Machine Learning Basics

Prof. Yunzhu Li
Spring 2024

CS598YL: Deep Learning for Robotic Manipulation
* Course materials adapted from Prof. Eric Eaton's CIS 419/519 at UPenn.

Logistics

- Since there are students registering/dropping this course
  - We have **reassigned the presentations**
  - **Please check the website** and let us know if you have questions
- We now have two presentations per lecture
  - 25 minutes pre + 10 minutes Q&A
- Questions regarding Websites / Campuswire / Gradescope
  - Please reach out to Mingtong Zhang

What is Machine Learning?

"Learning is any process by which a system improves performance from experience."

- Herbert Simon

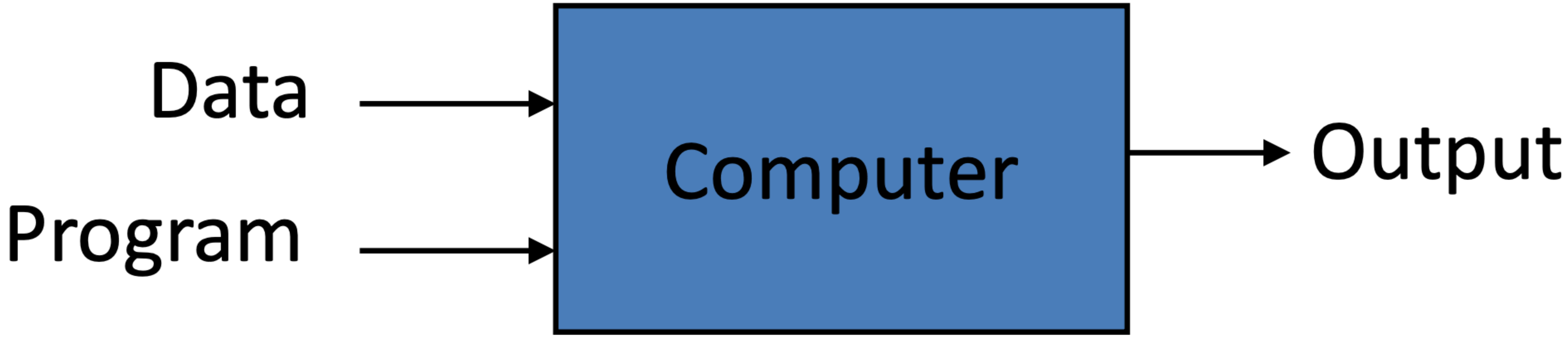
Definition by Tom Mitchell (1998):

Machine Learning is the study of algorithms that

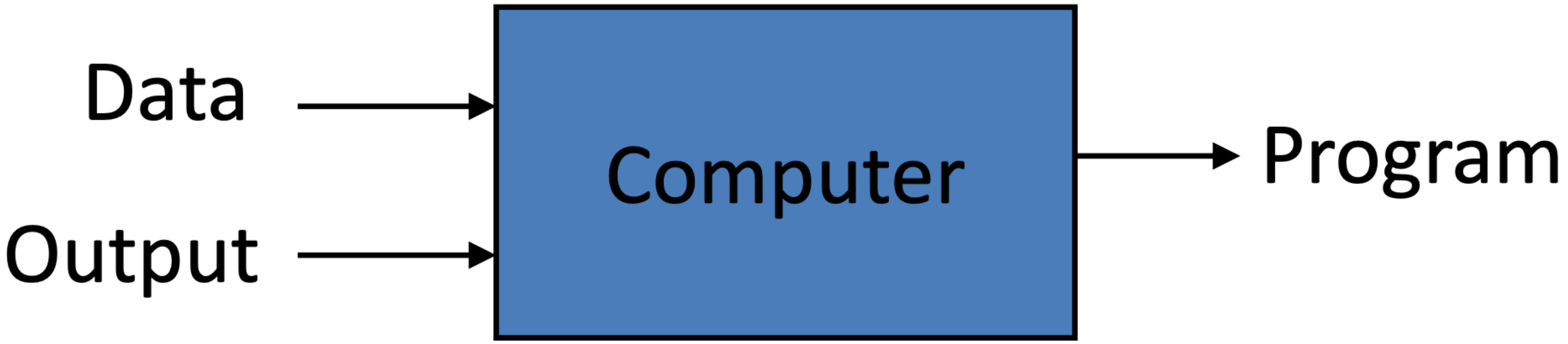
- improve their performance P
- at some task T
- with experience E.

A well-defined learning task is given by <P, T, E>.

Traditional Programming



Machine Learning



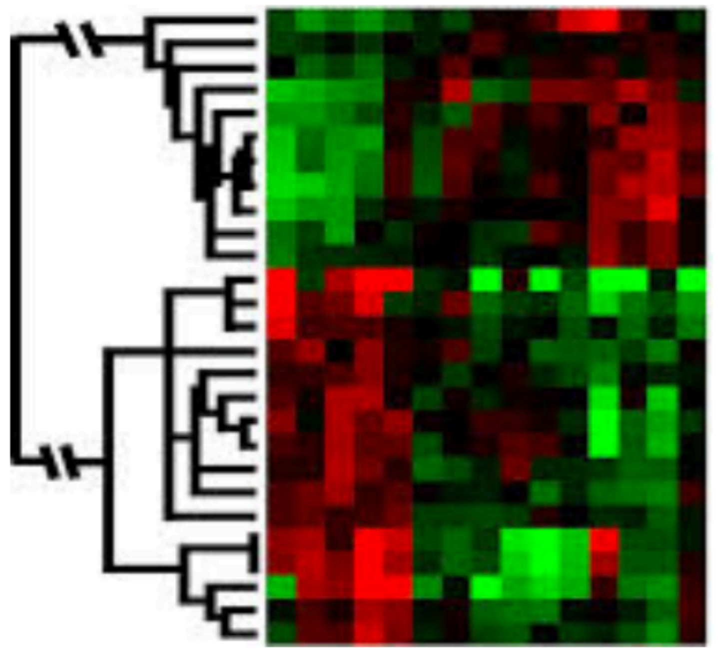
Slide credit: Pedro Domingos

When Do We Use Machine Learning?

ML is used when:

- Human expertise does not exist (navigating on Mars)
- Humans can't explain their expertise (speech recognition)
- Models must be customized (personalized medicine)
- Models are based on huge amounts of data (genomics)



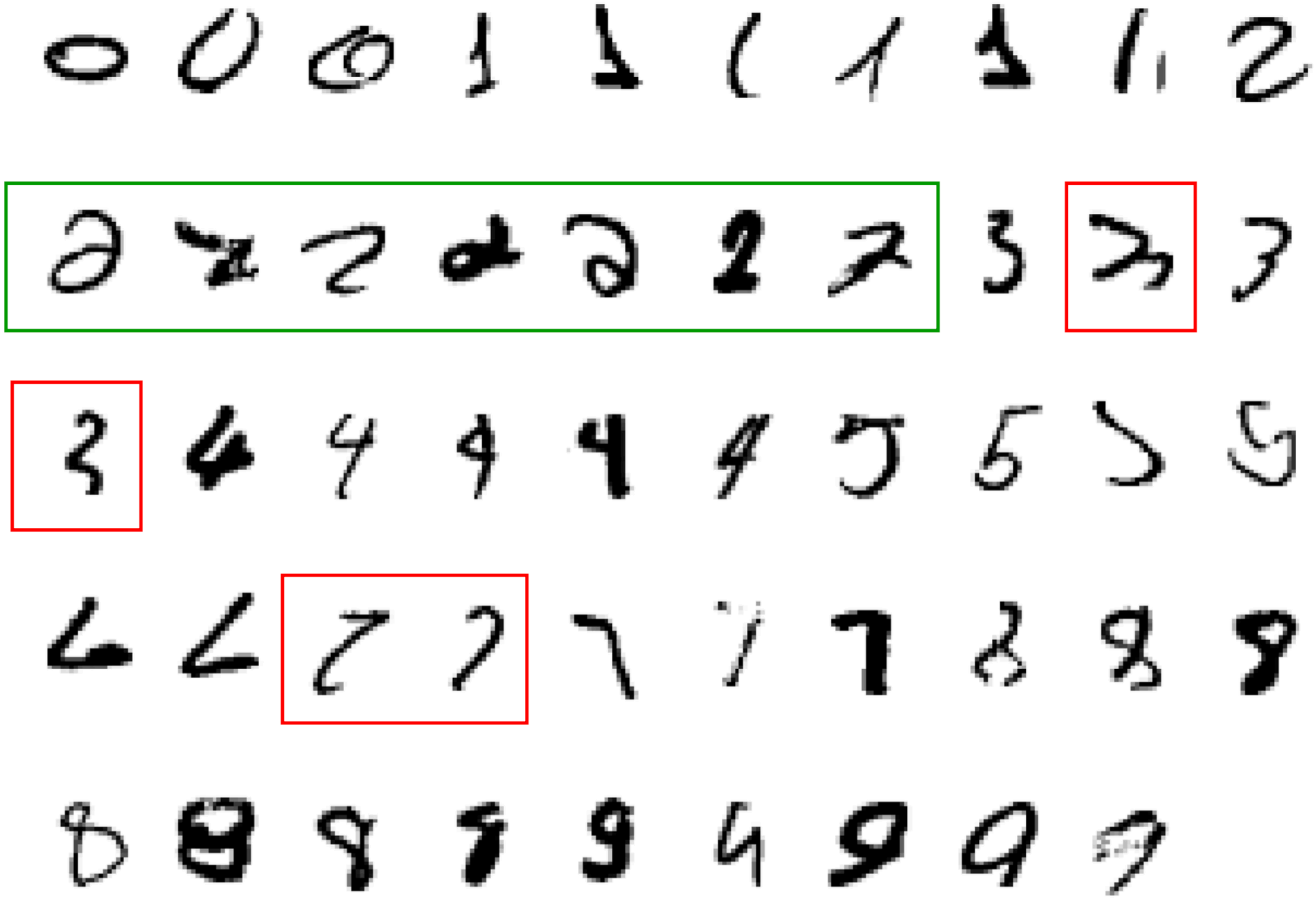



Learning isn't always useful:

- There is no need to "learn" to calculate payroll

Based on slide by E. Alpaydin

A classic example of a task that requires machine learning:
It is very hard to say what makes a 2



Slide credit: Geoffrey Hinton

Some more examples

- Recognizing patterns:
  - Facial identities or facial expressions
  - Handwritten or spoken words
  - Medical images
- Generating patterns:
  - Generating images or motion sequences
- Recognizing anomalies:
  - Unusual credit card transactions
  - Unusual patterns of sensor readings in a nuclear power plant
- Prediction:
  - Future stock prices or currency exchange rates

Sample Applications

- Web search
- Computational biology
- Finance
- E-commerce
- Space exploration
- Robotics
- Information extraction
- Social networks
- Debugging software
- [Your favorite area]

Challenges in Robotics

Real-world variations

Environment uncertainty

Need for adaptation

Samuel’s Checkers-Player

“Machine Learning: Field of study that gives computers the ability to learn without being explicitly programmed.” -Arthur Samuel (1959)

Defining the Learning Task

Improve on task T, with respect to
performance metric P, based on experience E

Defining the Learning Task

Improve on task T, with respect to
performance metric P, based on experience E

T: Playing checkers
P: Percentage of games won against an arbitrary opponent
E: Playing practice games against itself

Defining the Learning Task

Improve on task T, with respect to
performance metric P, based on experience E

T: Playing checkers
P: Percentage of games won against an arbitrary opponent
E: Playing practice games against itself

T: Recognizing hand-written words
P: Percentage of words correctly classified
E: Database of human-labeled images of handwritten words

Defining the Learning Task

Improve on task T, with respect to
performance metric P, based on experience E

T: Playing checkers
P: Percentage of games won against an arbitrary opponent
E: Playing practice games against itself

T: Recognizing hand-written words
P: Percentage of words correctly classified
E: Database of human-labeled images of handwritten words

T: Driving on four-lane highways using vision sensors
P: Average distance traveled before a human-judged error
E: A sequence of images and steering commands recorded while
observing a human driver.

Defining the Learning Task

Improve on task T, with respect to performance metric P, based on experience E

T: Playing checkers
P: Percentage of games won against an arbitrary opponent
E: Playing practice games against itself

T: Recognizing hand-written words
P: Percentage of words correctly classified
E: Database of human-labeled images of handwritten words

T: Driving on four-lane highways using vision sensors
P: Average distance traveled before a human-judged error
E: A sequence of images and steering commands recorded while observing a human driver.

T: Categorize email messages as spam or legitimate.
P: Percentage of email messages correctly classified.
E: Database of emails, some with human-given labels

Slide credit: Ray Mooney

State of the Art Applications of Machine Learning

Robotaxis Can Now Work the Streets of San Francisco 24/7

Robotaxis can offer paid rides in San Francisco around the clock after Alphabet's Waymo and GM's Cruise got approval from the California Public Utilities Commission.

PHOTOGRAPH: SHIIKO ALEXANDER/ALAMY

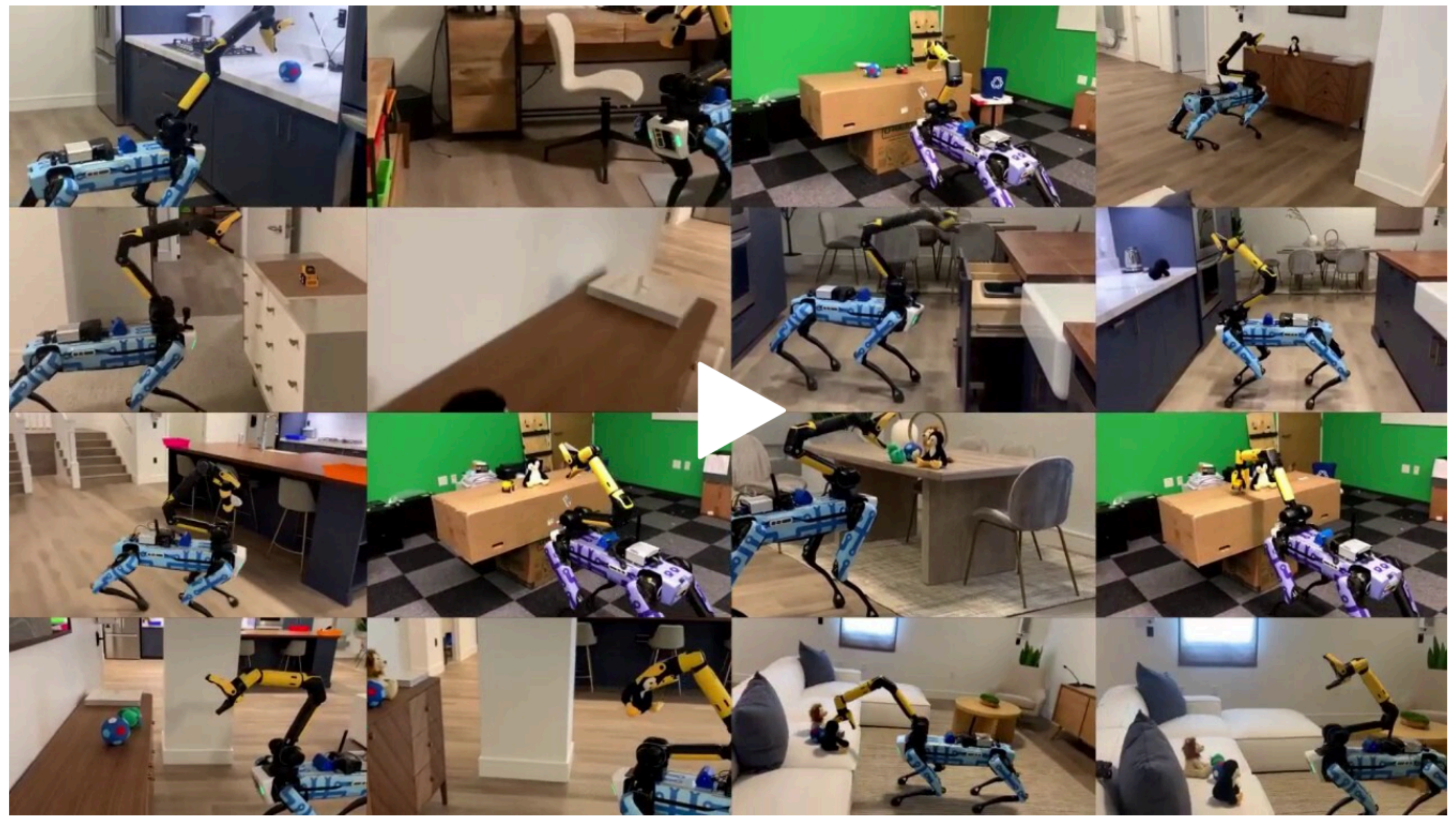

cruise

FACEBOOK · **Published** April 1, 2023 5:08pm EDT

Facebook parent Meta touts Artificial Intelligence robot that can learn from humans

Meta said that its robot was able to rearrange objects inside of a lab and apartment

By **Adam Sabes** | **FOXBusiness** |



Meta's AI Robot rearranges a 'variety of objects'

In announcing the second development, Meta's FAIR team says that it has used adaptive (sensorimotor) skill coordination (ASC) on a Boston Dynamics' Spot robot to "rearrange a variety of objects" in a "185-square-meter apartment and a 65-square-meter university lab." (Credit: Meta)

U.S. Stock Market Quotes

Quotes displayed in real-time

Quote Lookup

DOW JONES AVERAGES	(I:DJI)
34,789.49	▲67.58 (+0.19%)
NASDAQ COMPOSITE INDEX	(I:COMP)
14,023.75	▼-11.21 (-0.08%)
S&P 500	(SP500)
4,511.80	▲4.14 (+0.09%)

Robotics

Google's DeepMind team highlights new system for teaching robots novel tasks

Brian Heater @bheater / 12:18 PM CDT • July 28, 2023

Comment

WILL KNIGHT BUSINESS AUG 16, 2022 10:00 AM

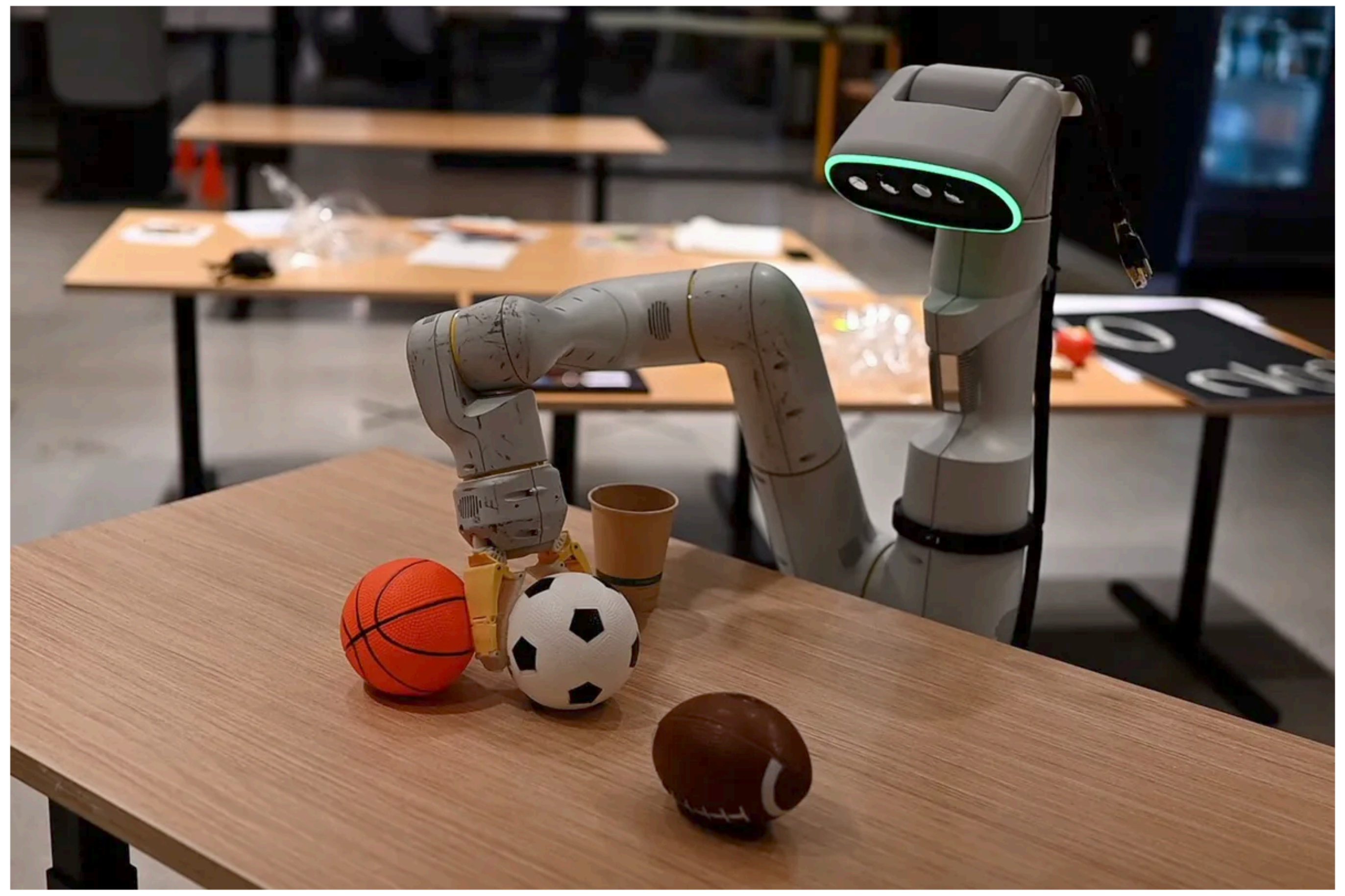
Google's New Robot Learned to Take Orders by Scraping the Web

The machine learning technique that taught notorious text generator GPT-3 to write can also help robots make sense of spoken commands.

COURTESY OF GOOGLE

ARTIFICIAL INTELLIGENCE / TECH

Google is training robots the way it trains AI chatbots



Google robot choosing a ball. Image: Google

/ Google's new robots don't need complex instructions now that they can access large language models.

By Emilia David, a reporter who covers AI. Prior to joining The Verge, she covered the intersection between technology, finance, and the economy.

Jul 28, 2023, 11:47 AM CDT | 6 Comments / 6 New

RT-2: Vision-Language-Action Models

Transfer Web Knowledge to Robotic Control

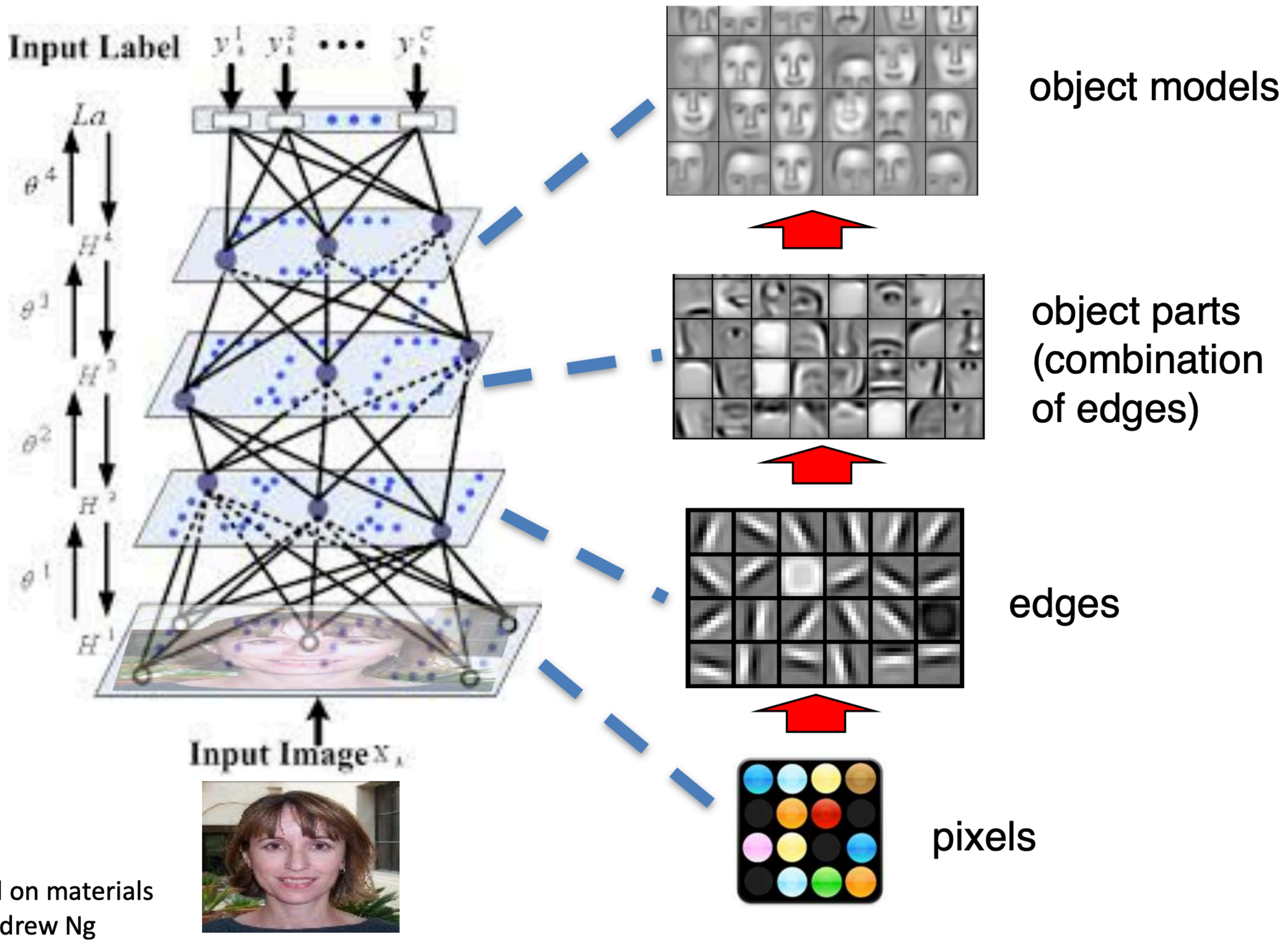
Anthony Brohan Noah Brown Justice Carbajal Yevgen Chebotar Xi Chen Krzysztof Choromanski Tianli Ding
Danny Driess Avinava Dubey Chelsea Finn Pete Florence Chuyuan Fu Montse Gonzalez Arenas Keerthana Gopalakrishnan
Kehang Han Karol Hausman Alex Herzog Jasmine Hsu Brian Ichter Alex Irpan Nikhil Joshi Ryan Julian
Dmitry Kalashnikov Yuheng Kuang Isabel Leal Lisa Lee Tsang-Wei Edward Lee Sergey Levine Yao Lu Henryk Michalewski
Igor Mordatch Karl Pertsch Kanishka Rao Krista Reymann Michael Ryoo Grecia Salazar Pannag Sanketi Pierre Sermanet
Jaspiar Singh Anikait Singh Radu Soricut Huong Tran Vincent Vanhoucke Quan Vuong Ayzaan Wahid Stefan Welker
Paul Wohlhart Jialin Wu Fei Xia Ted Xiao Peng Xu Sichun Xu Tianhe Yu Brianna Zitkovich

Authors listed in alphabetical order (see paper appendix for contribution statement).

Google DeepMind

Paper Blogpost Demo Videos RT-1

Deep Belief Net on Face Images



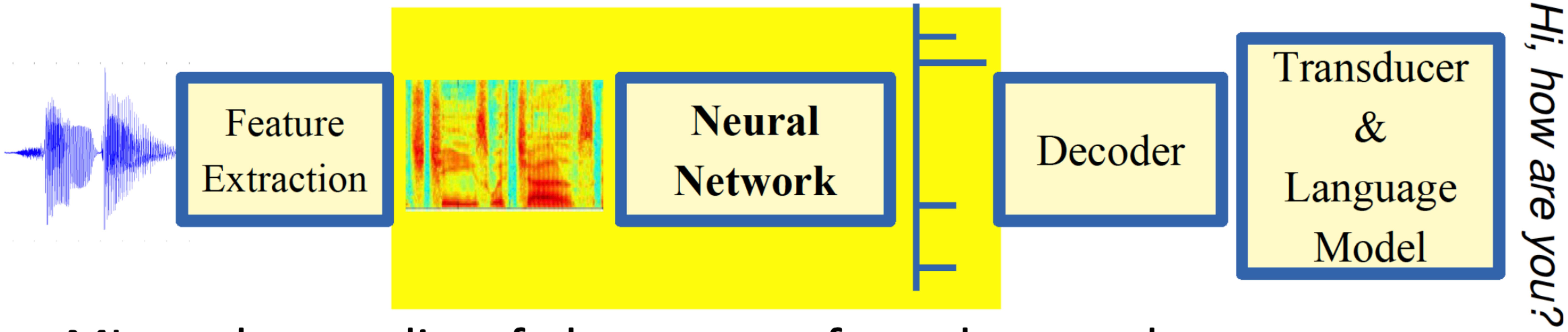
Based on materials
by Andrew Ng

Scene Labeling via Deep Learning

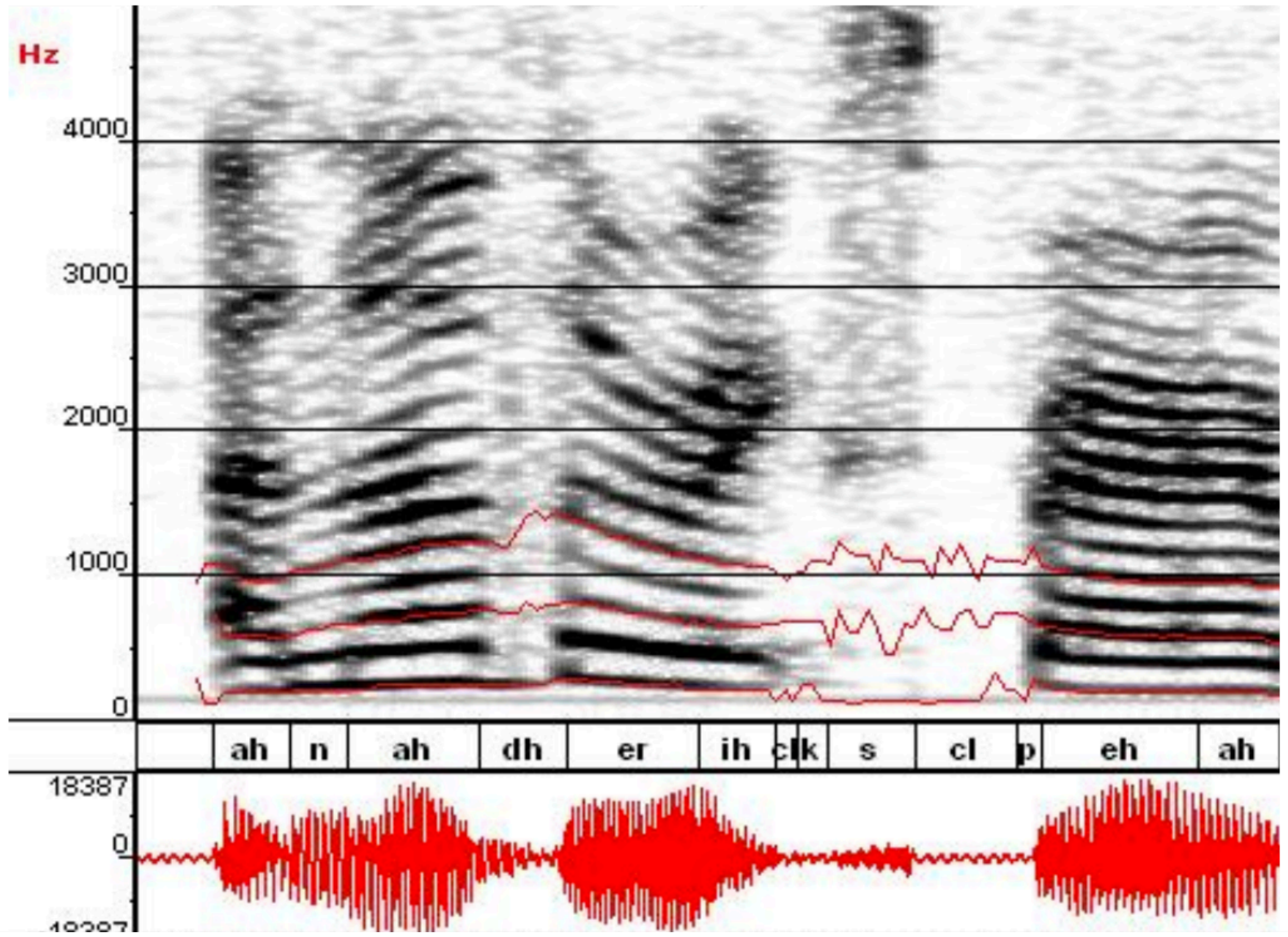
[Farabet et al. ICML 2012, PAMI 2013]

Machine Learning in Automatic Speech Recognition

A Typical Speech Recognition System



ML used to predict of phone states from the sound spectrogram



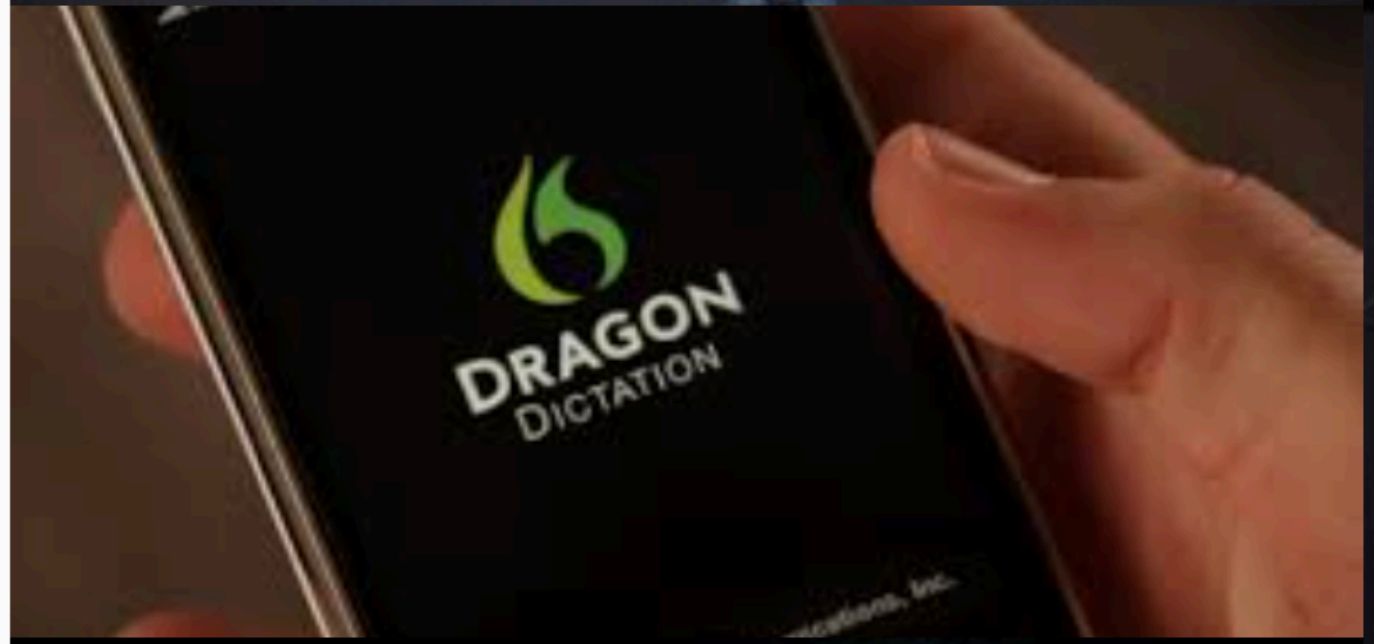
Deep learning has state-of-the-art results

# Hidden Layers	1	2	4	8	10	12
Word Error Rate %	16.0	12.8	11.4	10.9	11.0	11.1

Baseline GMM performance = 15.4%

[Zeiler et al. “On rectified linear units for speech recognition” ICASSP 2013]

Impact of Deep Learning in Speech Technology



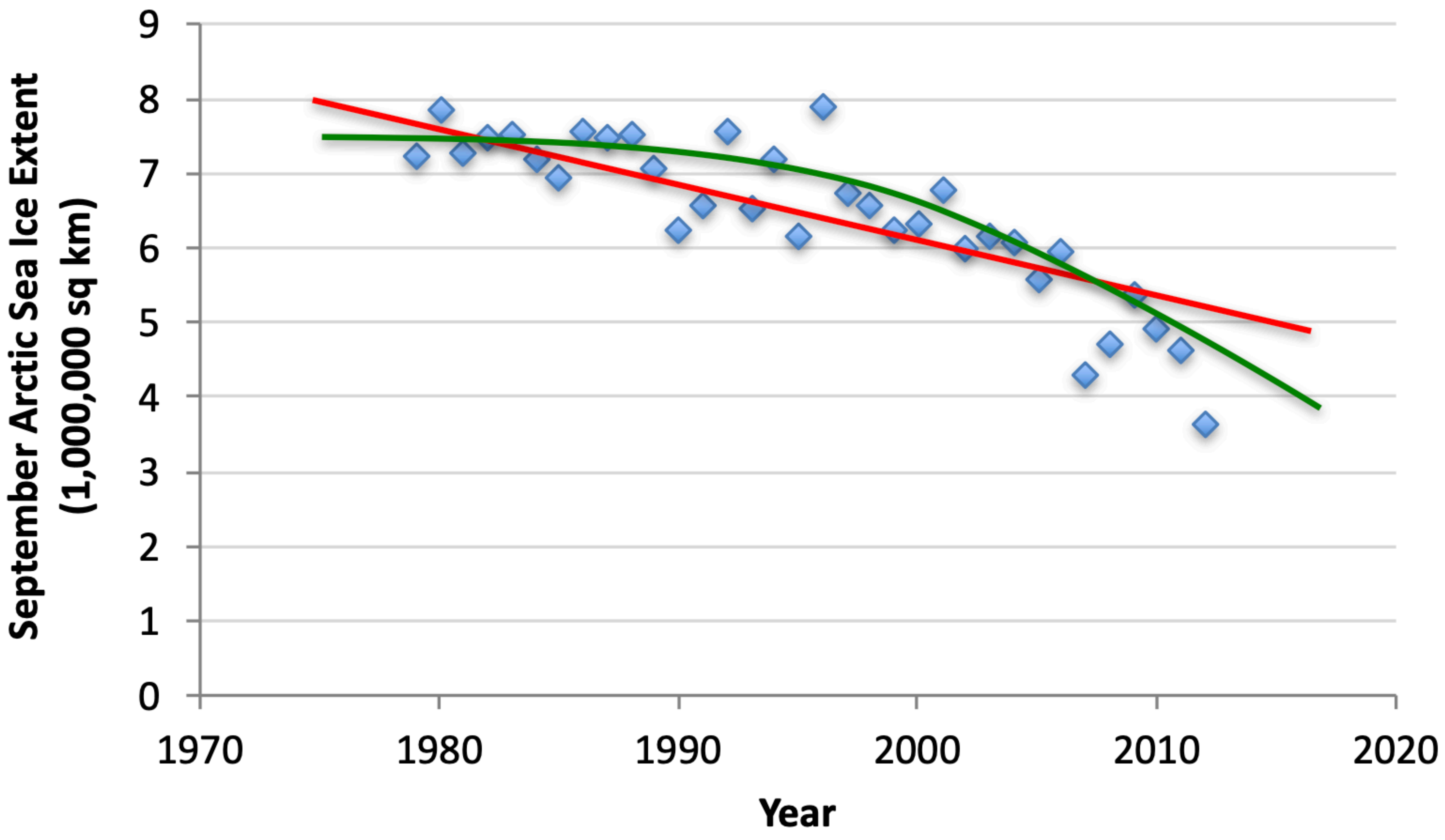
Slide credit: Li Deng, MS Research

Types of Learning

- **Supervised learning**
  - Given: training data + desired outputs (labels)
- **Unsupervised learning**
  - Given: training data (without desired outputs)
- **Semi-supervised learning**
  - Given: training data + a few desired outputs
- **Reinforcement learning**
  - Rewards from sequence of actions

Supervised Learning: Regression

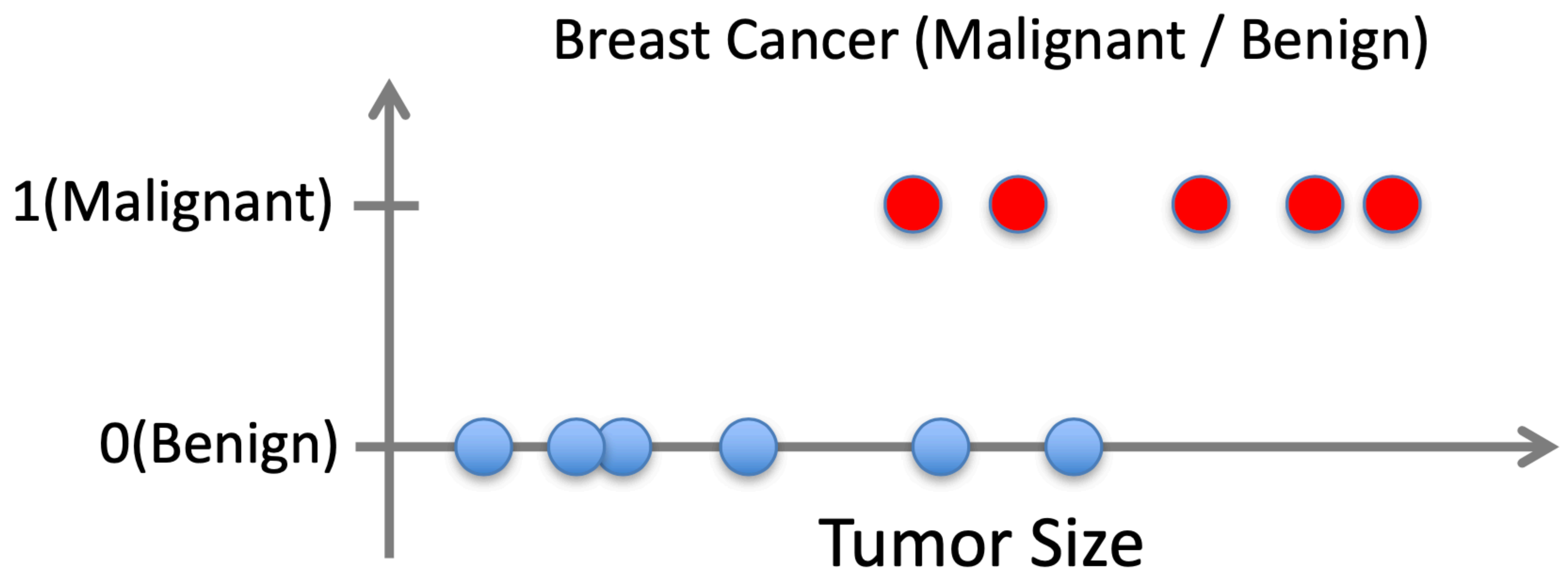
- Given $(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)$
- Learn a function $f(x)$ to predict y given x
  - y is real-valued == regression



Data from G. Witt. Journal of Statistics Education, Volume 21, Number 1 (2013)

Supervised Learning: Classification

- Given $(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)$
- Learn a function $f(x)$ to predict y given x
  - y is categorical == classification

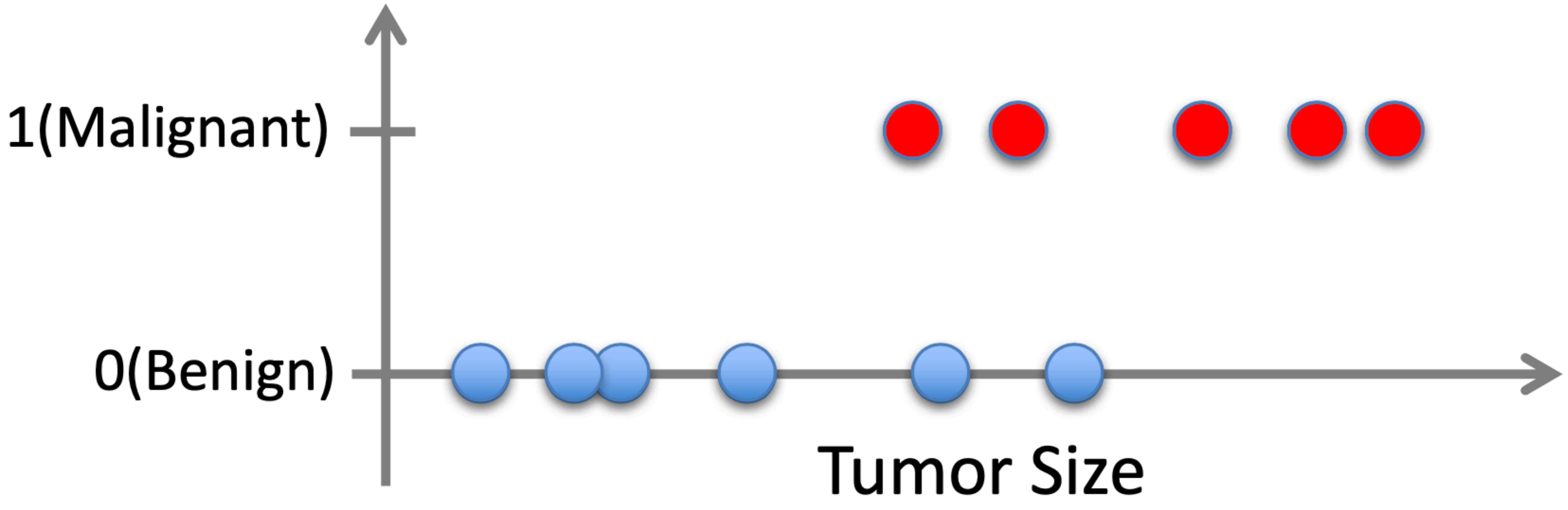


Based on example by Andrew Ng

Supervised Learning: Classification

- Given $(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)$
- Learn a function $f(x)$ to predict y given x
  - y is categorical == classification

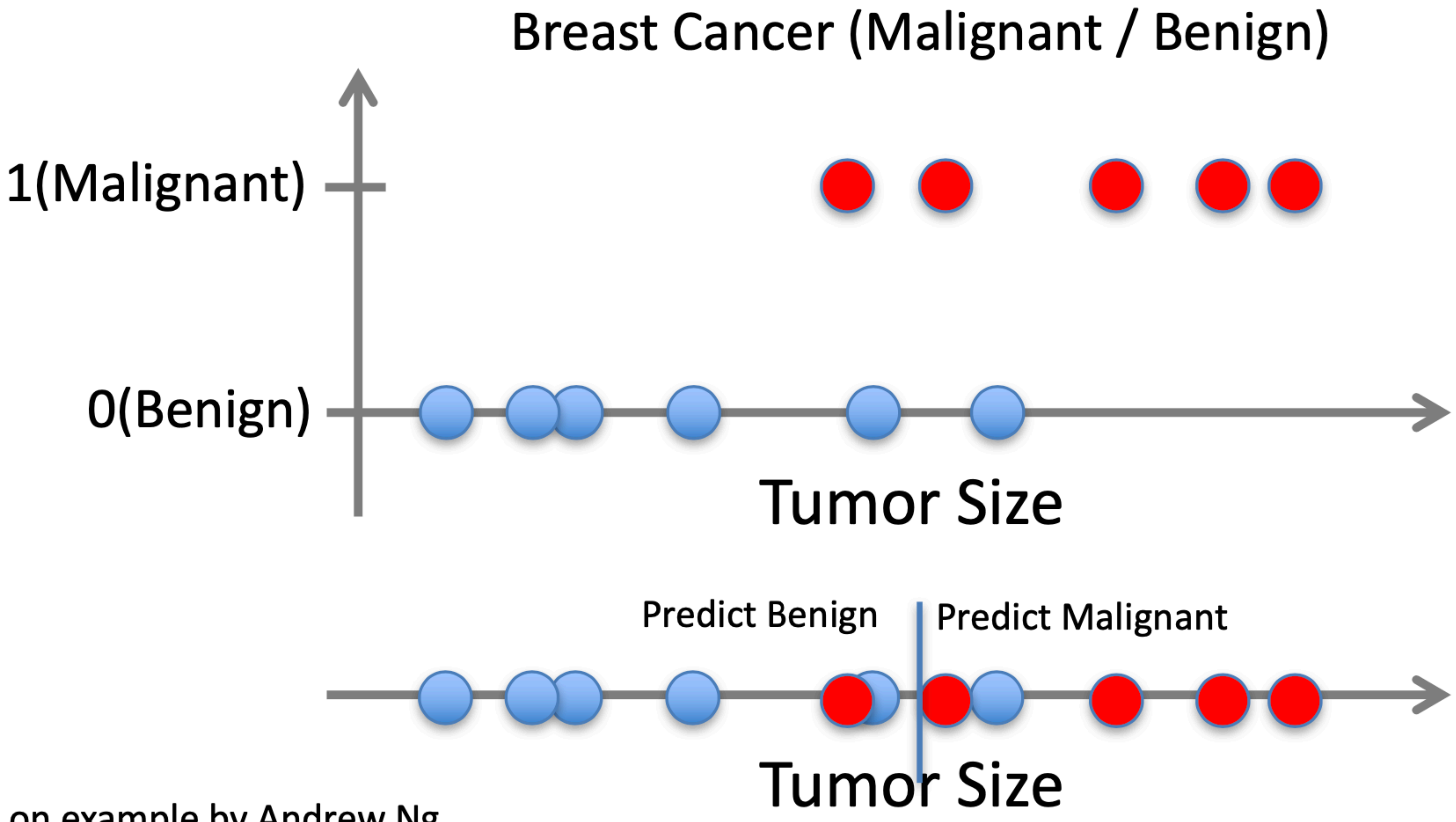
Breast Cancer (Malignant / Benign)



Based on example by Andrew Ng

Supervised Learning: Classification

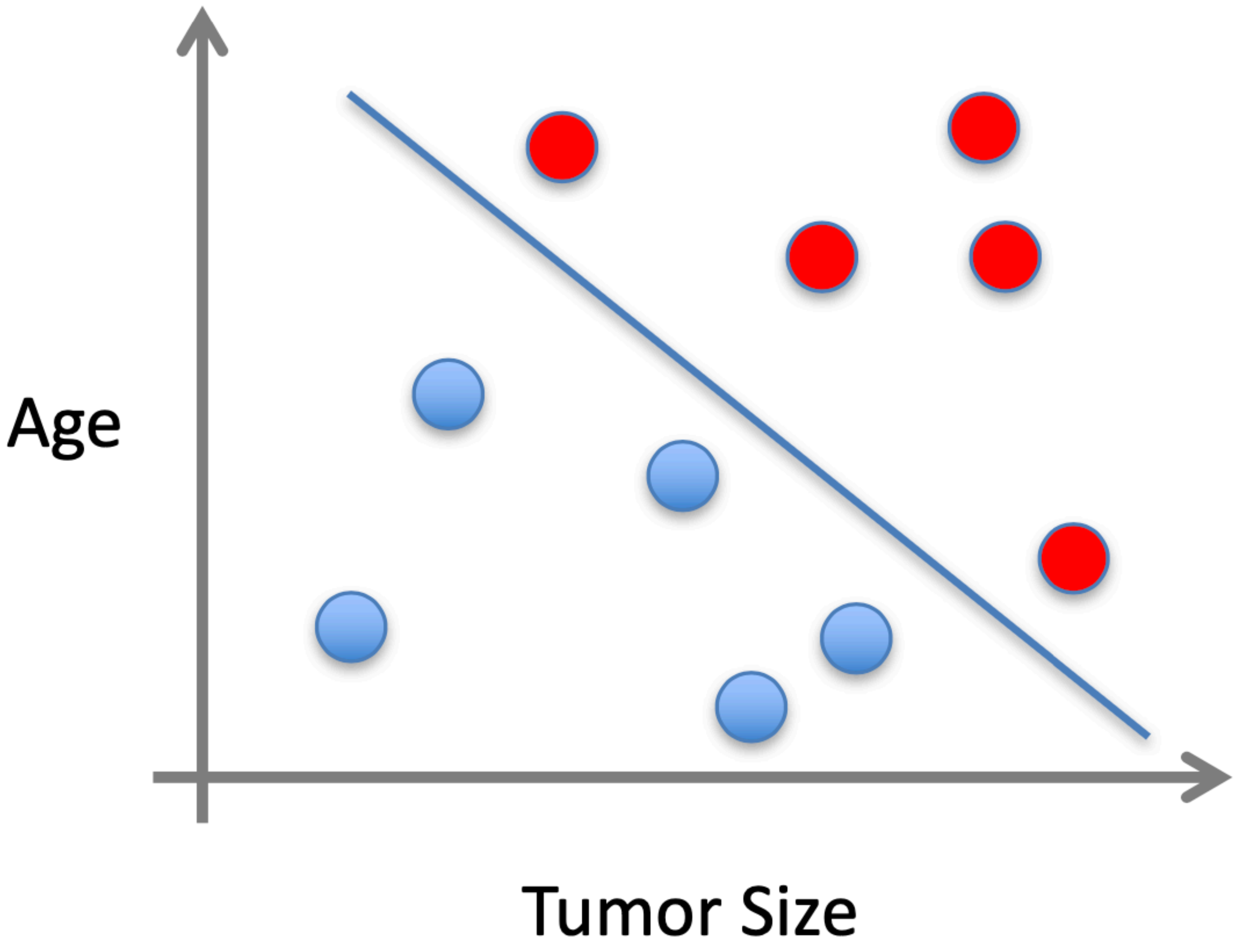
- Given $(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)$
- Learn a function $f(x)$ to predict y given x
  - y is categorical == classification



Based on example by Andrew Ng

Supervised Learning

- x can be multi-dimensional
  – Each dimension corresponds to an attribute



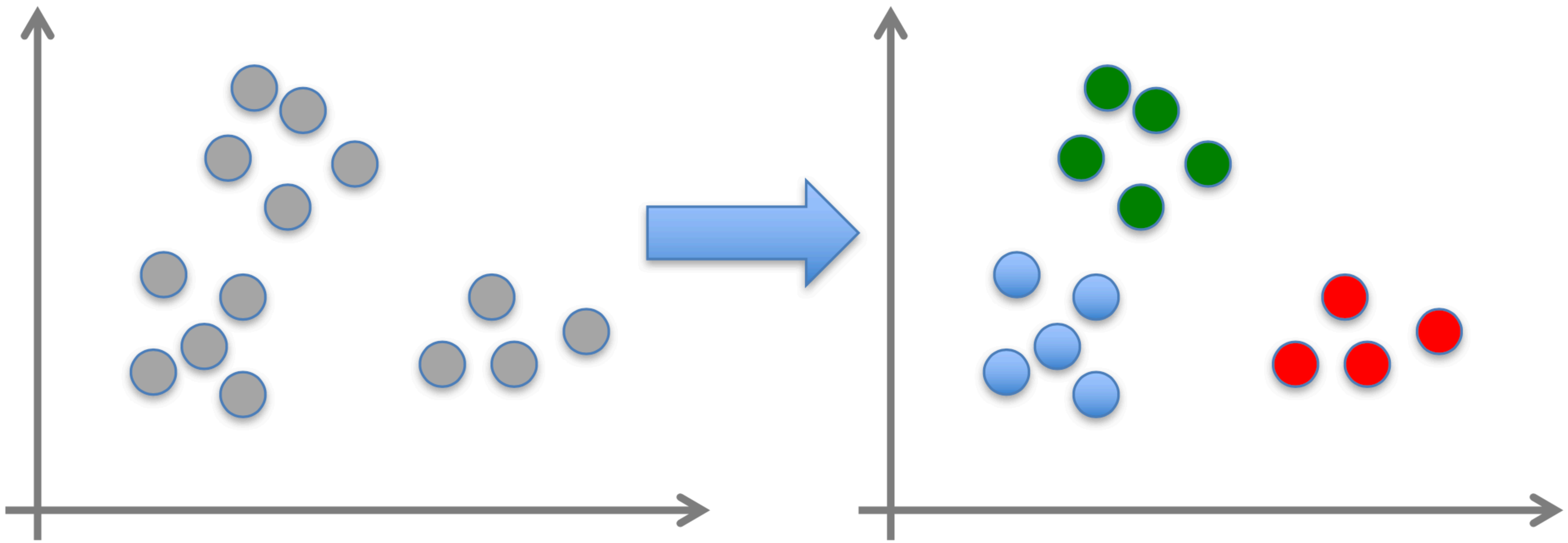
- Clump Thickness
- Uniformity of Cell Size
- Uniformity of Cell Shape

...

Based on example by Andrew Ng

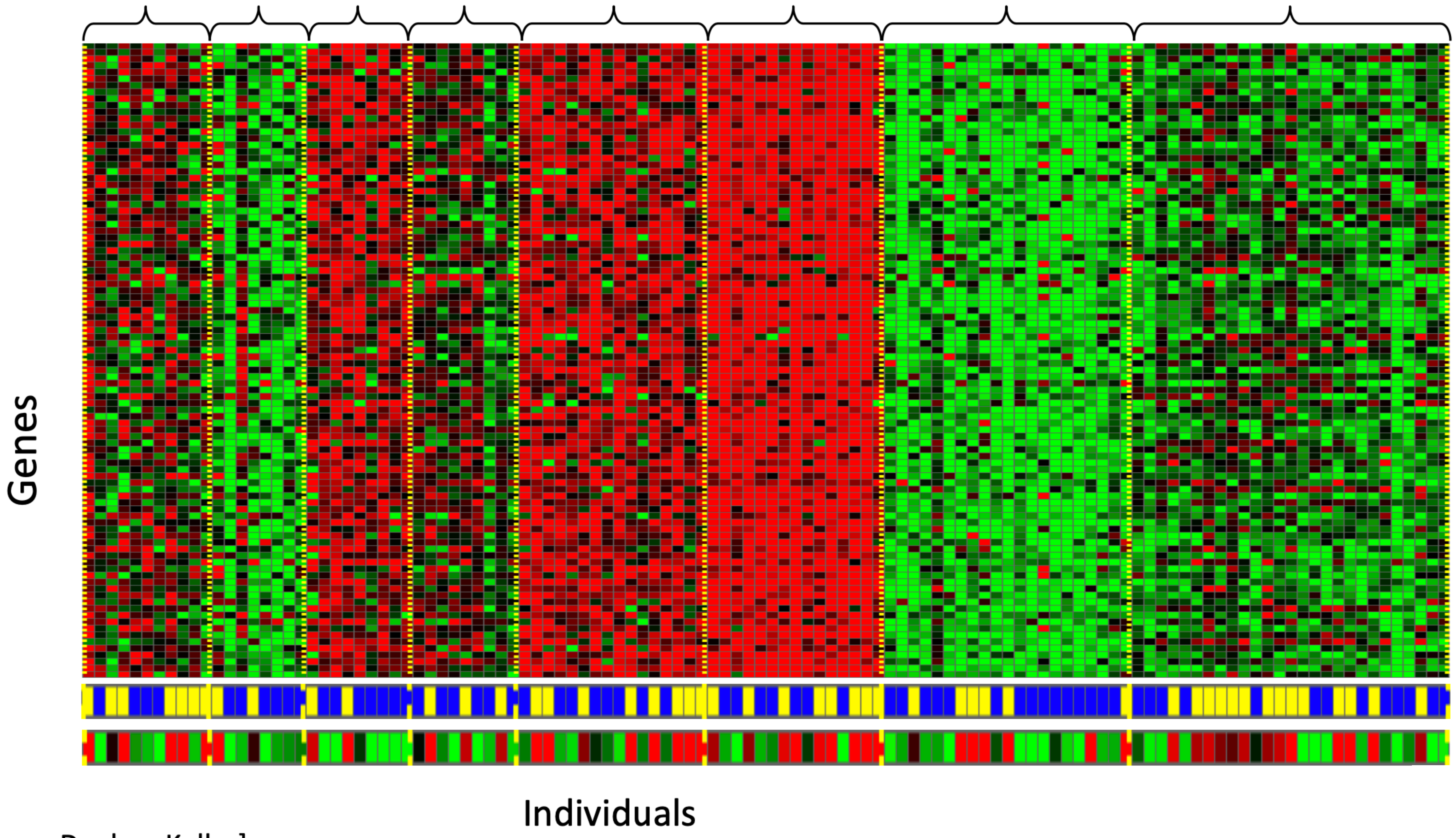
Unsupervised Learning

- Given x_1, x_2, ..., x_n (without labels)
- Output hidden structure behind the x's
  - E.g., clustering



Unsupervised Learning

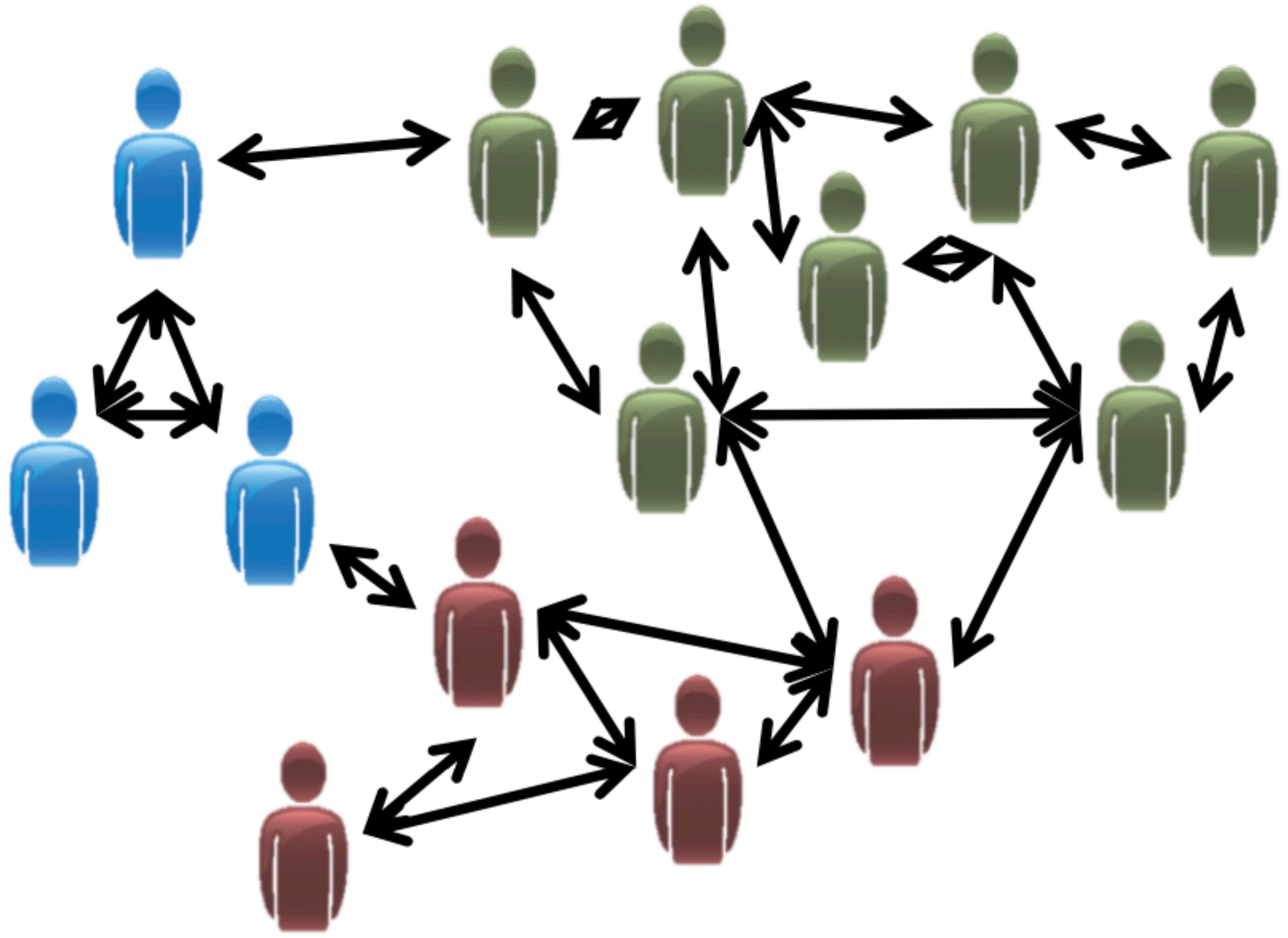
Genomics application: group individuals by genetic similarity



[Source: Daphne Koller]

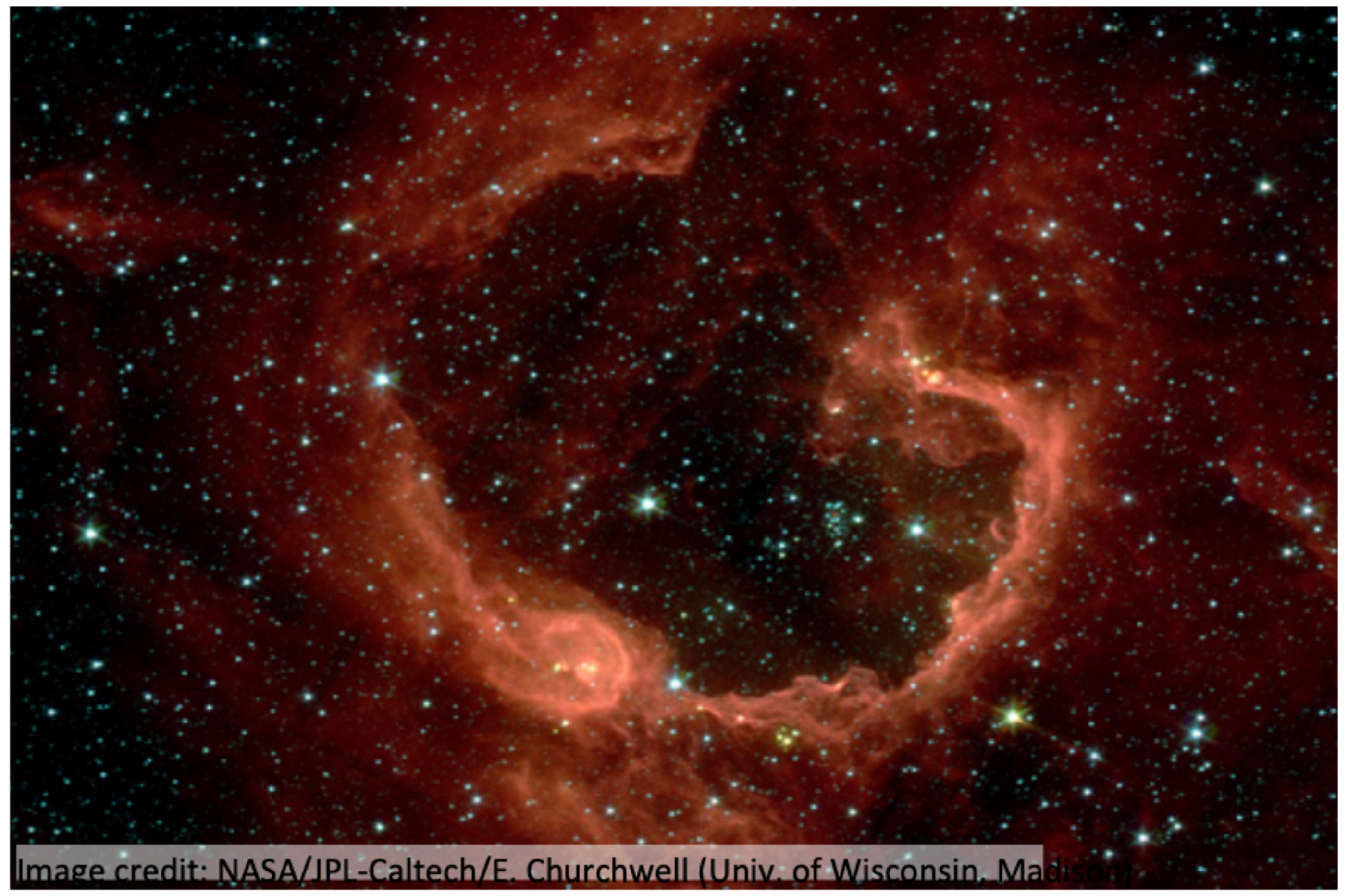
Unsupervised Learning

Organize computing clusters



Social network analysis

Market segmentation



Astronomical data analysis

Slide credit: Andrew Ng

Unsupervised Learning

- Independent component analysis – separate a combined signal into its original sources

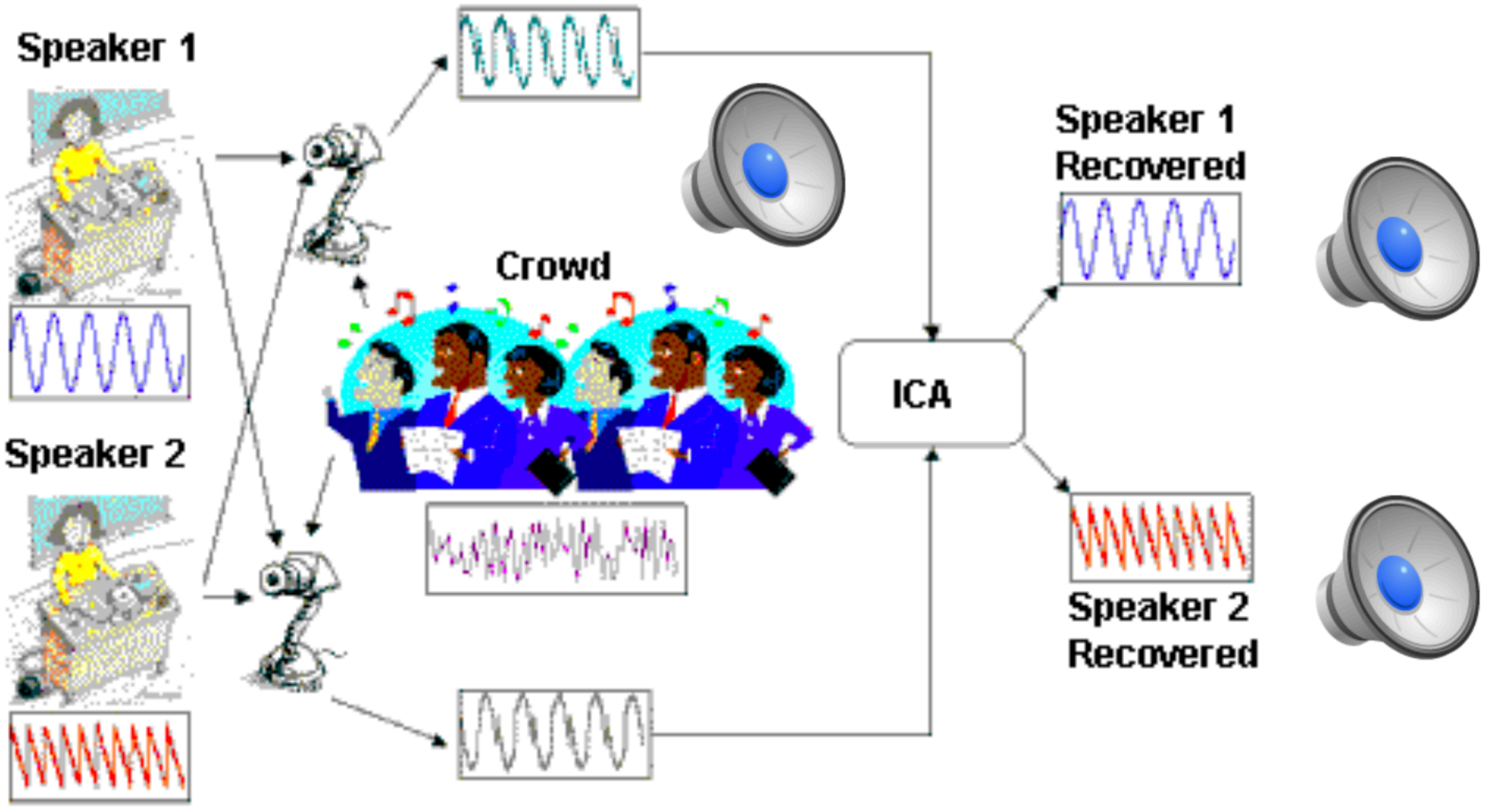
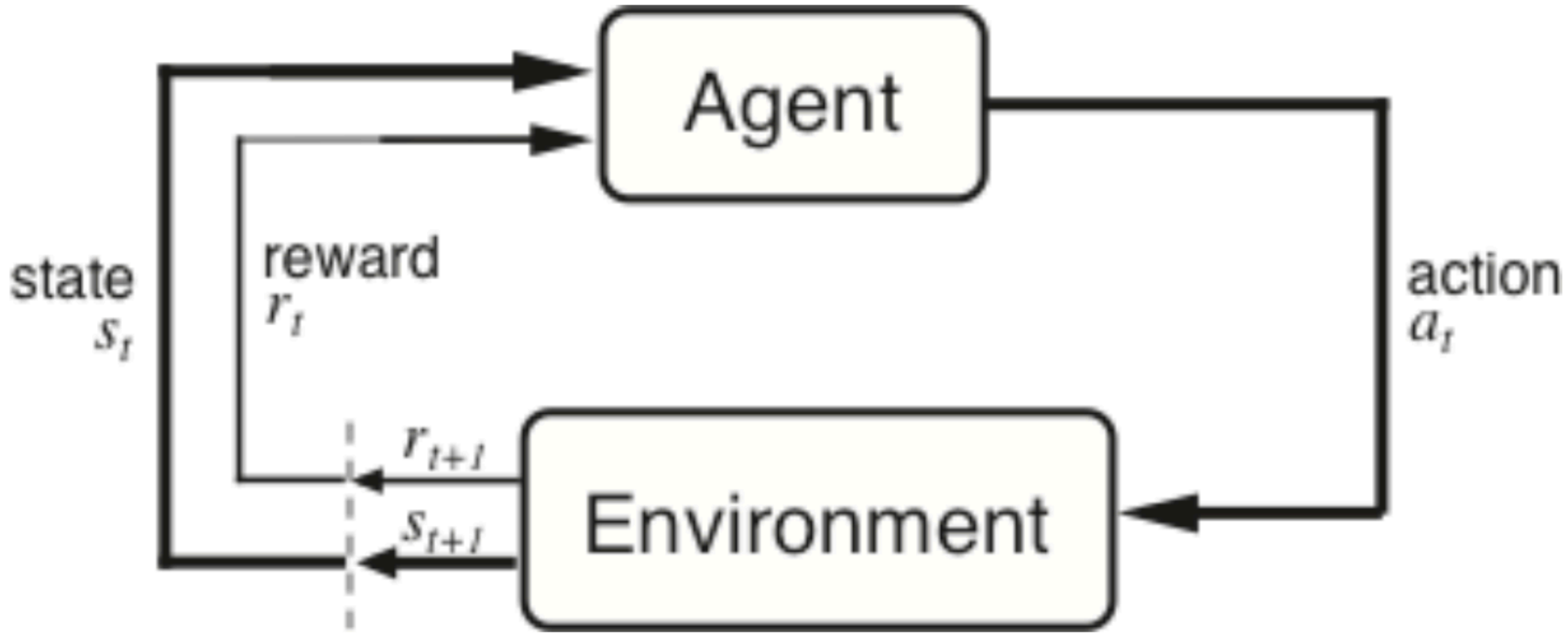


Image credit: statsoft.com Audio from http://www.ism.ac.jp/~shiro/research/blindsep.html

Reinforcement Learning

- Given a sequence of states and actions with (delayed) rewards, output a policy
  - Policy is a mapping from states → actions that tells you what to do in a given state
- Examples:
  - Credit assignment problem
  - Game playing
  - Robot in a maze
  - Balance a pole on your hand

The Agent-Environment Interface



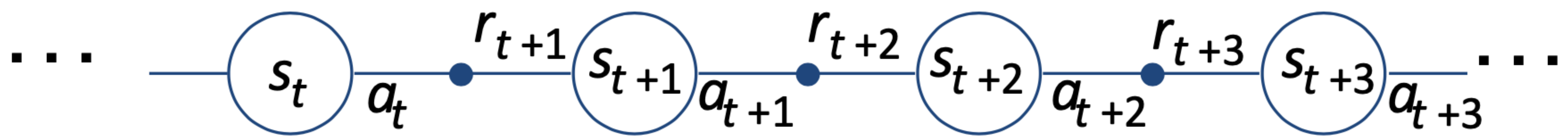
Agent and environment interact at discrete time steps : $t = 0, 1, 2, K$

Agent observes state at step t: $s_t \in S$

produces action at step t: $a_t \in A(s_t)$

gets resulting reward : $r_{t+1} \in \Re$

and resulting next state : s_{t+1}



Slide credit: Sutton & Barto


[~ Mario AI Benchmark ~ 0.1.9]
DIFFICULTY: 00
SEED:6
TYPE:Overground(0)
LENGTH:2 of 256
HEIGHT:2 of 15
Agent: QLearner
PRESSED KEYS:
ALL KILLS: 0
by Fire : 0
by Shell : 0
by Stomp : 0
TIME
199
FPS:
24

Inverse Reinforcement Learning

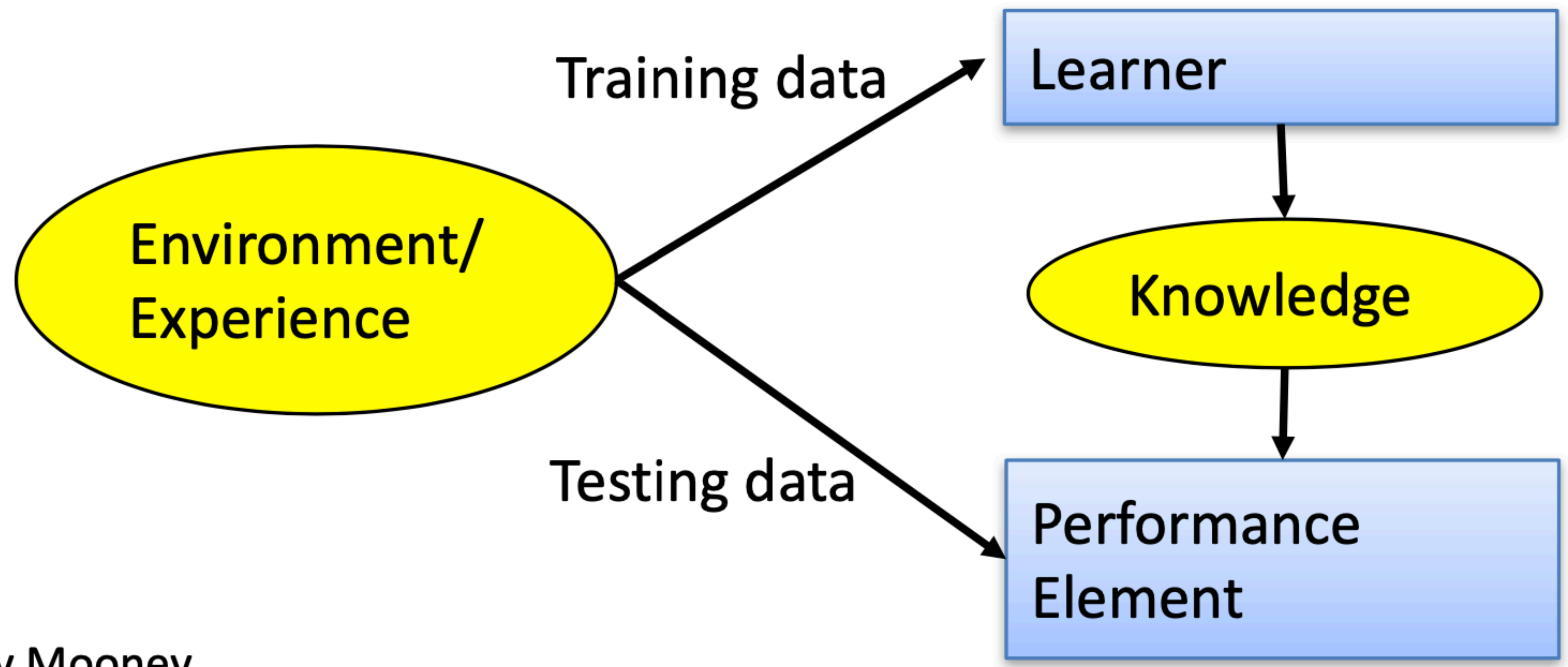
- Learn policy from user demonstrations

Stanford Autonomous Helicopter
http://heli.stanford.edu/
https://www.youtube.com/watch?v=VCdxqn0fcnE

Framing a Learning Problem

Designing a Learning System

- Choose the training experience
- Choose exactly what is to be learned
  - i.e. the ***target function***
- Choose how to represent the target function
- Choose a learning algorithm to infer the target function from the experience



Based on slide by Ray Mooney

Training vs. Test Distribution

- We generally assume that the training and test examples are independently drawn from the same overall distribution of data
  - We call this “i.i.d” which stands for “independent and identically distributed”

- If examples are not independent, requires ***collective classification***

- If test distribution is different, requires ***transfer learning***

Slide credit: Ray Mooney

ML in a Nutshell

- Tens of thousands of machine learning algorithms
  - Hundreds new every year

- Every ML algorithm has three components:
  - **Representation**
  - **Optimization**
  - **Evaluation**

Slide credit: Pedro Domingos

Various Function Representations

Various Function Representations

- Numerical functions
  - Linear regression
  - Neural networks
  - Support vector machines

Various Function Representations

- Numerical functions
  - Linear regression
  - Neural networks
  - Support vector machines
- Symbolic functions
  - Decision trees
  - Rules in propositional logic
  - Rules in first-order predicate logic

Various Function Representations

- Numerical functions
  - Linear regression
  - Neural networks
  - Support vector machines
- Symbolic functions
  - Decision trees
  - Rules in propositional logic
  - Rules in first-order predicate logic
- Instance-based functions
  - Nearest-neighbor
  - Case-based

Various Function Representations

- Numerical functions
  - Linear regression
  - Neural networks
  - Support vector machines
- Symbolic functions
  - Decision trees
  - Rules in propositional logic
  - Rules in first-order predicate logic
- Instance-based functions
  - Nearest-neighbor
  - Case-based
- Probabilistic Graphical Models
  - Naïve Bayes
  - Bayesian networks
  - Hidden-Markov Models (HMMs)
  - Probabilistic Context Free Grammars (PCFGs)
  - Markov networks

Slide credit: Ray Mooney

Various Search/Optimization Algorithms

Various Search/Optimization Algorithms

- Gradient descent
  - Perceptron
  - Backpropagation

Various Search/Optimization Algorithms

- Gradient descent
  - Perceptron
  - Backpropagation
- Dynamic Programming
  - HMM Learning
  - PCFG Learning

Various Search/Optimization Algorithms

- Gradient descent
  - Perceptron
  - Backpropagation
- Dynamic Programming
  - HMM Learning
  - PCFG Learning
- Divide and Conquer
  - Decision tree induction
  - Rule learning

Various Search/Optimization Algorithms

- Gradient descent
  - Perceptron
  - Backpropagation
- Dynamic Programming
  - HMM Learning
  - PCFG Learning
- Divide and Conquer
  - Decision tree induction
  - Rule learning
- Evolutionary Computation
  - Genetic Algorithms (GAs)
  - Genetic Programming (GP)
  - Neuro-evolution

Slide credit: Ray Mooney

Evaluation

- Accuracy
- Precision and recall
- Squared error
- Likelihood
- Posterior probability
- Cost / Utility
- Margin
- Entropy
- K-L divergence
- etc.

Slide credit: Pedro Domingos

ML in Practice

Loop

- Understand domain, prior knowledge, and goals
- Data integration, selection, cleaning, pre-processing, etc.
- Learn models
- Interpret results
- Consolidate and deploy discovered knowledge

Based on a slide by Pedro Domingos

Lessons Learned about Learning

- Learning can be viewed as using direct or indirect experience to approximate a chosen target function.

Lessons Learned about Learning

- Learning can be viewed as using direct or indirect experience to approximate a chosen target function.

- Function approximation can be viewed as a search through a space of hypotheses (representations of functions) for one that best fits a set of training data.

Lessons Learned about Learning

- Learning can be viewed as using direct or indirect experience to approximate a chosen target function.
- Function approximation can be viewed as a search through a space of hypotheses (representations of functions) for one that best fits a set of training data.
- Different learning methods assume different hypothesis spaces (representation languages) and/or employ different search techniques.

Slide credit: Ray Mooney

A Brief History of Machine Learning

History of Machine Learning

- 1950s
  - Samuel's checker player
  - Selfridge's Pandemonium
- 1960s:
  - Neural networks: Perceptron
  - Pattern recognition
  - Learning in the limit theory
  - Minsky and Papert prove limitations of Perceptron
- 1970s:
  - Symbolic concept induction
  - Winston's arch learner
  - Expert systems and the knowledge acquisition bottleneck
  - Quinlan's ID3
  - Michalski's AQ and soybean diagnosis
  - Scientific discovery with BACON
  - Mathematical discovery with AM

Slide credit: Ray Mooney

History of Machine Learning (cont.)

- 1980s:
  - Advanced decision tree and rule learning
  - Explanation-based Learning (EBL)
  - Learning and planning and problem solving
  - Utility problem
  - Analogy
  - Cognitive architectures
  - Resurgence of neural networks (connectionism, backpropagation)
  - Valiant's PAC Learning Theory
  - Focus on experimental methodology
- 1990s
  - Data mining
  - Adaptive software agents and web applications
  - Text learning
  - Reinforcement learning (RL)
  - Inductive Logic Programming (ILP)
  - Ensembles: Bagging, Boosting, and Stacking
  - Bayes Net learning

Slide credit: Ray Mooney

History of Machine Learning (cont.)

- 2000s
  - Support vector machines & kernel methods
  - Graphical models
  - Statistical relational learning
  - Transfer learning
  - Sequence labeling
  - Collective classification and structured outputs
  - Computer Systems Applications (Compilers, Debugging, Graphics, Security)
  - E-mail management
  - Personalized assistants that learn
  - Learning in robotics and vision
- 2010s
  - Deep learning systems
  - Learning for big data
  - Bayesian methods
  - Multi-task & lifelong learning
  - Applications to vision, speech, social networks, learning to read, etc.
  - ???

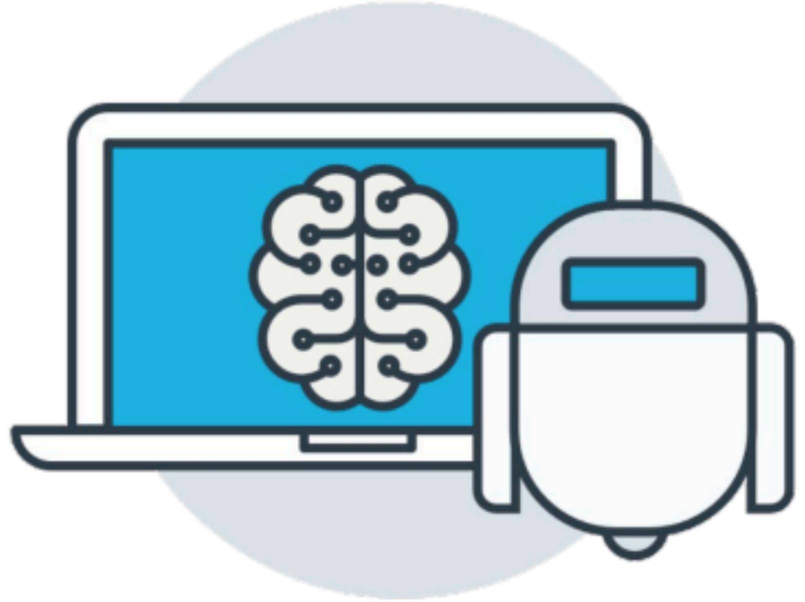
Based on slide by Ray Mooney

Machine Learning for Robotics

General-Purpose Robot Autonomy

general-purpose
robots

Robot Learning



general-purpose
behaviors

Logistics

- 1/30 (Tue)
  - **Kulbir Singh Ahluwalia**
    - DINOv2: Learning Robust Visual Features without Supervision
  - **Xin Xu**
    - SAM: Segment Anything
- 2/1 (Thu)
  - **Jose Cuaran**
    - PoseCNN: A Convolutional Neural Network for 6D Object Pose Estimation in Cluttered Scenes
  - **Kaifeng Zhang**
    - NOCS: Normalized Object Coordinate Space for Category-Level 6D Object Pose and Size Estimation