Course Overview and Introduction

CE-717 : Machine Learning

Sharif University of Technology

M. Soleymani

Fall 2019

Course Info

- Instructor: Mahdieh Soleymani
  - Email: soleymani@sharif.edu

- Website: http://ce.sharif.edu/courses/98-99/1/ce717-1/
  - Tentative schedule
  - Slides and notes
  - Policies and rules
  -
- Discussions: On Piazza

- TAs:
  - **Head TA**: Sina Hajimiri
  - **TAs**: Rassa Ghavami, Parishad Behnam Ghader, Amir Dalili, Faezeh Ghorbanpour, Ali Karimi, Mohammad Ostad Mohammadi, Sorena Salari, Mohammad Ali Samiei

Text Books

- Pattern Recognition and Machine Learning, C. Bishop, Springer, 2006.
- Machine Learning, T. Mitchell, MIT Press, 1998.

- Other books:
  - The elements of statistical learning, T. Hastie, R. Tibshirani, J. Friedman, Second Edition, 2008.
  - Machine Learning: A Probabilistic Perspective, K. Murphy, MIT Press, 2012.
  - Richard Sutton and Andrew Barto, Reinforcement Learning: An introduction. MIT Press, Second edition, 2017.

Prerequisites:

- Programming skills
- Probability and statistics
- Basic linear algebra
  - We'll go over it in the review sections.

Assignments

- 7 Problem sets
  - The first one is on prerequisites.
  - Other sets contain both theoretical and programming assignments
- Exams
  - Midterm and final exams covering all topics taught in class
  - Two mini-exams

Marking Scheme

Midterm Exam:	25%
Final Exam:	30%
Homeworks (written & programming) :	35%
Mini-exams:	10%

Machine Learning (ML) and Artificial Intelligence (AI)

- ML appears first as a branch of AI

- ML is a preferred approach to other subareas of AI
  - Computer Vision
  - Natural Language Processing
  - Robotics
  - Speech Recognition

- ML is a strong driver in many applications

A Definition of ML

- Tom Mitchell (1998):
  - A computer program is said to learn from <u>experience</u> if its performance improves with experience
- Using the observed data to make better decisions
  - Generalizing from the observed data

ML Definition: Example

- Consider an email program that learns how to filter spam according to emails you do or do not mark as spam.

  - Task: Classifying emails as spam or not spam.
  - Experience: Watching you label emails as spam or not spam.
  - Performance: The number (or fraction) of emails correctly classified as spam/not spam.

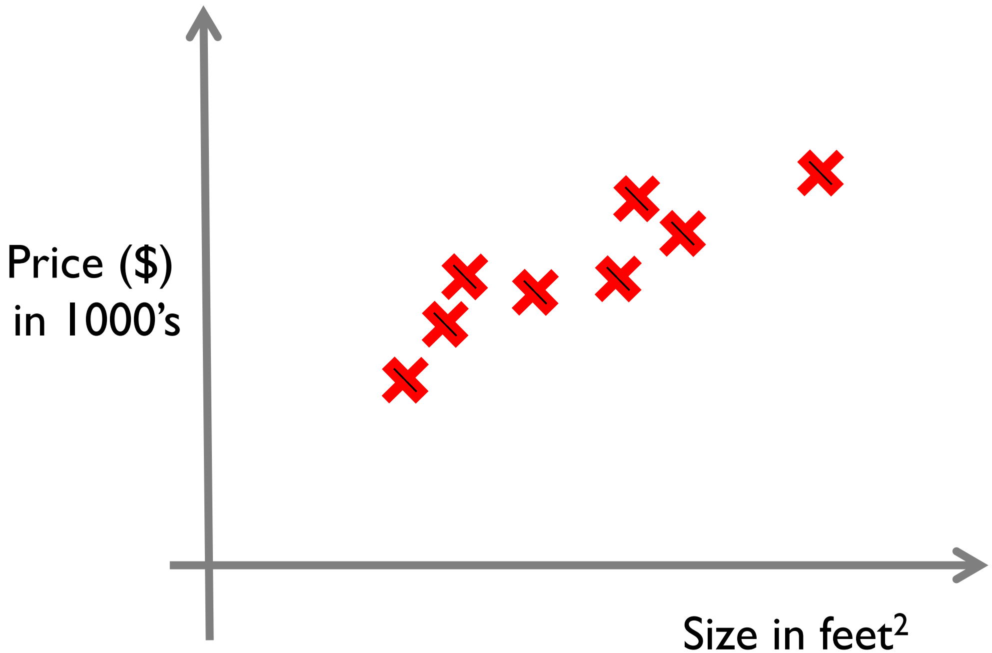
The essence of machine learning

- A pattern exist
- We do not know it mathematically
- We have data on it

[Abu Mostafa, 2012]

Example: Home Price

- Housing price prediction



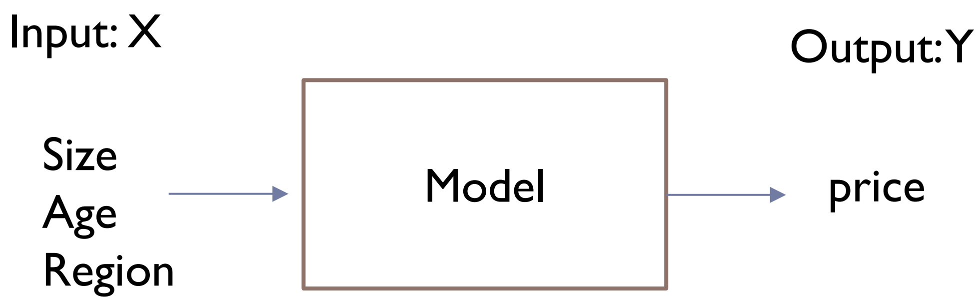
Regression problem

- The goal is to make (real valued) predictions given features

- Example: predicting house price from 3 attributes

Size (m^2)	Age (year)	Region	Price (10^6T)
100	2	5	500
80	25	3	250
…	…	…	…

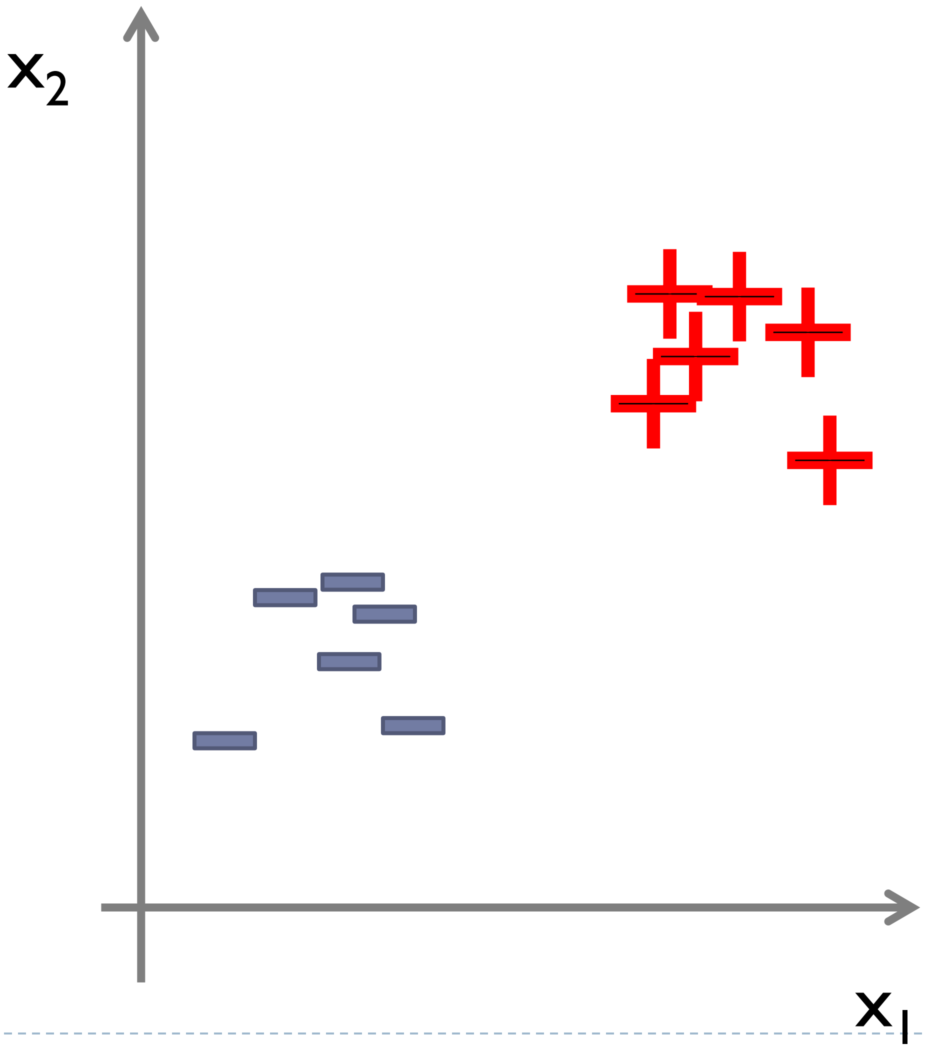
Handwritten Digit Recognition Example



Example: Bank loan

- Applicant form as the input:
  - salary
  - age
  - gender
  - current debt
  - …
- Output: approving or denying the request

Training data: Example

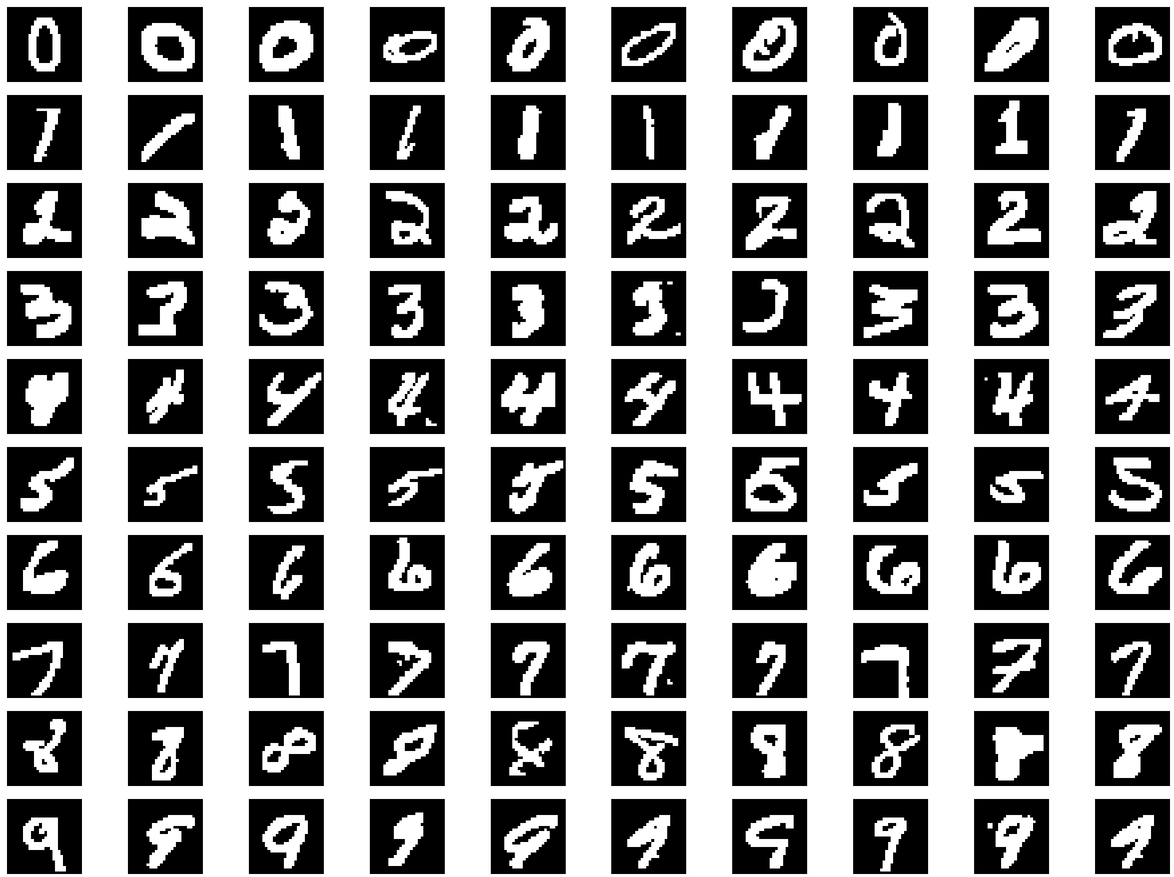


Training data

x_1	x_2	y
0.9	2.3	1
3.5	2.6	1
2.6	3.3	1
2.7	4.1	1
1.8	3.9	1
6.5	6.8	-1
7.2	7.5	-1
7.9	8.3	-1
6.9	8.3	-1
8.8	7.9	-1
9.1	6.2	-1

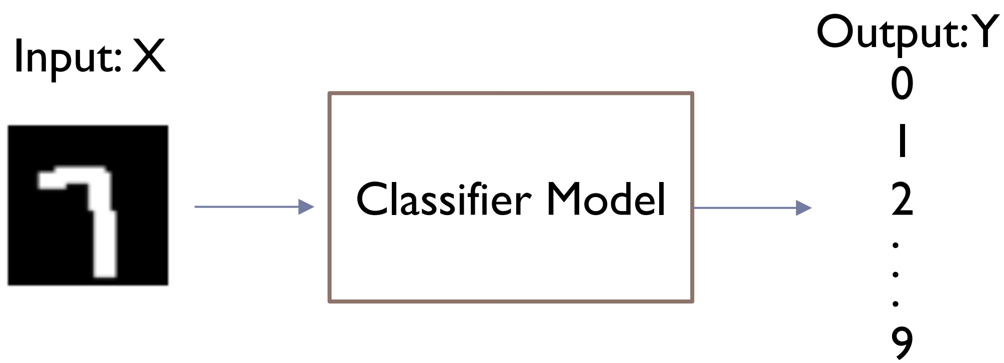
Handwritten Digit Recognition Example

- Data: labeled samples



0
1
2
3
4
5
6
7
8
9

Handwritten Digit Recognition Example



Experience (E) in ML

- Basic premise of learning:
  - "Using a set of observations to uncover an underlying process"

- We have different types of (getting) observations in different types or paradigms of ML methods

Paradigms of ML

- Supervised learning (regression, classification)
  - predicting a target variable for which we get to see examples.
- Unsupervised learning
  - revealing structure in the observed data
- Reinforcement learning
  - partial (indirect) feedback, no explicit guidance
  - Given rewards for a sequence of moves to learn a policy and utility functions

Supervised Learning: Regression vs. Classification

- Supervised Learning
  - **Regression**: predict a <u>continuous</u> target variable
    - E.g., $y \in [0,1]$
  - **Classification**: predict a <u>discrete</u> (unordered) target variable
    - E.g.,$y \in \{1,2, \dots, C\}$

Data in Supervised Learning

- Data are usually considered as vectors in a d dimensional space
  - Now, we make this assumption for illustrative purpose
    - We will see it is not necessary

Columns:
Features/attributes/dimensions

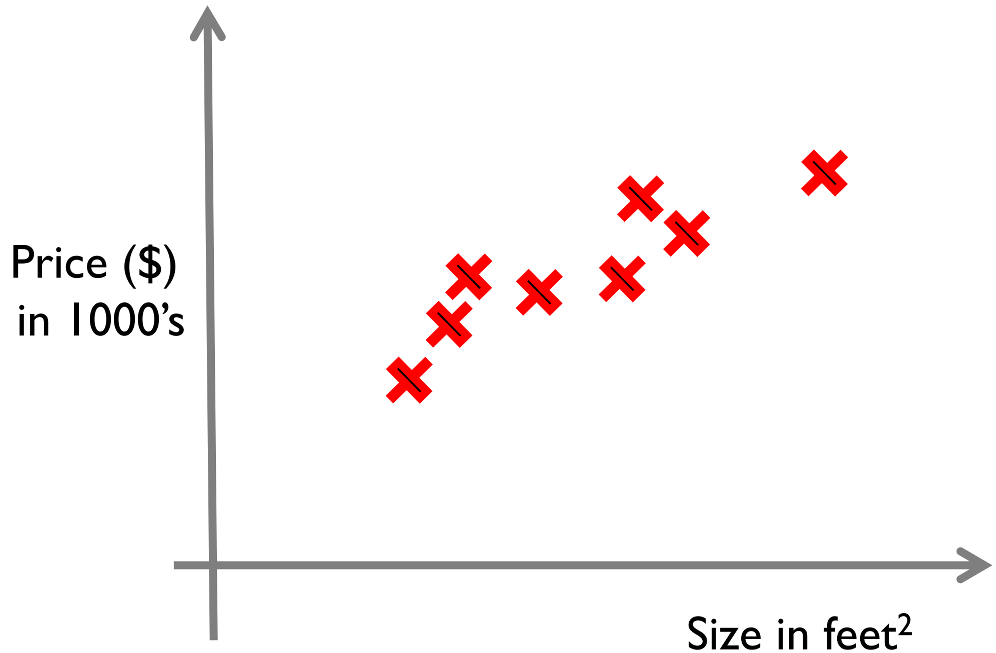
Rows:
Data/points/instances/examples/samples

Y column:
Target/outcome/response/label

	x_1	x_2	…	x_d	y (Target)
Sample 1					
Sample 2					
…					
Sample n-1					
Sample n					

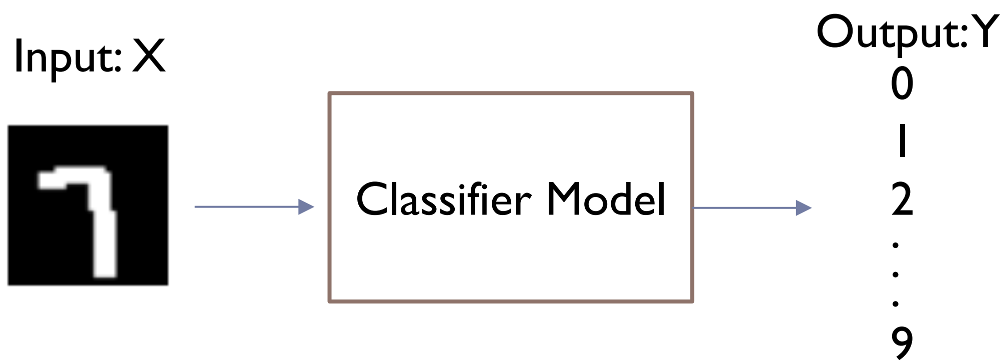
Regression: Example

- Housing price prediction



Classification: Example

- Handwritten Digit Recognition



Components of (Supervised) Learning

- Unknown target function: $f: \mathcal{X} \rightarrow \mathcal{Y}$
  - Input space: $\mathcal{X}$
  - Output space: $\mathcal{Y}$

- Training data: $(\boldsymbol{x}_1, y_1), (\boldsymbol{x}_2, y_2), \ldots, (\boldsymbol{x}_N, y_N)$

- Pick a formula $g: \mathcal{X} \rightarrow \mathcal{Y}$ that approximates the target function f
  - selected from a set of hypotheses $\mathcal{H}$

Components of (Supervised) Learning

- We have some example pairs of (input, output) called training samples
  - $\left(\boldsymbol{x}^{(1)}, y^{(1)}\right), \ldots, (\boldsymbol{x}^{(N)}, y^{(N)})$
- We want to select a function from the input space to the output space
  - $f: \mathcal{X} \rightarrow \mathcal{Y}$
- We choose a set of hypotheses (candidate formulas)
  - e.g., linear functions
- We use a learning algorithm to select a function from hypothesis set that approximates the target function

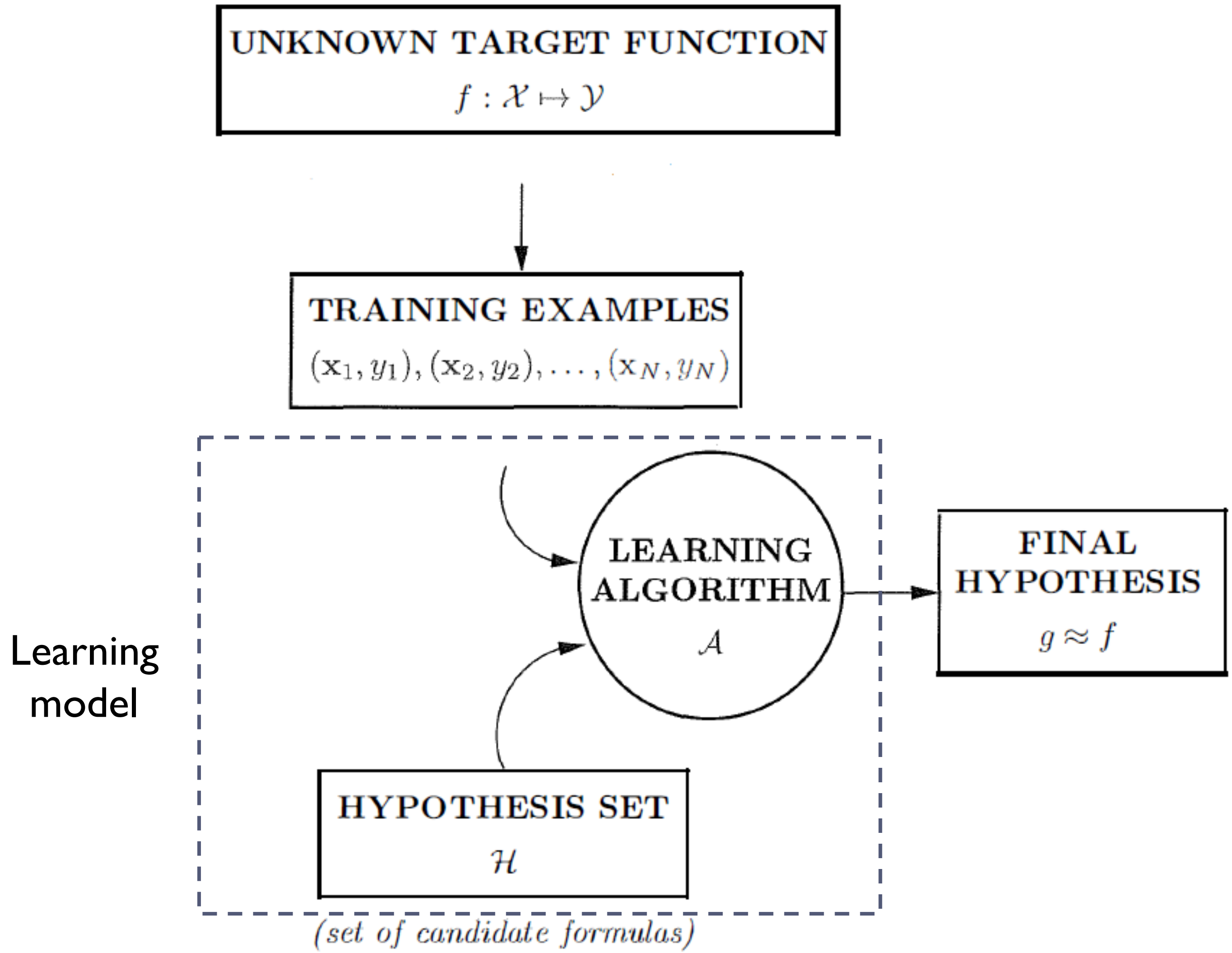
(Supervised) Learning problem

- Selecting a **hypothesis space**
  - Hypothesis space: a set of mappings from feature vector to target

- **Learning**: find mapping $\hat{f}$ (from hypothesis set) based on the training data
  - Which notion of error should we use? (loss functions)
  - Optimization of loss function to find mapping $\hat{f}$

- **Evaluation**: we measure how well $\hat{f}$ generalizes to unseen examples (generalization)

Components of (Supervised) Learning



https://work.caltech.edu/telecourse.html

(Supervised) Learning problem

- Selecting a **hypothesis space**
  - Hypothesis space: a set of mappings from feature vector to target

- **Learning (estimation)**: optimization of a cost function
  - Based on the training set $D = \{(\boldsymbol{x}^{(i)}, y^{(i)})\}_{i=1}^{n}$ and a cost function we find (an estimate) $f \in F$ of the target function

- **Evaluation**: we measure how well $\hat{f}$ generalizes to unseen examples

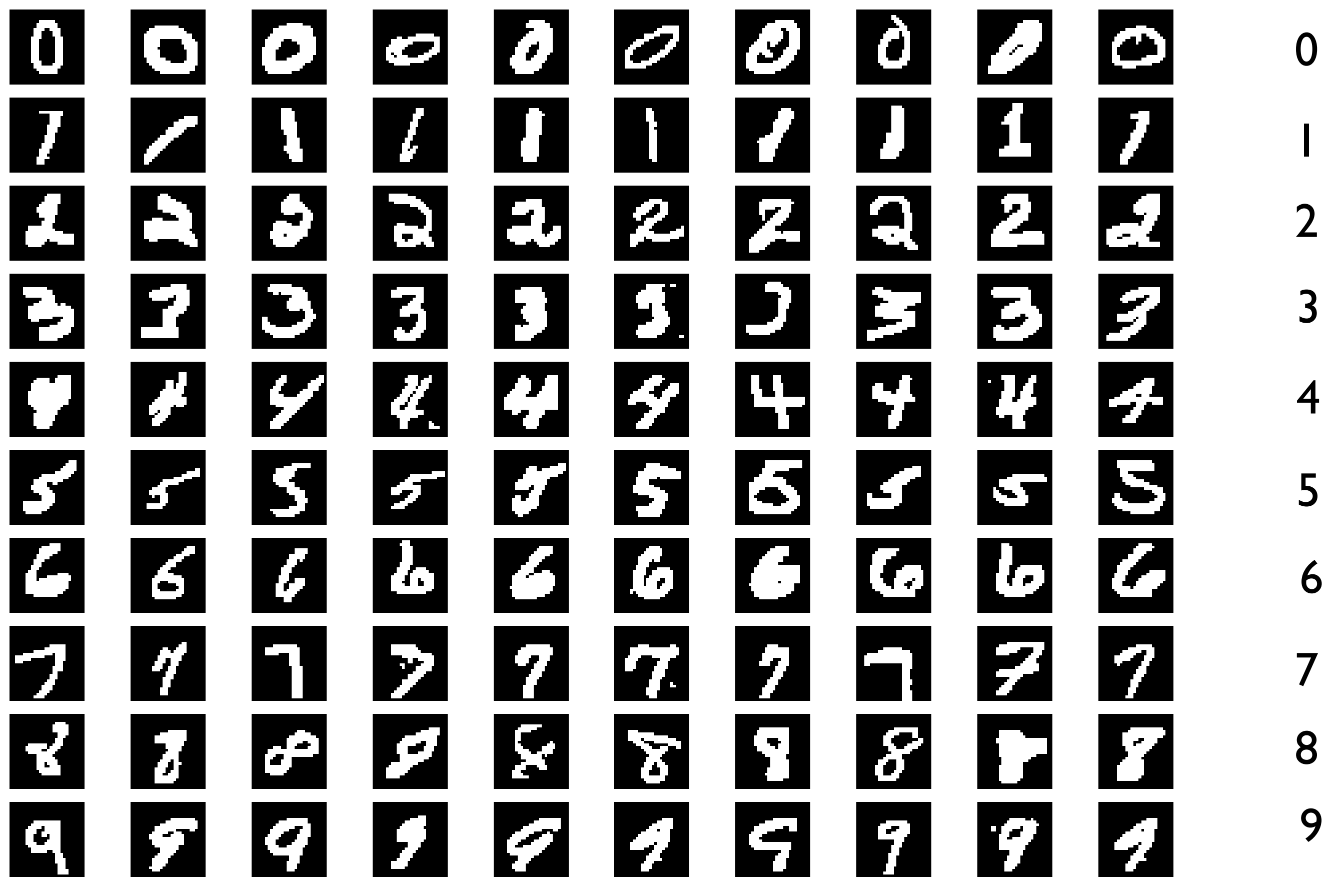
Solution Components

- Learning model composed of:
  - Hypothesis set
  - Learning algorithm

- Perceptron example

Handwritten Digit Recognition Example

- Data: labeled samples



Example: Input representation

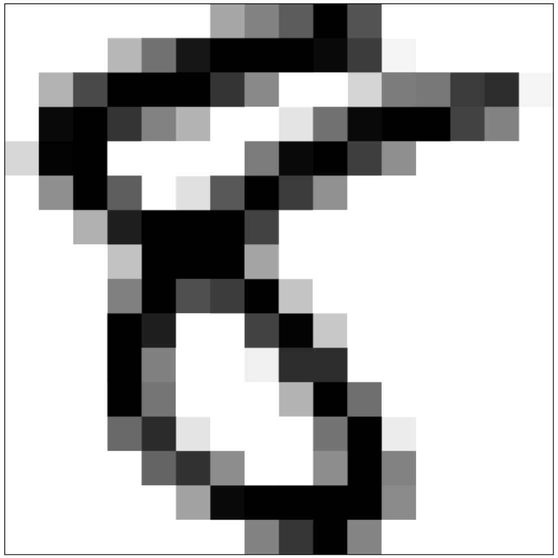
'raw' input $\mathbf{x} = (x_0, x_1, x_2, \cdots, x_{256})$

linear model: $(w_0, w_1, w_2, \cdots, w_{256})$

Features: Extract useful information, e.g.,

intensity and symmetry $\mathbf{x} = (x_0, x_1, x_2)$

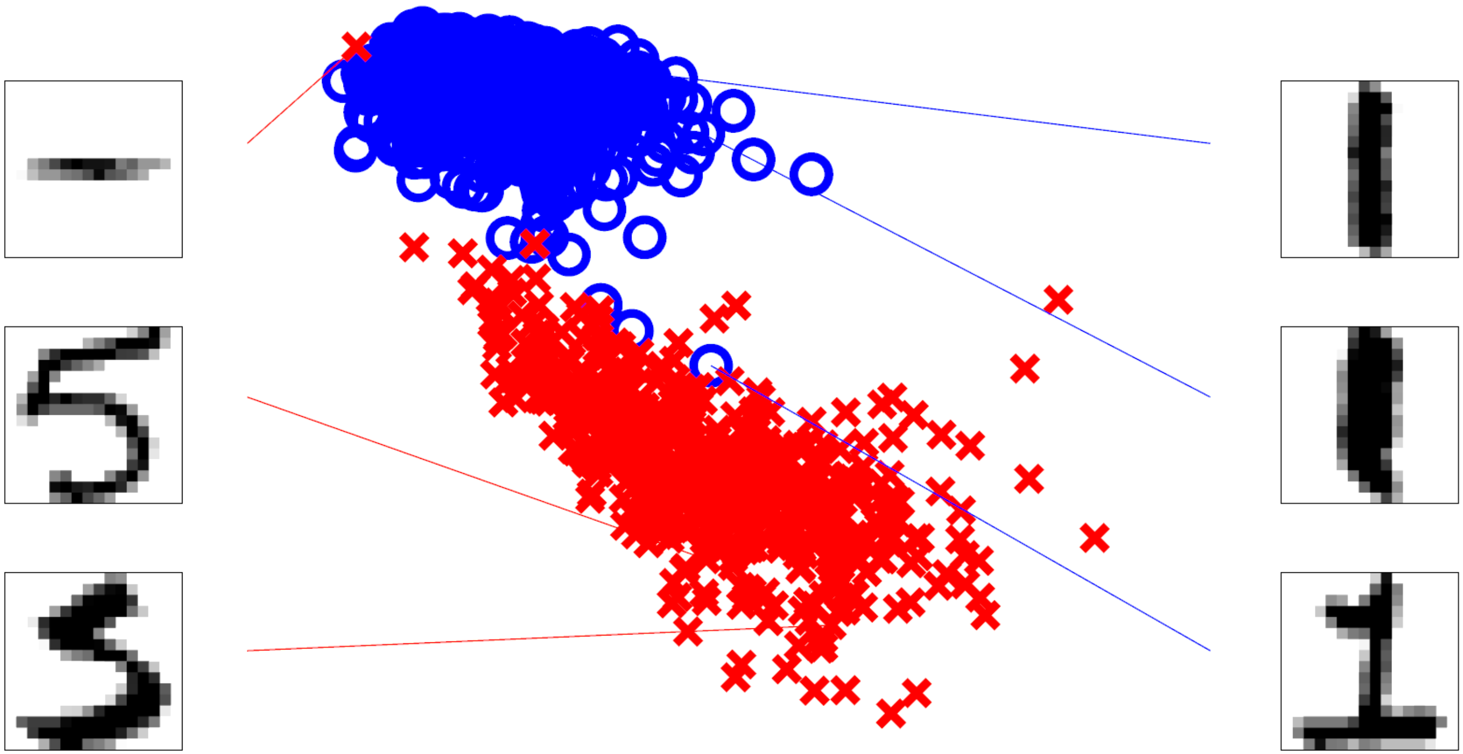
linear model: (w_0, w_1, w_2)



https://work.caltech.edu/telecourse.html

Example: Illustration of features

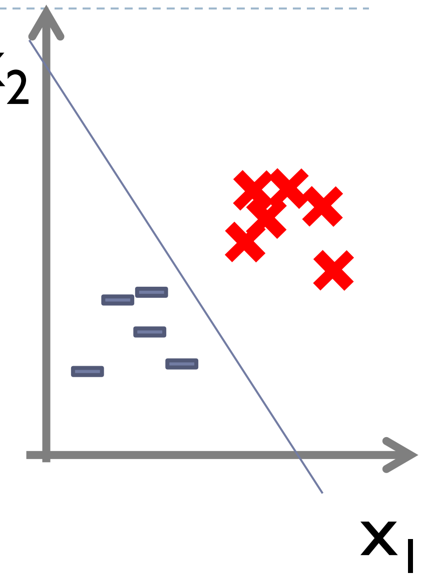
$\mathbf{x} = (x_0, x_1, x_2)$ x_1: intensity x_2: symmetry



https://work.caltech.edu/telecourse.html

Perceptron classifier

- Input $\boldsymbol{x} = [x_1, \dots, x_d]$
- Classifier:
  - If $\sum_{i=1}^{d} w_i x_i > \text{threshold}$ then output 1
  - else output -1



- The linear formula $g \in \mathcal{H}$ can be written:

$$g(\boldsymbol{x}) = \text{sign}\left(\sum_{i=1}^{d} w_i x_i \quad + \quad w_0 \quad \right)$$

If we add a coordinate $x_0 = 1$ to the input:

$$g(\boldsymbol{x}) = \text{sign}\left(\sum_{i=0}^{d} w_i x_i\right)$$

Vector form

$$g(\boldsymbol{x}) = \text{sign}(\boldsymbol{w}^T \boldsymbol{x})$$

Perceptron learning algorithm: linearly separable data

- Give the training data $\left(\boldsymbol{x}^{(1)}, y^{(1)}\right), \ldots, \left(\boldsymbol{x}^{(N)}, y^{(N)}\right)$

- Misclassified data $\left(\boldsymbol{x}^{(n)}, y^{(n)}\right)$:

$$\operatorname{sign}(\boldsymbol{w}^T \boldsymbol{x}^{(n)}) \neq y^{(n)}$$

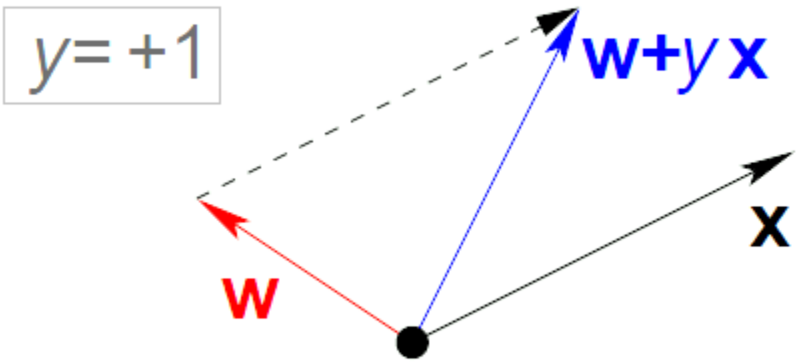
Repeat

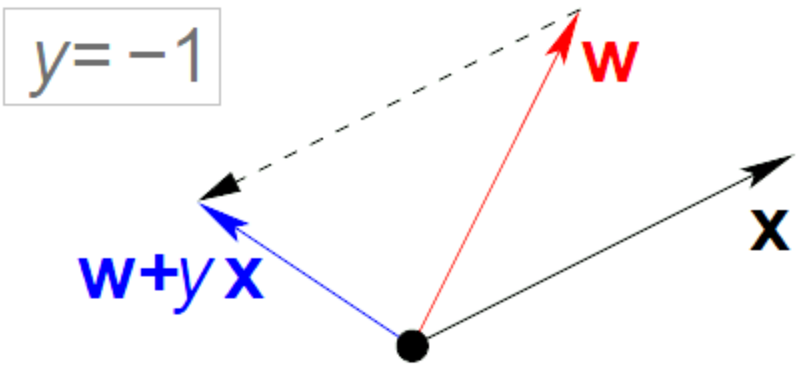
Pick a misclassified data $\left(\boldsymbol{x}^{(n)}, y^{(n)}\right)$ from training data and update $\boldsymbol{w}$:

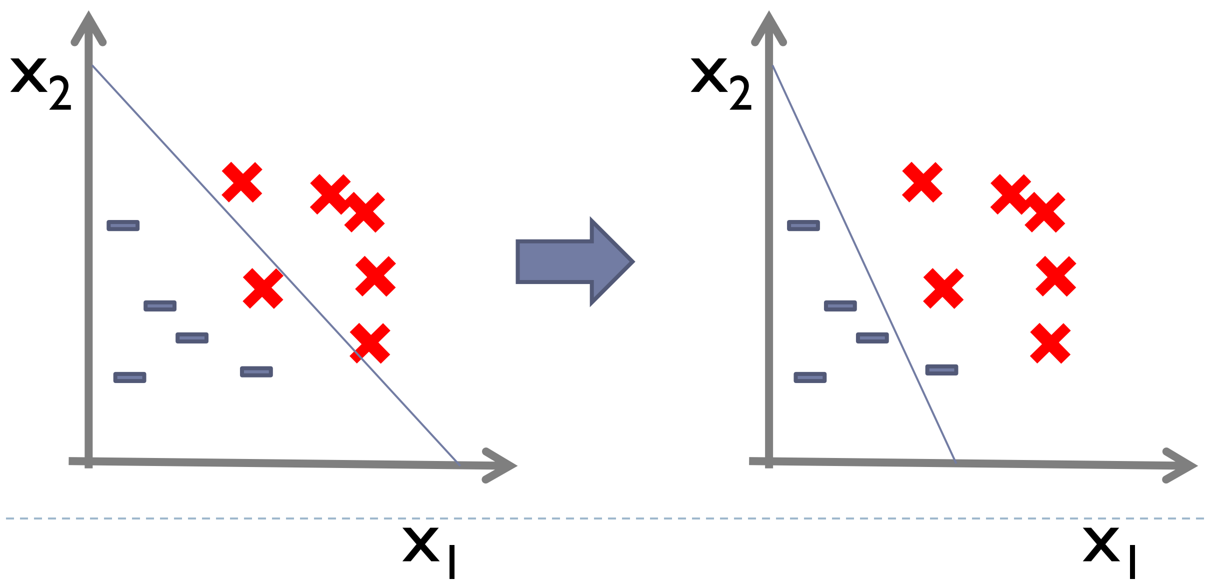
$$\boldsymbol{w} = \boldsymbol{w} + y^{(n)} \boldsymbol{x}^{(n)}$$

Until all training data points are correctly classified by g

Perceptron learning algorithm: Example of weight update

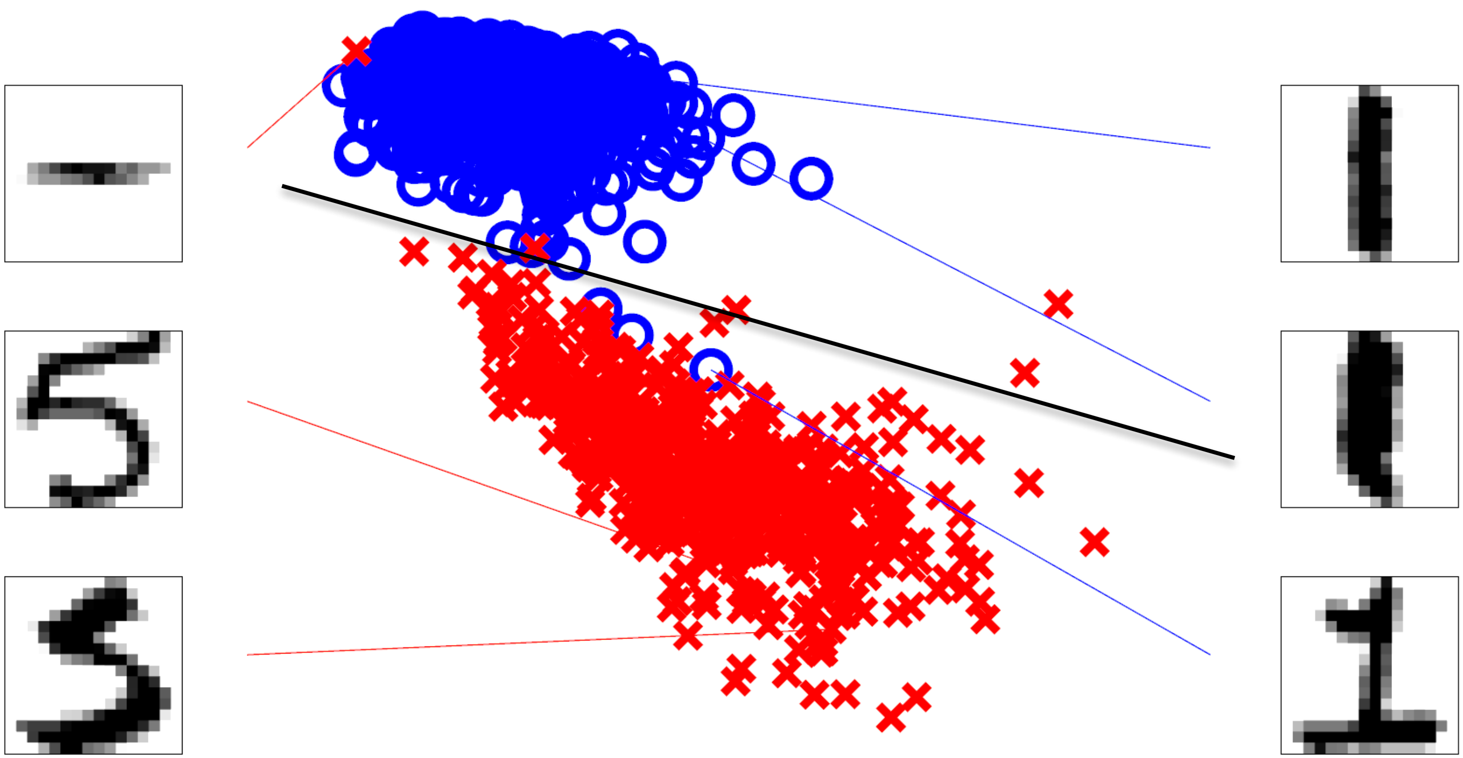






Example: linear classifier

$\mathbf{x} = (x_0, x_1, x_2)$ x_1: intensity x_2: symmetry



https://work.caltech.edu/telecourse.html

(Supervised) Learning problem

- Selecting a **hypothesis space**
  - Hypothesis space: a set of mappings from feature vector to target

- **Learning (estimation)**: optimization of a cost function
  - Based on the training set $D = \{(x^{(i)}, y^{(i)})\}_{i=1}^{n}$ and a cost function we find (an estimate) $f \in F$ of the target function

- **Evaluation**: we measure how well $\hat{f}$ generalizes to unseen examples

Generalization

- We don't intend to memorize data but want to distinguish the pattern.
- A core objective of learning is to generalize from the experience.
  - Generalization: ability of a learning algorithm to perform accurately on new, unseen examples after having experienced.

Paradigms of ML

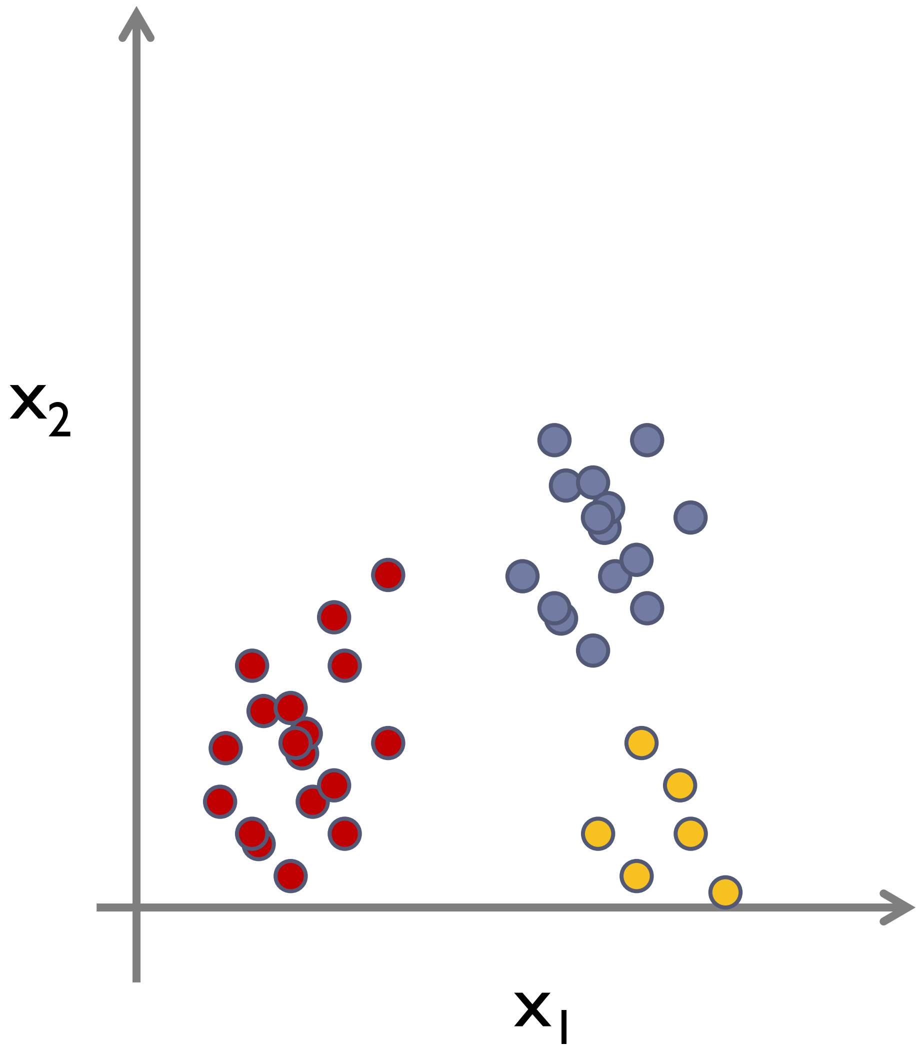
- Supervised learning (regression, classification)
  - predicting a target variable for which we get to see examples.
- Unsupervised learning
  - revealing structure in the observed data
- Reinforcement learning
  - partial (indirect) feedback, no explicit guidance
  - Given rewards for a sequence of moves to learn a policy and utility functions

Supervised Learning vs. Unsupervised Learning

- Supervised learning
  - Given: Training set
    - labeled set of N input-output pairs $D = \left\{\left(\boldsymbol{x}^{(i)}, y^{(i)}\right)\right\}_{i=1}^{N}$
  - Goal: learning a mapping from $\boldsymbol{x}$ to y

- Unsupervised learning
  - Given: Training set
    - $\left\{\boldsymbol{x}^{(i)}\right\}_{i=1}^{N}$
  - Goal: find groups or structures in the data
    - Discover the intrinsic structure in the data

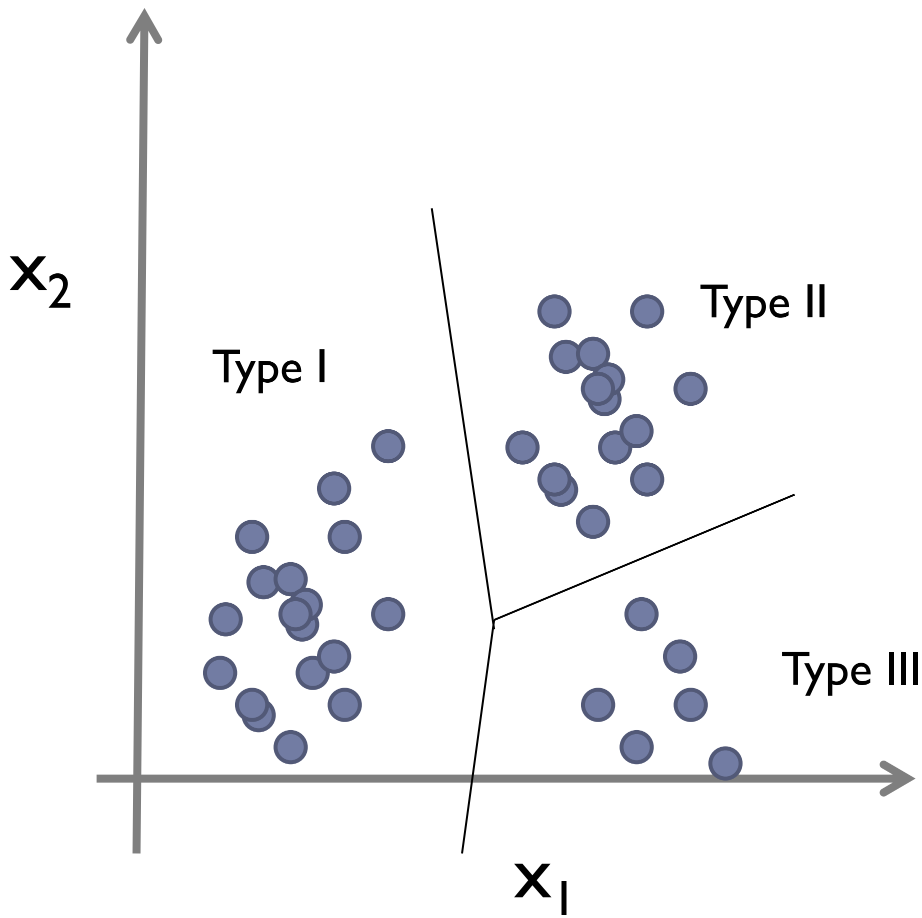
Supervised Learning: Samples



Classification

Unsupervised Learning: Samples

- Wants to use data to improve their knowledge on a task



Clustering

Sample Data in Unsupervised Learning

- Unsupervised Learning:

Columns:
Features/attributes/dimensions

Rows:
Data/points/instances/examples/samples

	x_1	x_2	…	x_d
Sample1				
Sample 2				
…				
Sample n-1				
Sample n				

Unsupervised learning

- **Clustering**: partitioning of data into groups of similar data points.

- **Dimensionality reduction**: data representation using a smaller number of dimensions while preserving (perhaps approximately) some properties of the data.

- **Density estimation**

Some clustering purposes

- **Preprocessing stage** to index, compress, or summarize the data
- As a tool to **understand the hidden structure** in data or to **group** them
  - To gain knowledge (insight into the structure of the data) or
  - To group the data when no label is available

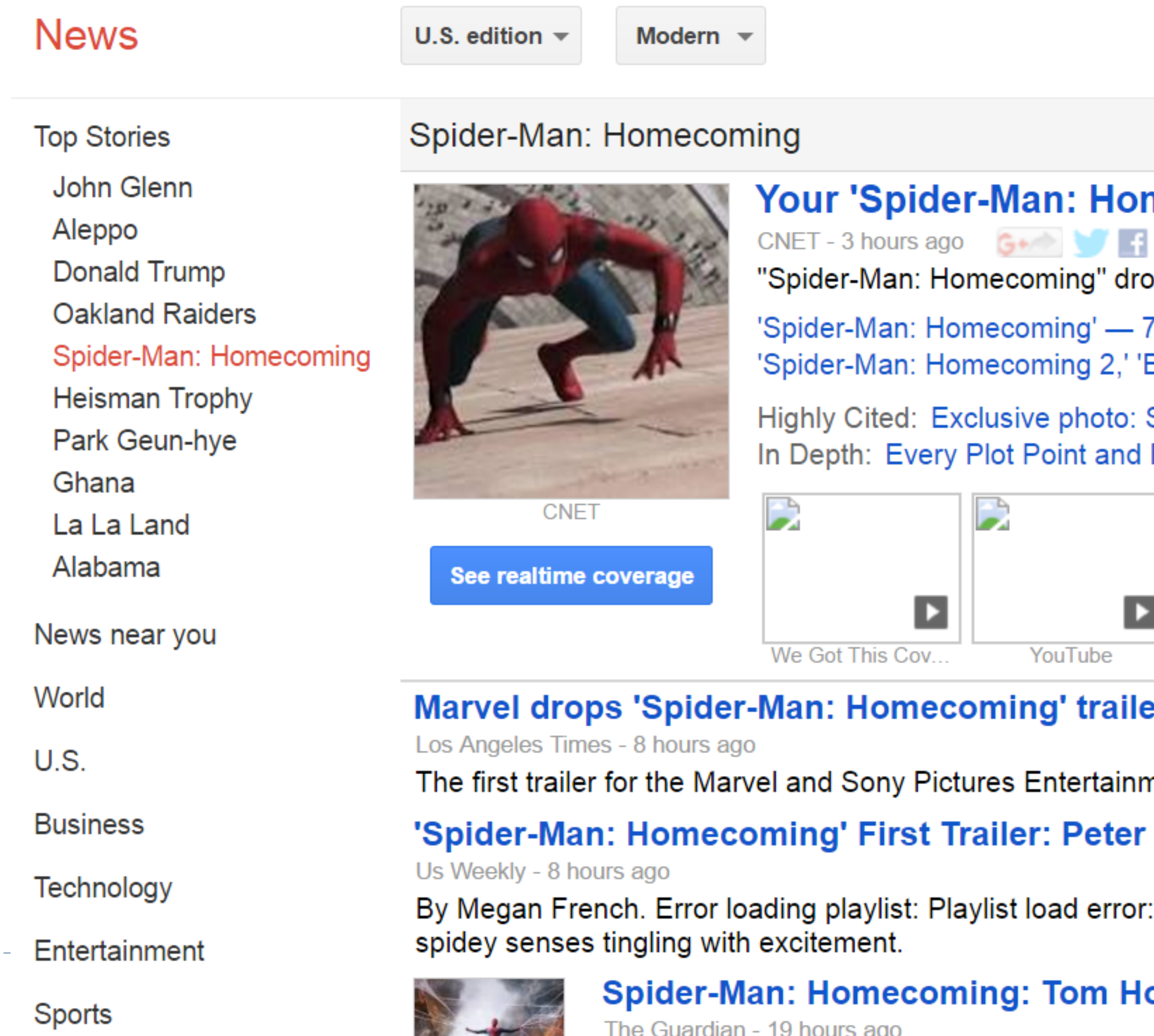
Clustering: Example Applications

- Clustering docs based on their similarities
  - Grouping new stories in the Google news site

- Market segmentation: group customers into different market segments given a database of customer data.

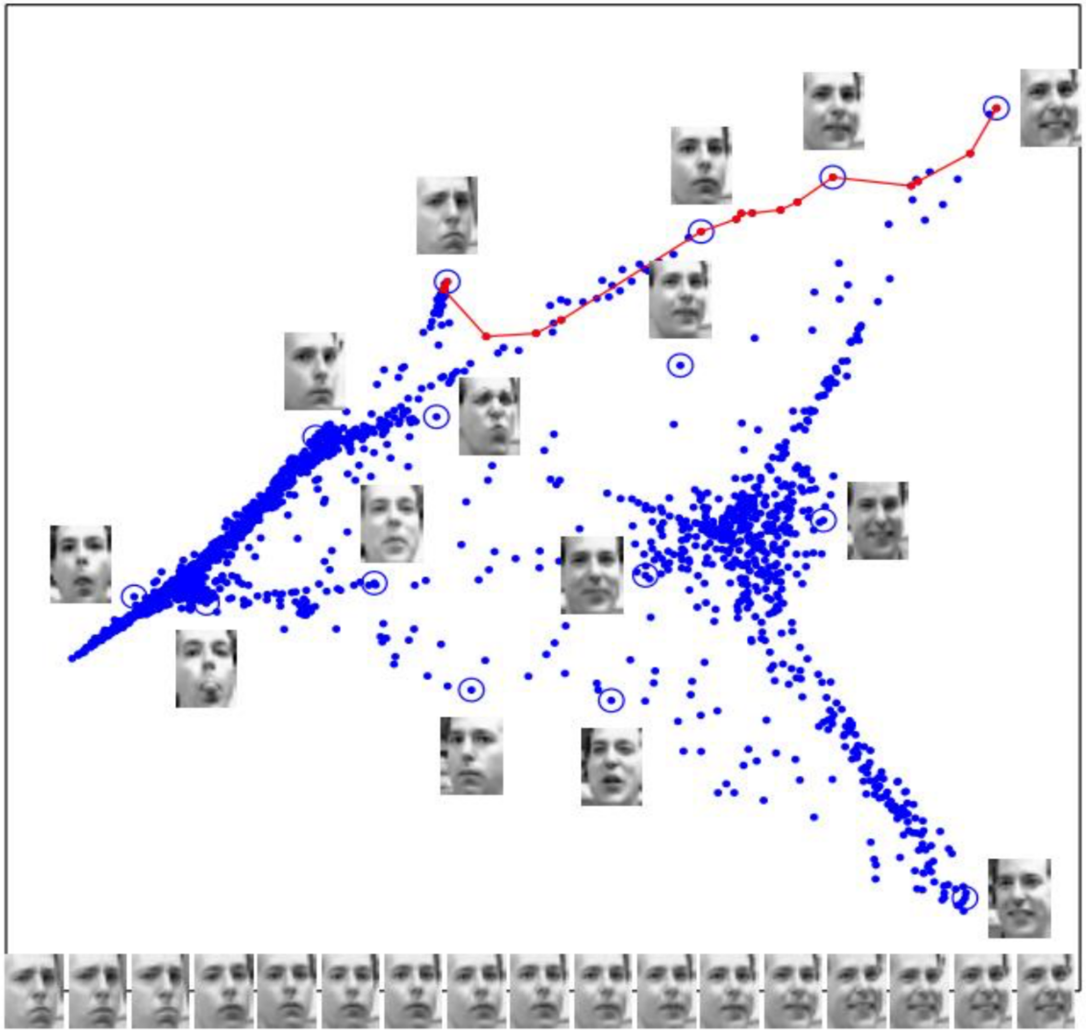
Clustering of docs

- Google news



Dimensionality reduction: Example

How to map the high dimensional data into a lower dimensional space in which the distance is more meaningful.



[Saul & Roweis 2003]

Paradigms of ML

- Supervised learning (regression, classification)
  - predicting a target variable for which we get to see examples.
- Unsupervised learning
  - revealing structure in the observed data
- Reinforcement learning
  - partial (indirect) feedback, no explicit guidance
  - Given rewards for a sequence of moves to learn a policy and utility functions

Reinforcement

- Provides only an indication as to whether an action is correct or not

Data in supervised learning:

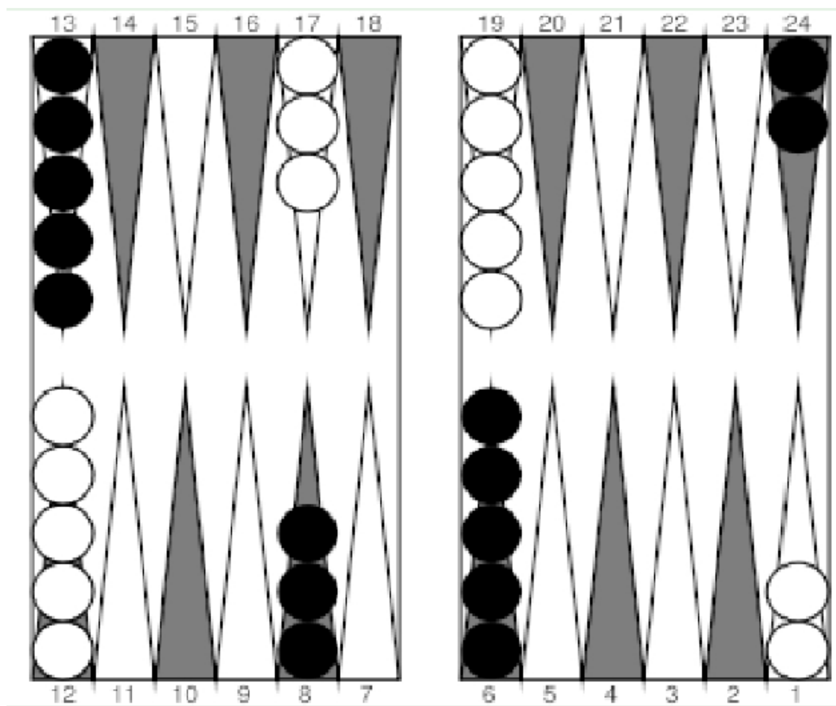
(input, correct output)

Data in Reinforcement Learning:

(input, some output, a reward for this output)

Reinforcement Learning

- Typically, we need to get a sequence of decisions
- Usually, need to decide under uncertainty



Learn a policy that specifies the action for each state

Paradigms of ML

- Supervised learning (regression, classification)
  - predicting a target variable for which we get to see examples.
- Unsupervised learning
  - revealing structure in the observed data
- Reinforcement learning
  - Reasoning under uncertainty
  - partial (indirect) feedback, no explicit guidance
  - Given rewards for a sequence of moves to learn a policy and utility functions

- Other paradigms: semi-supervised learning, active learning, etc.

Active learning

- Select not only the model but also the most informative samples to be labeled

- learn a selection function to maximize the success of the supervised learning

Three axes of ML

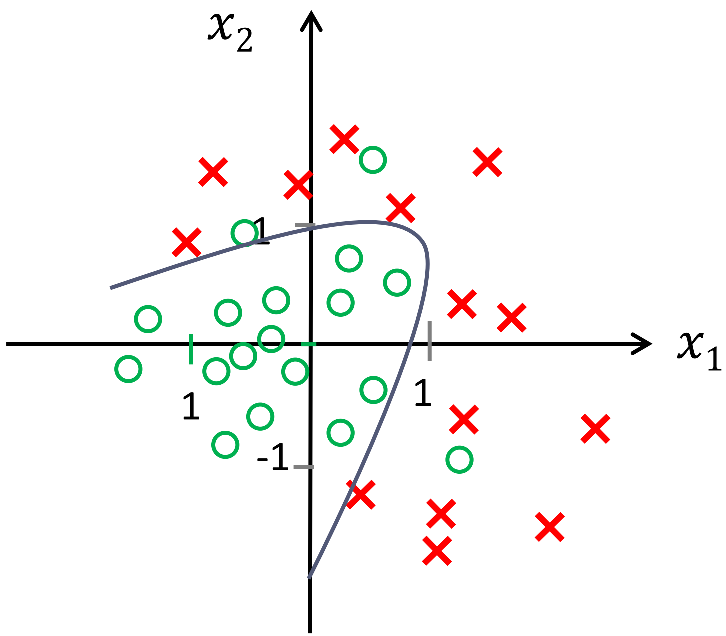
- Data
- Task (i.e. what is the type of knowledge that we seek from data)
- Algorithm

Three axes of ML

- Data
  - Fully observed
  - Partially observed
  - Actively collecting data
- Task (i.e. what is the type of knowledge that we seek from data)
  - Prediction (i.e. classification or regression)
  - Control
  - Description
- Algorithm
  - Parametric models
  - Nonparametric models

Parametric models

- We consider a parametric boundary (e.g., hyper-plane, hyperbola, …) and learn its parameters form data
  - The set of parameters does not grow with increasing the data

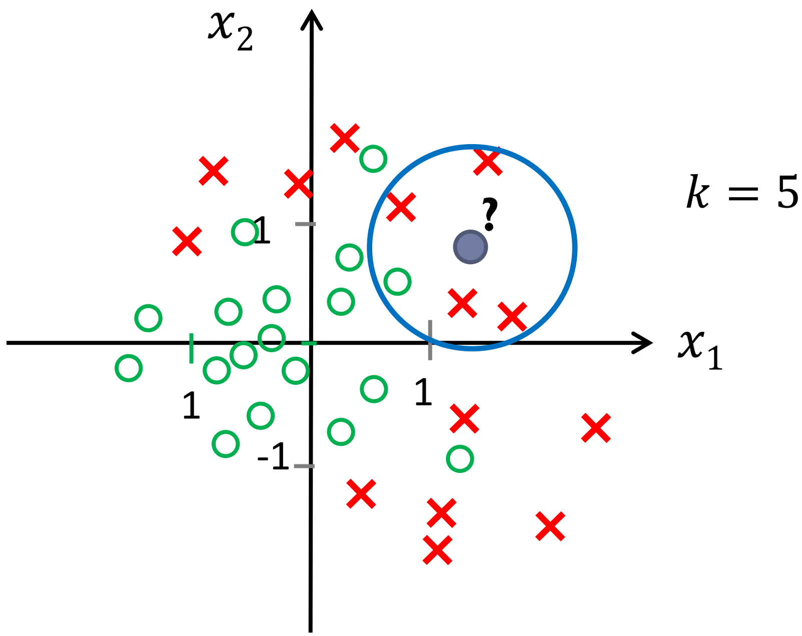


Nonparametric models

- We must store data and for each prediction, we need to process training data
- More data means a more complex model
  - Models that grow with the data

Nonparametric models

- k-NN classifier
  - Label for $\boldsymbol{x}$ predicted by majority voting among its k-NN.



Find k nearest training data to the new input and predict its label from the labels of its k nearest neighbors

The number of points to search scales with the training data

Some Learning Application Areas

- Computer Vision (Photo tagging, face recognition, …)
- Natural language processing (e.g., machine translation)
- Robotics
- Speech recognition
- Autonomous vehicles
- Social network analysis
- Web search engines
- Medical outcomes analysis
- Market prediction (e.g., stock/house prices)
- Computational biology (e.g., annotation of biological sequences)
- Self-customizing programs (recommender systems)

ML in Computer Science

- Why ML applications are growing?
  - Improved machine learning algorithms
  - Availability of data (Increased data capture, networking, etc)
  - Software too complex to write by hand
    - Demand for complex systems (on high-dimensional, multi-modal, or heterogeneous data)
    - Demand for self-customization to user or environment

Relation to other fields

- **Statistics:** the goal is the understanding of the data at hand
- **Artificial Intelligence:** the goal is to build an intelligent agent
- **Data Mining:** the goal is to extract patterns from large-scale data
- **Data Science:** the science encompassing collection, analysis, and interpretation of data

- The goal of machine learning is the underlying mechanisms and algorithms that allow improving our knowledge with more data

http://www.cs.cmu.edu/~pradeepr/701

Topics of this course

- Ml, Map, and Bayesian
- Regression & generalization
- Linear classifier
- Probabilistic classifiers
- SVM & kernel
- Neural Networks
- Decision tree
- Learning Theory
- Non-parametric methods
- Ensemble learning
- Dimensionality reduction
- Clustering
- Reinforcement Learning
- Advanced Topics