



# Machine Learning (CSC-465)

## Lec 1: Introduction

Dr. Wazir Ali  
Assistant Professor  
Department of Computer Science  
College of Computer Science and Information Systems  
[wazir.ali@iobm.edu.pk](mailto:wazir.ali@iobm.edu.pk)

The lectures may contain  
the content of  
CIS 419/519 by Eric Eaton

# What is Machine Learning?

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

- Herbert Simon

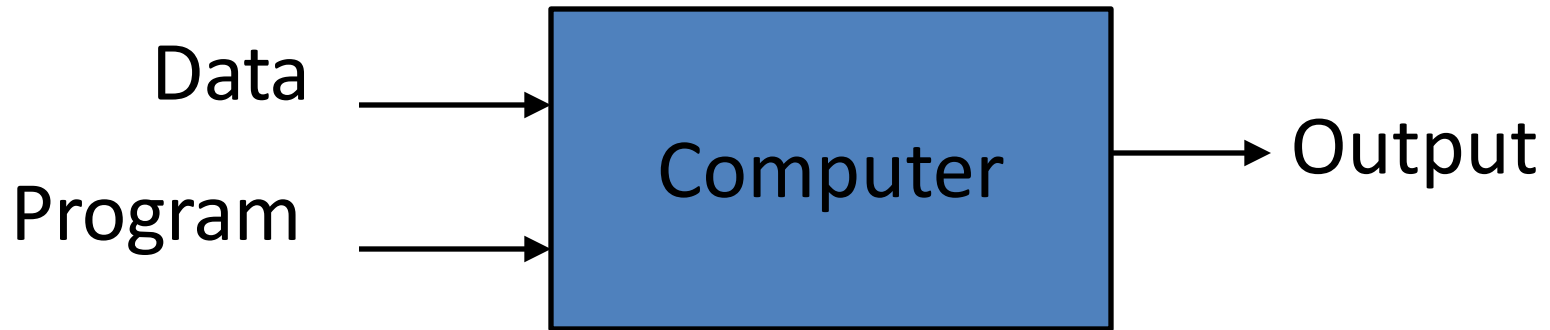
Definition by Tom Mitchell (1998):

Machine Learning is the study of algorithms that

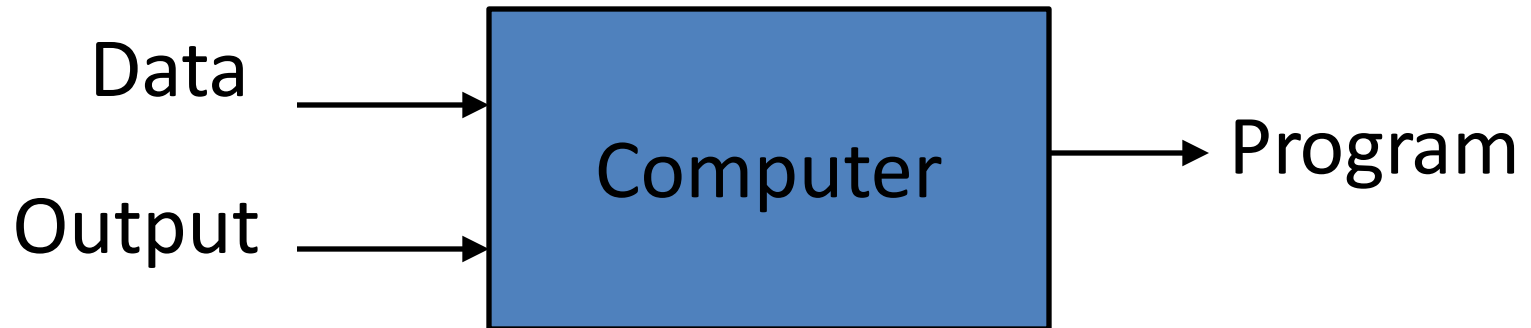
- improve their performance  $P$
- at some task  $T$
- with experience  $E$ .

A well-defined learning task is given by  $\langle P, T, E \rangle$ .

## Traditional Programming



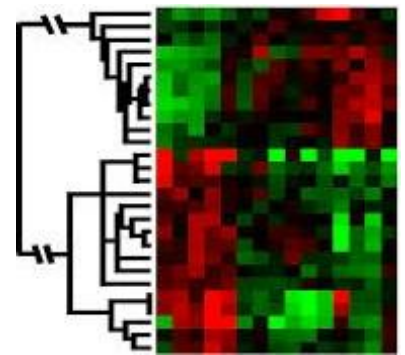
## Machine Learning



# 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

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

0 0 0 1 1 1 1 1 1 2

2 2 2 2 2 2 2 3 3 3

3 4 4 4 4 4 5 5 5 5

6 6 7 7 7 7 7 8 8 8

9 9 9 9 9 9 9 9 9

# Some more examples of tasks that are best solved by using a learning algorithm

- 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]

# 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

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.

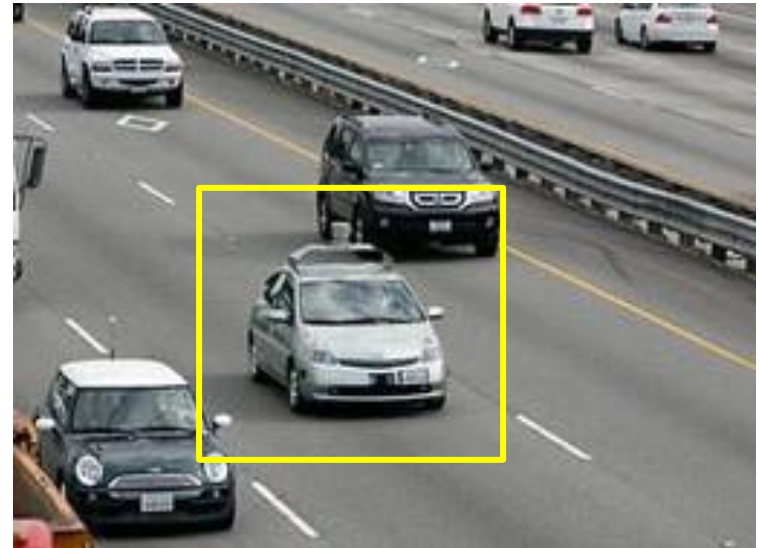
T: Categorize email messages as spam or legitimate.

P: Percentage of email messages correctly classified.

E: Database of emails, some with human-given labels

# State of the Art Applications of Machine Learning

# Autonomous Cars

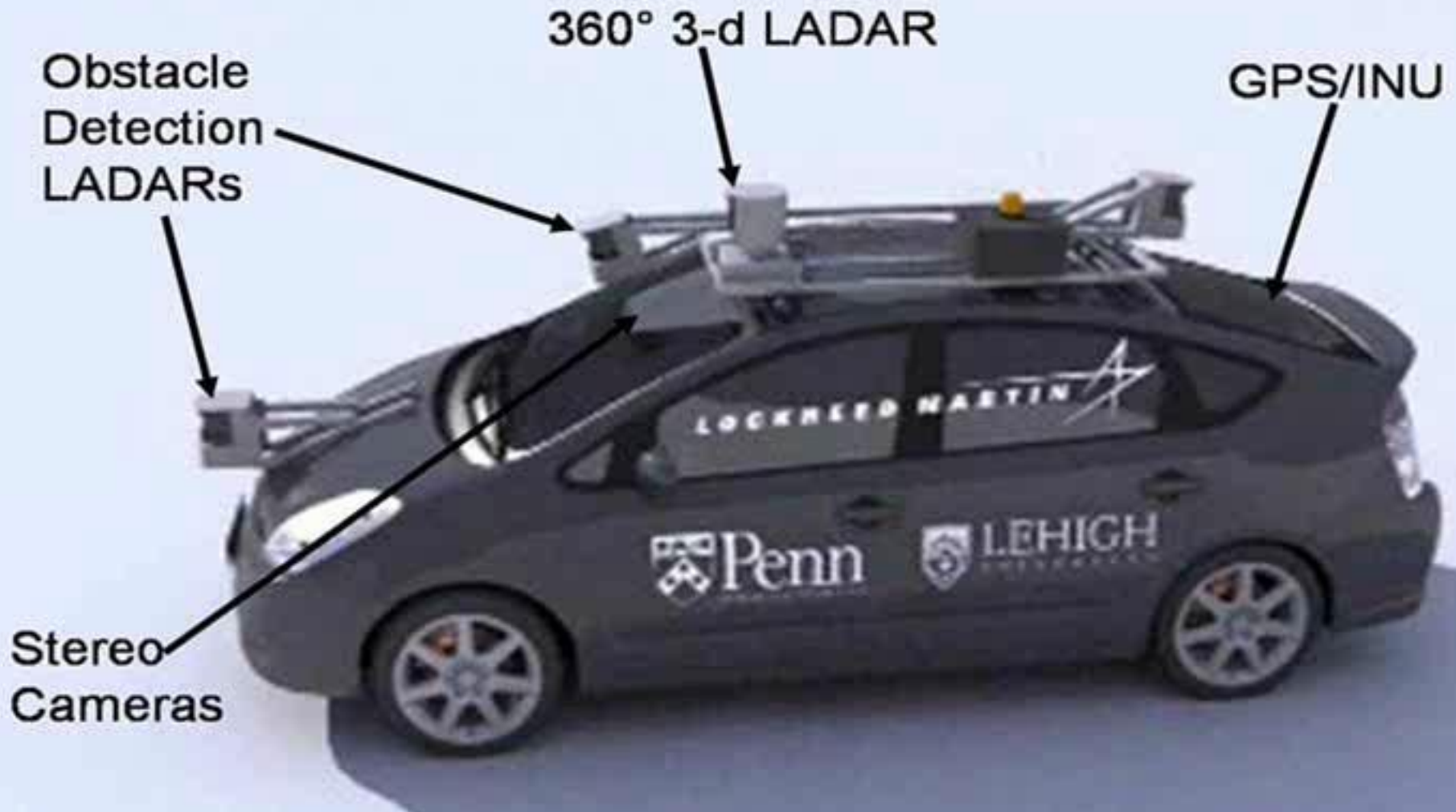


- Nevada made it legal for autonomous cars to drive on roads in June 2011
- As of 2013, four states (Nevada, Florida, California, and Michigan) have legalized autonomous cars

Penn's Autonomous Car →  
(Ben Franklin Racing Team)

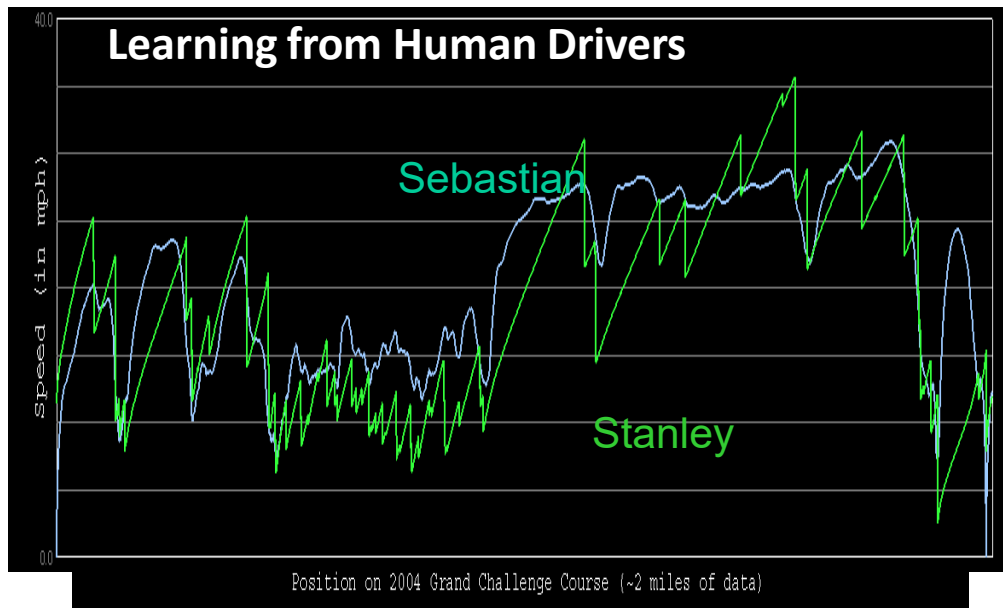
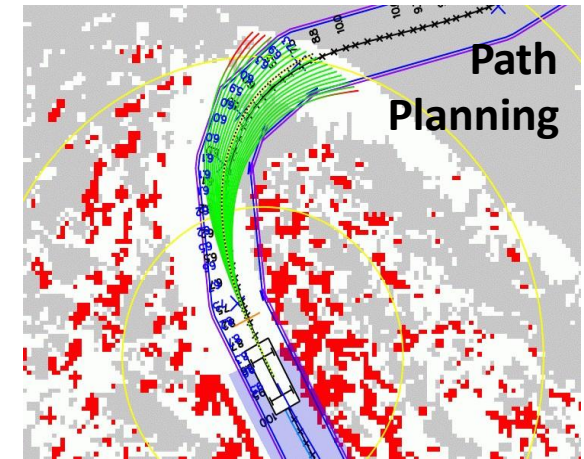


# Autonomous Car Sensors





# Autonomous Car Technology



# Deep Learning in the Headlines

BUSINESS NEWS

MIT  
Technology  
Review

## Is Google Cornering the Market on Deep Learning?

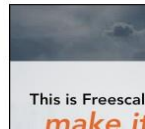
A cutting-edge corner of science is being wooed by Silicon Valley, to the dismay of some academics.

By Antonio Regalado on January 29, 2014



How much are a dozen deep-learning researchers worth? Apparently, more than \$400 million.

This week, Google [reportedly paid that much](#) to acquire [DeepMind Technologies](#), a startup based in



This is Freescale  
make it

## BloombergBusinessweek Technology

Acquisitions

## The Race to Buy the Human Brains Behind Deep Learning Machines

By Ashlee Vance | January 27, 2014

intelligence projects. “DeepMind is bona fide in terms of its research capabilities and depth,” says Peter Lee, who heads Microsoft Research.

According to Lee, Microsoft, Facebook ([FB](#)), and Google find themselves in a battle for deep learning talent. Microsoft has gone from four full-time deep learning experts to 70 in the past three years. “We would have more if the talent was there to

WIRED

GEAR SCIENCE ENTERTAINMENT BUSINESS SECURITY DESIGN

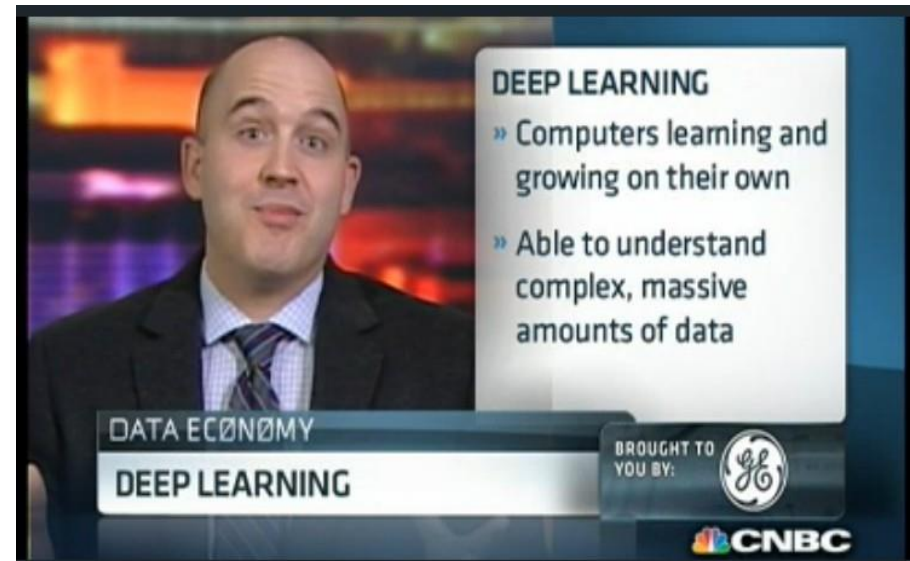
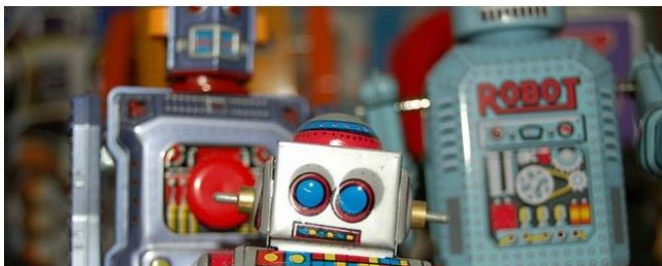
INNOVATION INSIGHTS

community content

featured

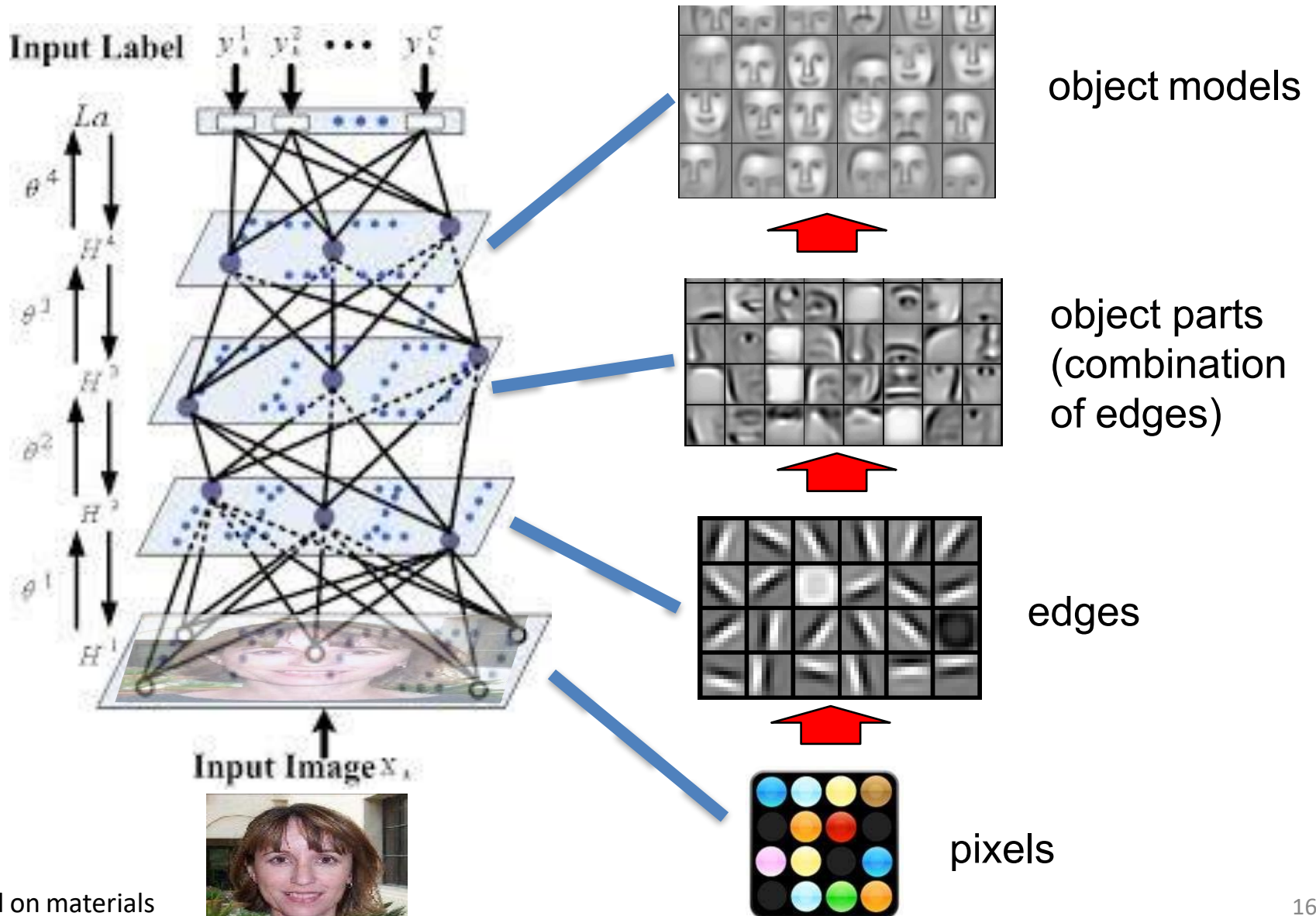
## Deep Learning's Role in the Age of Robots

BY JULIAN GREEN, JETPAC 05.02.14 2:56 PM

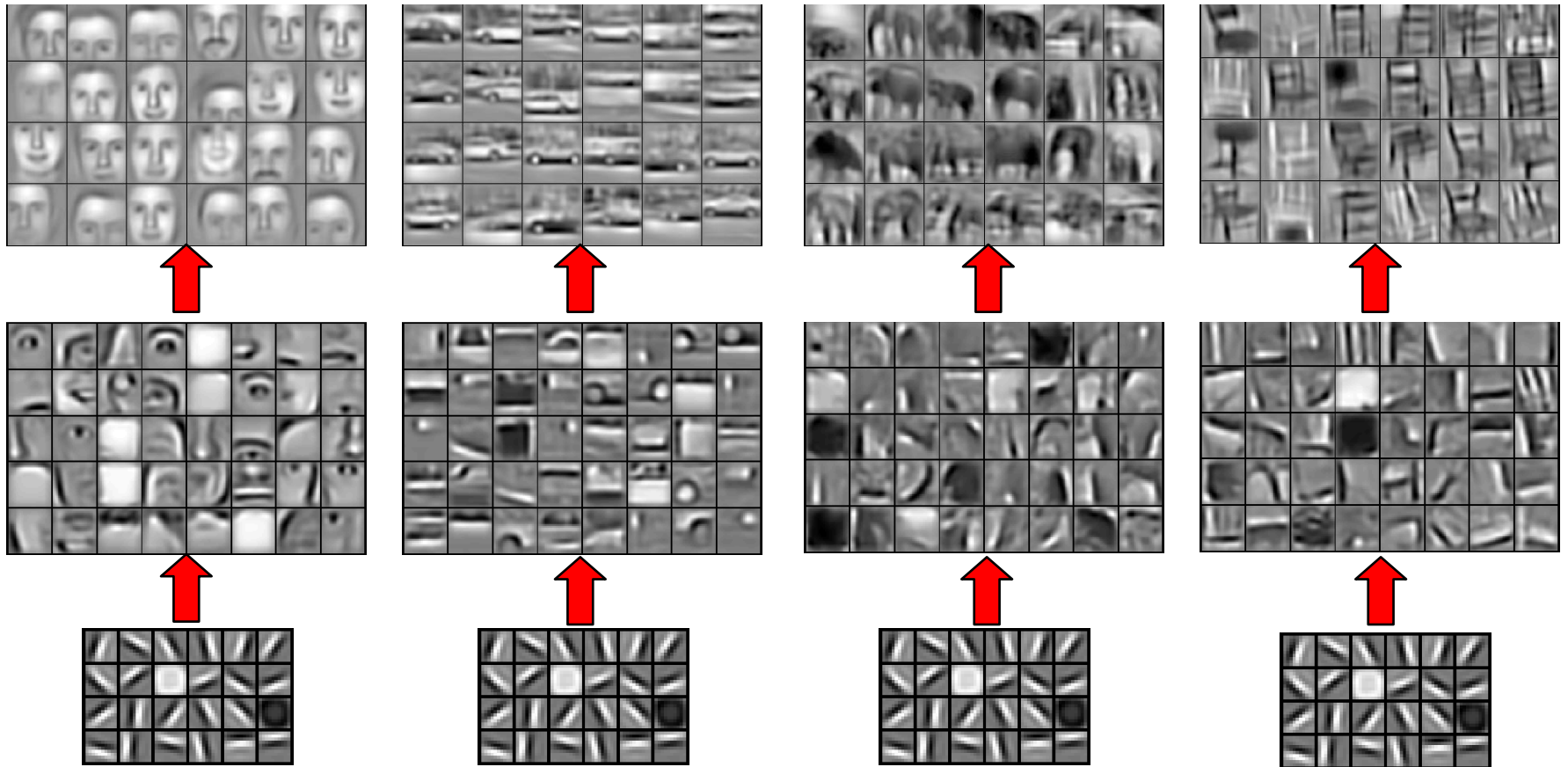




# Deep Belief Net on Face Images

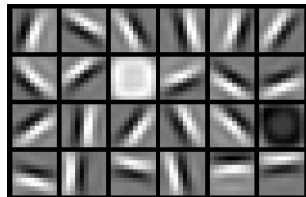


# Learning of Object Parts





# Training on Multiple Objects



Trained on 4 classes (cars, faces, motorbikes, airplanes).

Second layer: Shared-features and object-specific features.

Third layer: More specific features.

# Scene Labeling via Deep Learning



[Farabet et al. ICML 2012, PAMI 2013]

# Inference from Deep Learned Models

Generating posterior samples from faces by “filling in” experiments (cf. Lee and Mumford, 2003). Combine bottom-up and top-down inference.

Input images



Samples from  
feedforward  
Inference  
(control)

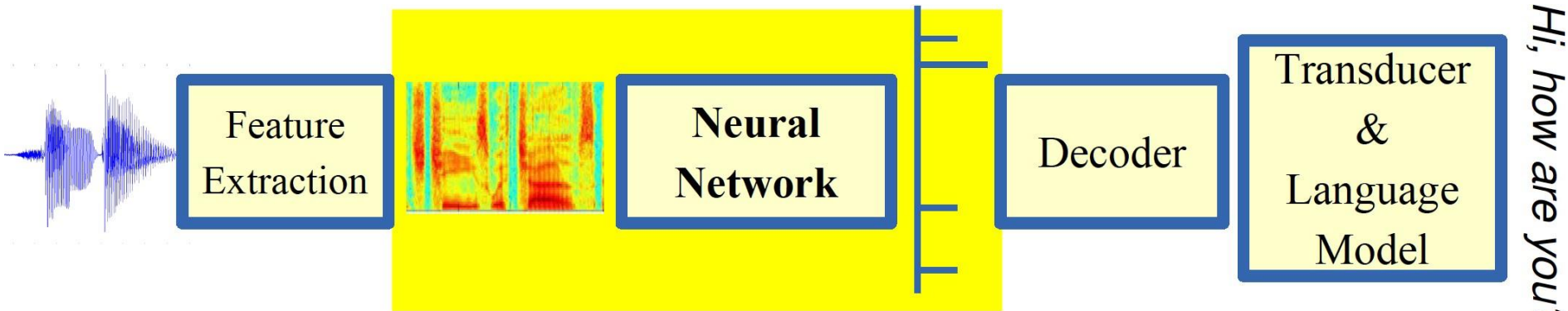


Samples from  
Full posterior  
inference

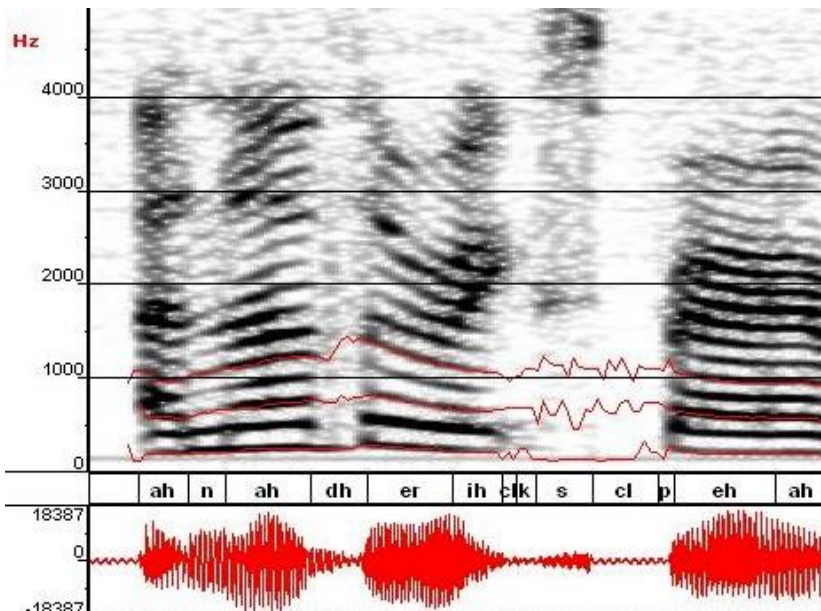


# Machine Learning in Automatic Speech Recognition

## A Typical Speech Recognition System



ML used to predict of phone states from the sound spectrogram



Deep learning has state-of-the-art results

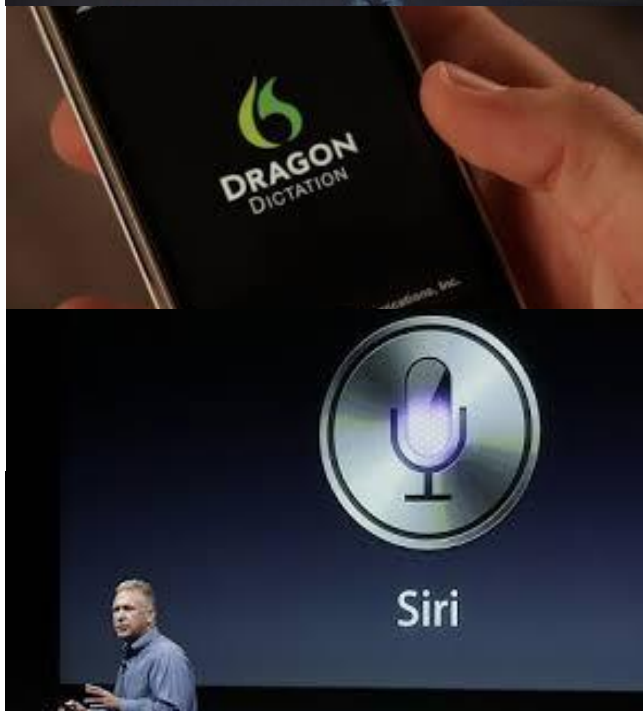
| # Hidden Layers   | 1    | 2    | 4    | 8    | 10   | 12   |
|-------------------|------|------|------|------|------|------|
| Word Error Rate % | 16.0 | 12.8 | 11.4 | 10.9 | 11.0 | 11.1 |

Baseline GMM performance = 15.4%

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



# Impact of Deep Learning in Speech Technology



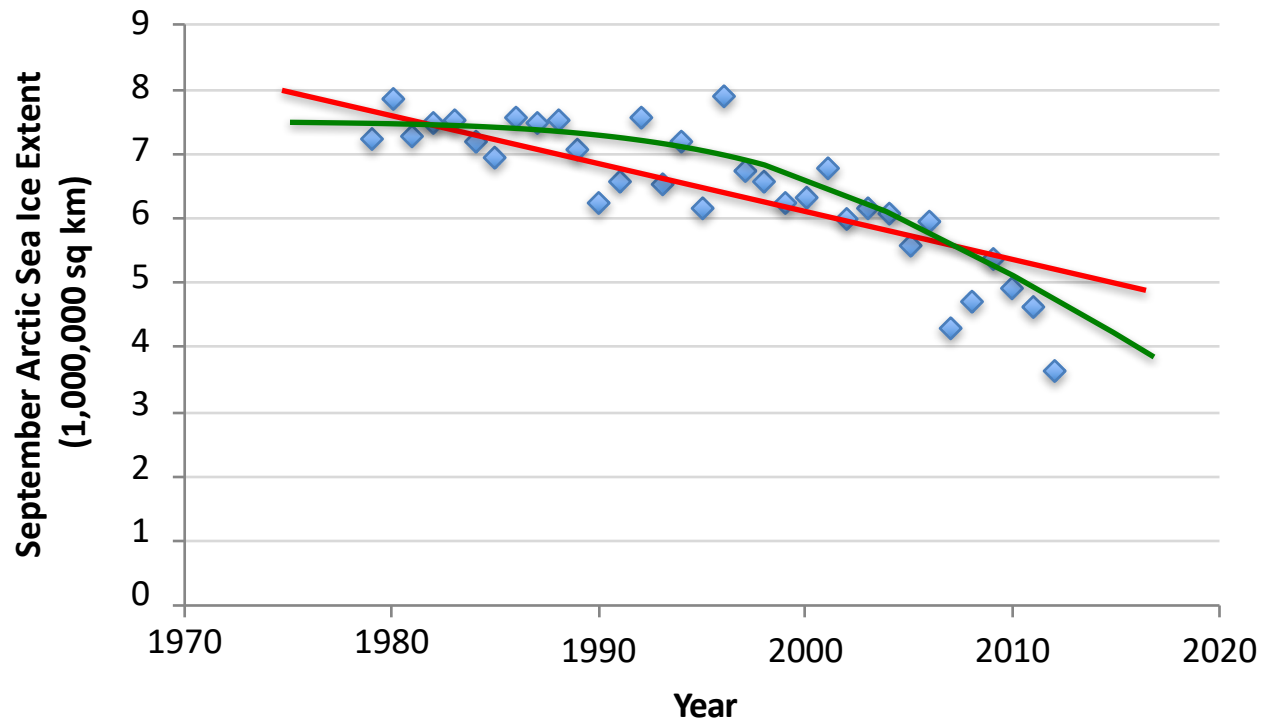
# Types of Learning

# Types of Learning

- **Supervised (inductive) 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

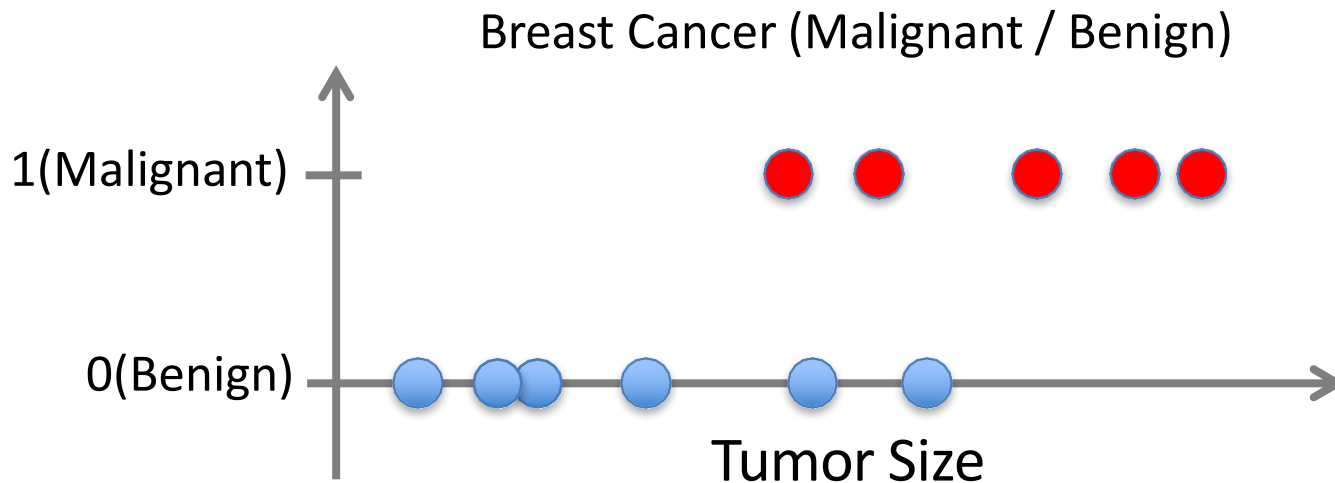
- Given  $(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)$
- Learn a function  $f(x)$  to predict  $y$  given  $x$ 
  - $y$  is real-valued == regression





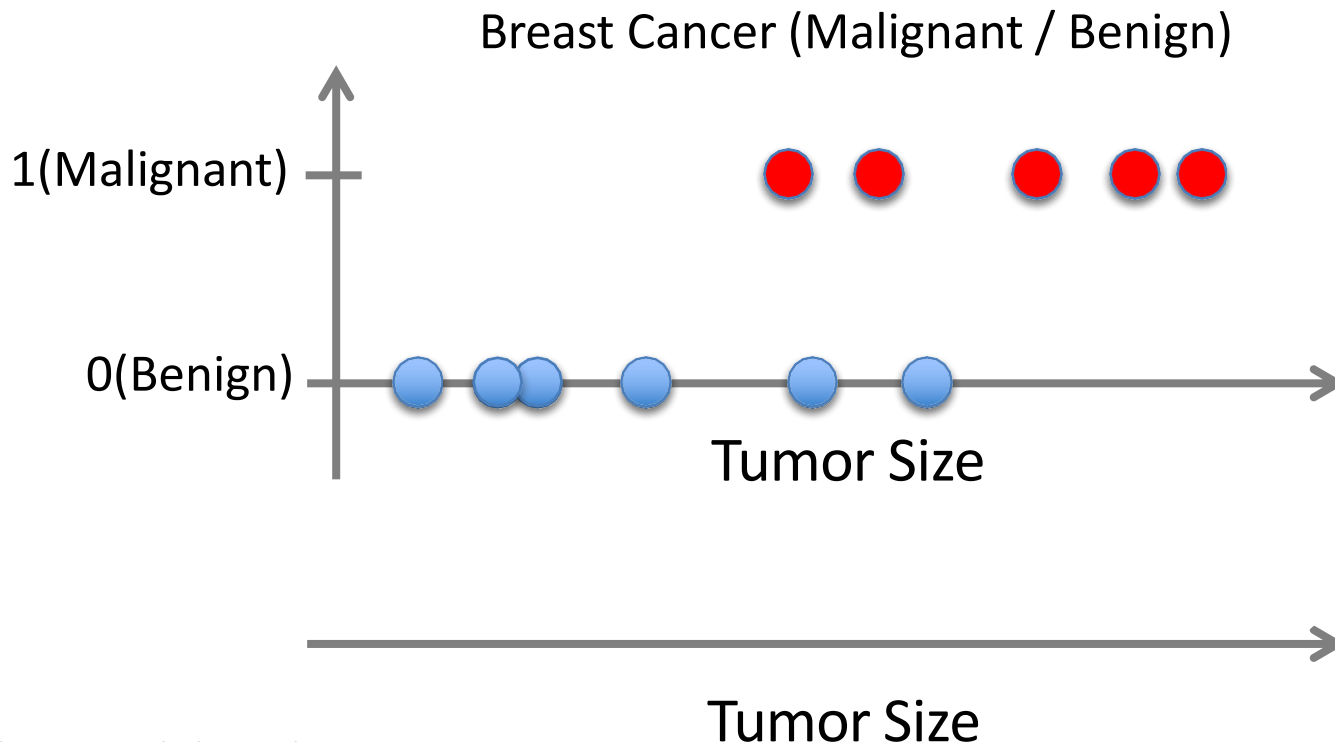
# Supervised Learning: Classification

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



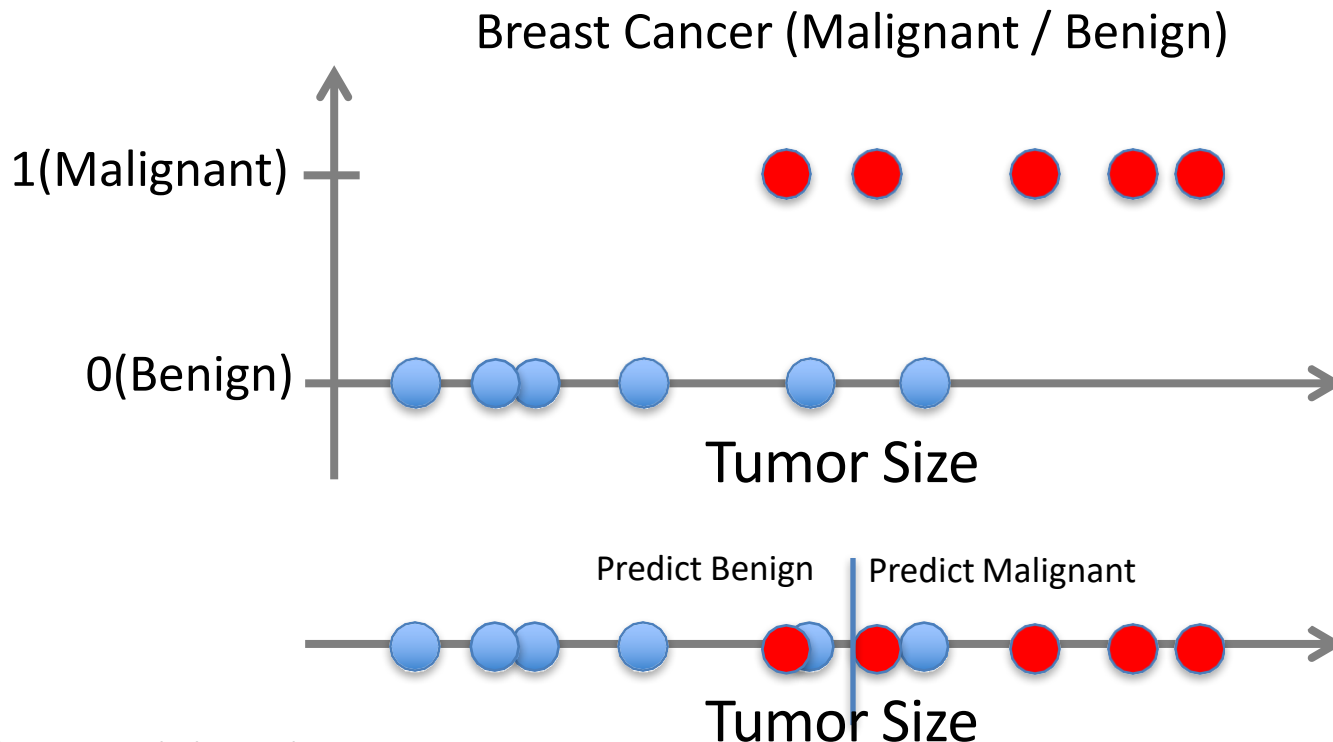
# Supervised Learning: Classification

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



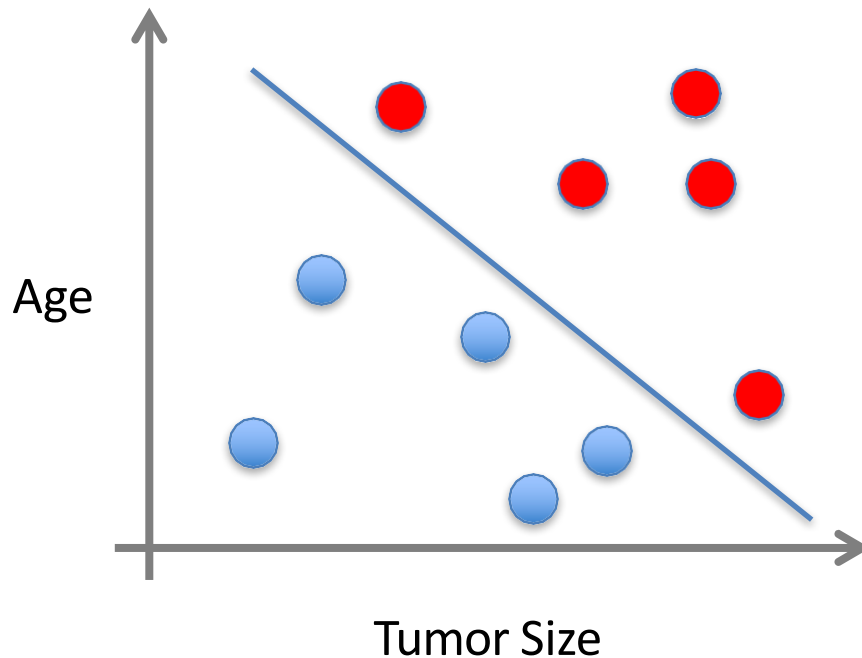
# Supervised Learning: Classification

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



# Supervised Learning

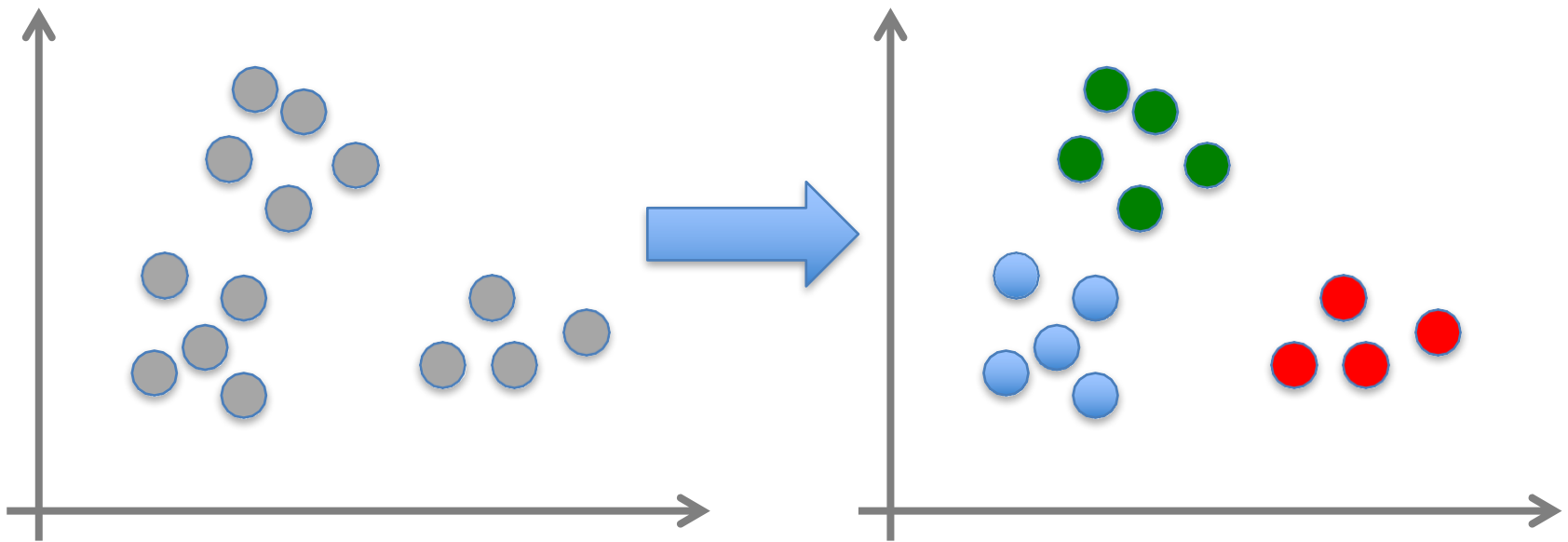
- $x$  can be multi-dimensional
  - Each dimension corresponds to an attribute



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

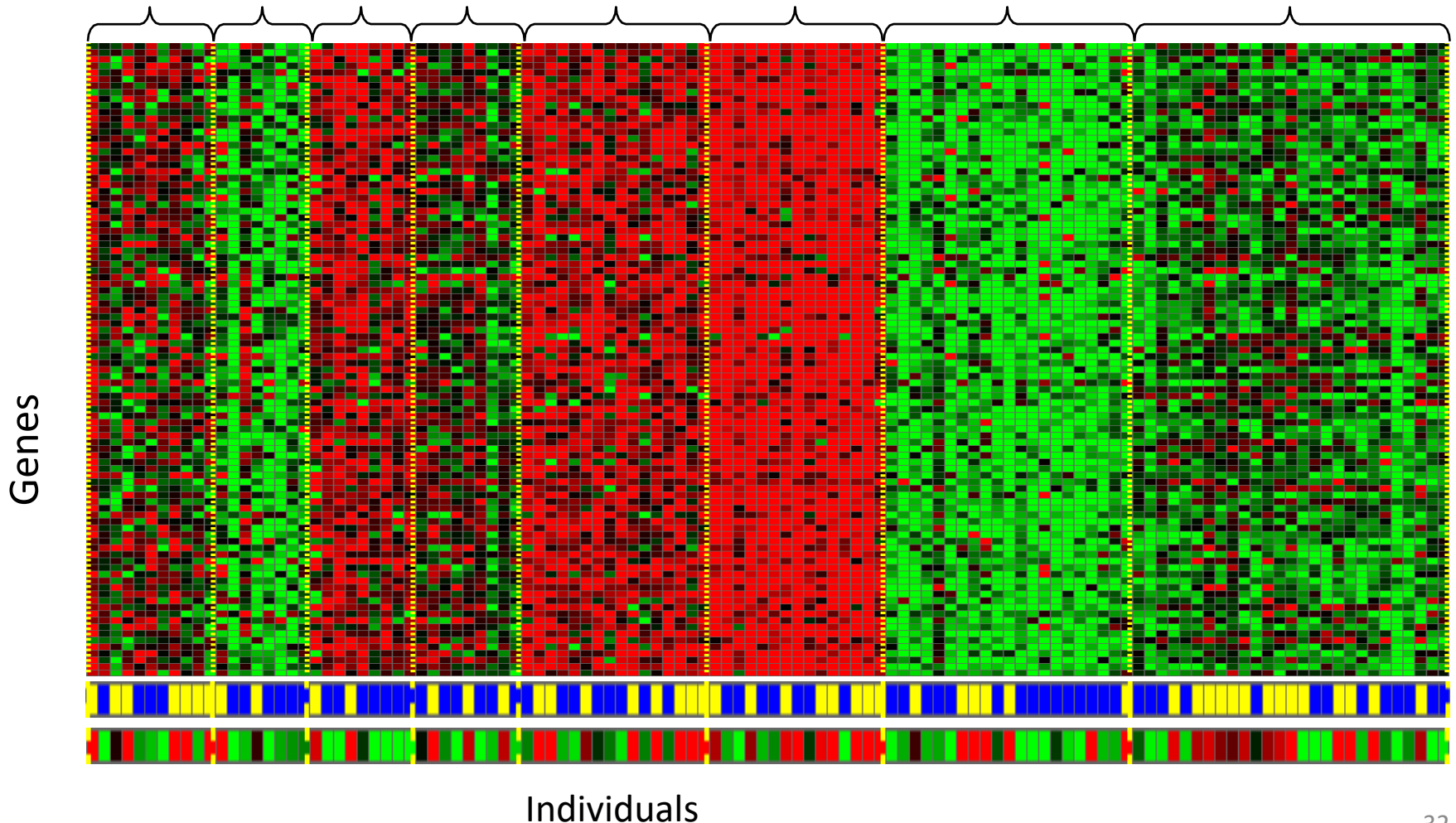
# Unsupervised Learning

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



# Unsupervised Learning

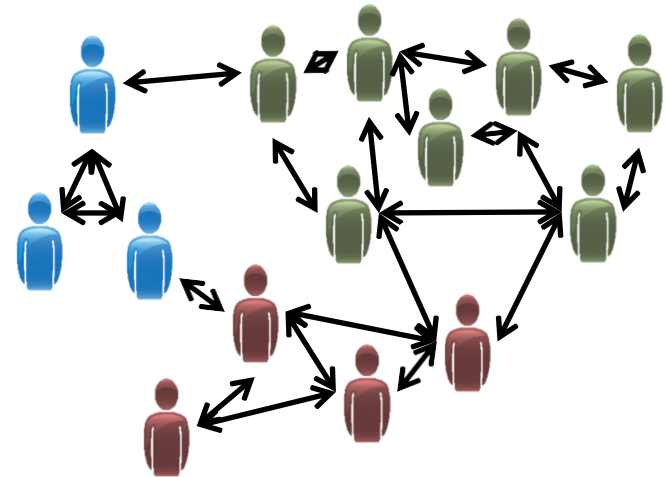
Genomics application: group individuals by genetic similarity



# Unsupervised Learning



Organize computing clusters



Social network analysis



Market segmentation

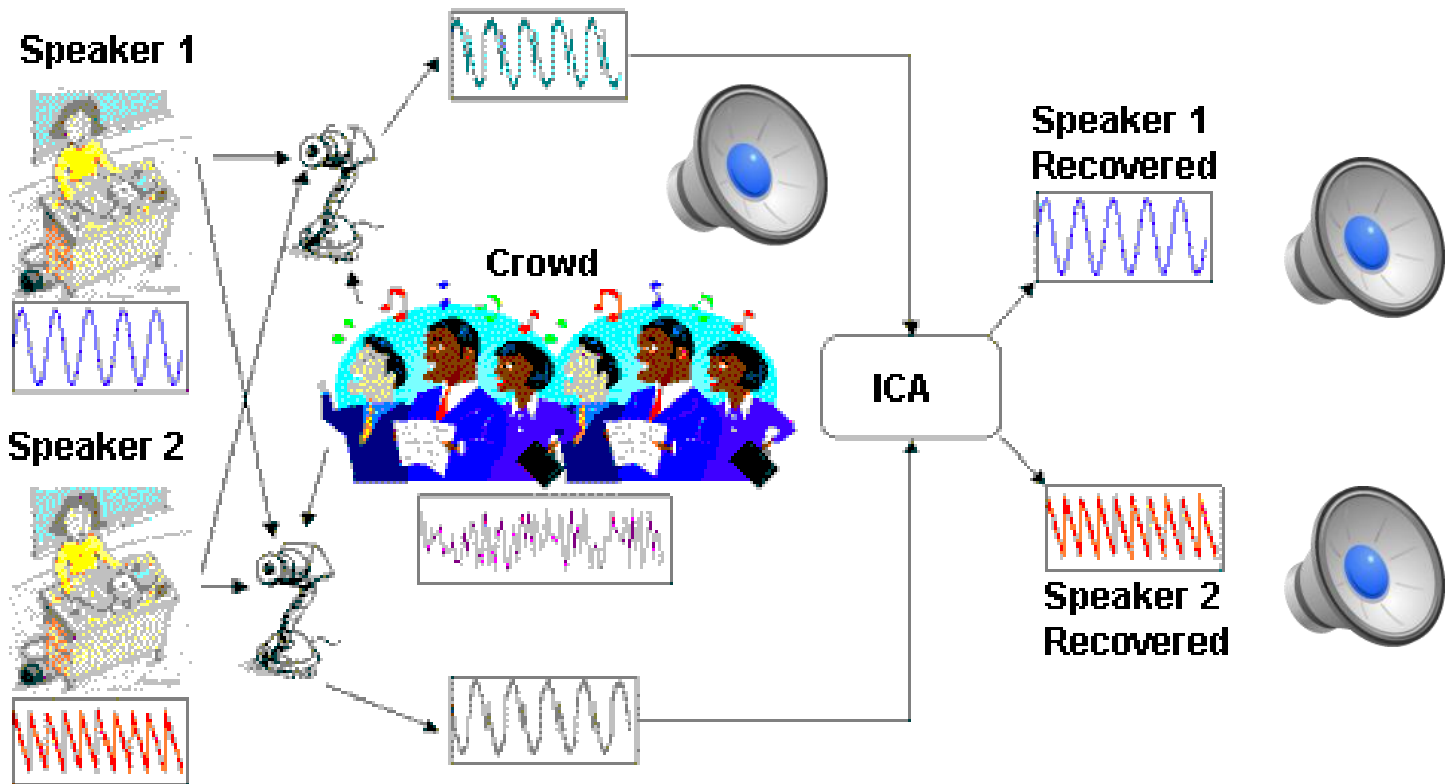


Image credit: NASA/JPL-Caltech/E. Churchwell (Univ. of Wisconsin, Madison)

Astronomical data analysis

# Unsupervised Learning

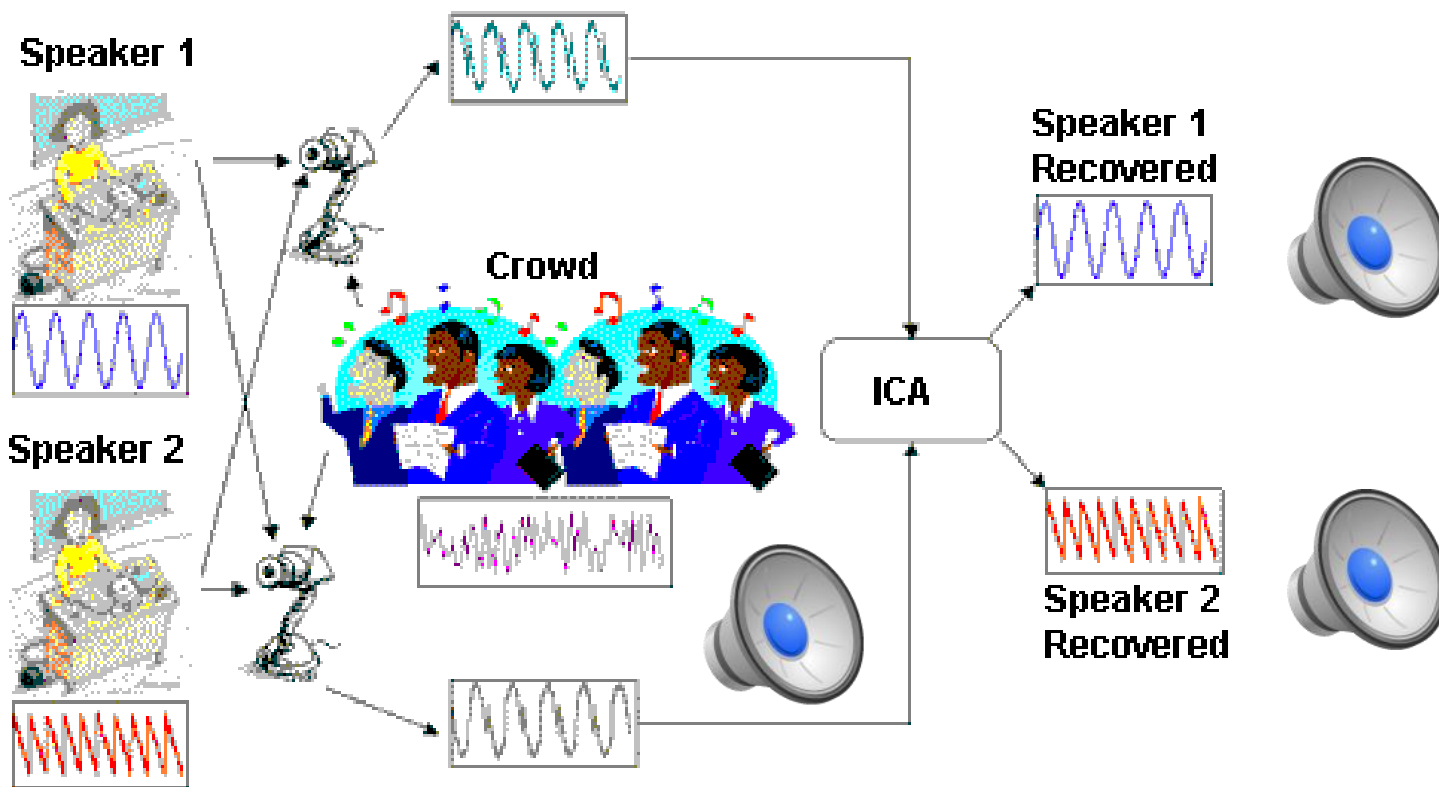
- Independent component analysis – separate a combined signal into its original sources





# Unsupervised Learning

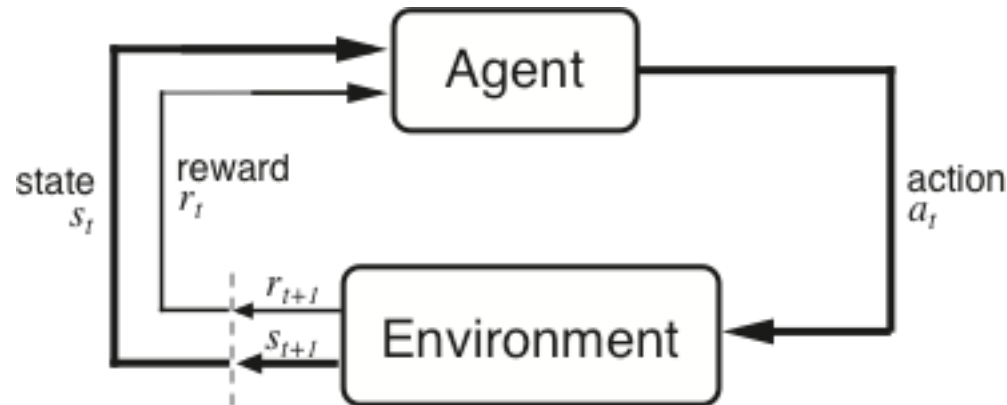
- Independent component analysis – separate a combined signal into its original sources



# Reinforcement Learning

- Given a sequence of states and actions with (delayed) rewards, output a policy
  - Policy is a mapping from states  $\rightarrow$  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



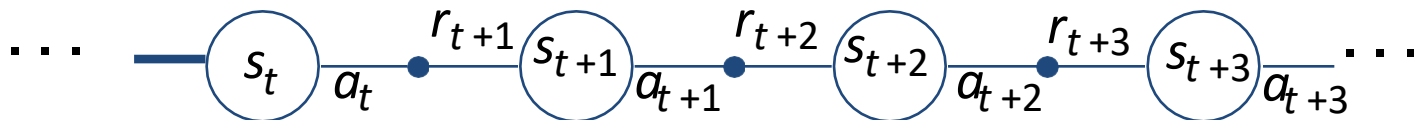
Agent and environment interact at discrete time steps :  $t = 0, 1, 2, K$

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

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

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

and resulting next state :  $s_{t+1}$



# Reinforcement Learning



<https://www.youtube.com/watch?v=4cgWya-wjgY>

# Inverse Reinforcement Learning

- Learn policy from user demonstrations



Stanford Autonomous Helicopter

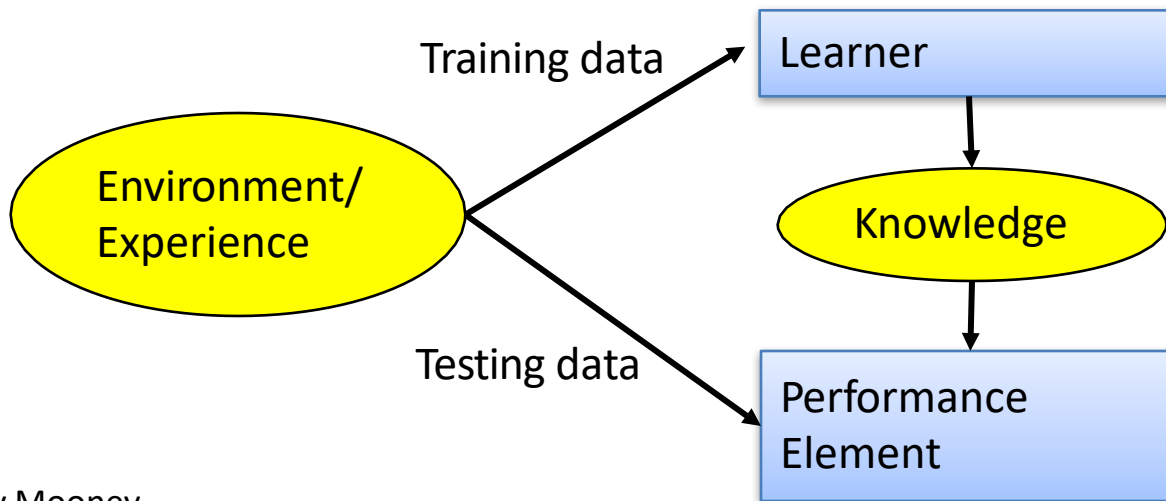
<http://heli.stanford.edu/>

<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





# 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*



# ML in a Nutshell

- Tens of thousands of machine learning algorithms
  - Hundreds new every year
- Every ML algorithm has three components:
  - **Representation**
  - **Optimization**
  - **Evaluation**

# 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

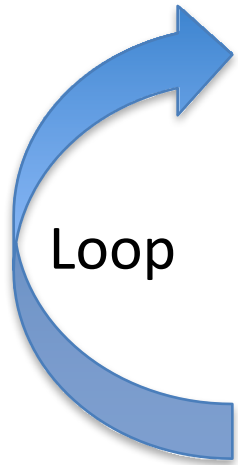
# Various Search/Optimization Algorithms

- 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

# Evaluation

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

# ML in Practice



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

# 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.

# 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



# 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

# 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.
  - ???



|                |               |                  |                          |                                |  |
|----------------|---------------|------------------|--------------------------|--------------------------------|--|
| Decision Rules | Hand-designed | Learnt           | Learnt                   | Learnt                         | Learnt   |
| Features       | Hand-designed | Hand-designed    | Learnt (Simple features) | Learnt (Hierarchy of features) | Learnt (Hierarchy of features)                 |
| Learning       | None          | Supervised       | Supervised               | Supervised                     | Self-supervised + in-context learning          |
| Data           | None          | Few labeled-data | Mid-sized labeled-data   | Large labeled-data             | Very large unlabeled data + small labeled data |
| Adaptability   | None          | None             | Little                   | Medium                         | Large  |

# What We'll Cover in this Course

- **Supervised learning**
  - Decision tree induction
  - Linear regression
  - Logistic regression
  - Support vector machines & kernel methods
  - Model ensembles
  - Bayesian learning
  - Neural networks & deep learning
  - Learning theory
- **Unsupervised learning**
  - Clustering
  - Dimensionality reduction
- **Reinforcement learning**
  - Temporal difference learning
  - Q learning
- **Evaluation**
- **Applications**

Our focus will be on applying machine learning to real applications