

# CS 2750 Machine Learning

## Lecture 1

# Machine Learning

**Milos Hauskrecht**

[milos@cs.pitt.edu](mailto:milos@cs.pitt.edu)

5329 Sennott Square, x4-8845

<http://www.cs.pitt.edu/~milos/courses/cs2750/>

---

CS 2750 Machine Learning

## Administration

**Instructor:**

**Milos Hauskrecht**

[milos@cs.pitt.edu](mailto:milos@cs.pitt.edu)

5329 Sennott Square, x4-8845

**TA:**

**Michael Moeng**

[moeng@cs.pitt.edu](mailto:moeng@cs.pitt.edu)

5802 Sennott Square

**Office hours:** Tuesday 11-12am and 2-4pm

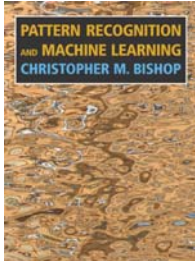
---

CS 2750 Machine Learning

## Administration

### Study material

- Handouts, your notes and course readings
- Primary textbook:



- Chris. Bishop. Pattern Recognition and Machine Learning. Springer, 2006.

## Administration

### Study material

- Other books:
  - Friedman, Hastie, Tibshirani. Elements of statistical learning. Springer, 2001.
  - Duda, Hart, Stork. Pattern classification. 2<sup>nd</sup> edition. J Wiley and Sons, 2000.
  - C. Bishop. Neural networks for pattern recognition. Oxford U. Press, 1996.
  - T. Mitchell. Machine Learning. McGraw Hill, 1997
  - J. Han, M. Kamber. Data Mining. Morgan Kauffman, 2001.

## Administration

- **Homeworks: weekly**
  - **Programming tool:** Matlab (CSSD machines and labs)
  - **Matlab Tutorial:** next week
- **Exams:**
  - **Midterm** (March)
  - **Final** (April)
- **Final project:**
  - **Proposals** (March)
  - **Written report + Oral presentation**  
(end of the semester)
- **Lectures:**
  - **Activity and random** short quizzes testing the understanding of basic concepts from previous lectures

---

CS 2750 Machine Learning

## Tentative topics

- Learning.
- Density estimation.
- Linear models for regression and classification.
- Multi-layer neural networks.
- Support vector machines. Kernel methods.
- Learning Bayesian networks.
- Clustering. Latent variable models.
- Dimensionality reduction. Feature extraction.
- Ensemble methods. Mixture models. Bagging and boosting.
- Hidden Markov models.
- Reinforcement learning

---

CS 2750 Machine Learning

## Machine Learning

- The field of **machine learning** studies the design of computer programs (agents) capable of learning from past experience or adapting to changes in the environment
- The need for building agents capable of learning is everywhere
  - predictions in medicine,
  - text and web page classification,
  - speech recognition,
  - image/text retrieval,
  - commercial software

## Learning

### Learning process:

Learner (a computer program) processes data  $D$  representing past experiences and tries to either develop an appropriate response to future data, or describe in some meaningful way the data seen

### Example:

Learner sees a set of patient cases (patient records) with corresponding diagnoses. It can either try:

- to predict the presence of a disease for future patients
- describe the dependencies between diseases, symptoms

## Types of learning

- **Supervised learning**
  - Learning mapping between input  $x$  and desired output  $y$
  - Teacher gives me  $y$ 's for the learning purposes
- **Unsupervised learning**
  - Learning relations between data components
  - No specific outputs given by a teacher
- **Reinforcement learning**
  - Learning mapping between input  $x$  and desired output  $y$
  - Critic does not give me  $y$ 's but instead a signal (reinforcement) of how good my answer was
- **Other types of learning:**
  - **Concept learning, explanation-based learning, etc.**

---

CS 2750 Machine Learning

## Supervised learning

**Data:**  $D = \{d_1, d_2, \dots, d_n\}$  a set of  $n$  examples

$$d_i = \langle \mathbf{x}_i, y_i \rangle$$

$\mathbf{x}_i$  is input vector, and  $y$  is desired output (given by a teacher)

**Objective:** learn the mapping  $f : X \rightarrow Y$

$$\text{s.t. } y_i \approx f(x_i) \text{ for all } i = 1, \dots, n$$

**Two types of problems:**

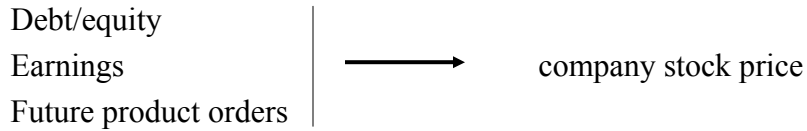
- **Regression:**  $X$  discrete or continuous  $\rightarrow$   
 $Y$  is **continuous**
- **Classification:**  $X$  discrete or continuous  $\rightarrow$   
 $Y$  is **discrete**

---

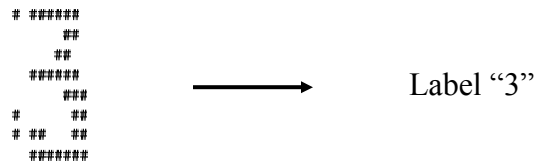
CS 2750 Machine Learning

## Supervised learning examples

- **Regression:** Y is **continuous**



- **Classification:** Y is **discrete**



Handwritten digit (array of 0,1s)

CS 2750 Machine Learning

## Unsupervised learning

- **Data:**  $D = \{d_1, d_2, \dots, d_n\}$   
 $d_i = \mathbf{x}_i$  vector of values  
No target value (output) y
- **Objective:**
  - learn relations between samples, components of samples

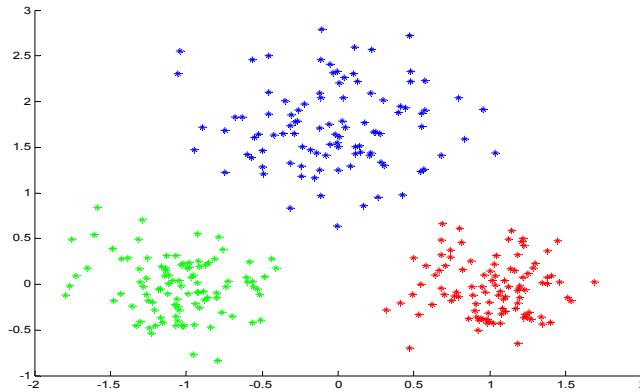
### Types of problems:

- **Clustering**
  - Group together “similar” examples, e.g. patient cases
- **Density estimation**
  - Model probabilistically the population of samples

CS 2750 Machine Learning

## Unsupervised learning example.

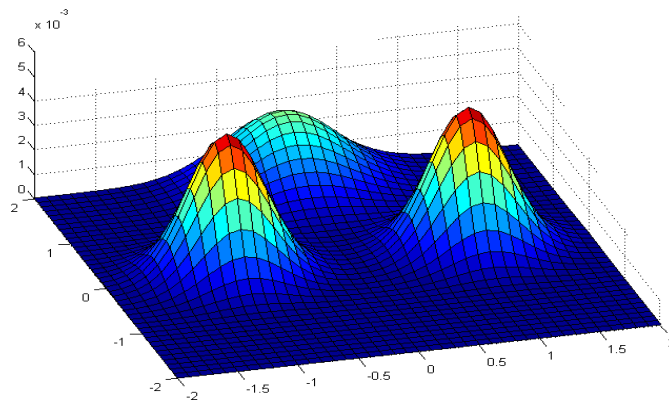
- **Density estimation.** We want to build the probability model of a population from which we draw samples  $d_i = \mathbf{x}_i$



CS 2750 Machine Learning

## Unsupervised learning. Density estimation

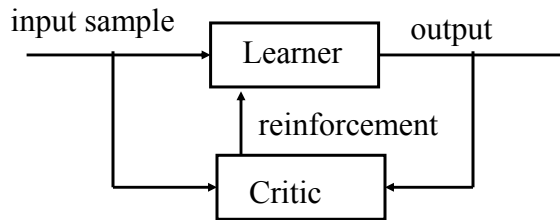
- A probability density of a point in the two dimensional space
  - Model used here: **Mixture of Gaussians**



CS 2750 Machine Learning

## Reinforcement learning

- We want to learn:  $f : X \rightarrow Y$
- We see samples of  $\mathbf{x}$  but not  $y$
- Instead of  $y$  we get a feedback (reinforcement) from a **critic** about how good our output was

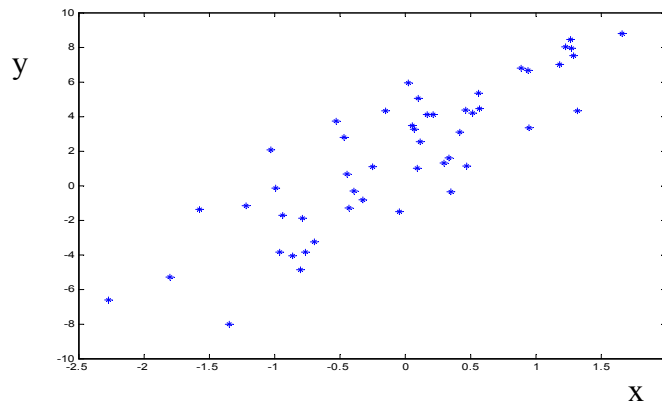


- The goal is to select outputs that lead to the best reinforcement

CS 2750 Machine Learning

## Learning

- Assume we see examples of pairs  $(\mathbf{x}, y)$  and we want to learn the mapping  $f : X \rightarrow Y$  to predict future  $y$ s for values of  $\mathbf{x}$
- We get the data what should we do?

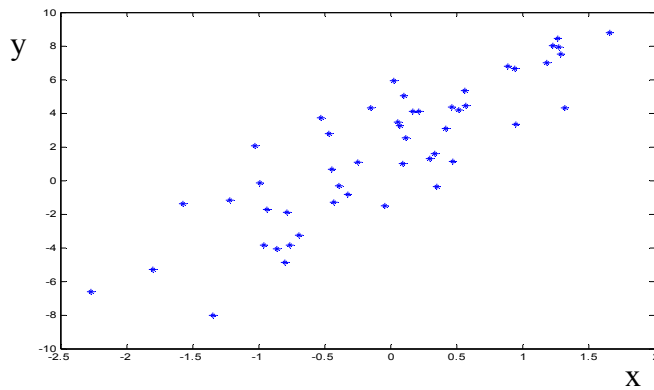


CS 2750 Machine Learning



## Learning bias

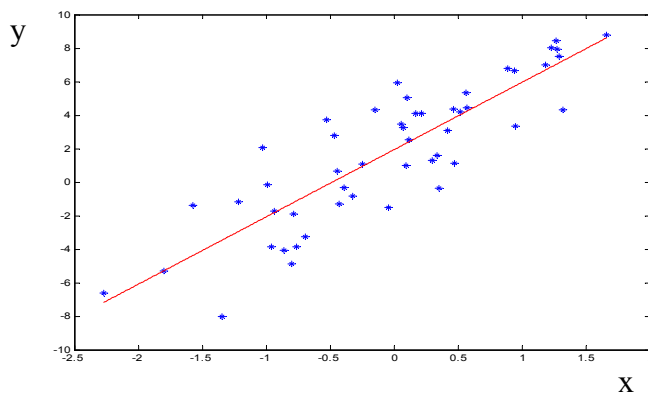
- **Problem:** many possible functions  $f : X \rightarrow Y$  exists for representing the mapping between  $x$  and  $y$
- Which one to choose? Many examples still unseen!



CS 2750 Machine Learning

## Learning bias

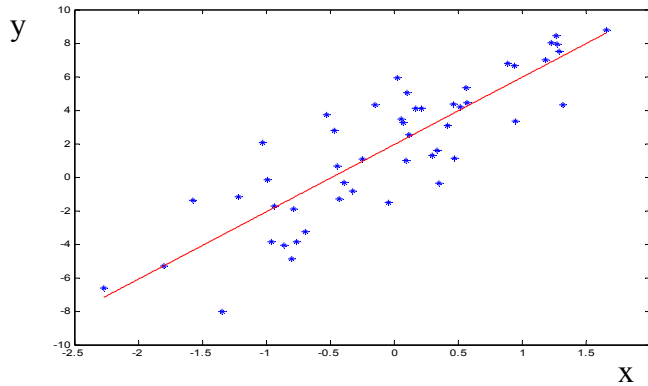
- Problem is easier when we make an assumption about the model, say,  $f(x) = ax + b + \varepsilon$   
 $\varepsilon = N(0, \sigma)$  - random (normally distributed) noise
- Restriction to a linear model is an example of learning bias



CS 2750 Machine Learning

## Learning bias

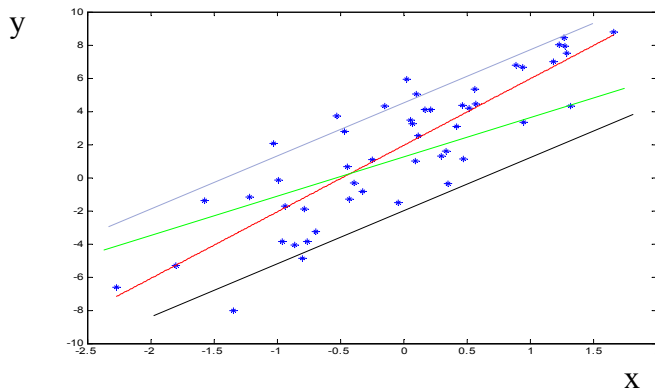
- **Bias** provides the learner with some basis for choosing among possible representations of the function.
- **Forms of bias:** constraints, restrictions, model preferences
- **Important:** There is no learning without a bias!



CS 2750 Machine Learning

## Learning bias

- Choosing a parametric model or a set of models is not enough  
Still too many functions  $f(x) = ax + b + \varepsilon$   $\varepsilon = N(0, \sigma)$ 
  - One for every pair of parameters  $a, b$



CS 2750 Machine Learning

## Fitting the data to the model

- We are interested in finding the **best set** of model parameters

**Objective:** Find the set of parameters that:

- reduces the misfit between the model and observed data
- Or, (in other words) that explain the data the best

**Error function:**

**Measure of misfit between the data and the model**

- **Examples of error functions:**

– Average square error  $\frac{1}{n} \sum_{i=1}^n (y_i - f(x_i))^2$

– Average misclassification error  $\frac{1}{n} \sum_{i=1}^n 1_{y_i \neq f(x_i)}$

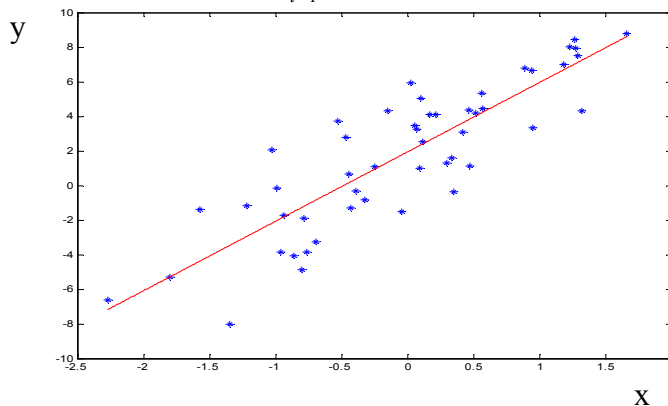
Average # of misclassified cases

## Fitting the data to the model

- **Linear regression**

- Least squares fit with the linear model

- minimizes  $\frac{1}{n} \sum_{i=1}^n (y_i - f(x_i))^2$



## Typical learning

### Three basic steps:

- **Select a model** or a set of models (with parameters)

E.g.  $y = ax + b$

- **Select the error function** to be optimized

E.g.  $\frac{1}{n} \sum_{i=1}^n (y_i - f(x_i))^2$

- **Find the set of parameters optimizing the error function**

– The model and parameters with the smallest error represent the best fit of the model to the data

But there are problems one must be careful about ...