

# DAYANANDA SAGAR UNIVERSITY

## School of Engineering

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

# Machine Learning Module 1

Presentation Material			
Department of Computer Science & Engineering			
Course Code:		Semester:	V
Course Title:	Machine Learning	Year:	2025-2026
Faculty Name:	Dr. Damodharan D		

# Machine Learning

## Syllabus

---

**UNIT - I****08 Hours**

---

**INTRODUCTION**

Introduction to Machine Learning, Types of ML, Applications of ML.

*(Text Book-1: Chapter 1: 1.1 to 1.2)*

**MATHEMATICS FOR MACHINE LEARNING**

Bayes' Theorem, Gaussian Distribution, Data, Models and Learning, Empirical Risk Minimization, Parameter Estimation, Probabilistic Modeling and Inference.

*(Reference Book-1: Chapter 6: 6.3, 6.5, Chapter 8: 8.1 to 8.4)*

---

# Introduction

- Introduction to Machine Learning
- Type of Machine Learning
- Application of ML

# 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 “Da Vinci Code” also bought “The Five People You Meet in Heaven” (www.amazon.com)*

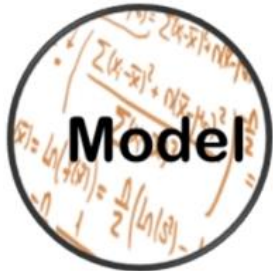
- Build a model that is *a good and useful approximation* to the data.

# What is Machine Learning?

*...creating and using models that learn from data...*



- **data:** anything you can *measure* or *record*



- **model:** specification of a (mathematical) *relationship* between different variables



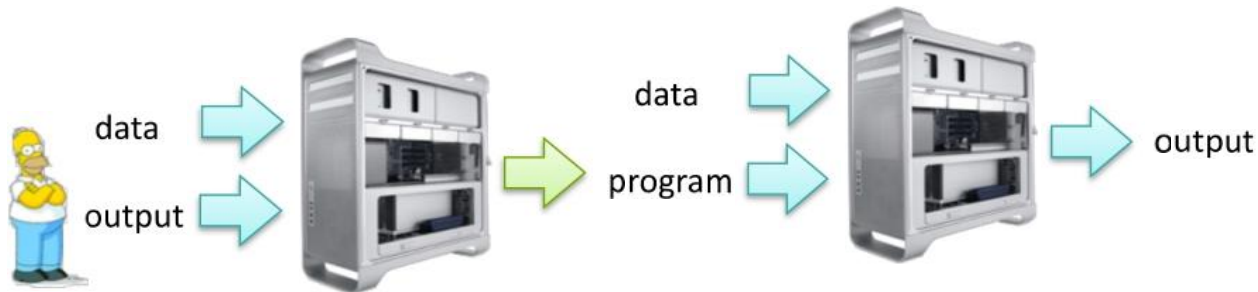
- **evaluation:** how well does the model *work?*

# What is Machine Learning?

- Traditional CS



- Machine Learning



# What is Machine Learning?

- Optimize a performance criterion using example data or past experience.
- Role of Statistics: Inference from a sample
- Role of Computer science: Efficient algorithms to
  - Solve the optimization problem
  - Representing and evaluating the model for inference



# Machine Learning Application

*...creating and using models that learn from data...*

## Examples

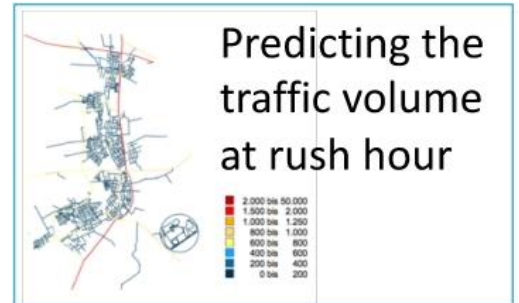
Identifying zip code  
from handwritten  
digits



Detecting  
communities  
in social  
networks



Predicting the  
traffic volume  
at rush hour



Detecting fraudulent  
credit card  
transactions



Determining the  
location of distribution  
centers based on  
customers' residence



# Type of Machine Learning

- Association
- Supervised Learning
  - Classification
  - Regression
- Unsupervised Learning
- Reinforcement Learning

# Learning Associations

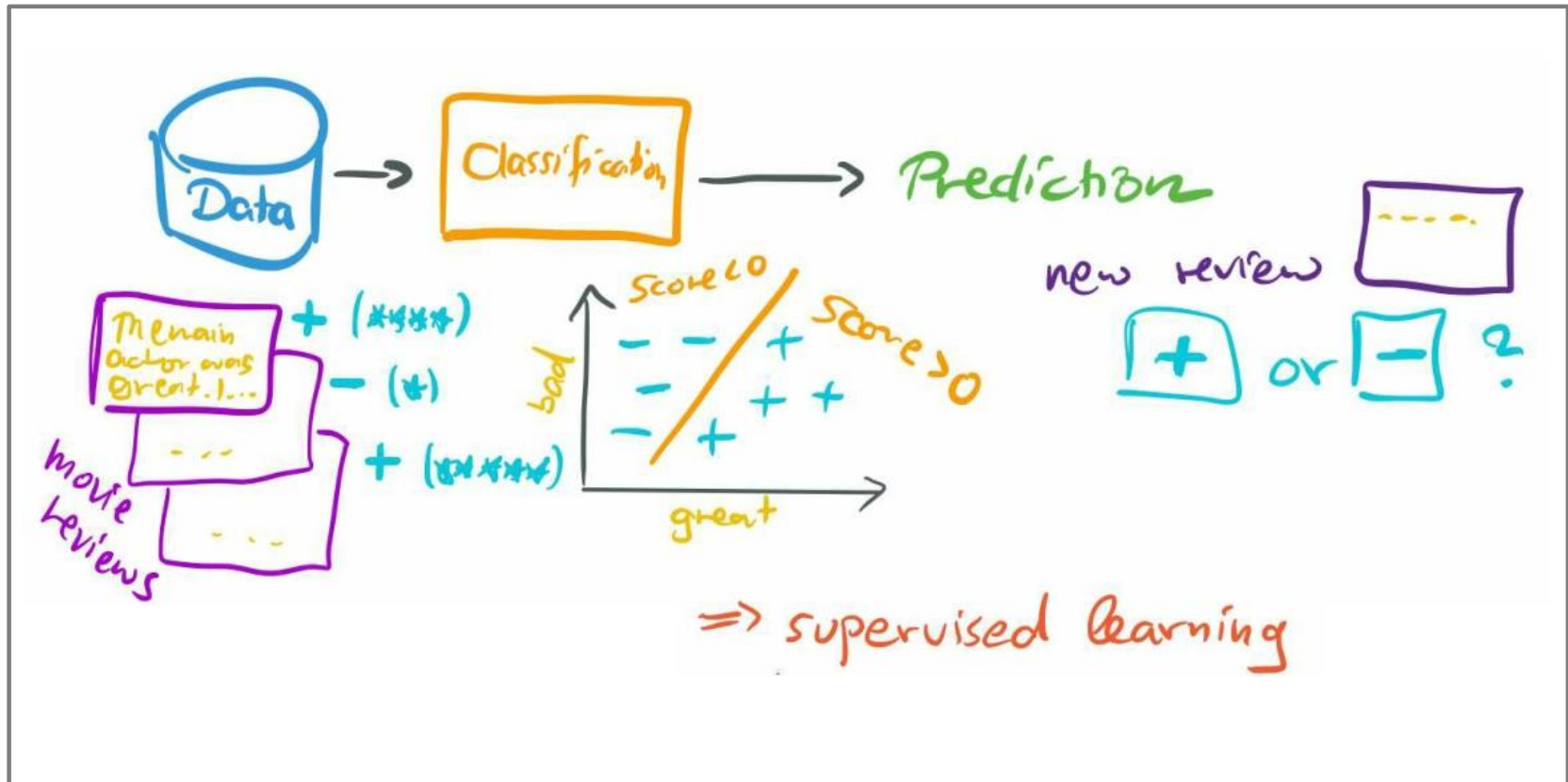
- Basket analysis:

$P(Y | X)$  probability that somebody who buys  $X$  also buys  $Y$  where  $X$  and  $Y$  are products/services.

Example:  $P(\text{chips} | \text{beer}) = 0.7$

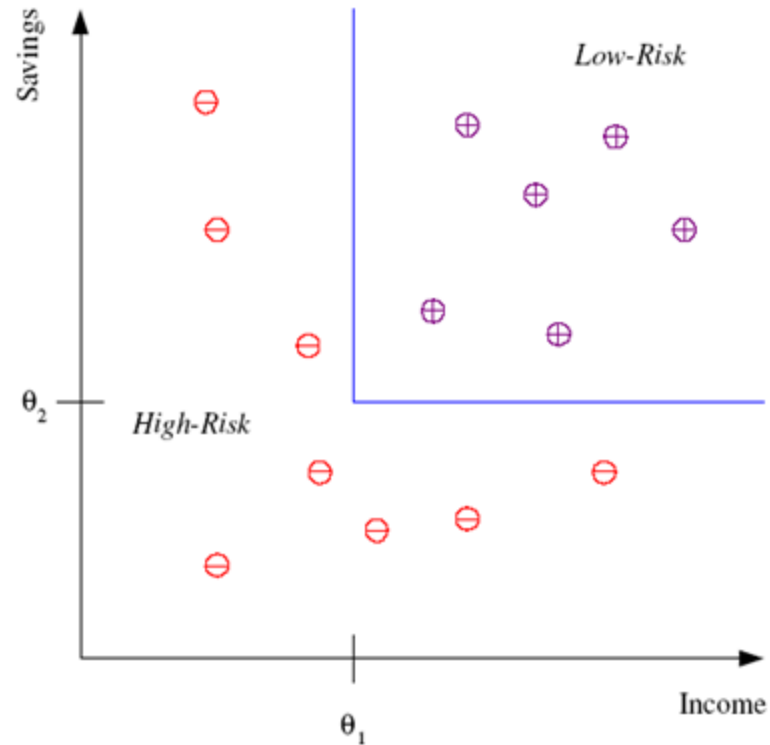
# Classification

Classification in machine learning is a predictive modeling process by which machine learning models use classification algorithms to predict the correct label for input data.



# Classification

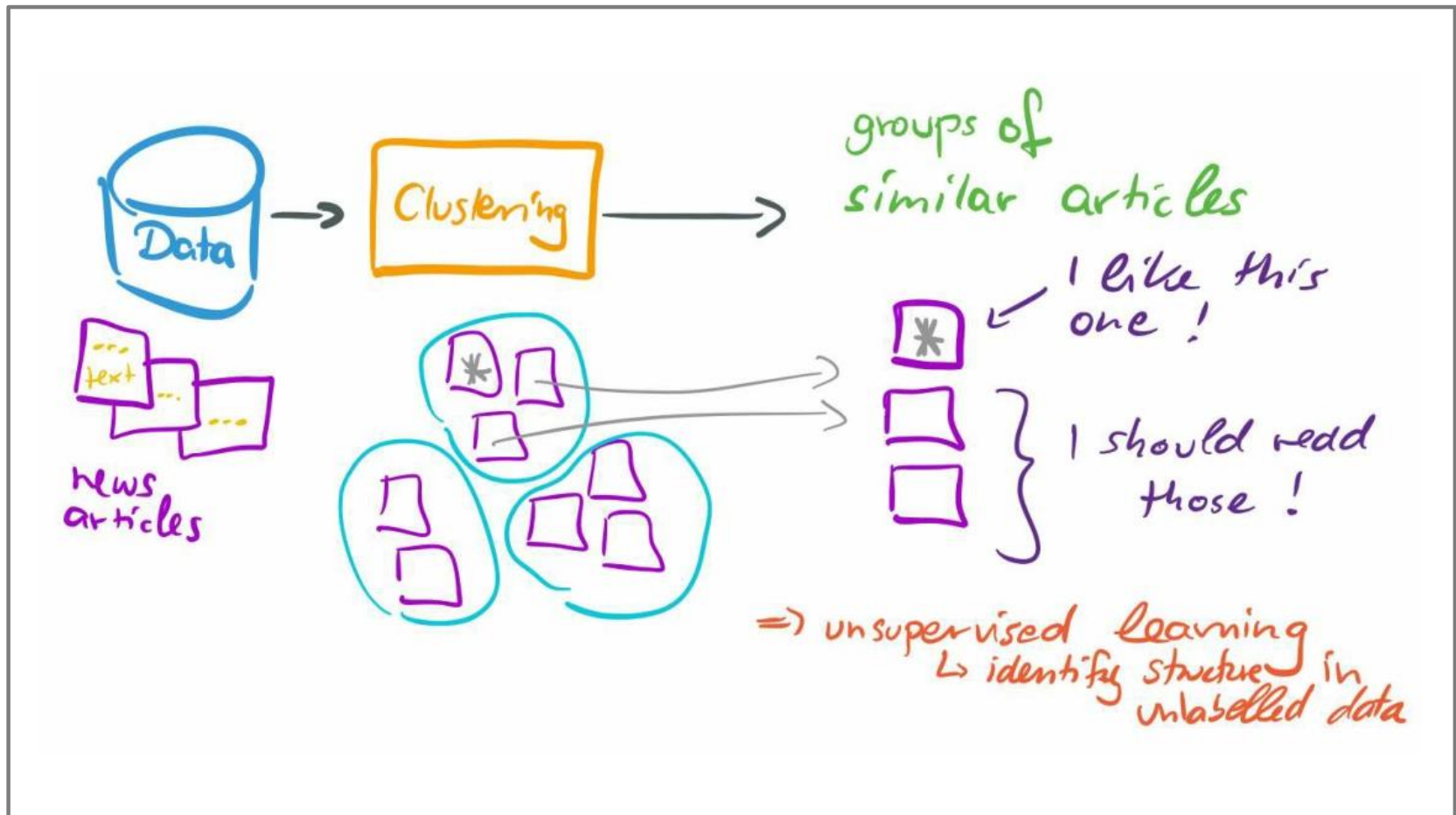
- Example: Credit scoring
- Differentiating between **low-risk** and **high-risk** customers from their *income* and *savings*



**Discriminant:** IF  $income > \theta_1$  AND  $savings > \theta_2$   
THEN **low-risk** ELSE **high-risk**

# LEARNING FROM DATA

- Clustering



# Face Recognition

Training examples of a person



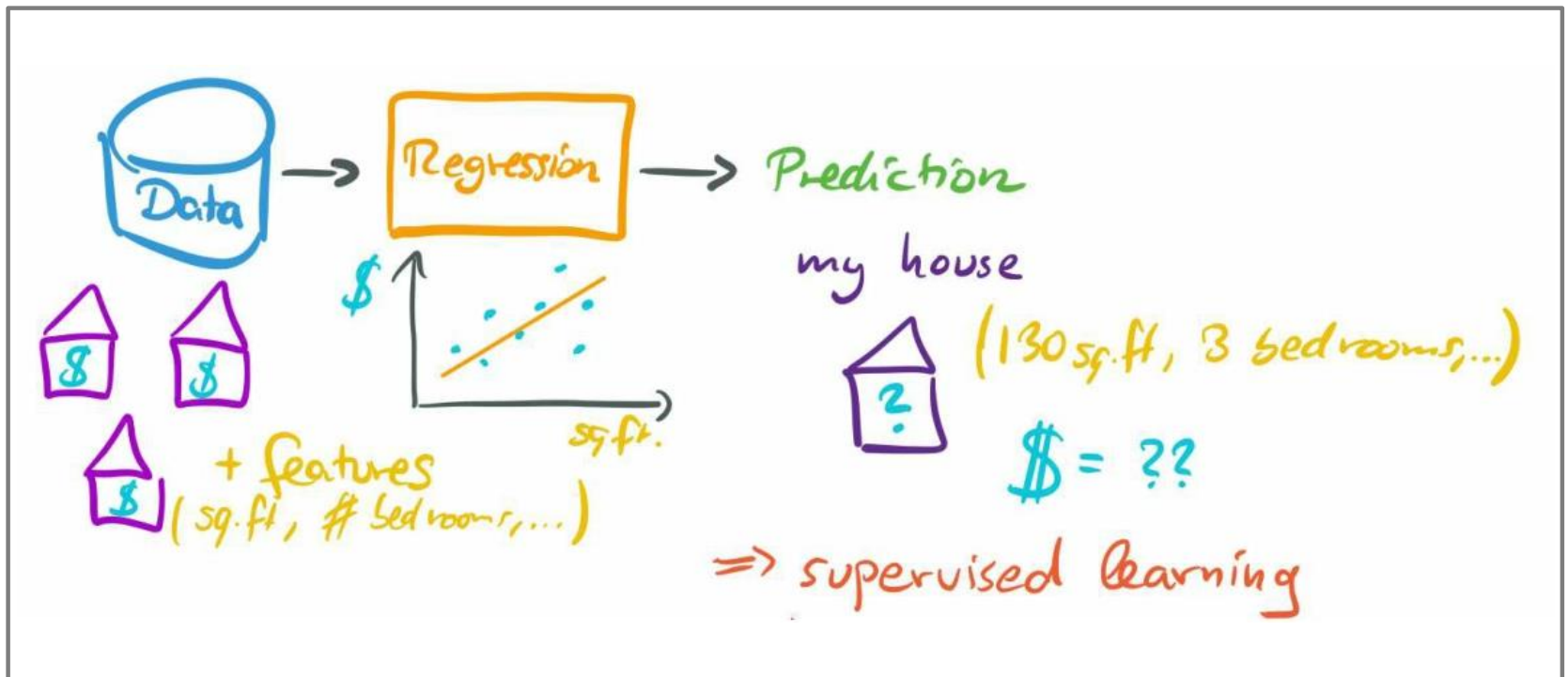
Test images



AT&T Laboratories, Cambridge UK  
<http://www.uk.research.att.com/facedatabase.html>

# Regression

Regression in machine learning is a technique used to capture the relationships between independent and dependent variables, with the main purpose of predicting an outcome.





# Regression

- Example: Price of a used car

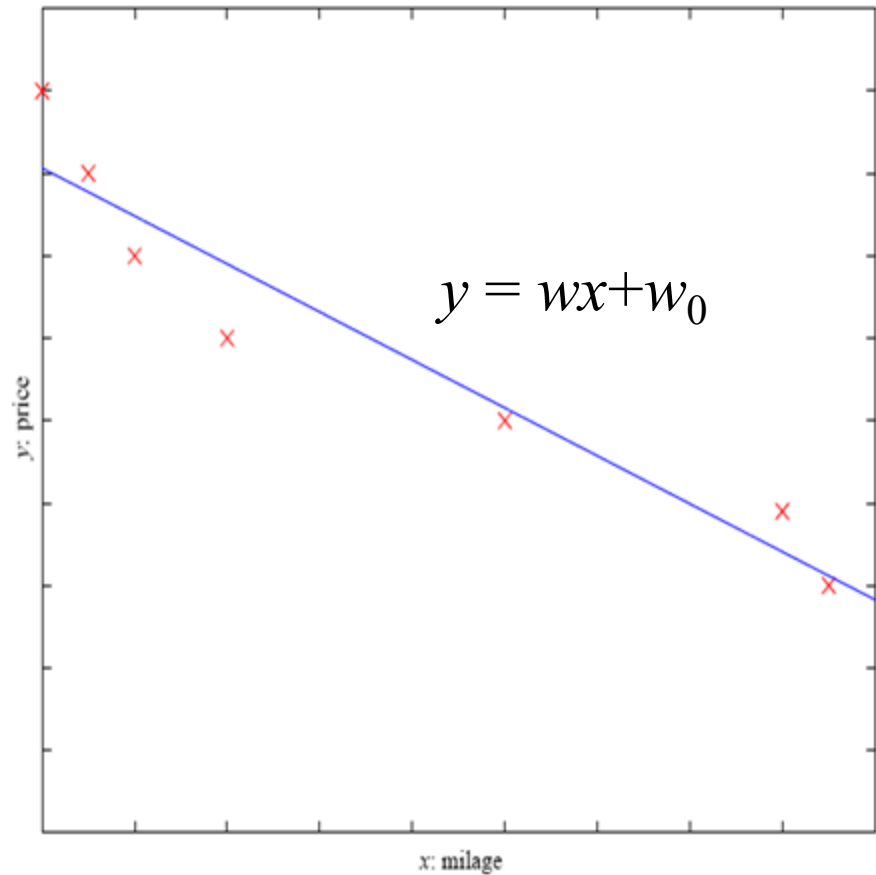
- $x$  : car attributes

$y$  : price

$$y = g(x | \theta)$$

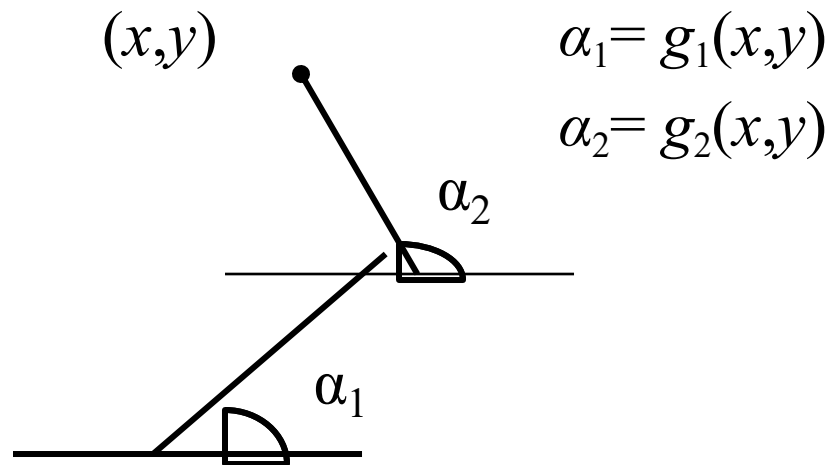
$g(\cdot)$  model,

$\theta$  parameters

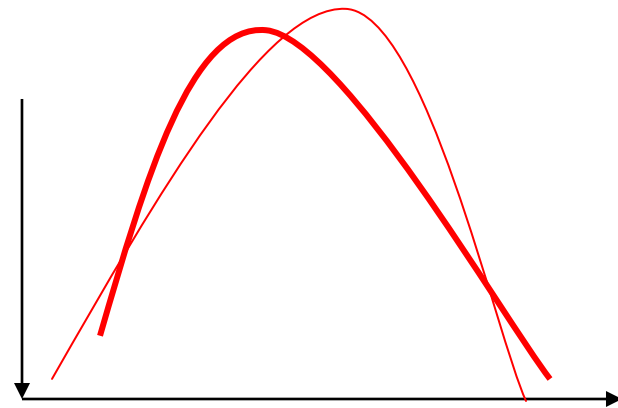


# Regression Applications

- Navigating a car: Angle of the steering wheel (CMU NavLab)
- Kinematics of a robot arm



■ Response surface design



# Supervised Learning: Uses

- **Prediction of future cases:** Use the rule to predict the output for future inputs
- **Knowledge extraction:** The rule is easy to understand
- **Compression:** The rule is simpler than the data it explains
- **Outlier detection:** Exceptions that are not covered by the rule, e.g., fraud

# Unsupervised Learning

- Learning “what normally happens”
- No output
- Clustering: Grouping similar instances
- Example applications
  - Customer segmentation in CRM
  - Image compression: Color quantization
  - Bioinformatics: Learning motifs

# Reinforcement Learning

- Learning a policy: A **sequence** of outputs
- No supervised output but delayed reward
- Credit assignment problem
- Game playing
- Robot in a maze
- Multiple agents, partial observability, ...

# Applications

- Aka Pattern recognition
- **Face recognition:** Pose, lighting, occlusion (glasses, beard), make-up, hair style
- **Character recognition:** Different handwriting styles.
- **Speech recognition:** Temporal dependency.
  - Use of a dictionary or the syntax of the language.
  - Sensor fusion: Combine multiple modalities; eg, visual (lip image) and acoustic for speech
- **Medical diagnosis:** From symptoms to illnesses

# MATHEMATICS FOR MACHINE LEARNING

- Bayes' Theorem,
- Gaussian Distribution,
- Data, Models and Learning,
- Empirical Risk Minimization,
- Parameter Estimation,
- Probabilistic Modeling and Inference.

# Bayes' Theorem

- **Bayes' Theorem** is a mathematical formula used to determine the **conditional probability** of an event based on prior knowledge and new evidence.
- It adjusts probabilities when new information comes in and helps make better decisions in uncertain situations.



## Guessing the Pet

Is it a cat or a dog in the box?


$P(\text{Dog}) = 0.5$



$P(\text{Cat}) = 0.5$



**Initial Belief:** 50/50 chances

**We have a CLUE** 

The Pet is  
very Quiet



→  $(P(\text{Quiet} \mid \text{Cat})) = 80\% \text{ or } 0.8$

→  $(P(\text{Quiet} \mid \text{Dog})) = 30\% \text{ or } 0.3$

## Applying Bayes Theorem to Find Probability



$$P(\text{Cat} \mid \text{Quiet}) = \frac{P(\text{Quiet Cat}) \times P(\text{Cat})}{P(\text{Quiet})} = 72.7\%$$



$$P(\text{Dog} \mid \text{Quiet}) = \frac{P(\text{Quiet Dog}) \times P(\text{Dog})}{P(\text{Quiet})} = 27.3\%$$

## Bayes Theorem Formula

For any two events A and B, **Bayes's** formula for the Bayes theorem is given by:

Formula for the Bayes theorem

$$P(A|B) = \frac{P(B|A) P(A)}{P(B)}$$

Where,

- **P(A)** and **P(B)** are the probabilities of events A and B; also, P(B) is never equal to zero.
- **P(A|B)** is the probability of event A when event B happens,
- **P(B|A)** is the probability of event B when A happens.

# Bayes' Theorem

- $P(A|B)$ : Conditional probability of event A occurring, given the event B

$P(A)$ : Probability of event A occurring

$P(B)$ : Probability of event B occurring

$P(B|A)$ : Conditional probability of event B occurring, given the event A

Formally, the terminologies of the Bayesian Theorem are as follows:

- A is known as the proposition and B is the evidence

$P(A)$  represents the prior probability of the proposition

$P(B)$  represents the prior probability of evidence

$P(A|B)$  is called the posterior

$P(B|A)$  is the likelihood

Therefore, the Bayes theorem can be summed up as:

- ***Posterior = (Likelihood) . (Proposition prior probability) / Evidence prior probability***

## Examples of Bayes' Theorem

**Example 1:** A person has undertaken a job. The probabilities of completion of the job on time with and without rain are 0.44 and 0.9, and 5, respectively. If the probability that it will rain is 0.45, then determine the probability that the job will be completed on time.

### Solution:

Let  $E_1$  be the event that the mining job will be completed on time and  $E_2$  be the event that it rains. We have,

$$P(A) = 0.45,$$

$$P(\text{no rain}) = P(B) = 1 - P(A) = 1 - 0.45 = 0.55$$

By multiplication law of probability,

$$P(E_1) = 0.44, \text{ and } P(E_2) = 0.95$$

Since, events A and B form partitions of the sample space S, by total probability theorem, we have

$$P(E) = P(A) P(E_1) + P(B) P(E_2)$$

$$\Rightarrow P(E) = 0.45 \times 0.44 + 0.55 \times 0.95$$

$$\Rightarrow P(E) = 0.198 + 0.5225 = 0.7205$$

So, the probability that the job will be completed on time is 0.7205

**Example 2:** There are three urns containing 3 white and 2 black balls, 2 white and 3 black balls, and 1 black and 4 white balls, respectively. There is an equal probability of each urn being chosen. One ball is equal probability chosen at random. What is the probability that a white ball will be drawn?

Solution:

Let  $E_1$ ,  $E_2$ , and  $E_3$  be the events of choosing the first, second, and third urn respectively. Then,

$$P(E_1) = P(E_2) = P(E_3) = 1/3$$

Let  $E$  be the event that a white ball is drawn. Then,  
 $P(E/E_1) = 3/5$ ,  $P(E/E_2) = 2/5$ ,  $P(E/E_3) = 4/5$

By theorem of total probability, we have

$$P(E) = P(E/E_1) \cdot P(E_1) + P(E/E_2) \cdot P(E_2) + P(E/E_3) \cdot P(E_3)$$

$$\Rightarrow P(E) = (3/5 \times 1/3) + (2/5 \times 1/3) + (4/5 \times 1/3)$$

$$\Rightarrow P(E) = 9/15 = 3/5$$

# Bayes Theorem Applications

Bayesian inference is very important and has found application in various activities, including medicine, science, philosophy, engineering, sports, law, etc., and Bayesian inference is directly derived from Bayes theorem.

## Some of the Key Applications are:

- **Medical Testing** → Finding the real probability of having a disease after a positive test.
- **Spam Filters** → Checking if an email is spam based on keywords.
- **Weather Prediction** → Updating the chance of rain based on new data.
- **AI & Machine Learning** → Used in Naïve Bayes classifiers to predict outcomes.



## Practice Problem Based on Bayes' Theorem

**Question 1:** A medical test for a disease is 95% accurate in detecting the disease (True Positive Rate). The probability of a person having the disease is 0.01 (1%). If a person tests positive for the disease, what is the probability that they actually have the disease? (Assume that the false positive rate is 5%).

**Question 2:** A bag contains 4 red balls and 6 blue balls. Two balls are drawn at random, and one of them is red. What is the probability that the second ball drawn is also red, given that the first ball was red?

**Question 3:** In a factory, 80% of the products are produced by Machine A and 20% by Machine B. Machine A produces 2% defective items, while Machine B produces 5% defective items. If a product is found to be defective, what is the probability that it was produced by Machine A?

**Question 4:** A survey shows that 70% of people like ice cream, and 40% of people like both ice cream and chocolate. What is the probability that a person likes chocolate, given that they like ice cream?

**Answer:-**

**1.16.1%.**

**2.33.33%.**

**3.61.5%.**

**4.57.1%.**

# Gaussian Distribution

Gaussian distribution arises naturally when we consider sums of independent and identically distributed random variables. This is known as the central limit theorem (Grinstead and Snell, 1997).

The Gaussian distribution is the most well-studied probability distribution for continuous-valued random variables. It is also referred to as the normal distribution. Its importance originates from the fact that it has many computationally convenient properties.

There are many other areas of machine learning that also benefit from using a Gaussian distribution, for example Gaussian processes, variational inference, and reinforcement learning.

# Normal (also called Gaussian) Random Variable

Why important?

Central limit theorem

- One of the most remarkable findings in the probability theory

Convenient analytical properties

Modeling aggregate noise with many small, independent noise terms

- Standard Normal  $\mathcal{N}(0, 1)$

$$f_X(x) = \frac{1}{\sqrt{2\pi}} e^{-x^2/2}$$

- $\mathbb{E}[X] = 0$
- $\text{var}[X] = 1$

- General Normal  $\mathcal{N}(\mu, \sigma^2)$

$$f_X(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-(x-\mu)^2/2\sigma^2}$$

- $\mathbb{E}[X] = \mu$
- $\text{var}[X] = \sigma^2$

# Gaussian Distribution,

## What is Normal (Gaussian) Distribution?

The normal distribution is a descriptive model that describes real world situations.

It is defined as a continuous frequency distribution of infinite range (can take any values not just integers as in the case of binomial and Poisson distribution).

This is the most important probability distribution in statistics and important tool in analysis of epidemiological data and management science.

# Characteristics of Normal Distribution

- It links frequency distribution to probability distribution
- Has a Bell Shape Curve and is Symmetric
- It is Symmetric around the mean:

## Characteristics of Normal Distribution

In a Standard Normal Distribution:

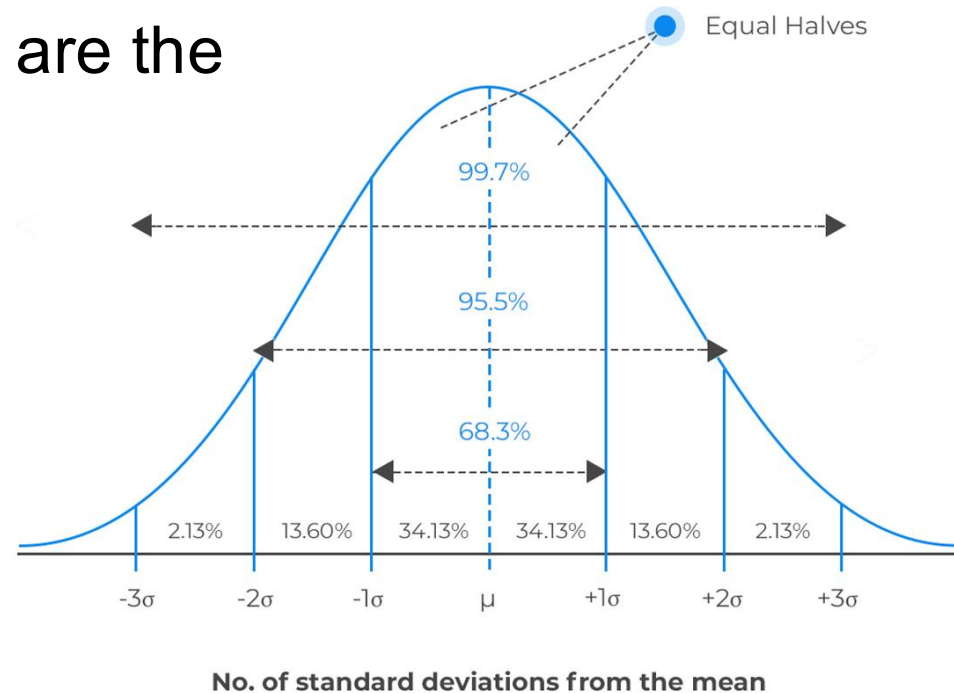
The mean ( $\mu$ ) = 0      and

Standard deviation ( $\sigma$ ) = 1



## Shape of the normal distribution

Two halves of the curve are the same (mirror images)



Hence Mean = Median

The total area under the curve is 1 (or 100%)

Normal Distribution has the same shape as Standard Normal Distribution.

# Data, Models and Learning

Three major components of a machine learning system:  
data, models, and learning.

The main question of machine learning is “**What do we mean by good models?**”.

One of the guiding principles of machine learning is that good models should perform well on unseen data. This requires us to define some performance metrics, such as accuracy or distance from ground truth, as well as figuring out ways to do well under these performance metrics.

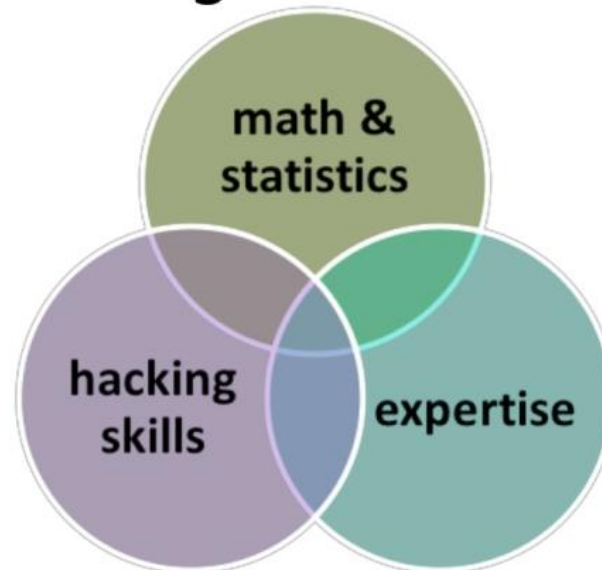
# WHAT IS DATA ?

*...solving problems with data...*



*...sounds cool!*

*What makes a good data scientist?*



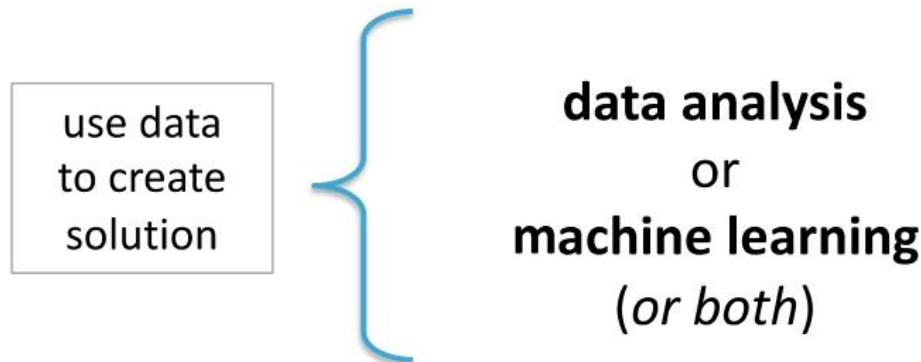


# WHAT IS DATA?

*...solving problems with data...*



*...which step is most challenging?*



# Data, Models and Learning

Three major components of a machine learning system

1. Data:  $\{(x_1, y_1), \dots, (x_n, y_n), \dots, (x_N, y_N)\}$
  2. Models: deterministic functions or probabilistic models
  3. Learning: Training, and prediction/inference
- Good machine learning models: Perform well for unseen (untrained) data
  - Machine learning algorithm: training and prediction

# Data as Vectors

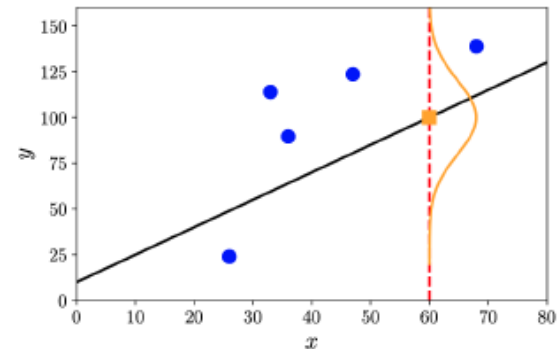
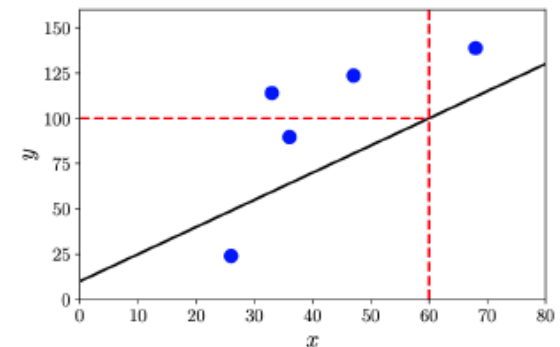
- Tabular format or not, numerical or not, good feature extraction etc.
- Assume that data is given as D-dimensional vector  $x_n$  of real numbers, each called **features**, **attributes**, or **covariates**.
- Dataset: consisting of data points or examples  $\{x_1, x_2, \dots, x_N\}$
- In supervised learning,  $\{(x_1, y_1), \dots, (x_n, y_n), \dots, (x_N, y_N)\}$ , where  $y_n$  is the **label** (or target, response variable, or annotation).

## Better representation of data as vectors

- finding lower-dimensional approximations of the original feature vector (e.g., PCA via SVD or EVD)
- using nonlinear higher-dimensional combinations of the original feature vector (e.g., feature map and kernel)

# Models: Functions vs. Probabilistic Models

- Now, the business of constructing a predictor
- Models as functions
  - $f : \mathbb{R}^D \mapsto \mathbb{R}$ .
  - **Example.**  $f(\mathbf{x}) = \theta^\top \mathbf{x} + \theta_0$ , Unknown parameter:  $\theta, \theta_0$
- Models as probabilistic models
  - model our uncertainty due to the observation process and our uncertainty in the parameters of our model
  - predictors should be able to express some sort of uncertainty via probabilistic models
  - Parameters: parameters of a chosen probabilistic model (e.g., mean and variance of Gaussian)



# Learning Algorithms

Three algorithmic phases

(1) Prediction or inference: via function or probabilistic models

(2) Training or parameters estimation

- fixed parameter assumption (non-probabilistic) or Bayesian approach (probabilistic)
- non-probabilistic: e.g., empirical risk minimization
- probabilistic: e.g., ML (Maximum Likelihood), MAP (Maximum A Posteriori)
- cross-validation: simulation of performing for unseen data
- regularization/prior: balancing models between training and unseen data

(3) Hyperparameter tuning or model selection

# Models as Functions: Empirical Risk Minimization

Empirical Risk Minimization (ERM) is a fundamental principle in statistical learning theory that defines a family of learning algorithms based on evaluating performance over a known and fixed dataset. The core idea is to estimate and optimize the performance of an algorithm on a known set of training data, referred to as the "empirical risk".

# Models as Functions: Empirical Risk Minimization

- Predictor as a function
- Given  $\{(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_n, y_n), \dots, (\mathbf{x}_N, y_N)\}$ , estimate a predictor  $f(\cdot, \boldsymbol{\theta}) : \mathbb{R}^D \mapsto \mathbb{R}$
- Find a good parameter  $\boldsymbol{\theta}^*$ , such that  $f(\mathbf{x}_n, \boldsymbol{\theta}^*) = \hat{y}_n \approx y_n$ , for all  $n = 1, \dots, N$

# Loss Function

- Training set:  $\{(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_n, y_n), \dots, (\mathbf{x}_N, y_N)\}$ , an example matrix<sup>1</sup>  
 $\mathbf{X} := [\mathbf{x}_1, \dots, \mathbf{x}_N]^T \in \mathbb{R}^{N \times D}$ , a label vector  $\mathbf{y} := [y_1, \dots, y_N]^T$ ,
- Average loss, empirical risk

$$R_{\text{emp}}(f, \mathbf{X}, \mathbf{y}) = \frac{1}{N} \sum_{n=1}^N \ell(y_n, \hat{y}_n)$$

- Goal: Minimizing empirical risk



# Overfitting and Regularization

- The predictor fits too closely to the training data and does not generalize well to new data
- Need to somehow bias the search for the minimizer of empirical risk by introducing a **penalty term**
- **Regularization**: compromise between accurate solution of empirical risk minimization and the size or complexity of the solution.

# Mean Average Precision (mAP)

In Machine Learning (ML), the term mAP stands for Mean Average Precision, which is a performance metric commonly used in tasks like object detection and information retrieval. Here's a concise explanation of both:

## Machine Learning (ML)

**Definition:** ML is a subset of artificial intelligence (AI) that enables systems to learn patterns from data and make predictions or decisions without being explicitly programmed.

**Applications:** Image recognition, natural language processing, recommendation systems, fraud detection, etc.

## Types:

**Supervised Learning:** Models learn from labeled data (e.g., classification, regression).

**Unsupervised Learning:** Models find patterns in unlabeled data (e.g., clustering, dimensionality reduction).

**Reinforcement Learning:** Models learn by interacting with an environment to maximize rewards.

# Models as Probabilistic Models: **Parameter Estimation (ML and MAP)**

## MLE (Maximum Likelihood Estimation): Concept

- Idea: define a function of the parameters called **likelihood function**.
- Negative log-likelihood for data  $\mathbf{x}$  and a family of probability densities  $\mathbb{P}(\mathbf{x} \mid \theta)$  parameterized by  $\theta$ :

$$\mathcal{L}_{\mathbf{x}}(\theta) = \mathcal{L}(\theta) := -\log \mathbb{P}(\mathbf{x} \mid \theta)$$

- $\mathcal{L}(\theta)$ : how likely a particular setting of  $\theta$  is for the observations  $\mathbf{x}$ .
- **MLE**: Find  $\theta$  such that  $\mathcal{L}(\theta)$  is **minimized** (i.e., likelihood is **maximized**)

# MLE: Supervised Learning

- The set of iid examples  $(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_N, y_N)$
- $\mathcal{X} = \{\mathbf{x}_1, \dots, \mathbf{x}_N\}$  and  $\mathcal{Y} = \{y_1, \dots, y_N\}$
- Negative log-likelihood

$$\mathcal{L}(\theta) = -\log \mathbb{P}(\mathcal{Y} \mid \mathcal{X}, \theta) = \sum_{n=1}^N \log \mathbb{P}(y_n \mid \mathbf{x}_n, \theta)$$

# MAP (Maximum A Posteriori)

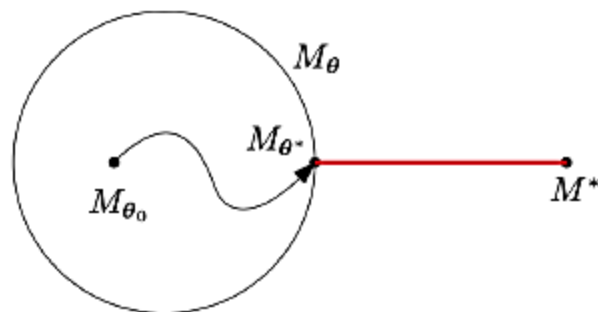
- What if we have some **prior knowledge** about  $\theta$ ? Then, how should we change our knowledge about  $\theta$  after observing data  $\mathbf{x}$ ?
- Compute a posteriori distribution (using Bayes' Theorem) and find  $\theta$  that maximizes the distribution:

$$\max_{\theta} \mathbb{P}(\theta \mid \mathbf{x}) = \max_{\theta} \frac{\mathbb{P}(\mathbf{x} \mid \theta) \mathbb{P}(\theta)}{\mathbb{P}(\mathbf{x})} \iff \min_{\theta} \left( -\log \mathbb{P}(\theta \mid \mathbf{x}) \right)$$

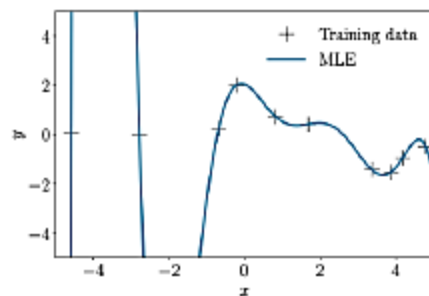
- In finding the optimal  $\theta$ ,  $\mathbb{P}(\mathbf{x})$  can be ignored
- ML and MAP: Bridging the non-probabilistic and probabilistic worlds as it explicitly acknowledges the need for a prior distribution, yet producing a **point estimate** (one single parameter return).

# Model Fitting

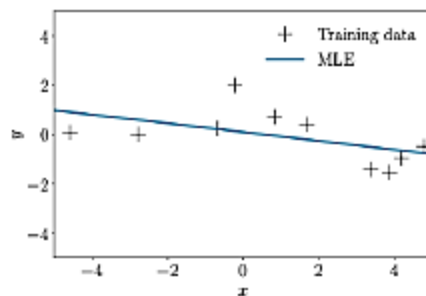
- Model class  $M_\theta$  vs. Right model  $M^*$



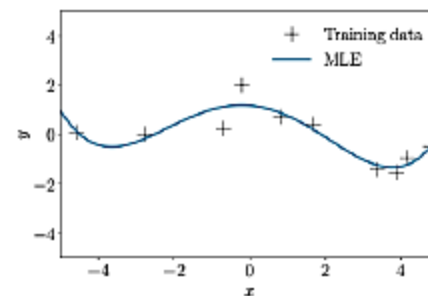
- Overfitting vs. Underfitting vs. Good fitting



(a) Overfitting



(b) Underfitting.



(c) Fitting well.

# Probabilistic Modeling and Inference

## Modeling Generative Process and Probabilistic Models

- For a data set  $\mathcal{X}$ , a parameter prior  $\mathbb{P}(\theta)$ , and a likelihood function, the posterior is:

$$\mathbb{P}(\theta \mid \mathcal{X}) = \frac{\mathbb{P}(\mathcal{X} \mid \theta)\mathbb{P}(\theta)}{\mathbb{P}(\mathcal{X})}, \quad \mathbb{P}(\mathcal{X}) = \int \mathbb{P}(\mathcal{X} \mid \theta)\mathbb{P}(\theta) \, d\theta$$

- Implementation hardness
  - Bayesian inference requires to solve integration, which is often challenging. In particular, a conjugate prior is not chosen, the integration is not analytically tractable.
  - Approximation techniques: MCMC (Markov Chain Monte Carlo), Laplace approximation, variational inference, expectation propagation

# Latent Variable Models

Latent variables are not directly observed but inferred from the data.

Used for clustering, dimensionality reduction, sequence modeling

Example: GMM, PCA, Hidden Markov Models

- In GMM, the cluster a data point belongs to is latent.

Use Case:

- Topic modeling in NLP using Latent Dirichlet Allocation (LDA).



# Latent-Variable Models (1)

- Including latent variables in the model  $\rightarrow$  contributing to the interpretability of the model
- General discussions here would be applied the following examples later
  - PCA for dimensionality reduction L10(7)
  - Gaussian mixture models for density estimation L11(3)
- In latent-variable models (LVMs)<sup>2</sup>,
  - Given: **prior**  $\mathbb{P}(\mathbf{z})$  and **likelihood**  $\mathbb{P}_{\theta}(\mathbf{x}|\mathbf{z})$
  - **Joint dist.** from prior and likelihood:  $\mathbb{P}_{\theta}(\mathbf{x}, \mathbf{z}) = \mathbb{P}_{\theta}(\mathbf{x}|\mathbf{z})\mathbb{P}(\mathbf{z})$
  - Our interest: **marginal likelihood**  $\mathbb{P}_{\theta}(\mathbf{x})$  and **posterior**  $\mathbb{P}_{\theta}(\mathbf{z}|\mathbf{x})$

---

<sup>2</sup>In our note, we express the dependence on the model parameters  $\theta$  using subscript notations, e.g.,  $\mathbb{P}_{\theta}(\mathbf{x}|\mathbf{z})$  rather than  $\mathbb{P}(\mathbf{x}|\mathbf{z}, \theta)$  to highlight the role of  $\mathbf{z}$ .

# LVM (2)

- Assuming we know  $\theta$ , to generate a data sample from the model (i) sample  $\mathbf{z}$  from  $\mathbb{P}(\mathbf{z})$  and (ii) sample  $\mathbf{x}$  from  $\mathbb{P}_{\theta}(\mathbf{x}|\mathbf{z})$
- **Inference.** computing the **posterior distribution**  $\mathbb{P}_{\theta}(\mathbf{z}|\mathbf{x})$ :

$$\mathbb{P}_{\theta}(\mathbf{z}|\mathbf{x}) = \frac{\mathbb{P}_{\theta}(\mathbf{x}, \mathbf{z})}{\mathbb{P}_{\theta}(\mathbf{x})} = \frac{\mathbb{P}_{\theta}(\mathbf{x}, \mathbf{z})}{\int \mathbb{P}_{\theta}(\mathbf{x}, \mathbf{z}) d\mathbf{z}}$$

- This requires to solve the sub-problem of computing the **marginal likelihood** of the observation:

$$\mathbb{P}_{\theta}(\mathbf{x}) = \int \mathbb{P}_{\theta}(\mathbf{x}, \mathbf{z}) d\mathbf{z}$$

# LVM (3): Why the posterior distribution $P_{\theta}(z|x)$ ?

- **Explanation of the observation.** Allows us to figure out which latent configurations could have plausibly generated the observation data samples.
- **Learning of model parameters  $\theta$ .** Training LVMs to estimate  $\theta$  (e.g., ML) requires  $\mathbb{P}_{\theta}(z|x)$  in its inner loops

marginal likelihood  $\mathbb{P}_{\theta}(\mathbf{x}) \implies$  posterior distribution  $\mathbb{P}_{\theta}(z|\mathbf{x}) \implies \theta_{\text{ML}}$

# LVM (4): How is $P_{\theta}(z|x)$ ? used for $\theta$ ML?

- Assuming we know  $\theta$ , to generate a data sample from the model (i) sample  $\mathbf{z}$  from  $\mathbb{P}(\mathbf{z})$  and (ii) sample  $\mathbf{x}$  from  $\mathbb{P}_{\theta}(\mathbf{x}|\mathbf{z})$
- **Inference.** computing the **posterior distribution**  $\mathbb{P}_{\theta}(\mathbf{z}|\mathbf{x})$ :

$$\mathbb{P}_{\theta}(\mathbf{z}|\mathbf{x}) = \frac{\mathbb{P}_{\theta}(\mathbf{x}, \mathbf{z})}{\mathbb{P}_{\theta}(\mathbf{x})} = \frac{\mathbb{P}_{\theta}(\mathbf{x}, \mathbf{z})}{\int \mathbb{P}_{\theta}(\mathbf{x}, \mathbf{z}) d\mathbf{z}}$$

- This requires to solve the sub-problem of computing the **marginal likelihood** of the observation:

$$\mathbb{P}_{\theta}(\mathbf{x}) = \int \mathbb{P}_{\theta}(\mathbf{x}, \mathbf{z}) d\mathbf{z}$$