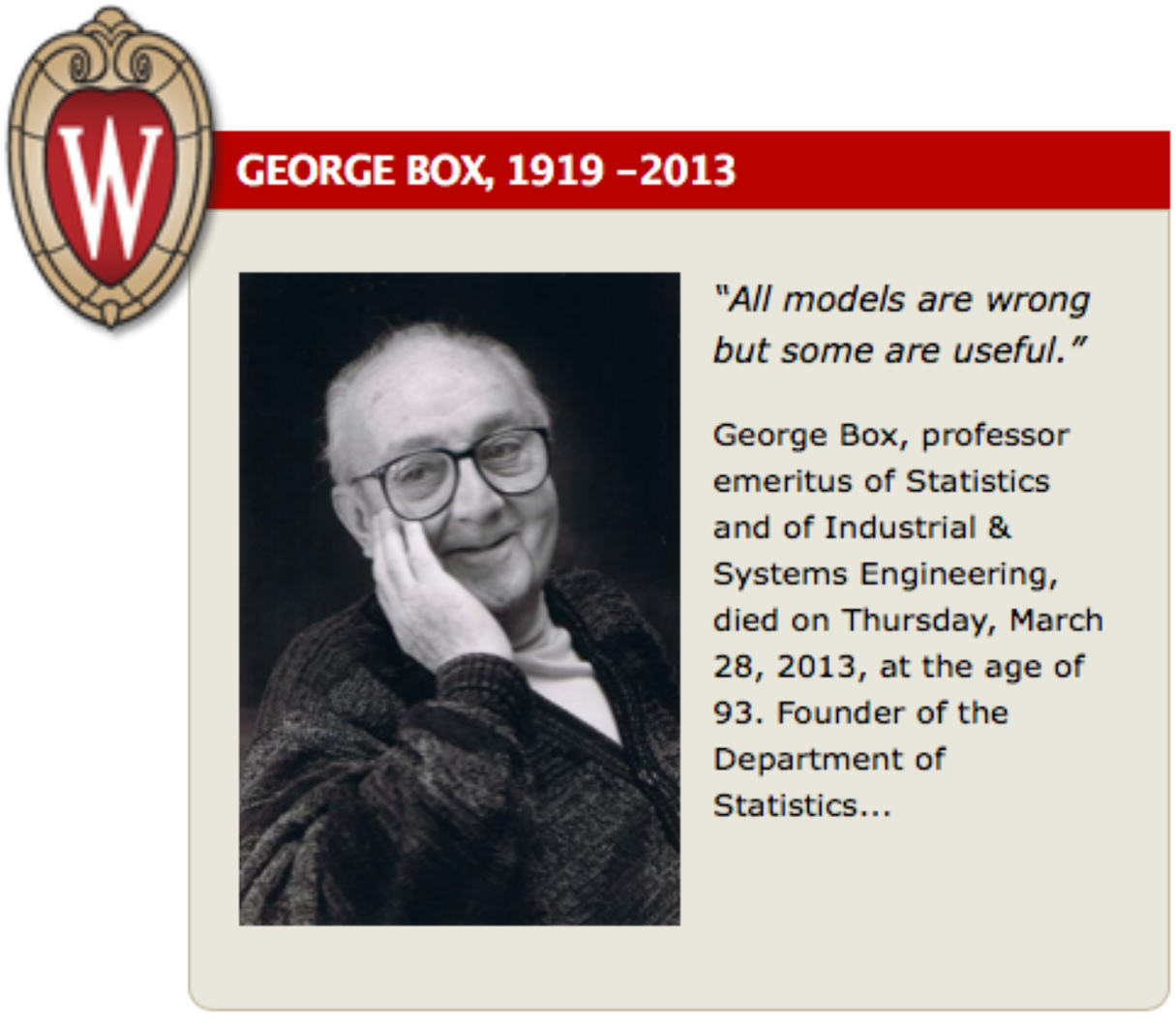
- **Classifier:** A classifier is a special case of a hypothesis (nowadays, often learned by a machine learning algorithm). A classifier is a hypothesis or discrete-valued function that is used to assign (categorical) class labels to particular data points


0
1
2 cm

Evolved antenna (Source: https://en.wikipedia.org/wiki/Evolved_antenna) via evolutionary algorithms; used on a 2006 NASA spacecraft.

Black Boxes vs Interpretability

Black Boxes vs Interpretability



Different Motivations for Studying Machine Learning

- Engineers:

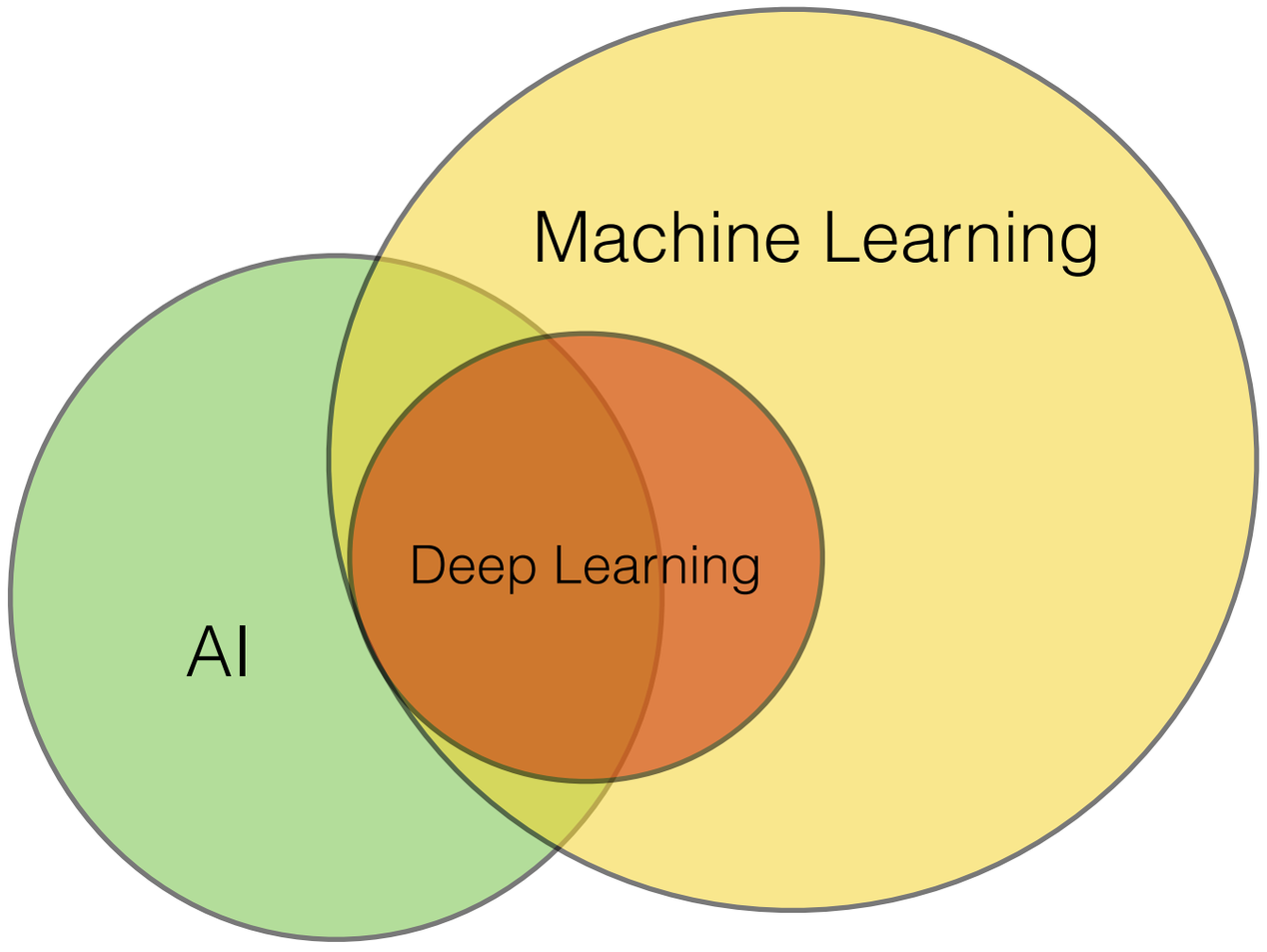
- Mathematicians, computer scientists, and statisticians:

- Neuroscientists:

The Relationship between Machine Learning and Other Fields

Machine Learning and Data Mining

Machine Learning, AI, and Deep Learning



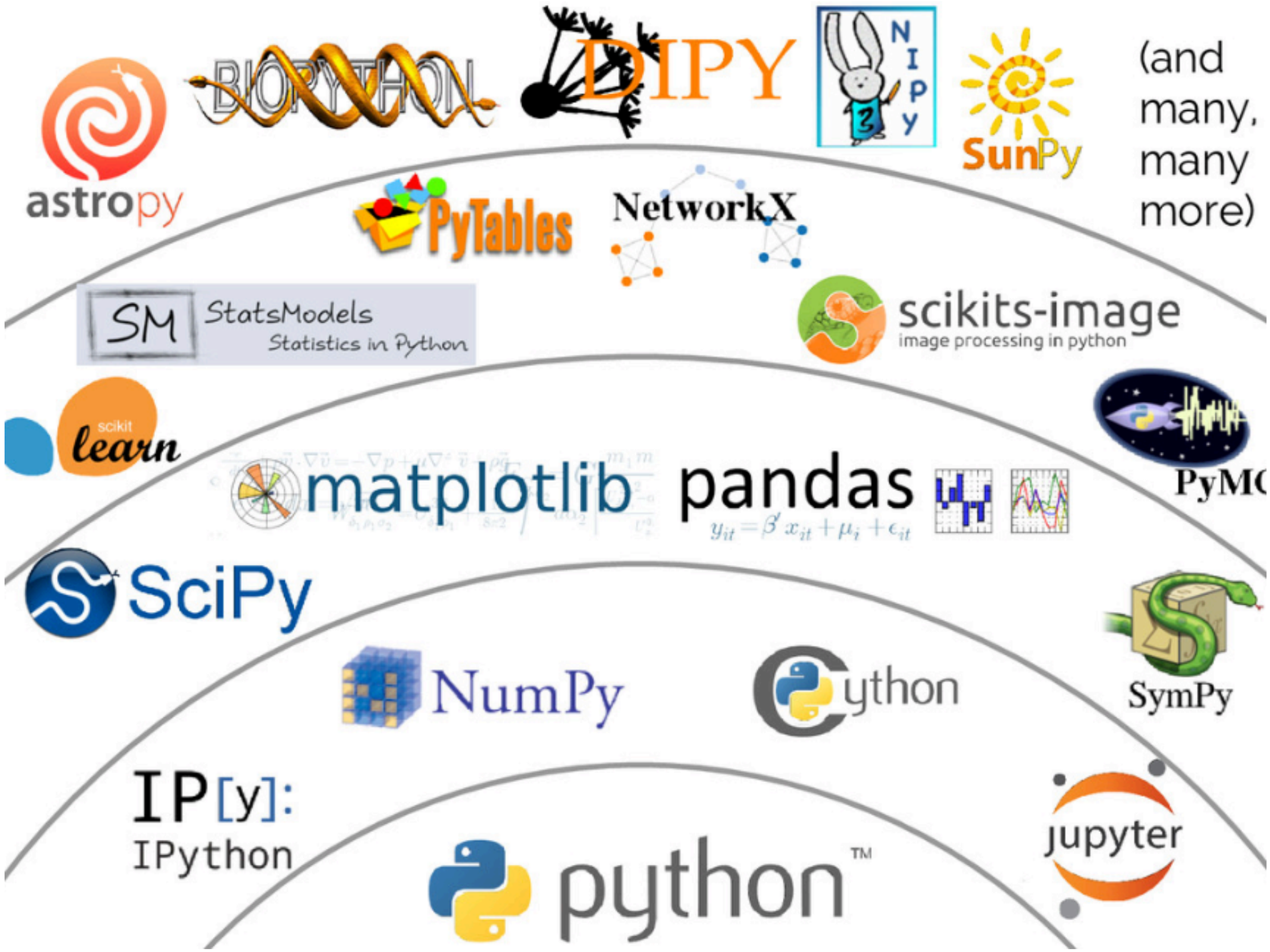


Image by Jake VanderPlas; Source: https://speakerdeck.com/jakevdp/the-state-of-the-stack-scipy-2015-keynote?slide=8)

TIOBE Index for September 2019

Aug 2019	Aug 2018	Change	Programming Language	Ratings	Change
1	1		Java	16.028%	-0.85%
2	2		C	15.154%	+0.19%
3	**4**	↑	**Python**	**10.020%**	**+3.03%**
4	3	↓	C++	6.057%	-1.41%
5	6	↑	C#	3.842%	+0.30%
6	5	↓	Visual Basic .NET	3.695%	-1.07%
7	8	↑	JavaScript	2.258%	-0.15%
8	7	↓	PHP	2.075%	-0.85%
9	14	⇈	Objective-C	1.690%	+0.33%
10	9	↓	SQL	1.625%	-0.69%
11	15	⇈	Ruby	1.316%	+0.13%
12	13	↑	MATLAB	1.274%	-0.09%
13	44	⇈	Groovy	1.225%	+1.04%
14	12	↓	Delphi/Object Pascal	1.194%	-0.18%
15	10	⇊	Assembly language	1.114%	-0.30%
16	19	↑	Visual Basic	1.025%	+0.10%
17	17		Go	0.973%	-0.02%
18	11	⇊	Swift	0.890%	-0.49%
19	16	↓	Perl	0.860%	-0.31%
20	18	↓	R	0.822%	-0.14%

Programming language "popularity"

https://www.tiobe.com/tiobe-index/

https://www.tiobe.com/tiobe-index/programming-languages-definition/

Roadmap for this Course

http://stat.wisc.edu/~sraschka/teaching/stat479-fs2019/#schedule

Reading Assignments

- Raschka and Mirjalili: Python Machine Learning, 2nd ed., Ch 1
- Elements of Statistical Learning, Ch 01 (https://web.stanford.edu/~hastie/ElemStatLearn/)
- Optional: Breiman, Leo. "Statistical modeling: The two cultures (with comments and a rejoinder by the author)". *Statistical science* 16.3 (2001): 199-231. https://projecteuclid.org/euclid.ss/1009213726