COMP4616 – Fundamentals of Machine Learning

ASST. PROF. TUĞBA ERKOÇ

SPRING 2025

WEEK 1: INTRODUCTION

COMP4616 - Fundamentals of Machine Learning

Course materials & Syllabus on Blackboard.

Check Blackboard Announcements & Messages regularly. I will use Blackboard to communicate with you.

Textbook

- Machine Learning, Tom Mitchell McGraw Hill (1997).
- Introduction to machine learning, Ethem Alpaydın. MIT press, 2020.

Workload & Grading

- In-class activities – 20 %
- Project – 20 %
- Midterm exam – 30%
- Final exam – 30%

Makeup Policy

- No makeups for in-class activities.

In case of any attempt of cheating in written examinations or project you will get an **F** for the course grade, and you will be reported to the Dean's office.

Key idea

Instead of trying to produce a programme to simulate the adult mind, why not rather try to produce one which simulates the child's? If this were then subjected to an appropriate course of education one would obtain the adult brain. Presumably the child brain is something like a notebook as one buys it from the stationer's. Rather little mechanism, and lots of blank sheets.

Alan Turing

What is Learning? What is Machine Learning?

"The activity or process of gaining knowledge or skill by studying, practicing, being taught, or experiencing something."

Merriam Webster dictionary

"A computer program is said to learn from experience **E** with respect to some class of tasks **T** and performance measure **P**, if its performance at tasks in **T**, as measured by **P**, improves with experience **E**."

Tom Mitchell

A well-defined learning task is given by <P,T,E>

Examples of Defining the Learning Task

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

T: Task
P: Performance
E: Experience

What is Machine Learning?

We try to uncover underlying patterns in data

Machine learning (ML) is concerned with building autonomous agents which uses the experience gained (underlying patterns) from the given data to perform some tasks.

Machine learning and **Artificial Intelligence** are generally confused

AI does not always imply a learning-based system

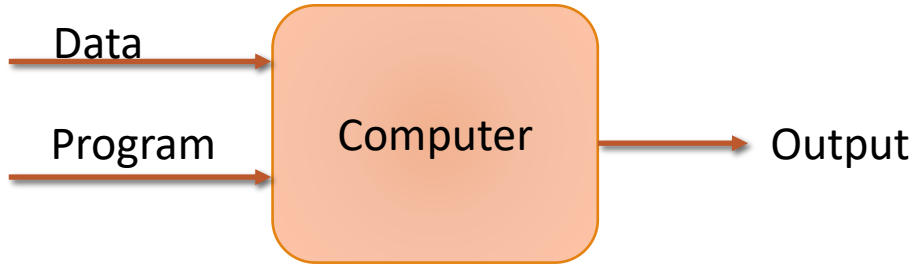
- Symbolic reasoning
- Rule based system
- Tree search
- etc.

Learning based system → learned based on the given data

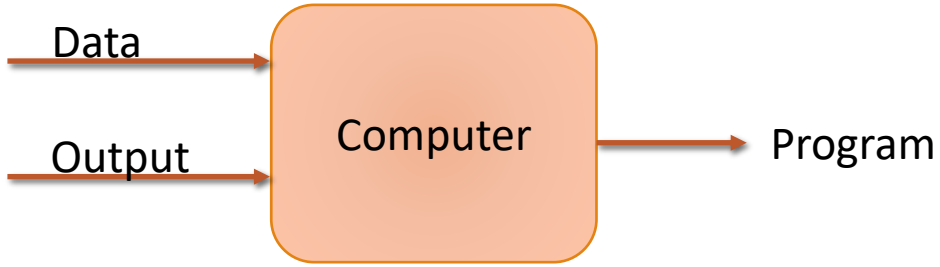
- more flexibility, good at solving pattern recognition problems

Programming vs Machine Learning

Traditional Programming



Machine Learning



When do we use Machine Learning?

For many problems, it's difficult to program the correct behavior by hand
- recognizing people and objects
- understanding human speech
- understanding huge genomic data

Machine learning approach:
- program an algorithm to automatically learn from data, or from experience

Why might you want to use a learning algorithm?
- hard to code up a solution by hand (e.g. vision, speech)
- system needs to adapt to a changing environment (e.g. spam detection, personalized medicine)
- want the system to perform better than the human programmers
- privacy/fairness (e.g. ranking search results)

Don't forget that not all problems need learning.
- No need to learn how to calculate the payroll of employees.

Machine Learning Workflow

1. Should I use ML on this problem?
   - Is there a pattern to detect?
   - Can I solve it analytically?
   - Do I have data?
2. Gather and organize data.
   - Preprocessing, cleaning, visualizing.
3. Establishing a baseline.
4. Choosing a model, loss, regularization, ...
5. Optimization (could be simple or hard).
6. Hyperparameter search.
7. Analyze performance & mistakes, and iterate back to step 4 (or 2)

Types of Machine Learning

Supervised learning

- have labeled examples of the correct behavior

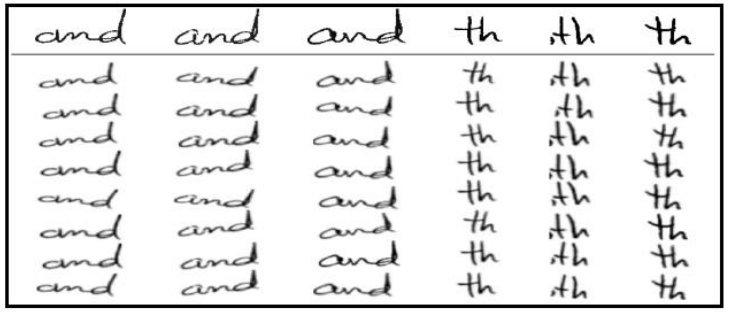
Reinforcement learning

- learning system (agent) interacts with the world and learns to maximize a scalar reward signal

Unsupervised learning

- no labeled examples – instead, looking for “interesting” patterns in the data

Supervised Learning



Objective is to learn an unknown **target function** f

Input is a **training set** of labelled examples. (x_i, y_i) where $y_i = f(x_i)$

Examples:

- x_i is an image, y_i is giraffe
- x_i is a seismic signal, y_i is explosion

Output is hypothesis h which is close to f (i.e. meaning that h predicts the correct outcome well on unseen examples)

The unseen examples forms the **test set**.

Many possible hypothesis families for h

- Linear models, logistic regression, neural networks, decision trees, nearest-neighbor, etc.

Supervised Learning

Training set examples for different tasks

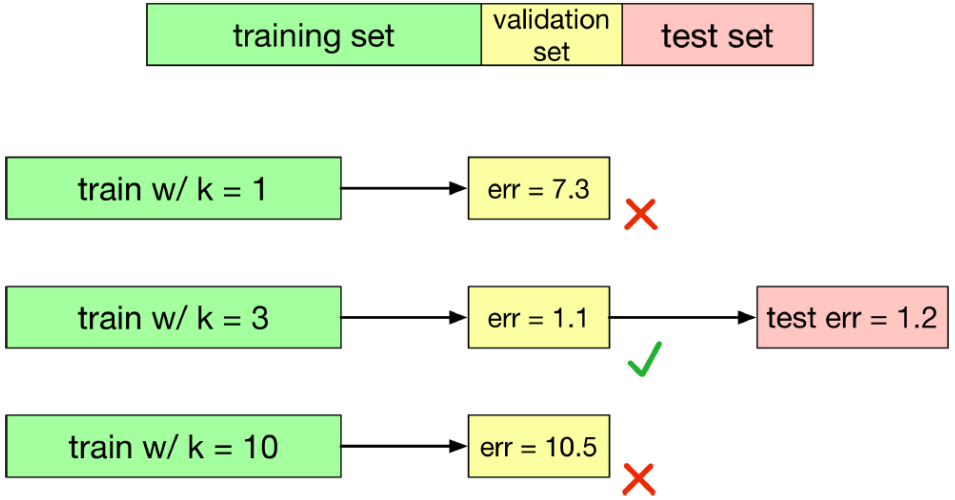
Task	Inputs	Labels
object recognition	image	object category
image captioning	image	caption
document classification	text	document category
speech-to-text	audio waveform	text
⋮	⋮	⋮

Supervised Learning

Most of the times, we would like to **fine tune** the **parameters** in our **hypothesis function**

We can do that by using a separate **validation set**.

Assume that **k** is a parameter that we want to fine tune.



Supervised Learning – Object Recognition

x_i






y_i

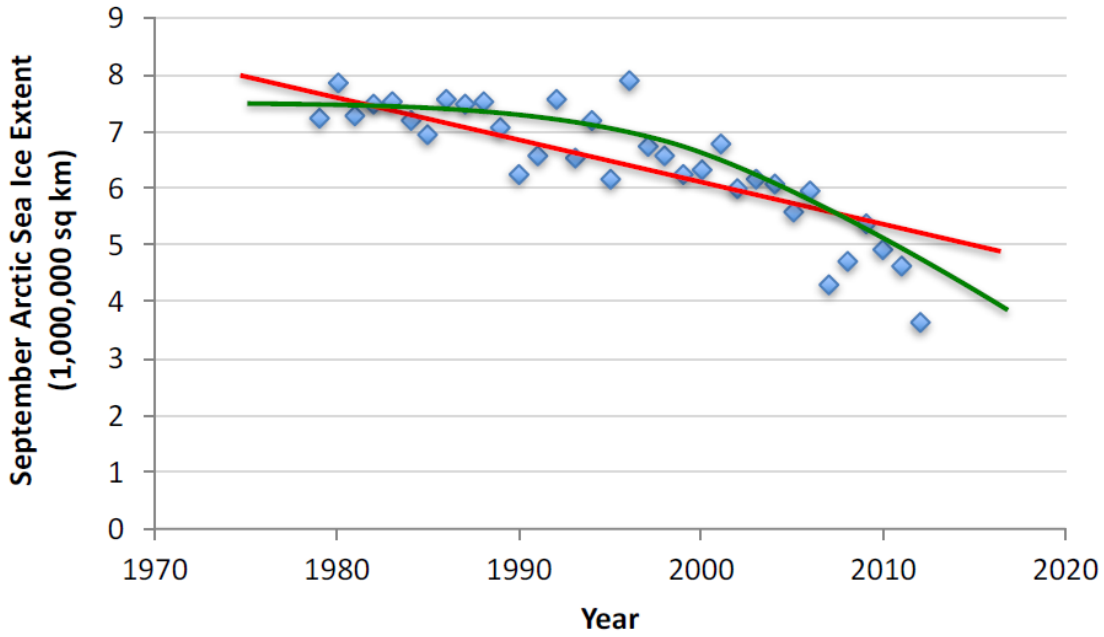
giraffe giraffe giraffe llama llama llama

$x =$

$f(x) = ?$

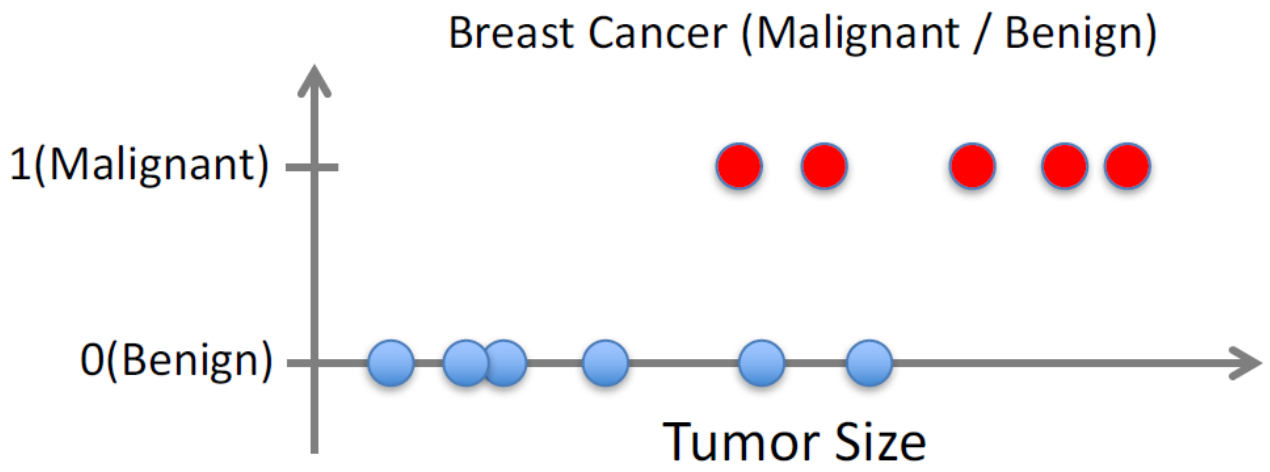
Supervised Learning

If y is real valued, we call the learning process as **regression**



Supervised Learning

If y is categorical, we call the learning process as **classification**



Supervised Learning

Which hypothesis space **H** to choose?

- **Example**: Linear combination of features

$$h_w(x) = w^T x$$

How to measure **degree of fit** between the hypothesis h and target function f?

- Loss function
- **Example**: Squared error

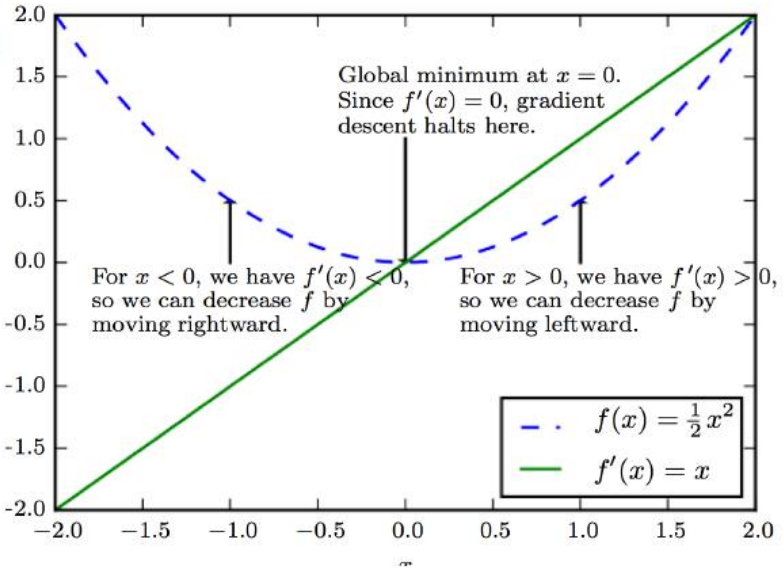
$$\sum_j (y_j - w^T x)^2$$

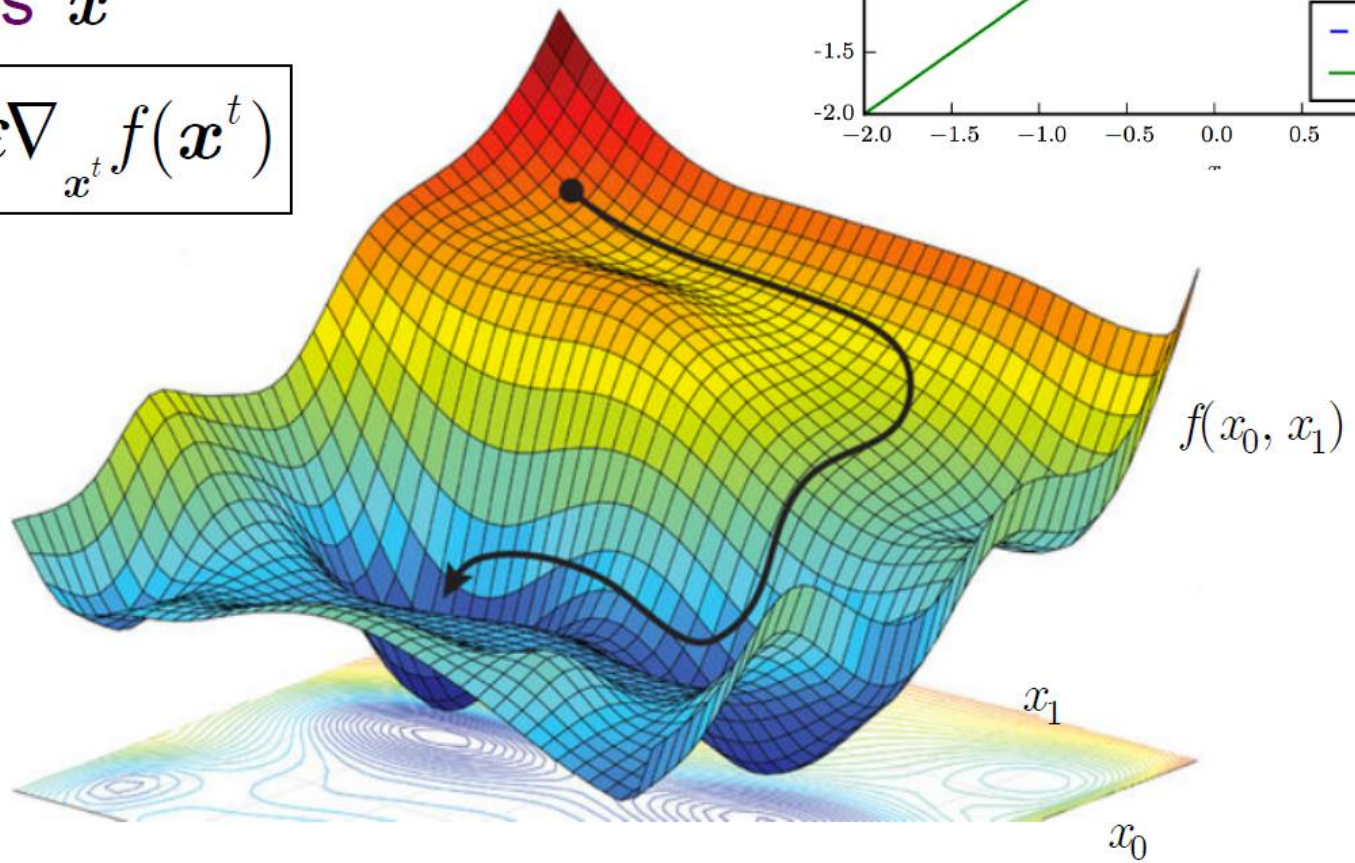
Supervised Learning by Gradient Descent

Classification Task:

Loss function $f(\boldsymbol{x})$,
e.g., sum of squared errors,
given weights $\boldsymbol{x}$

$$\boldsymbol{x}^{t+1} = \boldsymbol{x}^{t} - \varepsilon \nabla_{\boldsymbol{x}^{t}} f(\boldsymbol{x}^{t})$$



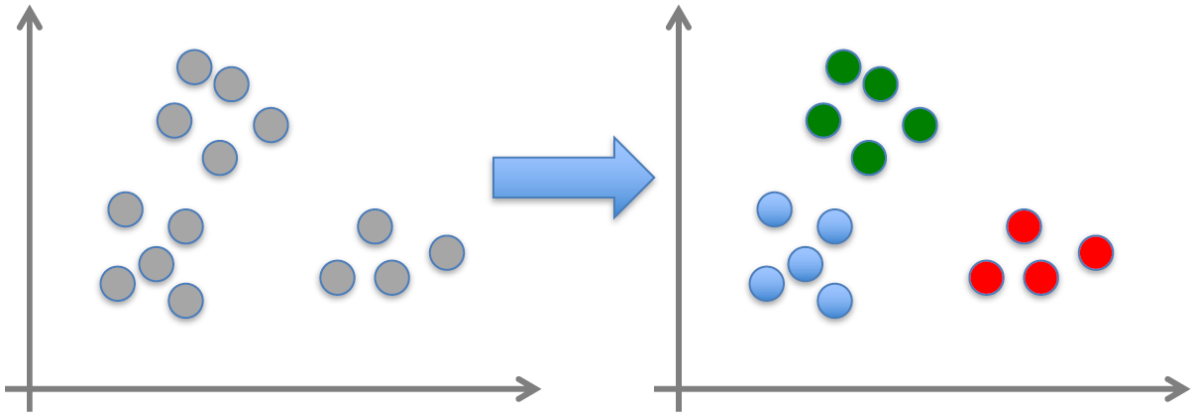


Unsupervised Learning

Given x_is without labels, we try to output the hidden structure of the data.

Clustering methods

- Used to group similar data points together based on their characteristics.
- They help identify patterns in unlabeled datasets.

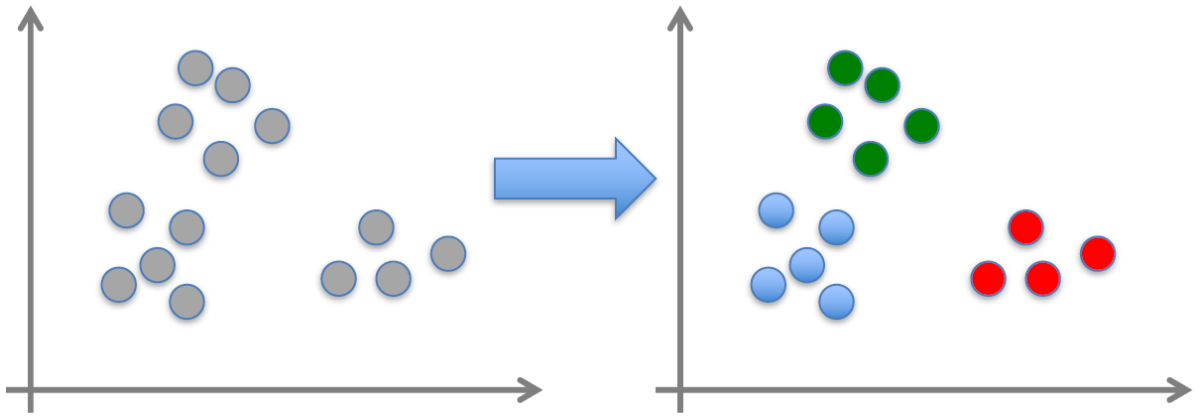


Market segmentation

Unsupervised Learning Methods

Clustering Methods

- Partitioning Methods → K-means or K-Medoids
- Hierarchical Clustering → Agglomerative (bottom-up) or Divisive (top-down)
- Density-Based Clustering → DBSCAN (Density-Based Spatial Clustering of Applications with Noise)
- Grid-Based Clustering → STING (Statistical Information Grid)
- Model-Based Clustering → GMM (Gaussian Mixture Model)

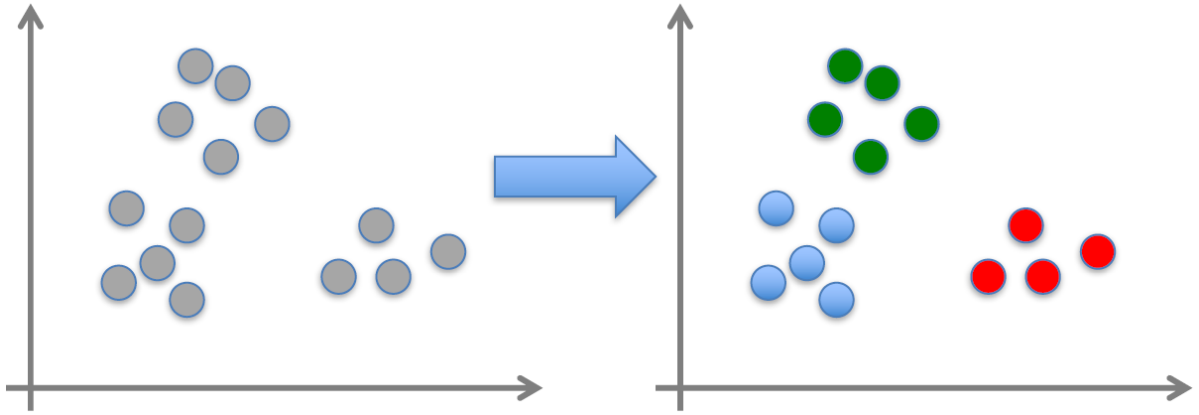


Market segmentation

Unsupervised Learning Methods

Clustering Methods

Method	Best For	Weakness
K-Means	Large datasets, fast processing	Sensitive to initial choice of k
Hierarchical Clustering	Small datasets, visualization	Slow for large datasets
DBSCAN	Clustering with noise & varying densities	Struggles with varying cluster densities
GMM	Data with overlapping clusters	Computationally expensive



Market segmentation

Unsupervised Learning Methods

We can identify a **low-dimensional manifold** by using methods like **Principal Component Analysis** (**PCA**) or **Autoencoders**.

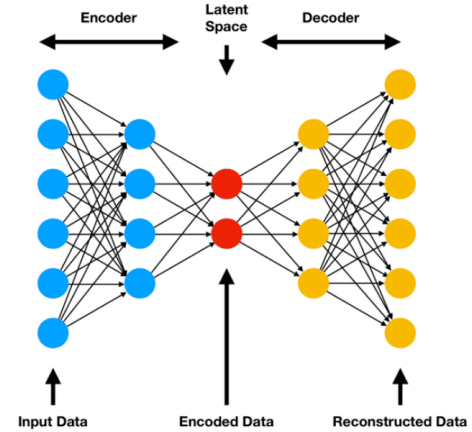
- We can find a simpler, lower-dimensional representation of high-dimensional data
- **High-dimensional data:** Data with many features (e.g., images, text, gene expression data).
- **Low-dimensional manifold:** A simplified version of the data that captures its essential structure with fewer dimensions.

PCA steps:

- **Standardization:** The data is first normalized so that all features have a mean of 0 and a standard deviation of 1
- **Covariance Matrix Computation:** PCA calculates the relationships between different features using a covariance matrix.
- **Eigenvalue & Eigenvector Calculation:** The eigenvectors (principal components) and their corresponding eigenvalues are determined. These represent the directions and magnitudes of variance in the data.
- **Feature Transformation:** The data is projected onto the new principal components, reducing dimensions while preserving maximum variance.

Unsupervised Learning Methods

Autoencoders are neural networks that learn a compressed representation of the data by encoding and then reconstructing it.



An **Autoencoder** consists of two main parts

- **Encoder:** Compresses the input into a lower-dimensional representation (latent space).
- **Decoder:** Reconstructs the input from the compressed representation

Input→Encoder→Latent Representation→Decoder→Reconstructed Output

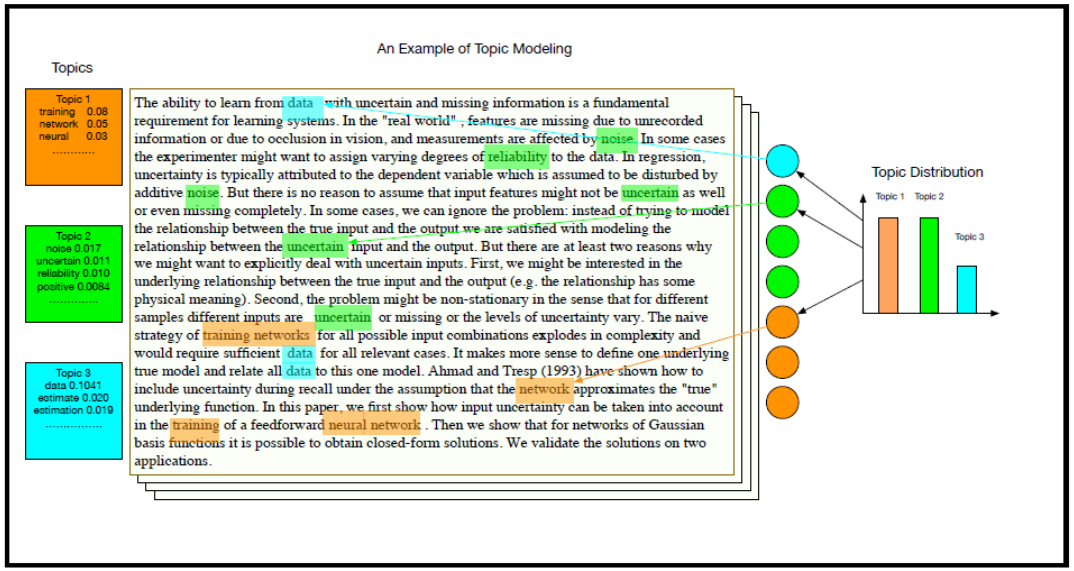
Imagine a dataset of handwritten digit images

- Each 28×28 pixel grayscale image has 784 pixels.
- If each pixel is treated as an individual feature (e.g., its intensity value as a feature in a vector), then the dataset represents each image as a 784-dimensional feature vector.
- Therefore, the data is 784-dimensional, meaning each sample (image) exists in a 784-dimensional feature space.
- An **autoencoder** can learn a more compact representation (e.g., a 32-dimensional encoding) and still reconstruct the digits well.

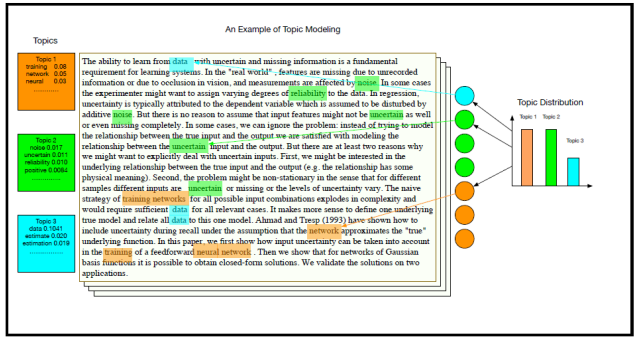
Unsupervised Learning Methods

We can use topic modeling by using **Singular Value Decomposition**(SVD) which is another method for reducing dimensionality and extracting meaningful patterns.

- Topics are distributions over words
- Document: a distribution across topics



Unsupervised Learning Methods



We can use topic modeling by using **Singular Value Decomposition**(SVD) which is another method for reducing dimensionality and extracting meaningful patterns.

- Topics are distributions over words
- Document: a distribution across topics

Steps of SVD for topic modelling

- **Text Representaiton:** Convert a set of documents into a large matrix
- **Apply SVD on the large matrix:** This will decompose the matrix into three matrices

$$A = U \cdot S \cdot V^T$$

  - U (left singular vectors) represents documents,
  - S (singular values) represents importance of topics
  - V^T (right singular vectors) represents words.
- **Apply Dimensionality Reduction:** By keeping only the top k singular values, we remove noise and capture the most significant underlying topics

Unsupervised Learning Methods

Method	Supervised?	Description
SVD (used in LSA for topic modeling)	❌ No	Finds patterns and reduces dimensions without labels.
PCA	❌ No	Identifies principal components without knowing classes.
Autoencoders	❌ No	Learn to compress data without labels.
Logistic Regression	✅ Yes	Requires labeled data to classify samples.
Neural Networks (for classification)	✅ Yes	Trains on labeled data to make predictions.

Reinforcement Learning

Training data in between supervised/unsupervised

- Indication of whether action is correct or not
- Reward signal may refer to entire input sequence
- Dog is given a reward/punishment for an action

Policies: what actions to take in a particular situation

Utility estimation: how good is state (used by policy)

No supervised output but delayed reward

Credit assignment

- what was responsible for outcome

Applications:

- Game playing, Robot in a maze

Reinforcement Learning - Learning to play ATARI

- Action(a)={left,right}
- Observation(s)=[image frame]
- Reward(r)= -100 if lose, -1 if win
- Policy(π)=$P_\pi(a|s)$
  - 10,000 states, 2 actions
- $Q(s,a)$= value (action,state)

$$Q_{i+1}(s,a) = \mathbb{E}_{s'}[r + \gamma \max_{a'} Q_i(s',a')|s,a]$$

- Loss = γ+E[$\max_{a'} Q(s',a')$- $Q_i(s',a')$