Machine Learning

Amirkabir University of Technology
(Tehran Polytechnic)

Lecture 1.
Course Overview & Introduction

Alireza Rezvanian

Fall 2023

Amirkabir University of Technology (Tehran Polytechnic)

Last update: Sep. 17, 2022

Outline

- Course overview
- Syllabus
- References
- Policies
- Evaluation & grading
- Research methods
- Machine learning

Course overview

- **Instructure**
  - Alireza Rezvanian
    - Email: rezvanlms[At]gmail[dot]com
    - URL: https://ce.aut.ac.ir/~rezvanian/
  - TA: Mr. Ahmadi & Babae
  - Class platform
    - https://courses.aut.ac.ir

Evaluation

- Discipline
  - 5% of the final grade (tentative)
- Homeworks and Quizzes
  - About 4 prearranged homework.
  - **Important!** no large delay is accepted.
  - 25% of the final grade (tentative)
- Research presentation, report and simulation
  - Select 1 WoS journal paper published after 2021 having IF>1.5 or SNIP>1.5 or Q1
  - 30% of the final grade (tentative)
- Exam
  - Conceptual questions about topics
  - 15% of the final grade for mid-exam (tentative)
  - 25% of the final grade for final exam (tentative)
- Evaluation on your works (not any other)!
- 20-90% penalty per day for late submission of homework
- Email policy (subject: **MLPR_4021_ID**, introduction, body, rezvanlms@gmail.com)

Policies & Assumptions

- Familiar with fundamental computer sciences, mathematics, probability, and statistics
- Love (like) mathematic and algorithms (at least don't hate)
- Interested in problem solving
- Everything is abstract model of real problem
- We don't discuss about details of models, algorithms and implementations.
- You have no problem with class time
- You can use Python libraries
- You as an engineer should behave like an engineer !
  - Accuracy, Time, Ethics

Topics of this course

- Feature engineering
- Regression & generalization
- Classification
- Probabilistic classifiers
- Support Vector Machine (SVM)
- Decision tree
- Neural Networks
- Genetic algorithm
- Non-parametric methods
- Ensemble learning
- Dimensionality reduction
- Clustering
- Reinforcement Learning

References

1) T. Mitchel, **Machine learning**, McGraw-Hill Education, 1998.
2) E. Alpaydin, **Introduction to Machine Learning**, 4th ed., The MIT Press, 2020.
3) C. M. Bishop, **Pattern recognition and machine learning**, Springer, 2006.
4) S. Theodoridis and K. Koutroumbas, **Pattern recognition**, 4th ed., Academic Press, 2008.
5) R. O. Duda, P. E. Hart. D. G. Stork, **Pattern classification**, 2nd ed., John Wiley & Sons, 2006.
6) A. Muller, S. Guido, **Introduction to Machine Learning with Python: A Guide for Data Scientists**, O'Reilly, 2106.

Resources: Datasets

- UCI Repository
  - http://archive.ics.uci.edu/ml/index.php
- UCI KDD Archive
  - https://kdd.ics.uci.edu
- Kaggle
  - https://www.kaggle.com/datasets
- Delve Datasets
  - https://www.cs.toronto.edu/~delve/data/datasets.html
- Statlib
  - http://lib.stat.cmu.edu/datasets/

Resources: Journals

- Journal of Machine Learning Research
  www.jmlr.org
- Machine Learning
- Neural Computation
- Neural Networks
- IEEE Transactions on Neural Networks and Learning Systems (NNLS)
- IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)
- Pattern Recognition (PR)
- Information Sciences (INS)
- Knowledge-based Systems (KNOSYS)

Resources: Conferences

- International Conference on Machine Learning (ICML)
- Neural Information Processing Systems (NIPS)
- European Conference on Machine Learning (ECML)
- The ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)
- International Conference on Pattern Recognition (ICPR)
- International Conference on Learning Representations (ICLR)
- Computational Learning Theory (COLT)
- International Conference on Artificial Neural Networks (ICANN)
- International Conference on AI & Statistics (AISTATS)

Famous researchers

vapnik 98, 289149

Professor of Columbia, Fellow of NEC Labs America,
Verified email at nec-labs.com
machine learning   statistics   computer science

Kevin Murphy 97, 95260

Research Scientist, Google
Verified email at google.com - Homepage
Artificial Intelligence   Machine Learning   Computer Vision
Natural Language Processing

Andrew Ng 142, 238311

Stanford University
Verified email at cs.stanford.edu - Homepage
Machine Learning   Deep Learning   AI

Jure Leskovec 141, 144627

Professor of Computer Science, Stanford University
Verified email at cs.stanford.edu - Homepage
Data mining   Machine Learning   Graph Neural Networks
Knowledge Graphs   Complex Networks

Christopher M. Bishop 70, 136605

Distinguished Scientist, Microsoft Research, Cambridge, U.K.
Verified email at microsoft.com - Homepage
Machine learning

Tom Mitchell 98, 75426

Founders University Professor of Machine Learning,
Carnegie Mellon University
Verified email at cs.cmu.edu - Homepage
Machine Learning   cognitive neuroscience
natural language understan...

Richard S. Sutton 100, 138741

DeepMind, Amii, and University of Alberta
Verified email at richsutton.com - Homepage
artificial intelligence   reinforcement learning   machin
computer science

Geoffrey Hinton 179, 713100

Emeritus Prof. Computer Science, University of Toronto
Verified email at cs.toronto.edu - Homepage
machine learning   psychology   artificial intelligence

What is machine learning?

- **Learning**: "the acquisition of knowledge or skills through experience, study, or by being taught."
- **Machine Learning**:
  - Arthur Samuel (1959): Field of study that gives computers, the ability to learn without being explicitly programmed.
  - Herbert Simon (1970): Any process by which a system improves its performance
  - Tom Mitchell (1998): A computer program that improves its performance with experience using the observed data to make better decisions (generalizing from the observed data)
  - Alpaydin (2004): programming computers to optimize a performance criterion using example data or past experience.
  - Kevin Murphy (2012): algorithms that automatically detect patterns in data use the uncovered patterns to predict future data or other outcomes of interest
- 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

Why "Learn" ?

- Machine learning is programming computers to optimize a performance criterion using example data or past experience.
- There is no need to "learn" to calculate payroll
- Learning is used when:
  - Human expertise does not exist (navigating on Mars),
  - Humans are unable to explain their expertise (speech recognition)
  - Solution changes in time (routing on a computer network)
  - Solution needs to be adapted to particular cases (user biometrics)

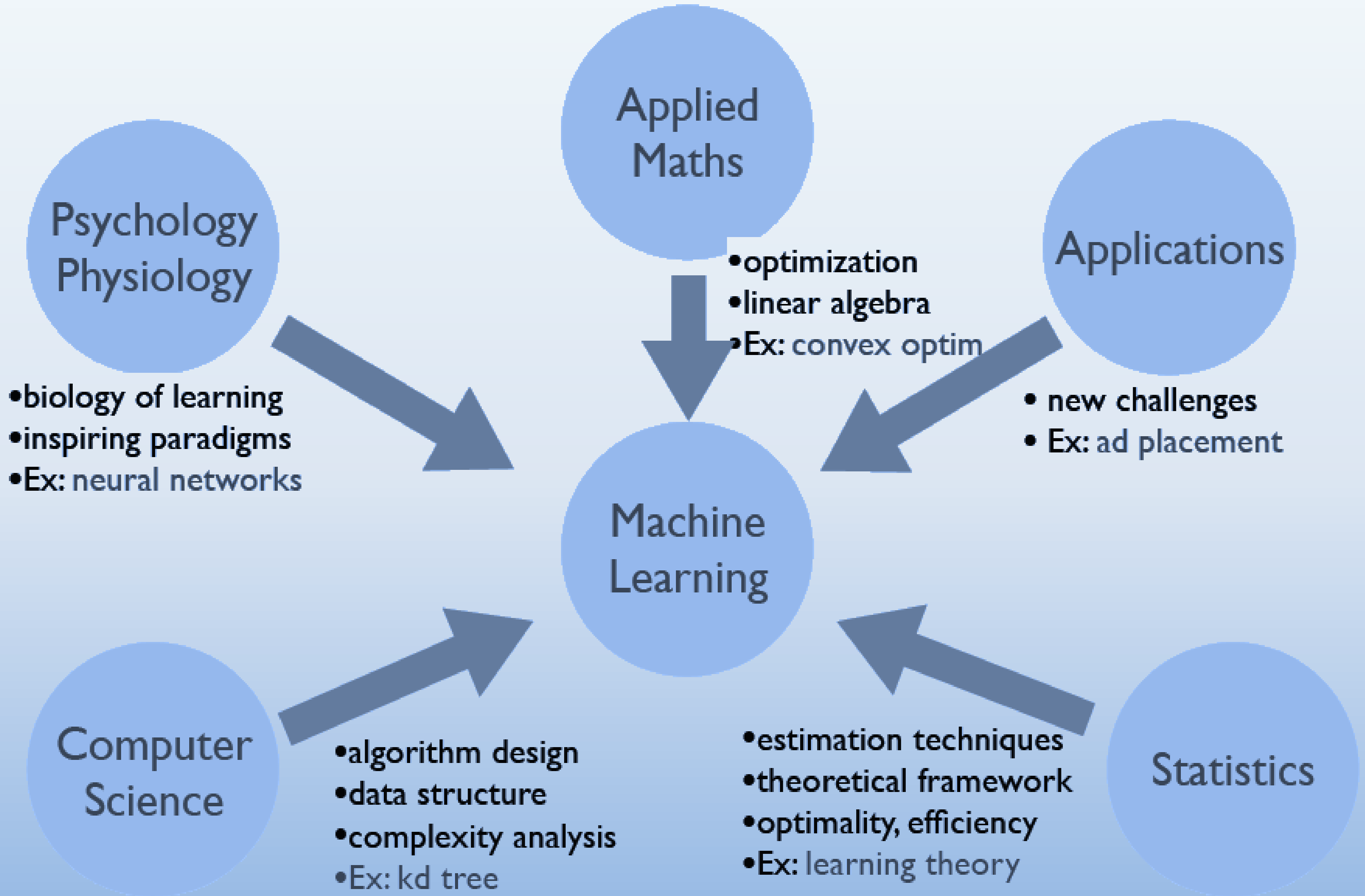
What We Talk About When We Talk About "Learning"

- Learning general models from a data of particular examples
- Data is cheap and abundant (data warehouses, data marts); knowledge is expensive and scarce.
- Example in retail: Customer transactions to consumer behavior:

  People who bought "X" also bought "Y" (www.amazon.com)

- Build a model that is *a good and useful approximation* to the data.

Where does ML fit in?



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

Some Learning Application Domains

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

ML unicorn business

- ByteDance
- SambaNova Systems
- UiPath
- Dataminr
- HighRadius
- SenseTime
- Feedzai Inc.
- Scale AI Inc.
- Automation Anywhere
- Pony.ai

https://www.analyticsinsight.net/top-10-ai-unicorns-that-are-setting-the-stage-on-fire/

ML in a Nutshell

- Every machine learning algorithm has three components:
  - Representation / feature set / Model Class (form)
  - Evaluation / Objective Function / Loss function
  - Optimization
    - Optimize a performance criterion using example data or past experience
    - Parameter estimation
    - Model selection and hyper-parameter tuning
- We have different types of (getting) observations in different types or paradigms of ML methods

Representation / Model Class

$$\hat{y} = f(x; w) = w^T x$$

- Decision trees
- Sets of rules / Logic programs
- Instances
- Graphical models (Bayes/Markov nets)
- Neural networks
- Support vector machines
- Model ensembles
- Etc.

Evaluation / Objective Function

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

$$L(y,\hat{y}) = \left||y,\hat{y}|\right|_2$$

Optimization

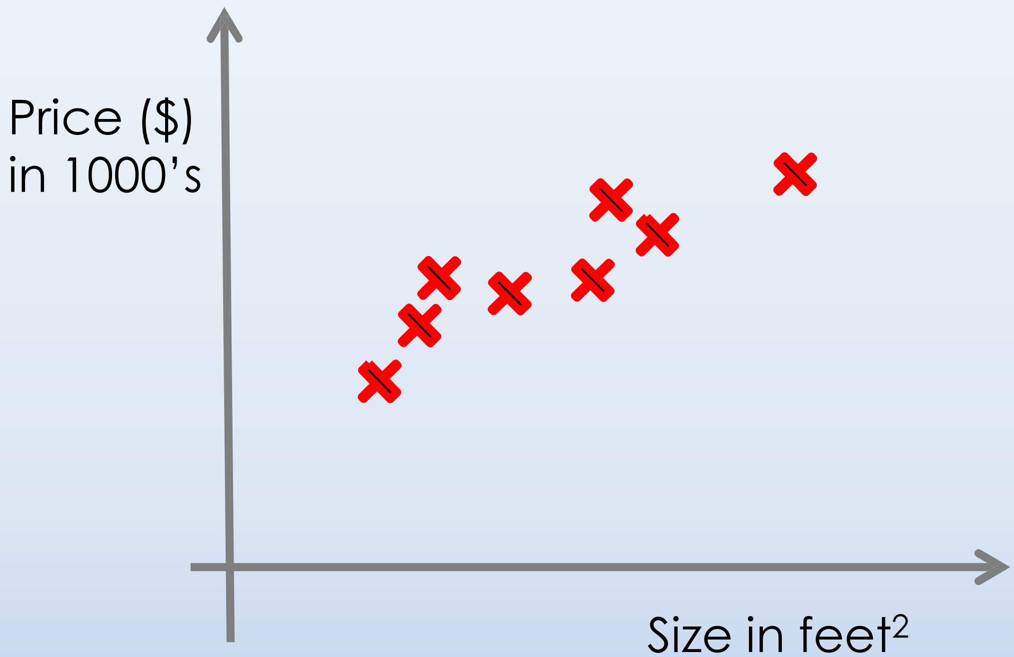
$$argmin_w\, L(y, \hat{y}(w))$$

- Discrete/Combinatorial optimization
  - Greedy search
  - Graph algorithms (cuts, flows, etc)

- Continuous optimization
  - Convex/Non-convex optimization
  - Linear programming

Example: Home Price

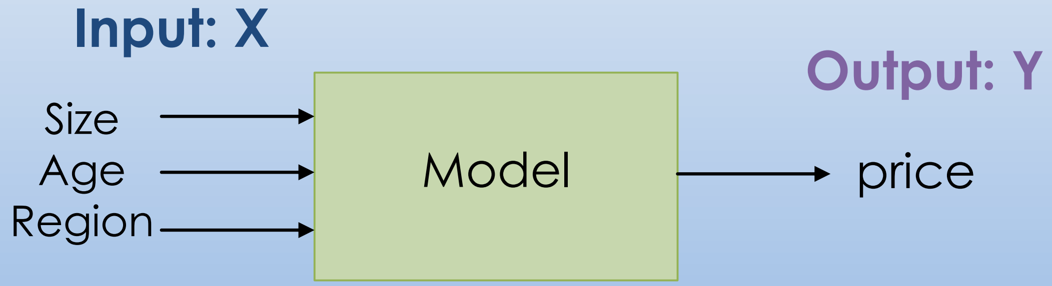
- Housing price prediction



Example: Home Price

- Predicting house price from 3 attributes

Size (m^2)	Age (year)	Region	Price (10^9T)
100	2	2	5
80	10	3	3
…	…	…	…

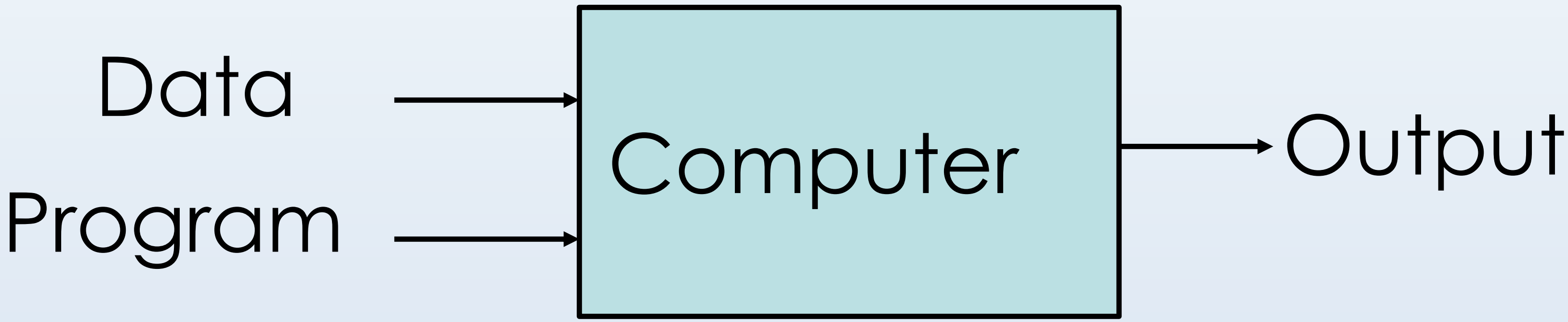


Example: Bank loan (Credit scoring)

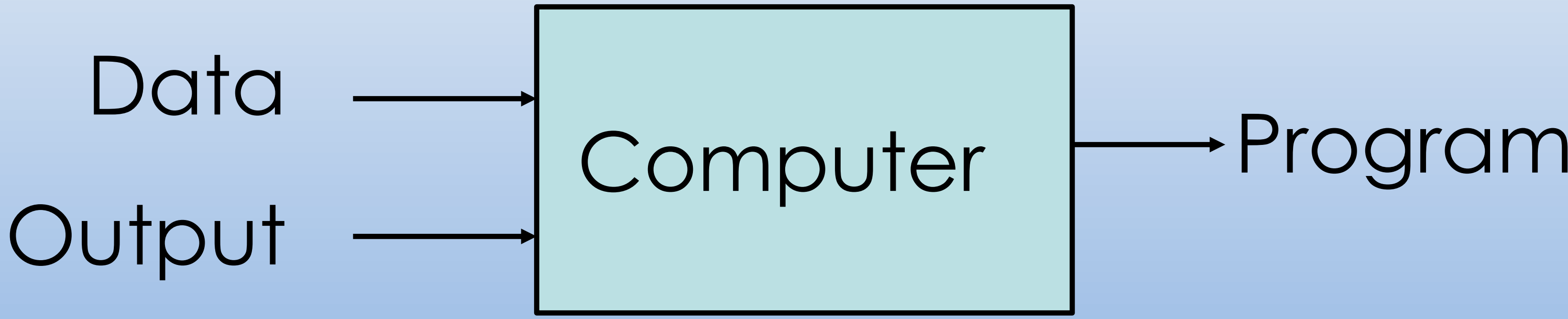
- Applicant form as the input:
  - Salary
  - Age
  - gender
  - current debt
  - …
- Output: **approving** or **denying** the request

Comparison

- **Traditional Programming**



- **Machine Learning**



Paradigms of ML

- **Supervised learning** (regression, classification)
  - Predicting a target variable for which we get to see examples.
  - Training data includes desired outputs
- **Unsupervised learning**
  - revealing structure in the observed data
  - Training data does not include desired outputs
- **Weakly** or **Semi-supervised learning**
  - Training data includes a few desired outputs
- **Reinforcement learning**
  - Partial (indirect) feedback, no explicit guidance
  - Given rewards for a sequence of moves to learn a policy and utility functions
  - Rewards from sequence of actions

Data in Supervised Learning

- Data are usually considered as vectors in a d dimensional space

$$\bar{X} = \begin{bmatrix} X_1 \\ X_2 \\ \cdot \\ \cdot \\ Xd \end{bmatrix} = [X_1, X_2, \dots\ X_d]^T$$

- **Columns**
  - Features
  - Attributes
  - Dimensions
- **Rows**
  - Data
  - Points
  - Instances
  - Examples
  - Samples
  - Records
- **Y column**
  - Class
  - Target
  - Outcome
  - Response
  - Label

	X_1	X_2	…	X_d	Y (Target)
Sample 1					
Sample 2					
…					
Sample n-1					
Sample n					

Y

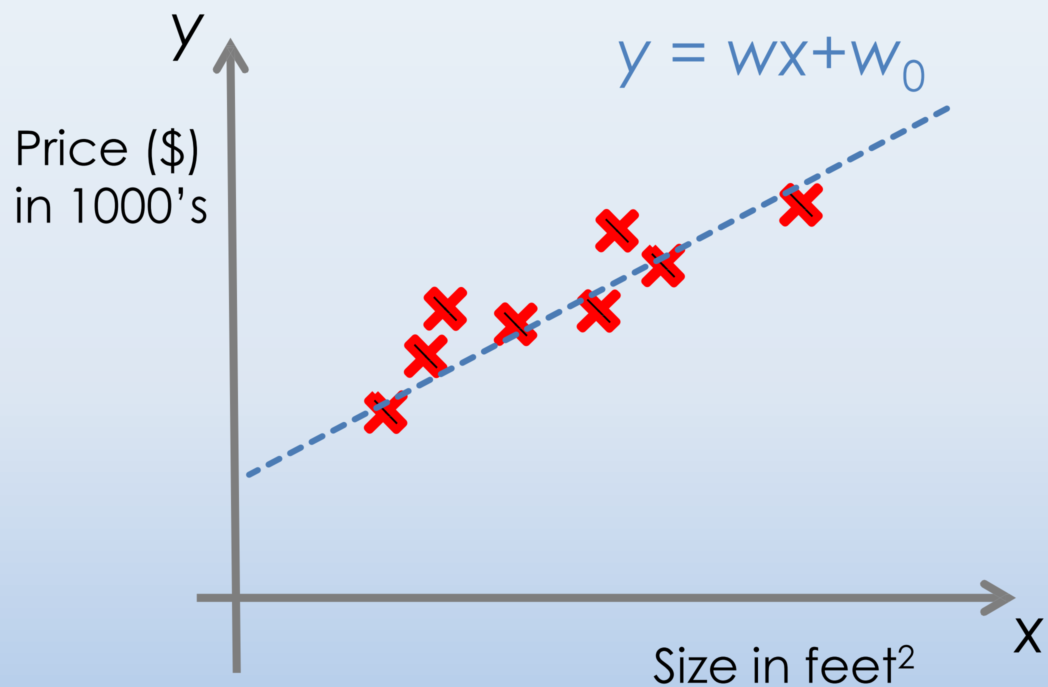
$$\mathbf{X} = [\mathbf{x}_1, \dots, \mathbf{x}_d]$$

Supervised Learning: Regression vs. Classification

- Supervised Learning
  - **Regression**: predict a continuous target variable
    - E.g., $y \in [0,1]$
    - Examples: stock market prediction, weather prediction, pose estimation.
  - **Classification**: predict a discrete (unordered) target variable
    - E.g., $y \in \{1,2, \ldots, C\}$
    - Examples: image classification, face recognition, speech recognition,

Regression: Example

- Housing price prediction



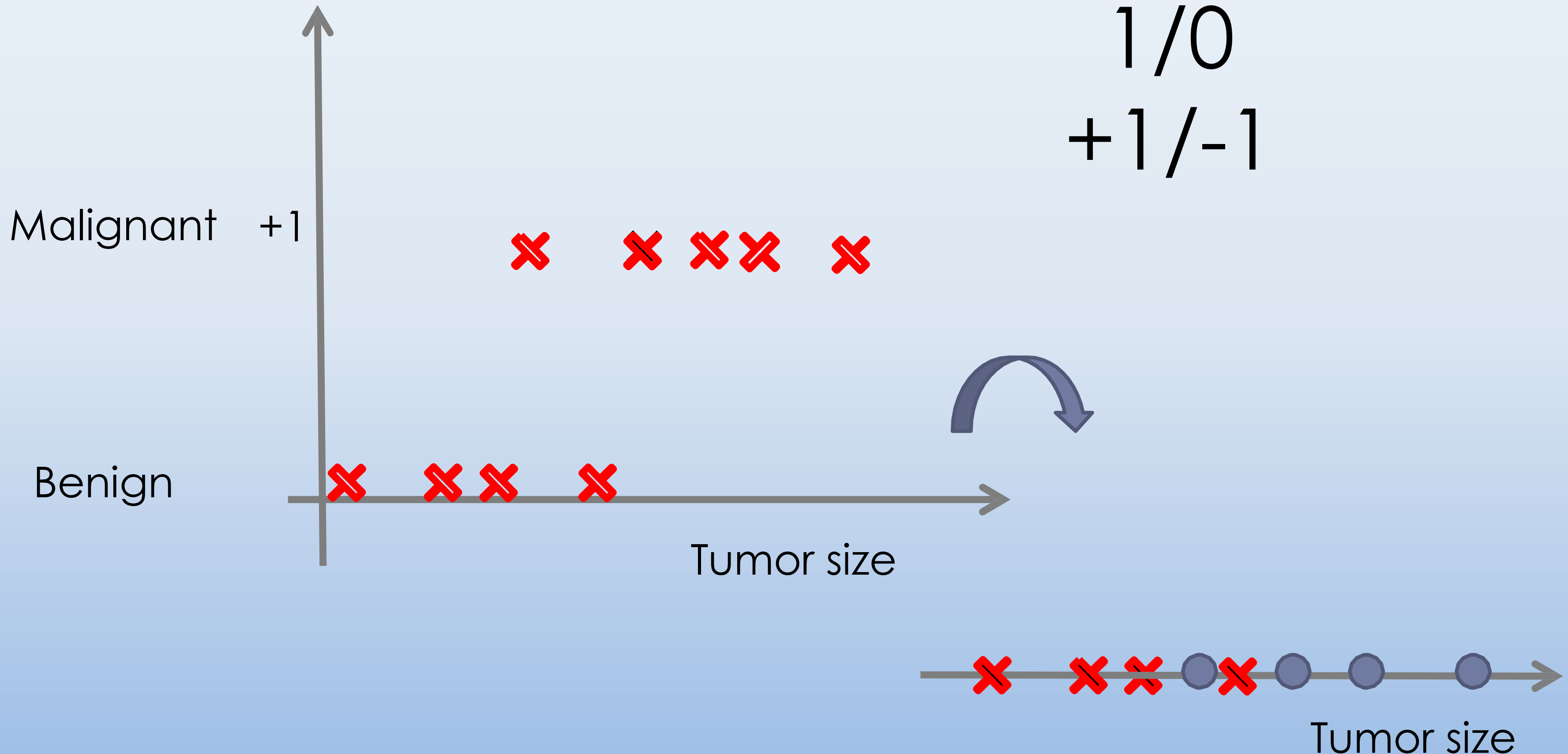
Classification: Example

- Classification of tumors to Benign/Malignant according to attributes

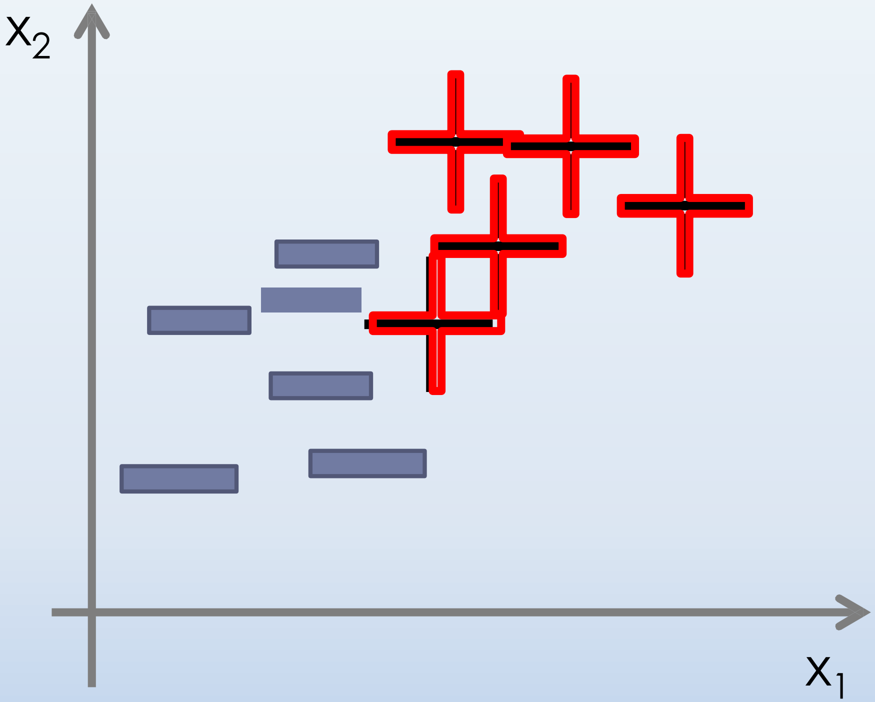
+/-

1/0

+1/-1



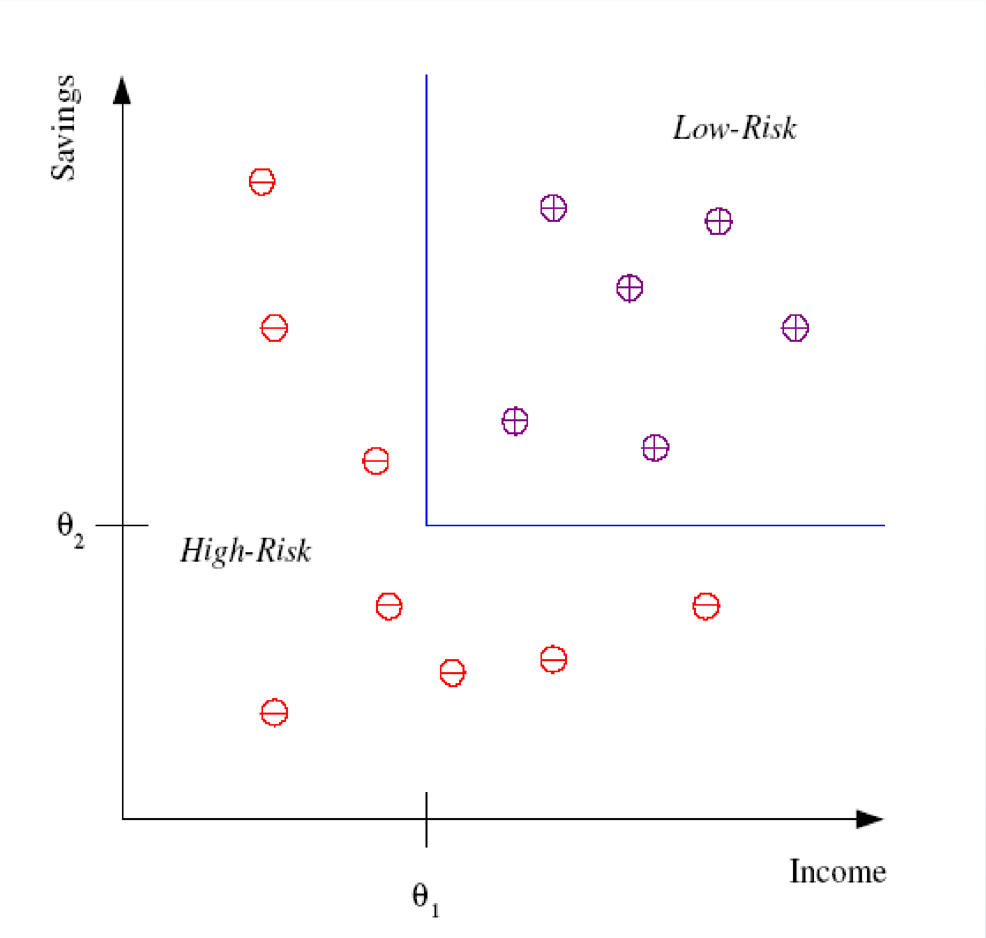
Training data: Example



X_1	X_2	Y
0.9	2.3	0
3.5	2.6	0
2.6	3.3	0
2.7	4.1	0
1.8	3.9	0
3.5	3	1
4.2	3.7	1
4.9	4.5	1
3.9	4.5	1
5.8	4.1	1
6.1	2.6	1

Classification: example

- Example: Credit scoring
- Differentiating between low-risk and high-risk customers from their *income* and *savings*

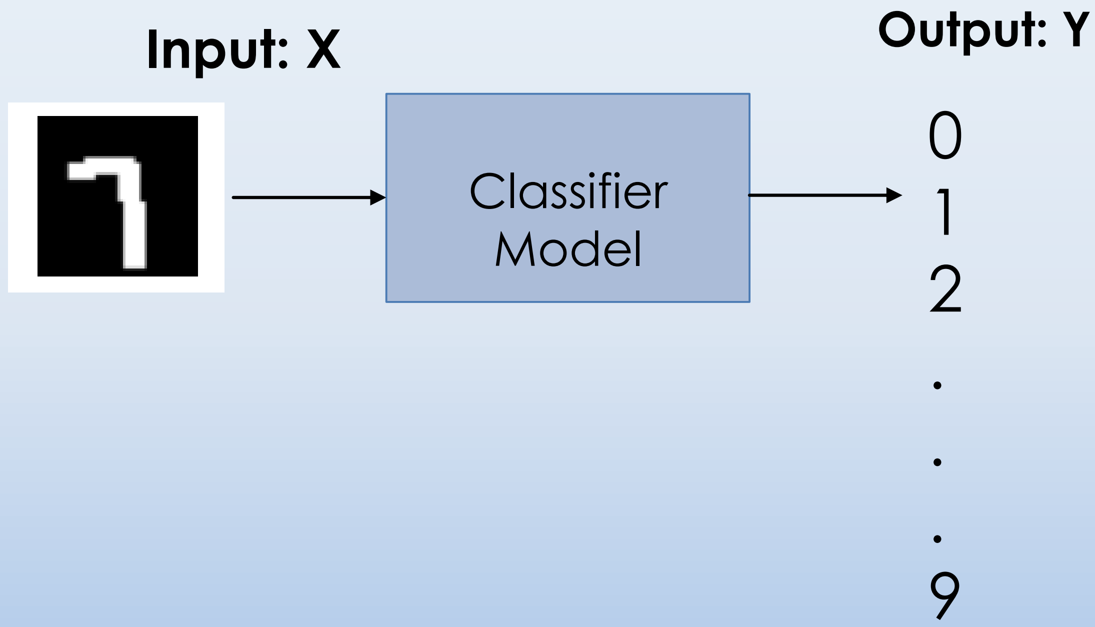


Discriminant: IF *income* > θ_1 AND *savings* > θ_2
THEN low-risk ELSE high-risk

Handwritten Digit Recognition Example

- Data: labeled samples

Handwritten Digit Recognition Example



Components of (Supervised) Learning

- Unknown target function: f: $X \rightarrow Y$
  - Input space/input feature: X
  - Output space/output feature: Y

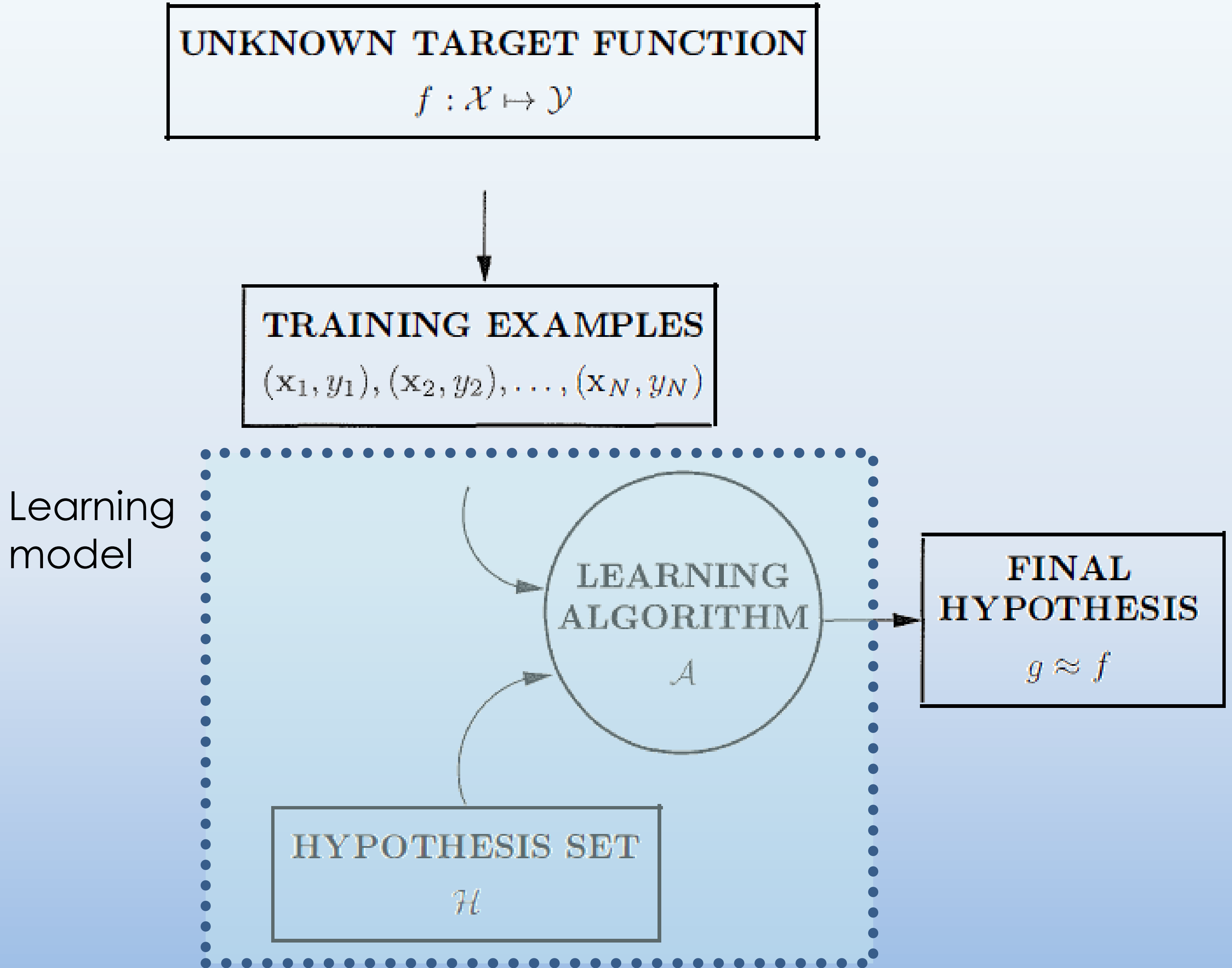
- Training data: $(x_1, y_1), (x_1, y_1), \ldots, (x_n, y_n)$

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

Components of (Supervised) Learning

- We have some example pairs of (input, output) called training samples
  - (x_1, y_1), (x_1, y_1) , … , (x_n, y_n)
- We want to select a function from the input space to the output space
  - $f: X \rightarrow Y$
- 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

Components of (Supervised) Learning

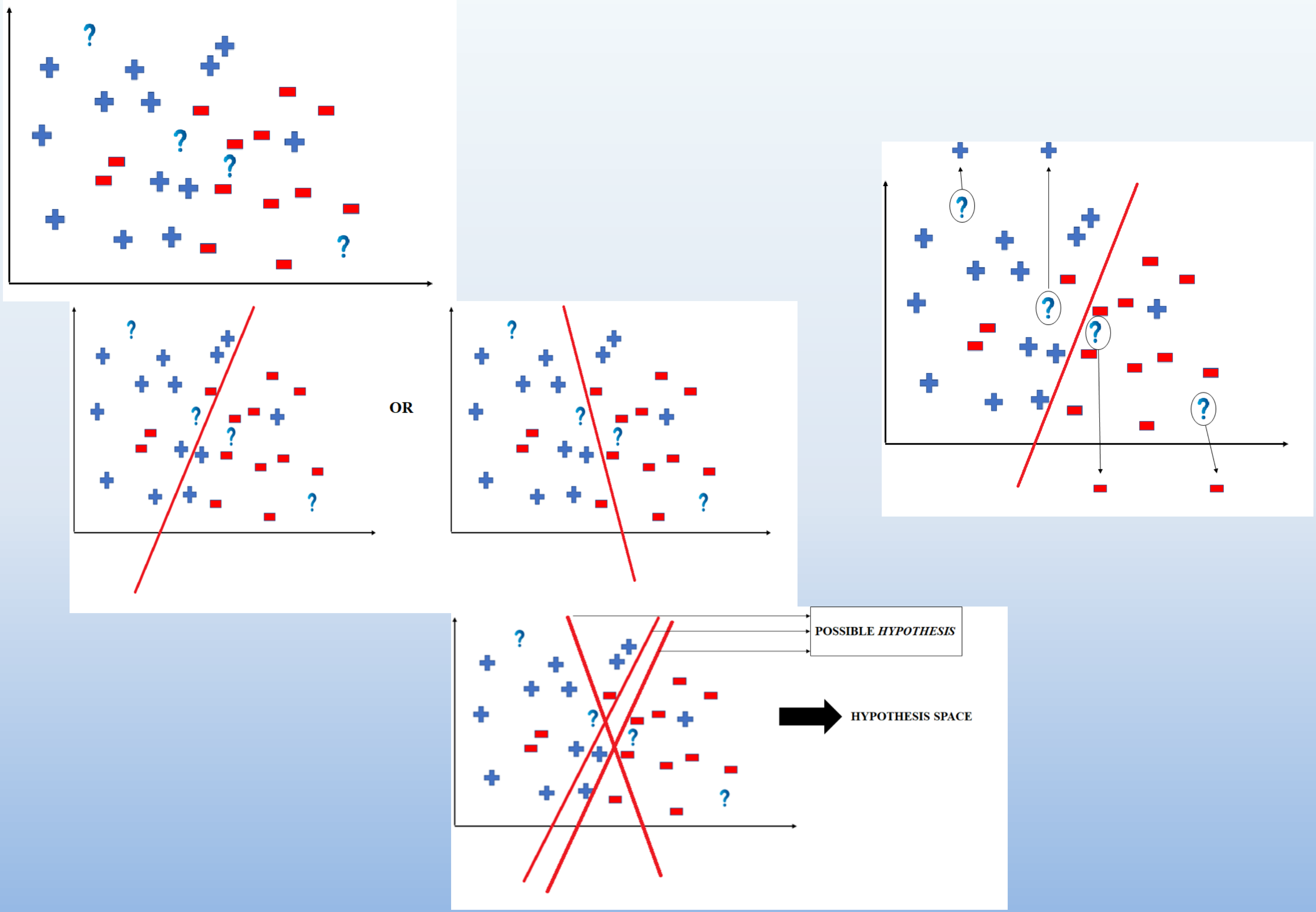


Learning From Data A short course, Y. S. Abu-Mostafa, M. Magdon-Ismail, HT. Lin - (2012)

(Supervised) Learning problem

- Selecting a hypothesis space
  - Hypothesis space: a set of mappings from feature vector to target
- Learning: find mapping g training data (from hypothesis set) based on the training data
  - Which notion of error should we use? (loss functions)
  - Optimization of loss function to find mapping g
- Evaluation: we measure how well g generalizes to unseen example (generalization)

hypothesis space

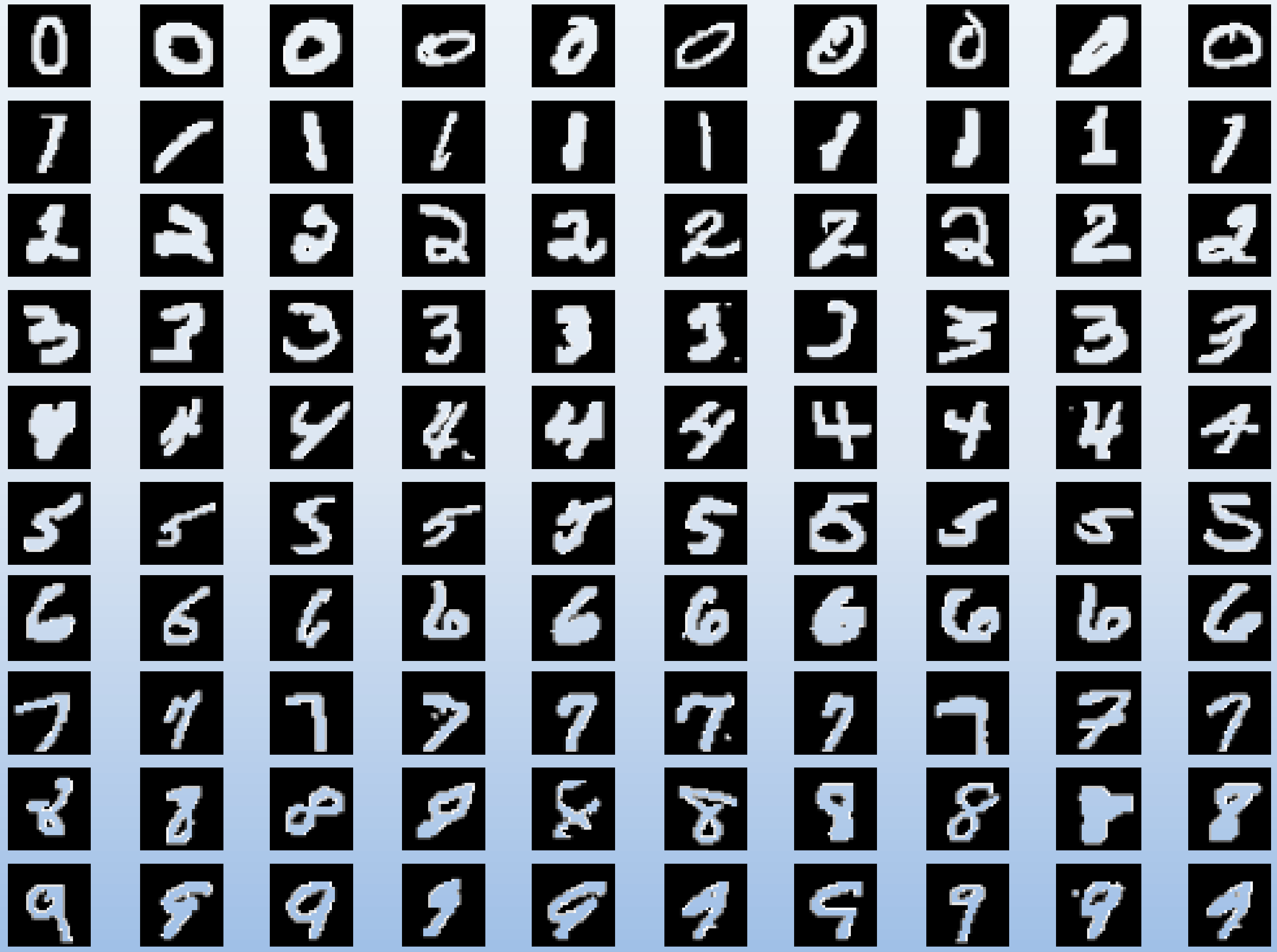


Solution Components

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

Handwritten Digit Recognition Example

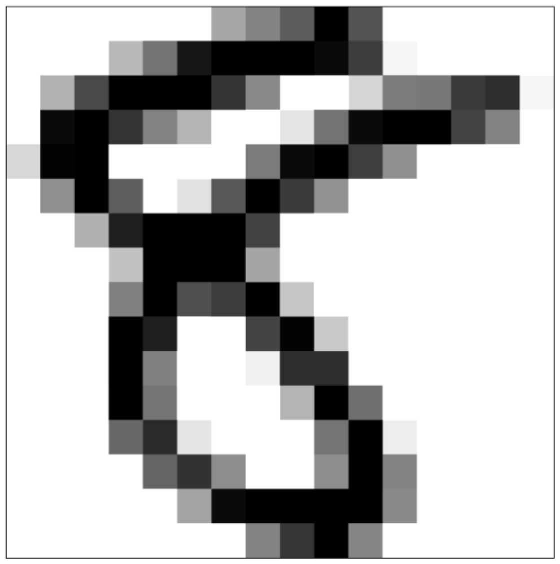
- Data: labeled samples



0
1
2
3
4
5
6
7
8
9

Example: Input representation

- Raw input $X=(x_0, x_1, \ldots x_{256})$
- Linear model input $W=(w_0, w_1, \ldots w_{256})$
- Features: extract useful information e.g.,
  - Intensity and symmetry $X(x_0, x_1, x_2)$
  - Linear model: (w_0, w_1, w_2)

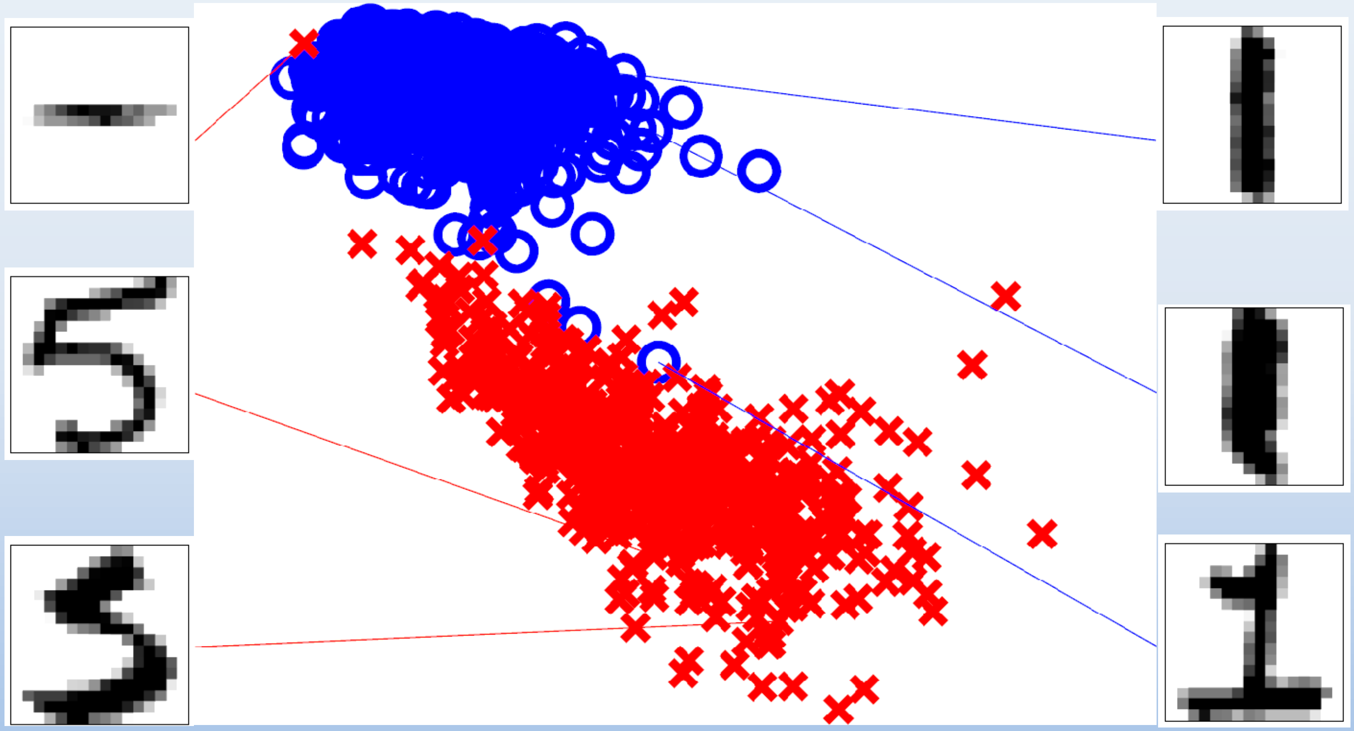


Feature

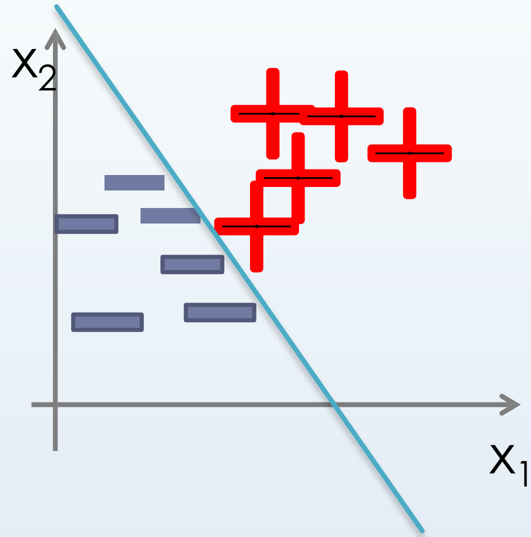
- Individual measurable property of a observable phenomenon
- Good feature: informative, discriminating and independent
- Examples:
  - character recognition:
    - histograms counting the number of black pixels along horizontal and vertical directions, number of internal holes, stroke detection
  - speech recognition,
    - noise ratios, length of sounds, relative power, filter matches
  - spam detection
    - presence or absence of certain email headers, the email structure, the language, the frequency of specific terms, the grammatical correctness of the text

Example: Illustration of features

- $X = (x_0, x_1, x_2)$ x_1: intensity x_2: symmetry



Perceptron classifier



- Input $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_d]$
- Classifier:

  If $\sum_{i=1}^{d} w_i x_i$ > threshold then output **1**

  else **-1**
- The linear formula $g \in \mathcal{H}$ can be written:
  - $G(x) = \text{sign}\left(\sum_{i=1}^{d} w_i x_i - \text{threshold}\right)$
  - $G(x) = \text{sign}\left(\sum_{i=1}^{d} w_i x_i - w_0\right)$
  - If we add a coordinate $x_0 = 1$ to the input
  - $G(x) = \text{sign}\left(\sum_{i=0}^{d} w_i x_i\right)$

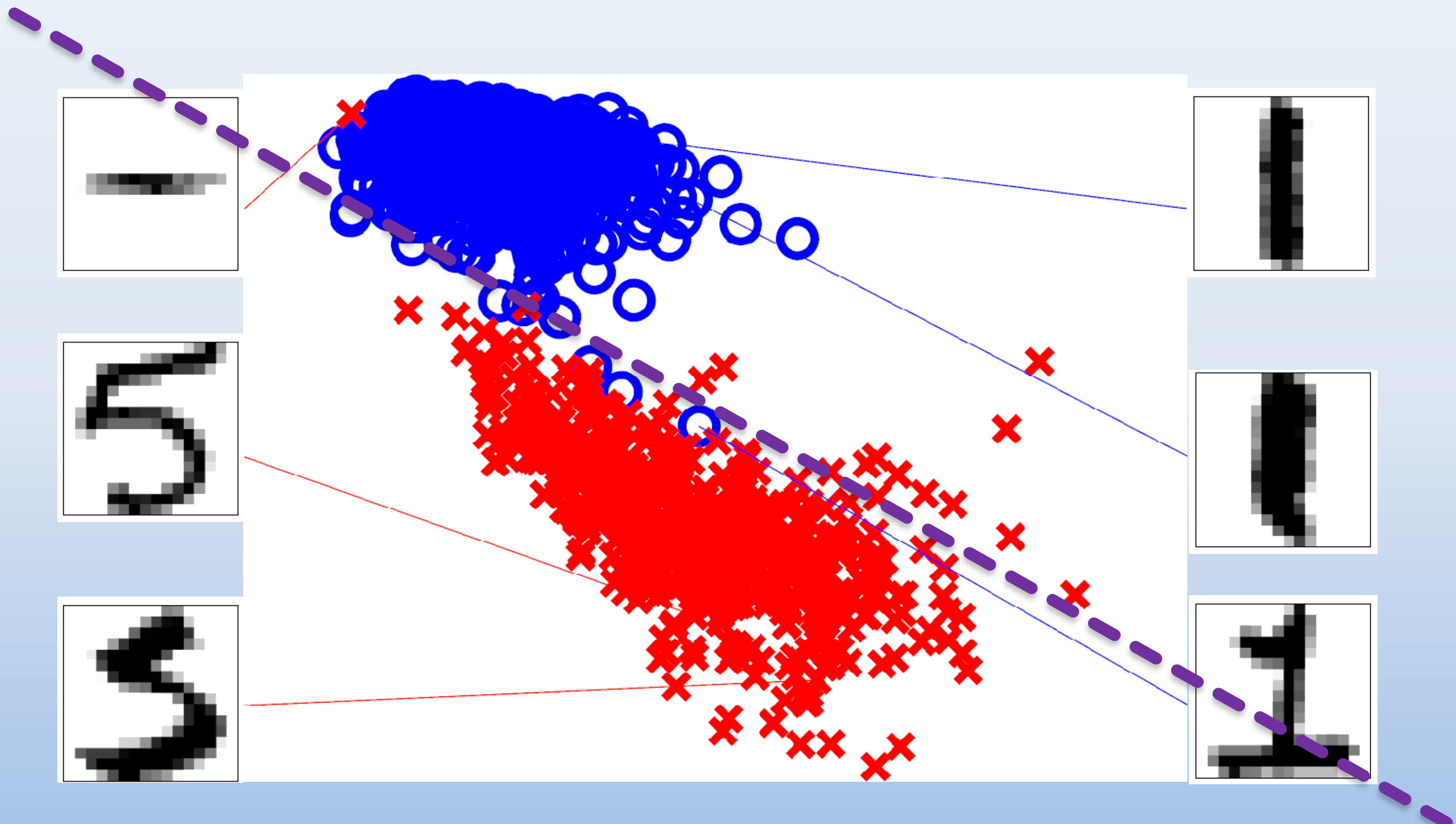
$\Rightarrow$ $g(x)=\text{sign}(W^TX)$

Perceptron learning algorithm: linearly separable data

- Give the training data $(x_1, y_1), \ldots, (x_n, y_n)$
- Misclassified data (x_i, y_i): sign $(W^T X_n) \neq y_n$
  - Repeat
    - Pick a misclassified data (x_i, y_i) from training data and update w:
      - $w = w + y_n x_n$
  - Until all training data points are correctly classified by g

Example: Illustration of features

- $X = (x_0, x_1, x_2)$ x_1: intensity x_2: symmetry



Generalization

- We don’t intend to memorize data but want to distinguish the pattern.

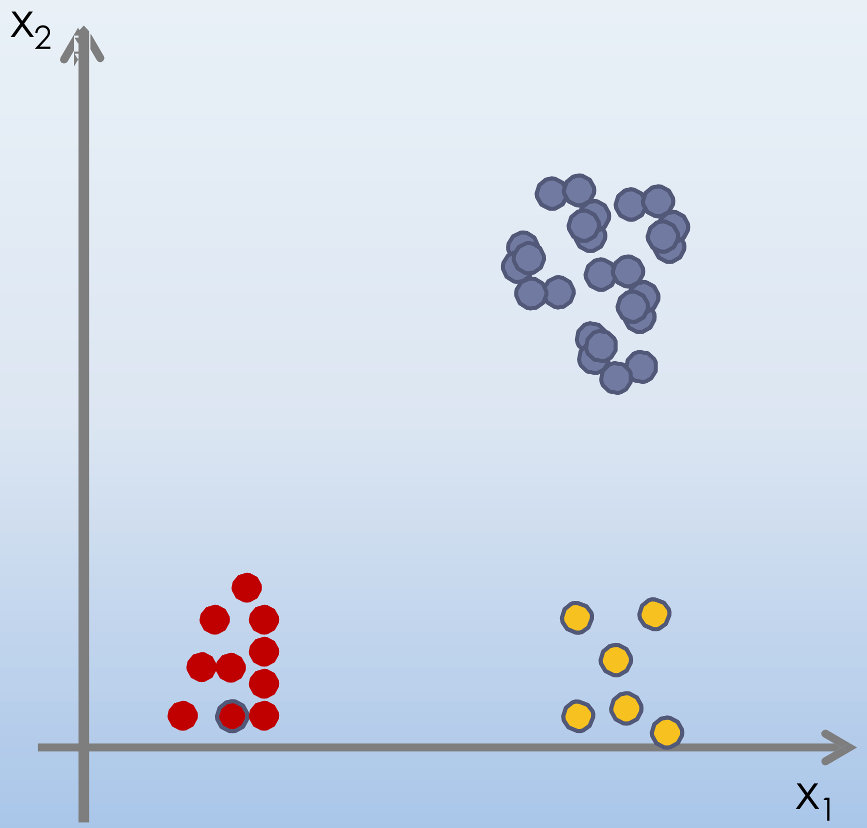
- A core objective of learning is to generalize from the experience.
  - Generalization: ability of a learning algorithm to perform accurately on new, unseen examples after having experienced.

Supervised Learning vs. Unsupervised Learning

- Supervised learning
  - Given: Training set
    - labeled set of n input-output pairs $D = \{(x_1, y_1), \ldots (x_n, y_n)\}$
- Goal: learning a mapping from X to Y
- Unsupervised learning
  - Given: Training set $D = \{(x_1), \ldots (x_n)\}$
- Goal: find groups or structures in the data
  - Discover the intrinsic structure in the data

Supervised Learning: Samples

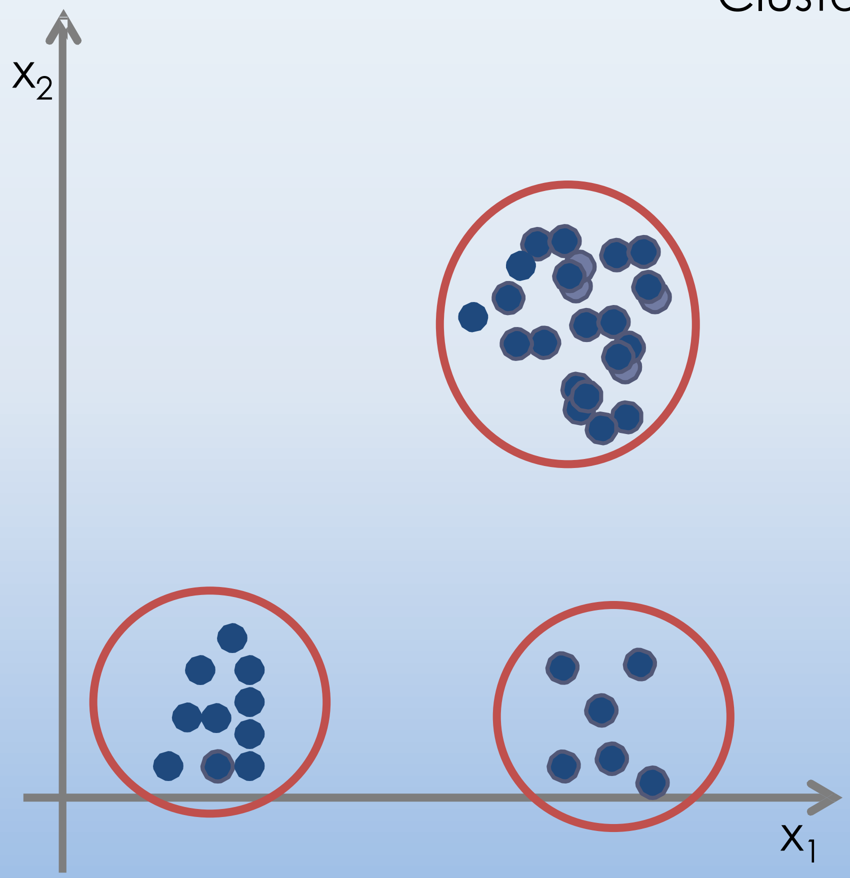
Classification



Unsupervised Learning: Samples

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

Clustering



Data in Unsupervised Learning

- Data are also usually considered as vectors in a d dimensional space

- Columns:
  - Features
  - Attributes
  - Dimensions
- Rows:
  - Data
  - Points
  - Instances
  - Examples
  - Samples
  - Records
- ~~Y column:~~
  - ~~Target~~
  - ~~Outcome~~
  - ~~Response~~
  - ~~Label~~

$$\bar{X} = \begin{bmatrix} X_1 \\ X_2 \\ . \\ . \\ Xd \end{bmatrix} = [X_1, X_2, \dots\ X_d]^T$$

	X_1	X_2	…	X_d
Sample 1				
Sample 2				
…				
Sample n-1				
Sample n				

$$X = [x_1, \dots, x_d]$$

Unsupervised learning

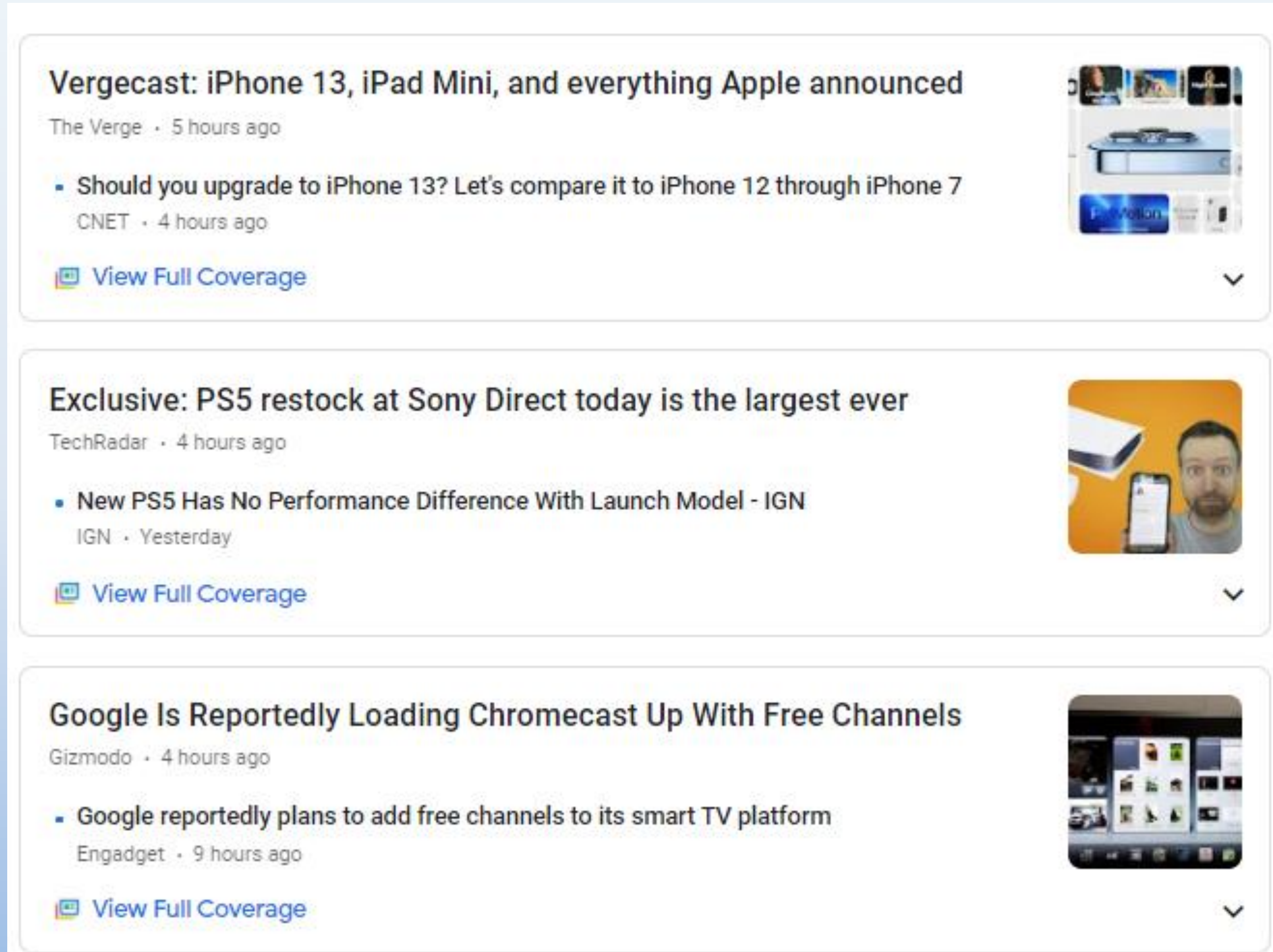
- **Clustering**: partitioning of data into groups of similar data points.
- **Dimensionality reduction/embedding**: data representation using a smaller number of dimensions while preserving (perhaps approximately) some properties of the data.
- Density estimation

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
- 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.
  - Community detection in social networks

Clustering of docs

- Google news



Vergecast: iPhone 13, iPad Mini, and everything Apple announced
The Verge · 5 hours ago
- Should you upgrade to iPhone 13? Let's compare it to iPhone 12 through iPhone 7
  CNET · 4 hours ago

View Full Coverage

Exclusive: PS5 restock at Sony Direct today is the largest ever
TechRadar · 4 hours ago
- New PS5 Has No Performance Difference With Launch Model - IGN
  IGN · Yesterday

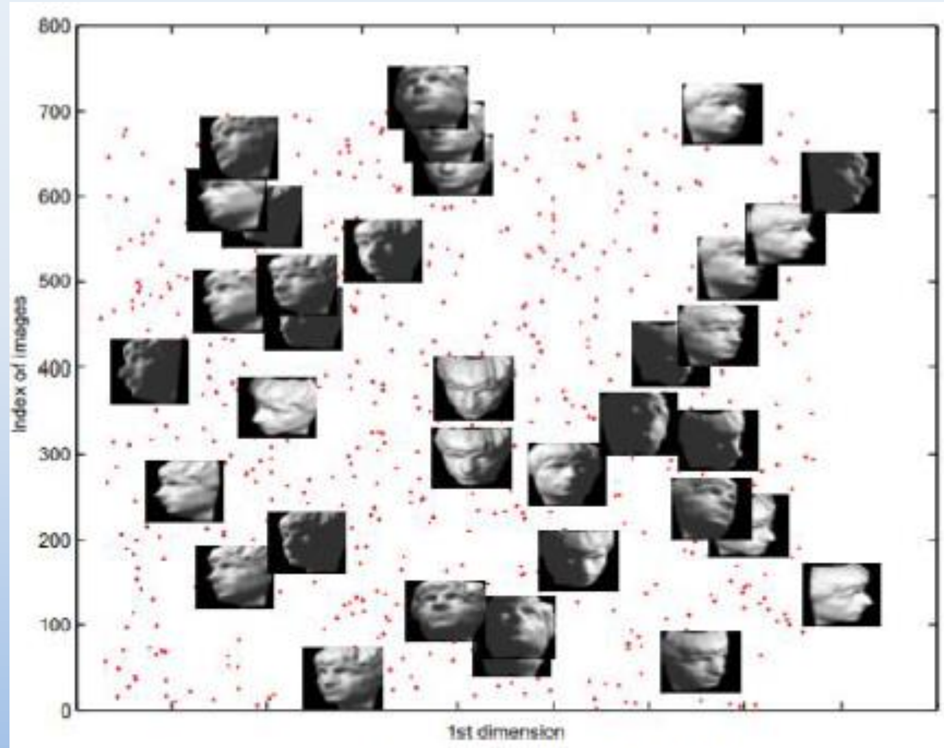
View Full Coverage

Google Is Reportedly Loading Chromecast Up With Free Channels
Gizmodo · 4 hours ago
- Google reportedly plans to add free channels to its smart TV platform
  Engadget · 9 hours ago

View Full Coverage

Dimensionality reduction: Example

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



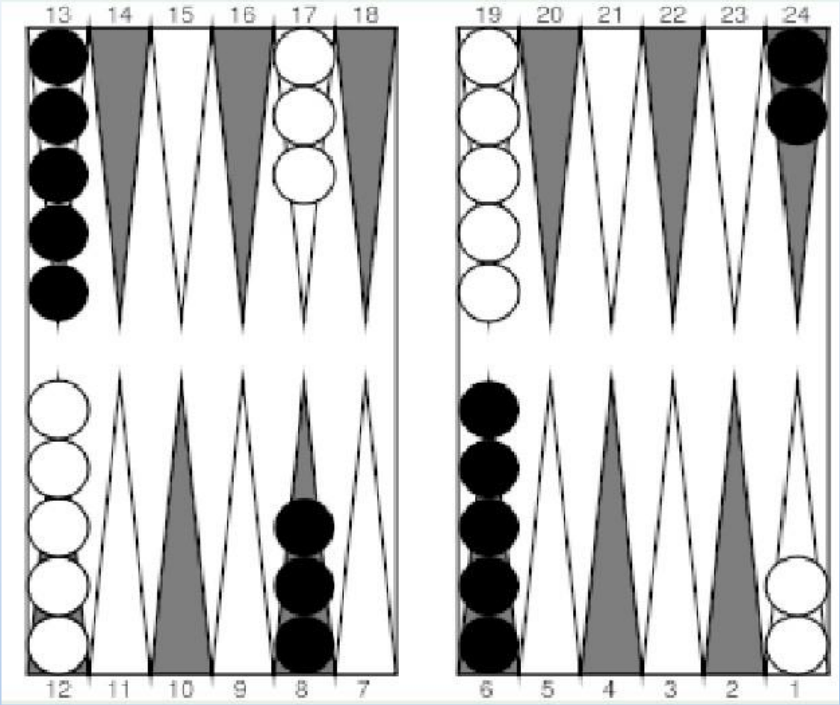
[Ali Ghodsi, 2006]

Reinforcement learning

- Provides only an indication as to whether an action is correct or not (feedback)
- Data in supervised learning:
  - (input, correct output)
- Data in Reinforcement Learning:
  - (input, a reward/penalty 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

Three dimensions 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
  - Non-parametric models

Parametric models

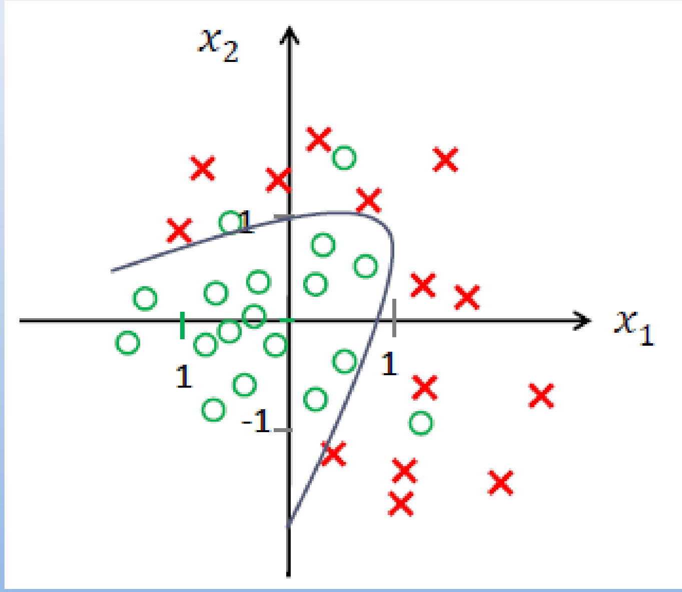
- We consider a parametric boundary (e.g., hyper-plane, hyperbola, …) and learn its parameters form data based on simplify or known form of model
  - The set of parameters does not grow with increasing the data

Benefits

- Simpler
- Speed
- Less Data

Limitations

- Constrained
- Limited Complexity
- Poor Fit



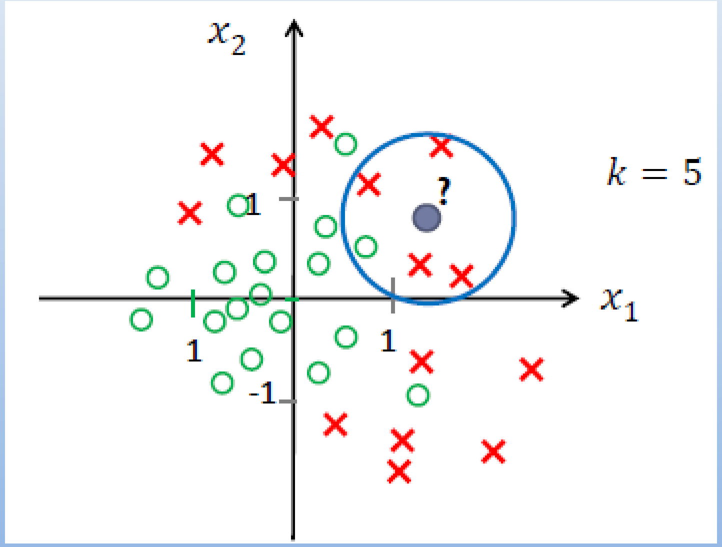
Non-parametric models

- 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

Benefits
- Flexibility
- Power
- Performance

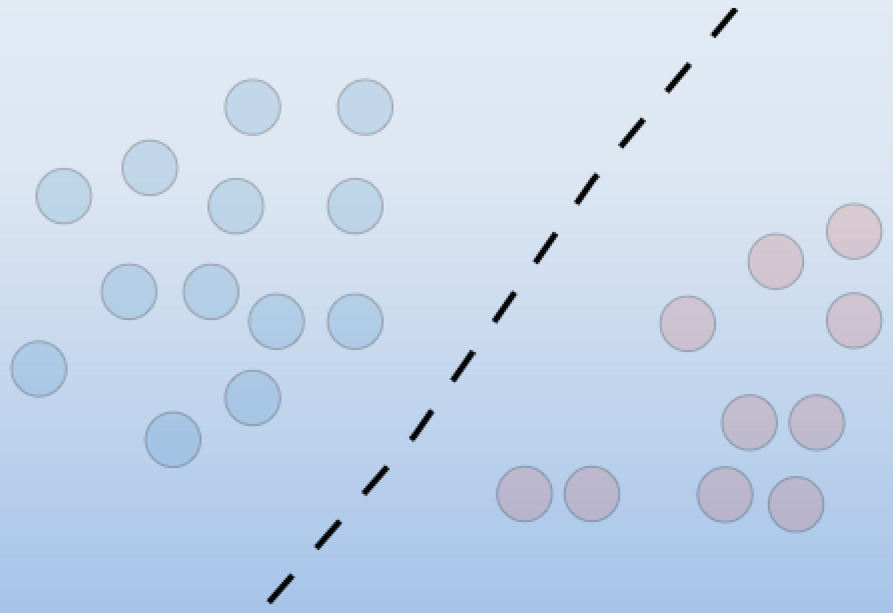
Limitations
- More data
- Slower
- Overfitting

- k-NN classifier
  - Label for 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



Discriminative model

- Goal
  - Directly estimate P(y | x)$P(y|x)$
  - Focuses on predicting the labels of the data
- What's learned
  - Decision boundary
- Examples
  - SVMs



Generative model

- Goal
  - Estimate $P(x \mid y)$P(x|y) to then deduce $P(y \mid x)$P(y|x)
  - Focuses on explaining how the data was generated
- What's learned
  - Probability distributions of the data
- Examples
  - Naive Bayes

Reading

- C. M. Bishop, **Pattern recognition and machine learning**, Springer, 2006. (ch. 1)
- E. Alpaydin, **Introduction to Machine Learning**, 4^{th} ed., The MIT Press, 2020. (ch. 1)
- Y. S. Abu-Mostafa, M. Magdon-Ismail, HT. Lin, **Learning From Data A short course**, 2012 (ch. 1)
- R. O. Duda, P. E. Hart. D. G. Stork, **Pattern classification**, 2^{nd} ed., John Wiley & Sons, 2006. (ch. 1)