Minia University
Faculty of Computers & information

Artificial Neural Networks and Deep Learning

Slides By:

T.A. Sarah Osama Talaat

E-mail: SarahOsama.fci@gmail.com

Slides were prepared based on set of references mentioned in the last slide

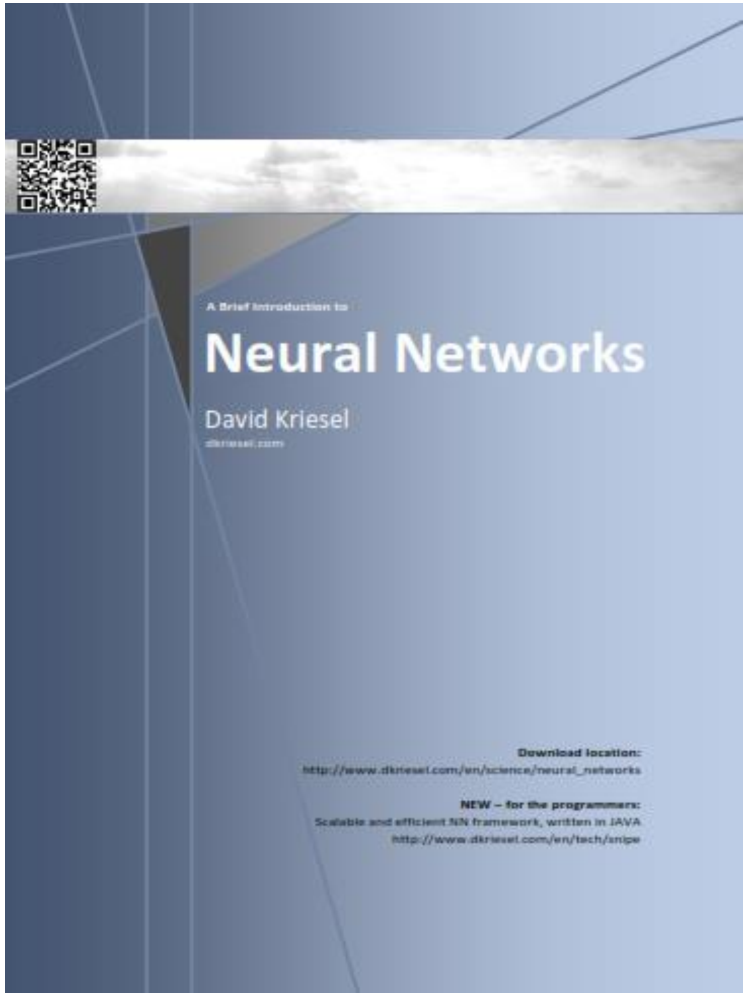
Agenda

- Logistic
- Rule and Regulations
- Course content
- Objectives and learning outcomes
- Machine Learning Basics

Logistic

❑ Textbook

Kriesel, David. "A Brief Introduction to Neural Networks. 2007." URL http://www. dkriesel. com (2007).



Logistic

❑ Online textbooks

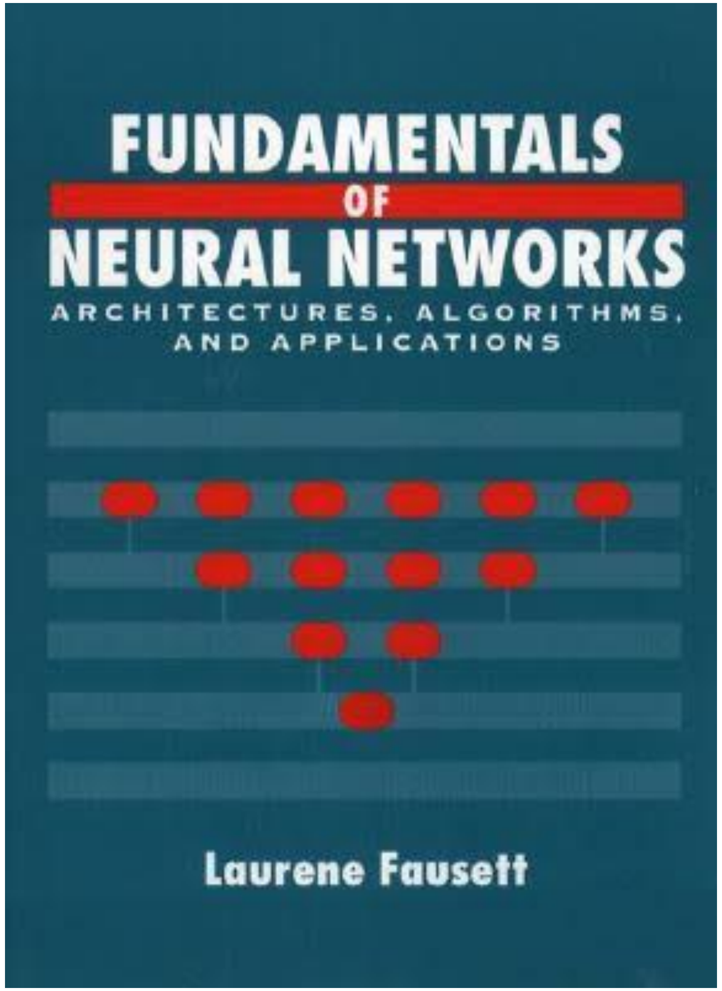
http:// www.deeplearningbook.org

http://neuralnetworksanddeeplearning.com/index.html

Logistic

❑ Textbook

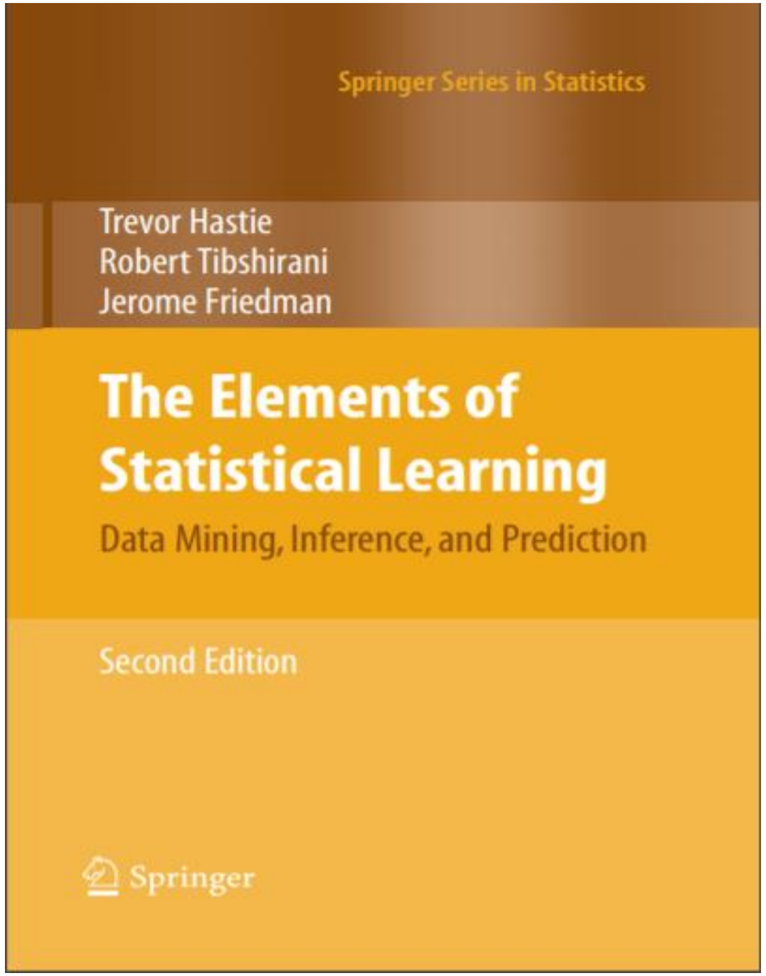
da Fontoura Costa, Luciano, and Gonzalo Travieso. "Fundamentals of neural networks: By Laurene Fausett. Prentice-Hall, 1994, pp. 461, ISBN 0-13-334186-0." (1996): 205-207.



Logistic

❑ Machine learning textbook

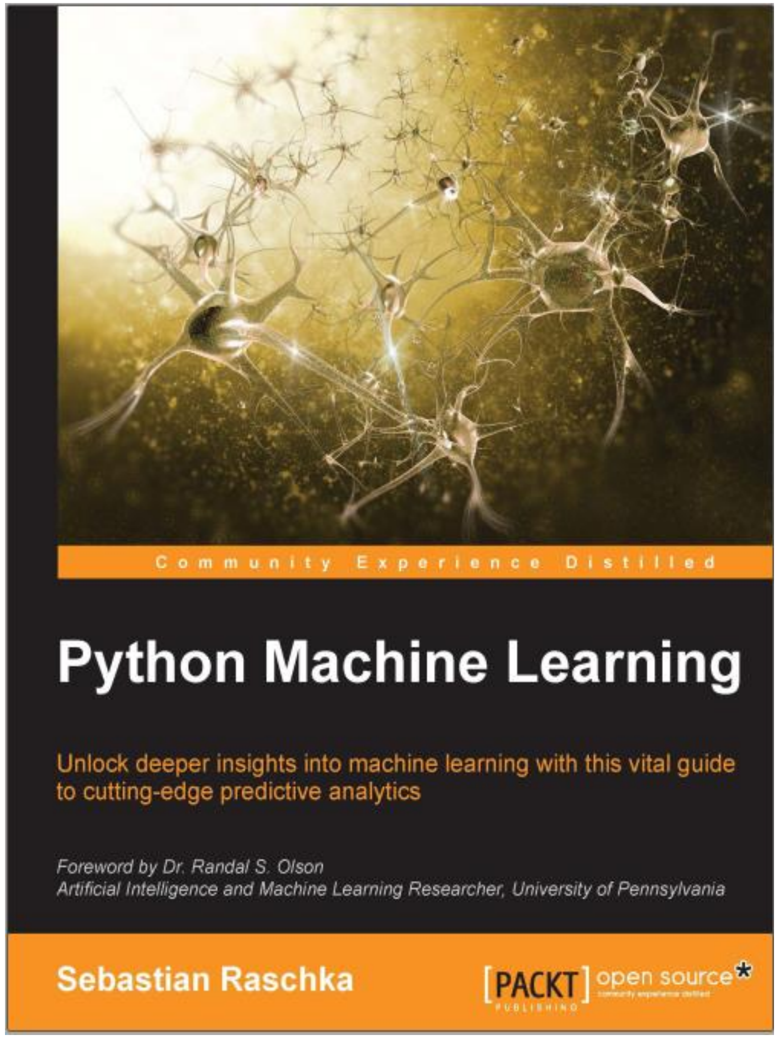
T. Hastie, R. Tibshirani and J. Fridmn, “The Elements of Statitical Learning”, second edition , Springer



Logistic

❑ Machine learning textbook

Raschka, Sebastian. “Python machine learning”. Packt Publishing Ltd, 2015.



Rule and Regulations

❑ Evaluation

<table>
<tr><th rowspan="2">Course Title</th><th colspan="2">Hours/week</th><th colspan="4">Marks</th><th colspan="2">Exams hours</th></tr>
<tr><th>Lecture</th><th>Lab</th><th>Final</th><th>Midterm</th><th>Lab exam and project</th><th>Total</th><th>Midterm</th><th>Final</th></tr>
<tr><td>Neural Networks</td><td>3</td><td>2</td><td>65</td><td>10</td><td>25</td><td>100</td><td>1</td><td>3</td></tr>
</table>

Course Content

- ❑Machine Learning Basics
- ❑Fundamentals on learning and training samples (fundamental)
- ❑Data Preprocessing
- ❑Biological neural networks
- ❑Components of artificial neural networks (fundamental)
- ❑Supervised learning network paradigms; The perceptron, backpropagation and its variants

Course Content

- Convolutional Neural Networks(CNN)
- Self Organization Maps (SOMs)
- Boltzmann Machines (BM)
- Evaluating, Improving and Tuning the (ANN, CNN, SMOs ,BM)

Objectives

1. Introduce the main fundamental principles and techniques of artificial neural network systems.
2. Understand the intuition behind artificial neural network.
3. Investigate the principal neural network models & applications
4. Introduce the main principles and techniques of deep learning neural networks
5. Understand the intuition behind set of main deep neural networks algorithms

Objectives

6. Describe the relation between real brains and simple artificial neural network models.
7. Explain and contrast the most common architectures and learning algorithms for different types of ANNs and deep learning DL algorithms.
8. Discuss the main factors involved in achieving good learning and generalization performance in neural network systems.
9. Identify the main implementational issues for common ANNs.
10. Evaluate the practical considerations in applying ANNs and DL to real classification and regression problems

Let’s Start

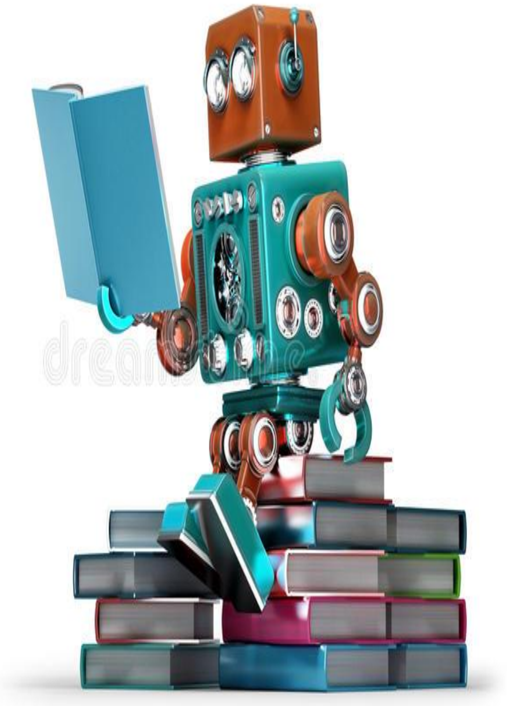
Introduction

- In the machine learning, pattern recognition and learning there are a relationship/nature between the observations (input features) and response (target) we need to understand this relationship .

Introduction
Machine learning algorithm

❑ Machine learning algorithm

- Machine learning algorithm is an algorithm that is able to learn from data. **But what do we mean by learning?**



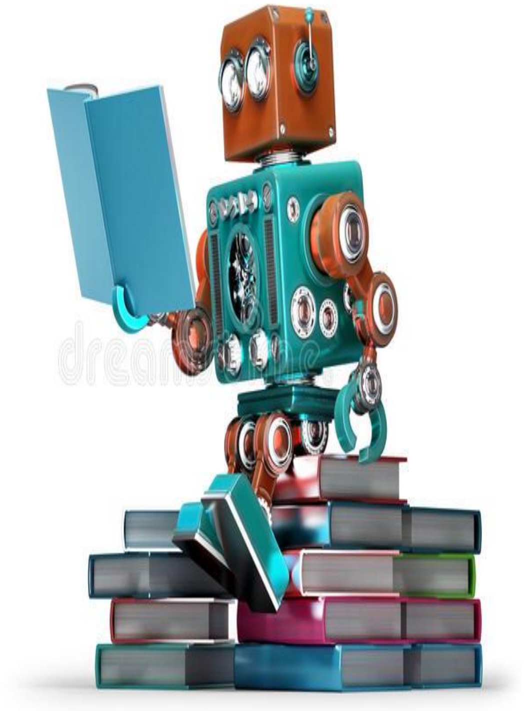
❑ Learning

- The learning process tends to understand and predict a new values based on finding a **mathematical model and relationship between** the given observations (i.e. inputs and output).

Introduction
Learning

❑ Definition 1.1 (Learning):

- 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** .



Introduction
Learning

❑ Definition 1.2 (The task T):

- Machine learning tasks are usually described in terms **of how the machine learning system should process an example**.
- An example is a collection of features that have been quantitatively measured from some object or event that **we want the machine learning system to process.** We typically represent an example as a vector $\mathbf{x} \in \boldsymbol{R}^n$ where each entry x_i of the vector is another feature.
- For example, the features of an image are usually the values of the pixels in the image.
- Many kinds of tasks can be solved with machine learning. Some of the most common machine learning tasks include; **classification, regression and clustering**.

Introduction
Learning

Definition 1.3 (The Performance Measure, P):

- In order to evaluate the abilities of a machine learning algorithm, we must design a quantitative measure of its performance. Usually this performance measure **P** is **specific** to the task being carried out by the system.
- For tasks T such as classification, we often measure the accuracy of the model. **Accuracy is just the proportion of examples for which the model produces the correct output**. We can also obtain equivalent information by measuring the error rate, the proportion of examples for which the model produces an incorrect output.

Introduction
Learning

Definition 1.3 (The Performance Measure, P):

- We often refer to **the error rate** as the expected **0-1** loss. The 0-1 loss on a particular example is 0 if it is correctly classified and 1 if it is not.
- Usually we are interested in how well the machine learning algorithm **performs on data that it has not seen before,** since this determines how well it will work when deployed in the real world. We therefore evaluate these performance measures using a **test set** of data that is separate from the data used for training the machine learning system.

Introduction
Learning

Definition 1.4 (The Experience, E):

- Machine learning algorithms can be broadly **categorized as unsupervised or supervised** by what kind of experience they are allowed to have during the learning process.
- Most of the learning algorithms in this course can be understood as being allowed to experience an entire **dataset**.

Introduction
Learning

Definition 1.4 (The Experience, E):

- **Most machine learning algorithms simply experience a dataset.**
- **Unsupervised learning algorithms experience** a dataset containing many features, then learn useful properties of the structure of this dataset. In the context of deep learning, we usually want to learn the entire probability distribution that generated a dataset, whether explicitly as in density estimation or implicitly for tasks like synthesis or denoising. Some other unsupervised learning algorithms perform other roles, like clustering, which consists of dividing the dataset into clusters of similar examples.

Introduction
Learning

Definition 1.4 (The Experience, E):

- **Supervised learning algorithms experience** a dataset containing features, but each example is also associated with a **label or target**. A supervised learning algorithm can study the input dataset and learn to classify the target (dependent variable) in this dataset into n different classes based on their features (independent variables).
- **Reinforcement learning algorithms do not just experience** a fixed dataset. These algorithms interact with an environment, so there is a feedback loop between the learning system and its experiences. Such algorithms are beyond the scope of this course.

Introduction
Learning

❑ Definition 1.5 (The Dataset):

- **A dataset** is a collection of many examples, sometimes we will also call examples **data points**. One common way of describing a dataset is with a **design matrix**.
- **A design matrix** is a matrix containing a different example in each row. Each column of the matrix corresponds to a different feature.

Air Temp (C)	Wind Direction (deg)	Global Light Energy (W.m2)	Wind speed
28.9	121	3	2.7
29	117	4	1.8
28.7	115	3	2.2
28.6	114	3	2.7

Introduction
Learning

The Dataset Example:

- For instance, the wind dataset contains 500 examples with four features for each example. This means we can represent the dataset with a design matrix x $\in \mathbb{R}^{500\times 4}$, where x_{i1} is the air temperature of instance (i.e. day) i, x_{i2} is the wind direction of day i, etc.
- Of course, to describe a dataset as a design matrix, it must be possible to describe each example as a vector, and each of these vectors must be the same size.
- This is not always possible. For example, if you have a collection of photographs with different widths and heights, then different photographs will contain different numbers of pixels, so not all of the photographs may be described with the same length of vector. This type of data is called **heterogeneous data**.

Introduction
Learning

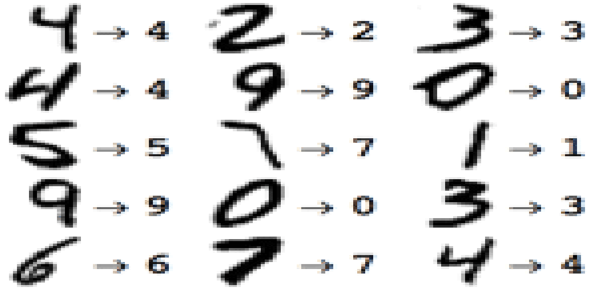
❑ Examples of learning problems:

- Predict a wind speed and a solar radiation.
- Detection of cancers; Brest, Leukemia and Spinal cancers.
- Estimate the amount of glucose in the blood.

Introduction

Machine Learning Real Application

Renewable Energy
Prediction



Handwritten
Digit Recognition

Cancer Detection

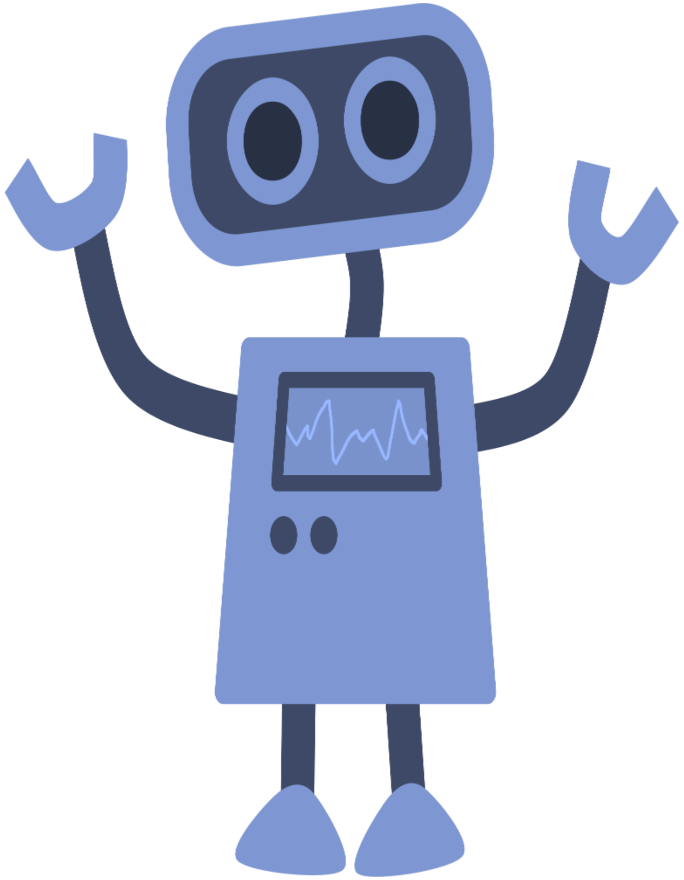
Email Spam Detection

Introduction

Machine Learning Importance

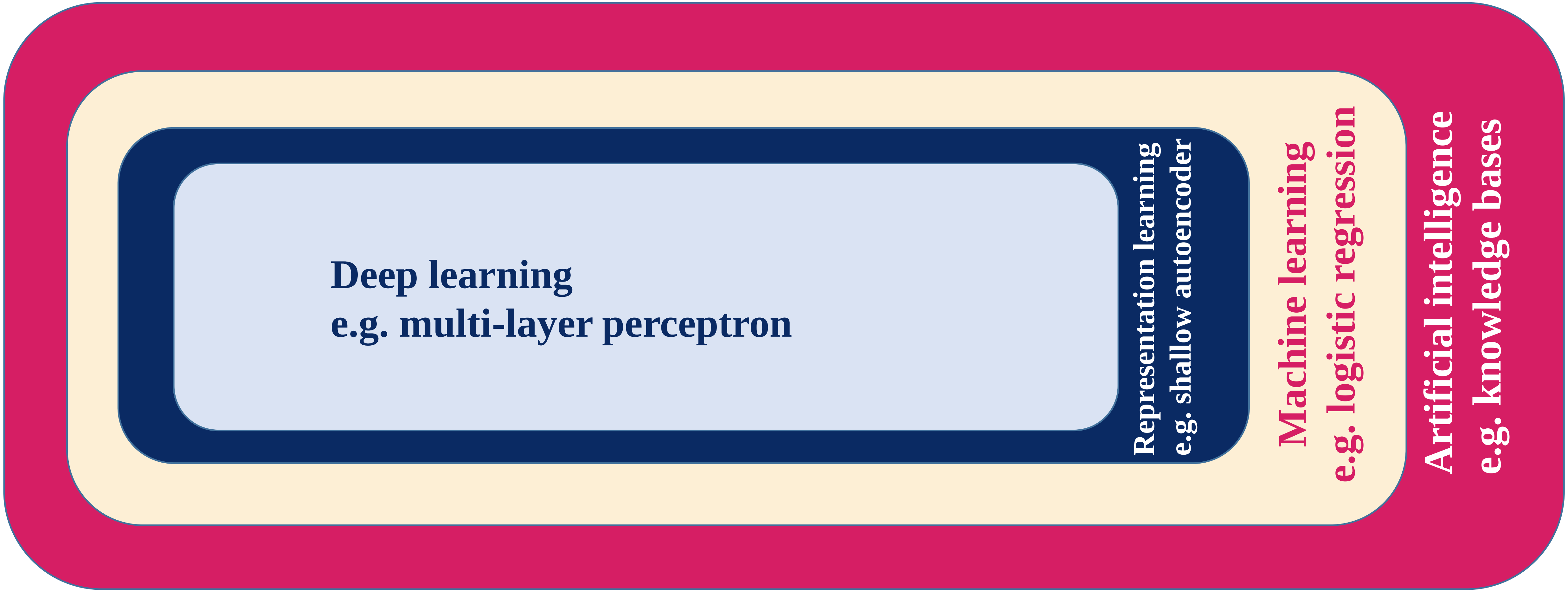
❑Machine Learning Importance:

- Machine learning offers a more efficient alternative of humans for **capturing the knowledge** in data to gradually **improve the performance of predictive models,** and make data-driven decisions
- Machine learning plays an ever greater role in our everyday life.



Introduction

Machine Learning and Artificial intelligence



Introduction
Variables Types

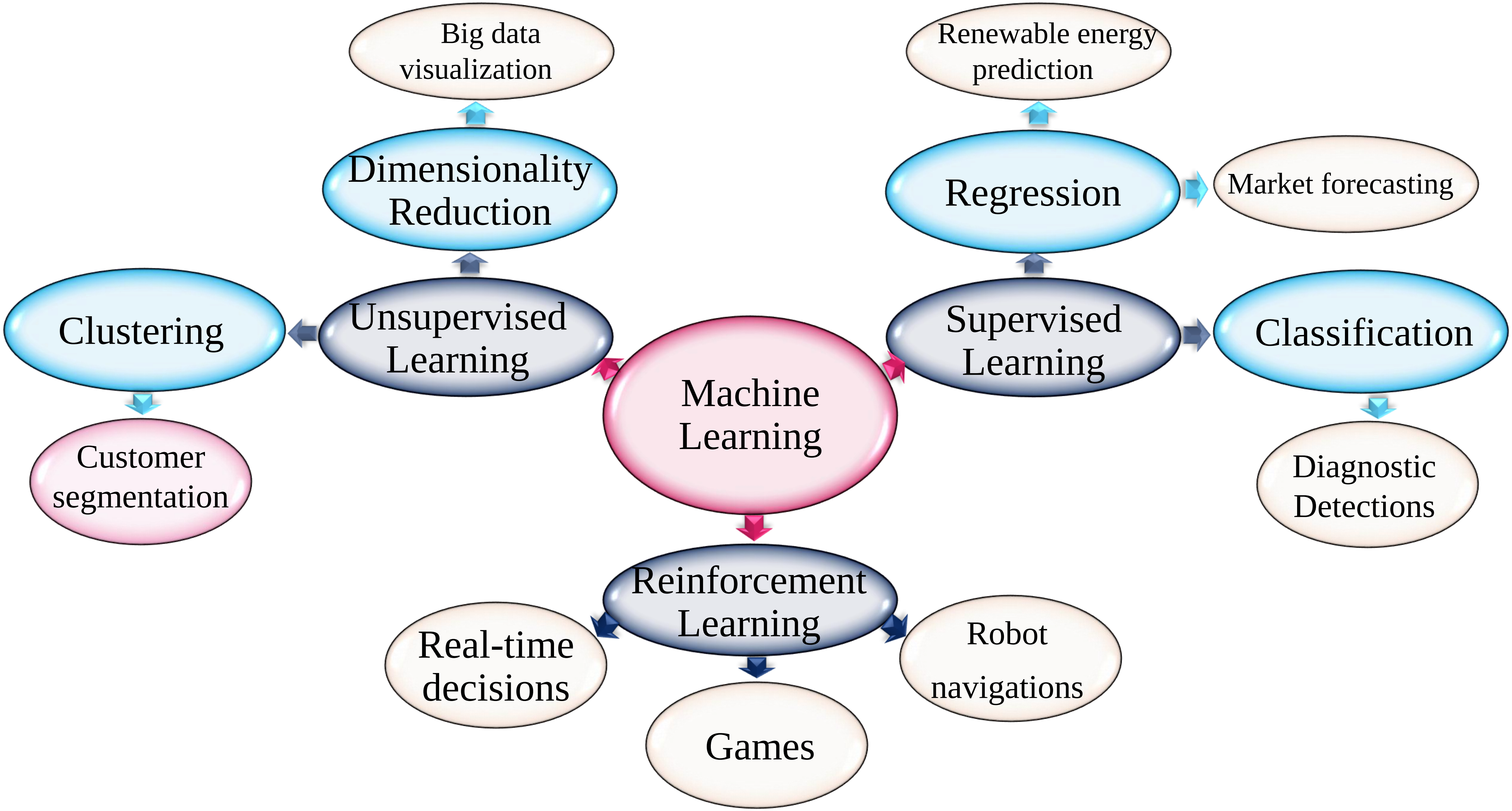
Variables Types

- Quantitative or continuous
- Qualitative, categorical or discreet
- Order categorical

Introduction
Variables Types

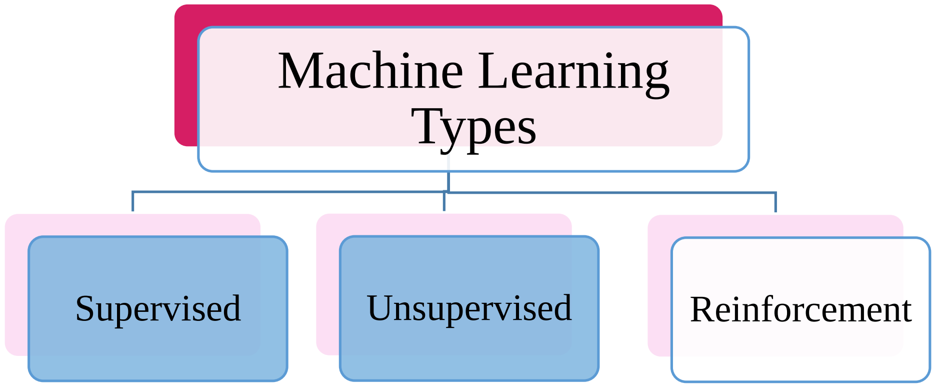
- **Quantitative or Continuous:**
  - Where a variables take a numeric value which is given as some measure, this value maybe positive, negative, integer, fraction ..etc.. ***For instance***: 6, 0.4, -5 or $\sqrt{8}$.
- **Qualitative, Categorical, Nominal or Discreet:**
  - Where a variable value belong to a specific set of categories. The variable value can not given as measure or matric but given as string . By other word, when the variable value is a label. ***For instance***: Gender = male or female, Gender $\in [male, female]$.
- **Ordered or Ordinal Categorical:**
  - This variable type take the characteristic of the two previous types, that mean the variable value is a category/label and can be sorted. The variable X $\in \mathcal{G}$, where $\mathcal{G}$ is a set of all possible values. ***For instance***: Size = small, medium, large or X-large

Some of the most common machine learning types (paradims) include the following:



Introduction
Machine Learning Types

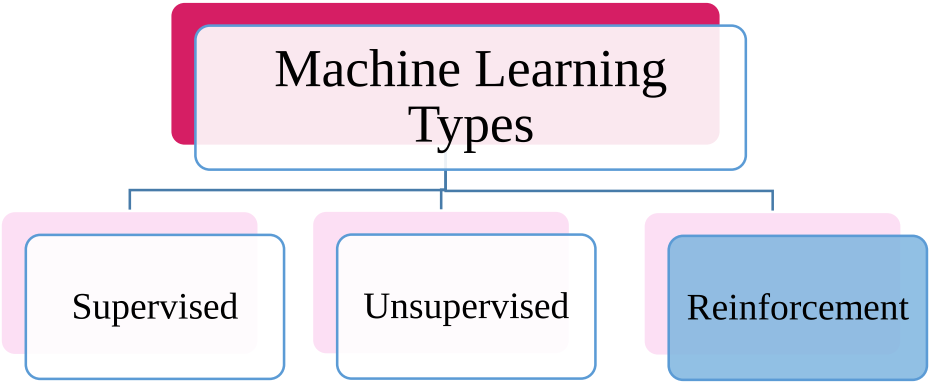
- **Supervised (predictive) learning:**
  - Build a model to predict a future data based on a training set which **includes** predictors (i.e. features) (**independent**) and response (i.e. outcome or output) (**dependent**) . It is like that a "teacher" gives the classes (supervision). **[The core of our course].**
- **Unsupervised (descriptive) learning:**
  - We need to organize, clustering the given data or establish the existence of the predictor. In this type of learning, we given a set of data and we don't know the response is **unknowns** and the predictor.



Introduction
Machine Learning Types

Reinforcement learning (RL):

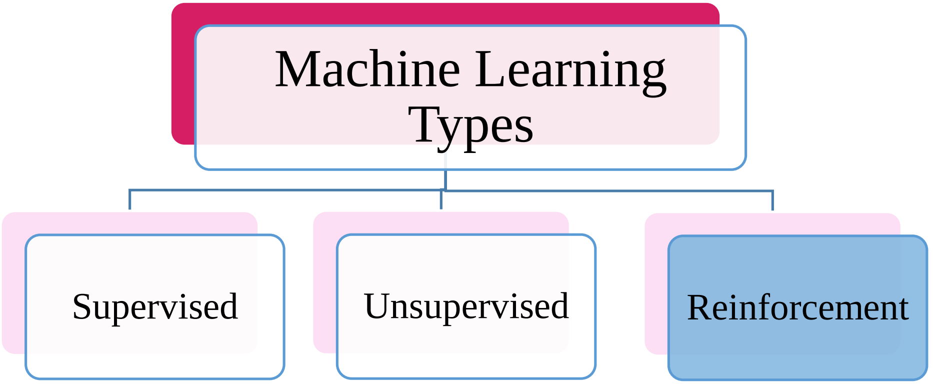
- This type is somewhat **less** commonly used.
- This is useful for learning how to act or behave when given occasional reward or punishment signals.
- In **reinforcement learning**, the **goal** is to develop a system (agent) that **improves its performance** based on **interactions** with the environment.



Introduction
Machine Learning Types

❑Reinforcement learning (RL):

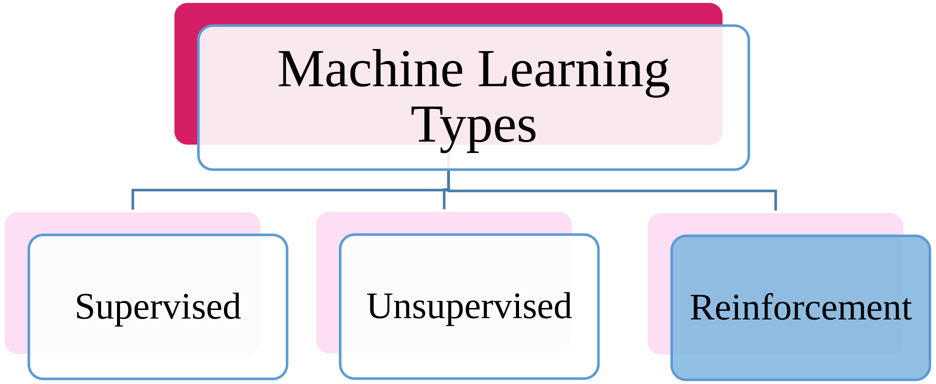
- Simple ***feedback*** is required for the agent to learn its behavior.
- Since the information about the current state of the environment typically also includes a so-called ***reward*** signal, we can think of reinforcement learning as a field related to **supervised learning**.
- For instance, consider how a baby learns to walk. Unfortunately, **RL** is beyond the scope of this course.



Introduction
Machine Learning Types

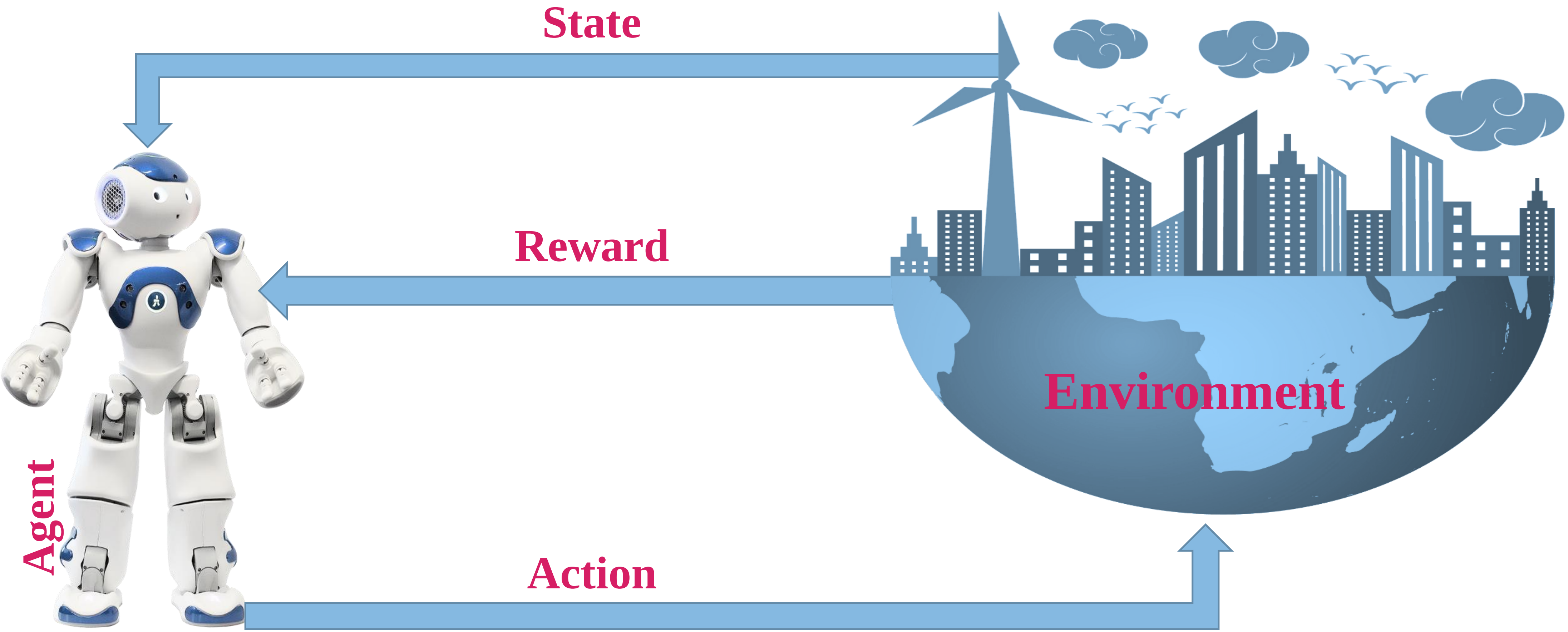
❑Reinforcement learning (RL):

- In reinforcement learning, this ***feedback*** is not the correct ground truth label or value, but a measure of how well the action was measured by a reward function.
- Through the **interaction** with the **environment**, an agent can then use reinforcement learning to learn a series of actions that maximizes this reward via an exploratory **trial-and-error** approach or deliberative planning.



Introduction
Machine Learning Types

- **Reinforcement learning (RL):**



Introduction
Machine Learning Types

- **Example(1.1):** A popular example of reinforcement learning is a chess engine. The agent decides upon a series of moves depending on the state of the board (the environment), and the reward can be defined as *win* or *lose* at the end of the game.

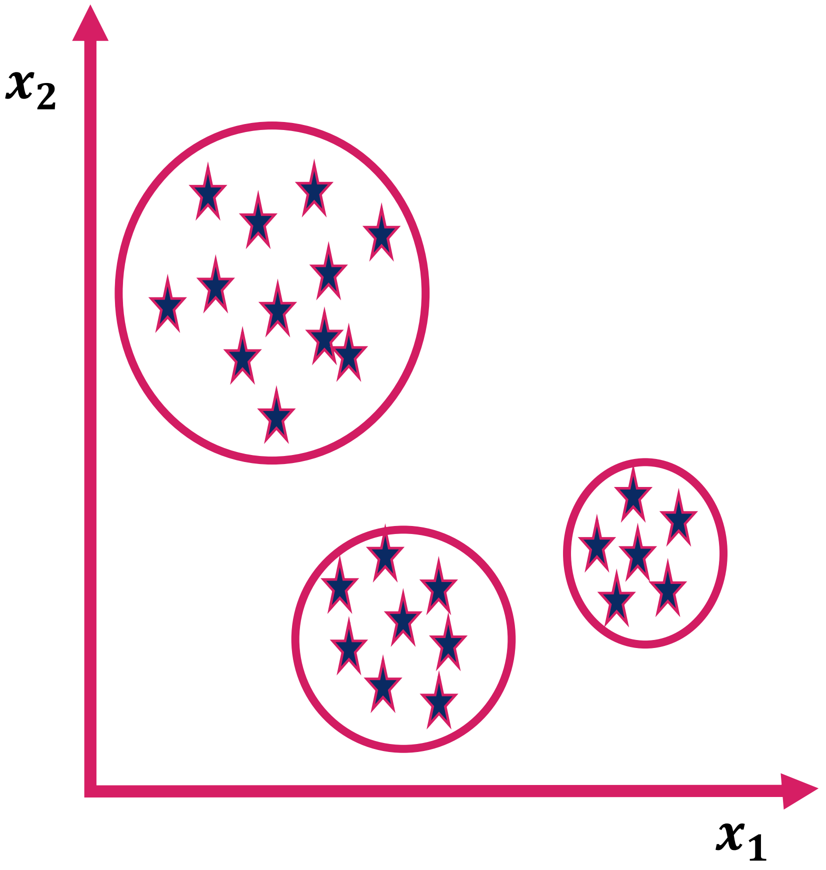
Introduction
Unsupervised Learning

- Unsupervised problems types
  - Clustering problems
  - Dimensionality reductions

Introduction
Unsupervised Problems Types

Clustering problems

- ***Clustering*** *is* ***an exploratory*** *data analysis technique that* ***allows*** *us to* ***organize*** pile of information into meaningful subgroups *(clusters)* ***without having any prior* knowledge** of their group memberships.
- Each cluster that may arise during the analysis defines a group of objects that share a certain degree of similarity but are more dissimilar to objects in other clusters, which is why clustering is also sometimes called "**unsupervised classification**“**.**
- Clustering is a great technique for **structuring information** and **deriving meaningful relationships among data**,



Introduction
Unsupervised Problems Types

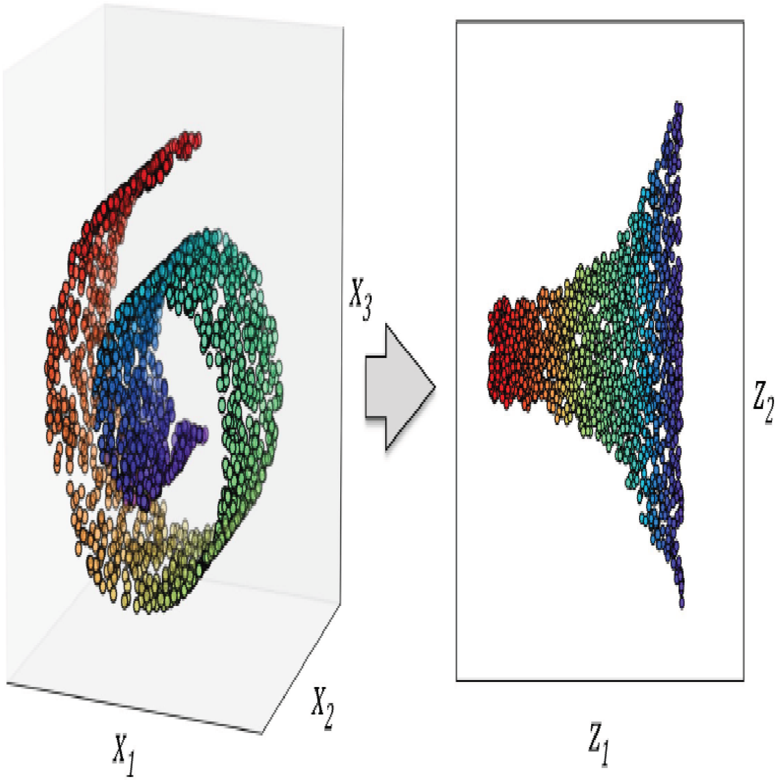
❑ Clustering problems

- ***For example***, it allows marketers to discover customer groups based on their interests in order to develop distinct marketing programs.

Introduction
Unsupervised Problems Types

Dimensionality reductions

- *Often we are* working with data of **high dimensionality** (*i.e. each observation comes with a high number of measurements*) that can present a challenge for limited storage space and the computational performance of machine learning algorithms.
- Unsupervised dimensionality reduction is a commonly used approach in **feature preprocessing** to **remove noise** from data, which can also degrade the predictive performance of certain algorithms, and **compress the data onto a smaller dimensional subspace while retaining most of the relevant information.**

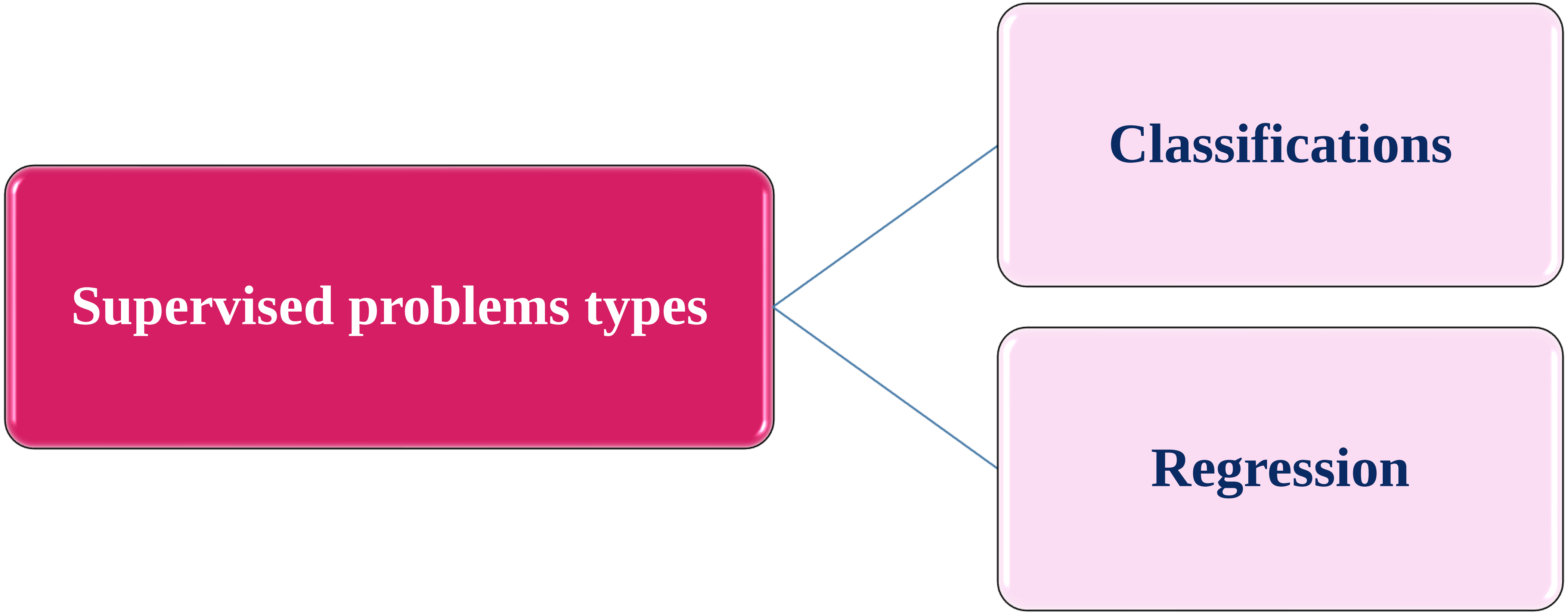


Introduction
Unsupervised Problems Types

❑ Dimensionality reductions

- Dimensionality reduction can also be useful for **visualizing data**.
- **For example**, a high-dimensional feature set can be projected onto one-, two-, or three-dimensional feature spaces in order to visualize it via 3D- or 2D-scatterplotsor histograms.

Introduction
Supervised Learning



Introduction
Supervised Problems Types

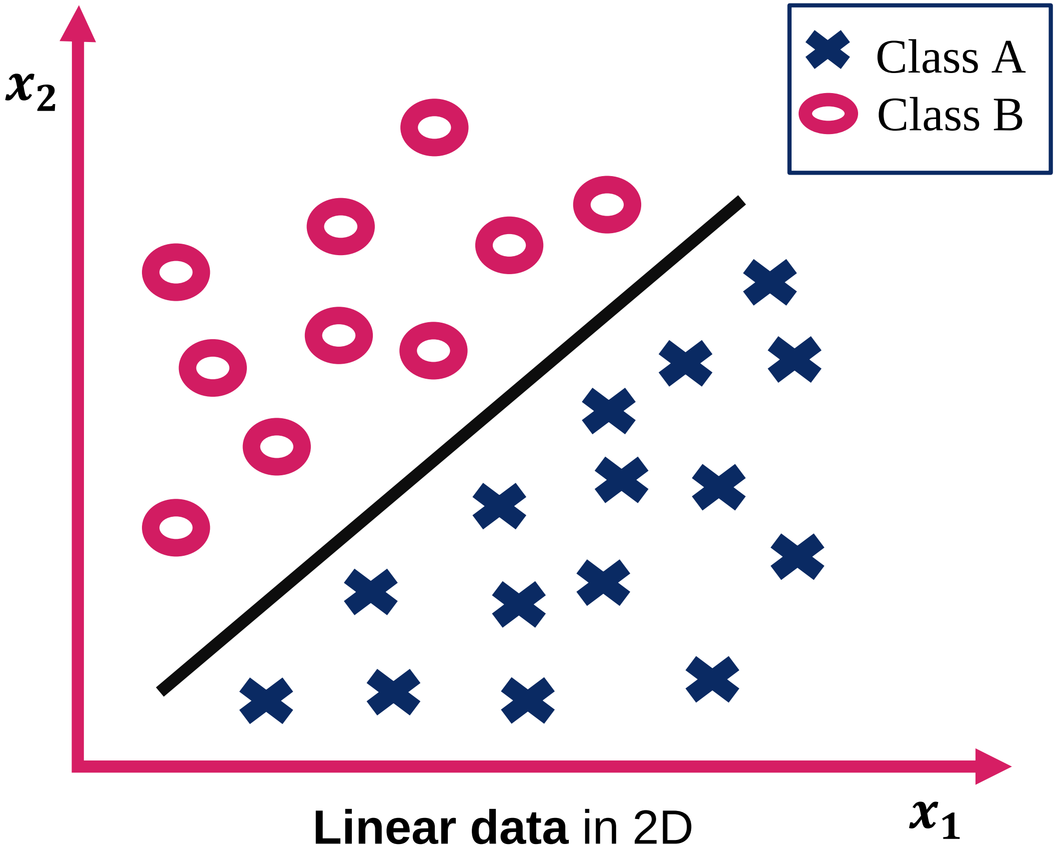
❑ Supervised Problem Types:

- The problem may be classification or regression problem.
- The problem type determine by the **response** variable (target/ outcome variable).
- If the outcome variable is a **quantitative** variable, the problem will be **regression**, for instance: wind speed prediction.
- If the outcome variable is a **qualitative** variable, the problem will be **classification**, for instance: cancers detection; Brest, Prostate or Leukemia .

Introduction
Supervised Problems Types

❑ Classification problems

- Classification is a subcategory of supervised learning where the goal is to **predict** the categorical class labels of new instances based on past observations.
- Those class labels are **discrete, unordered** values that can be understood as the ***group memberships* of the instances.**



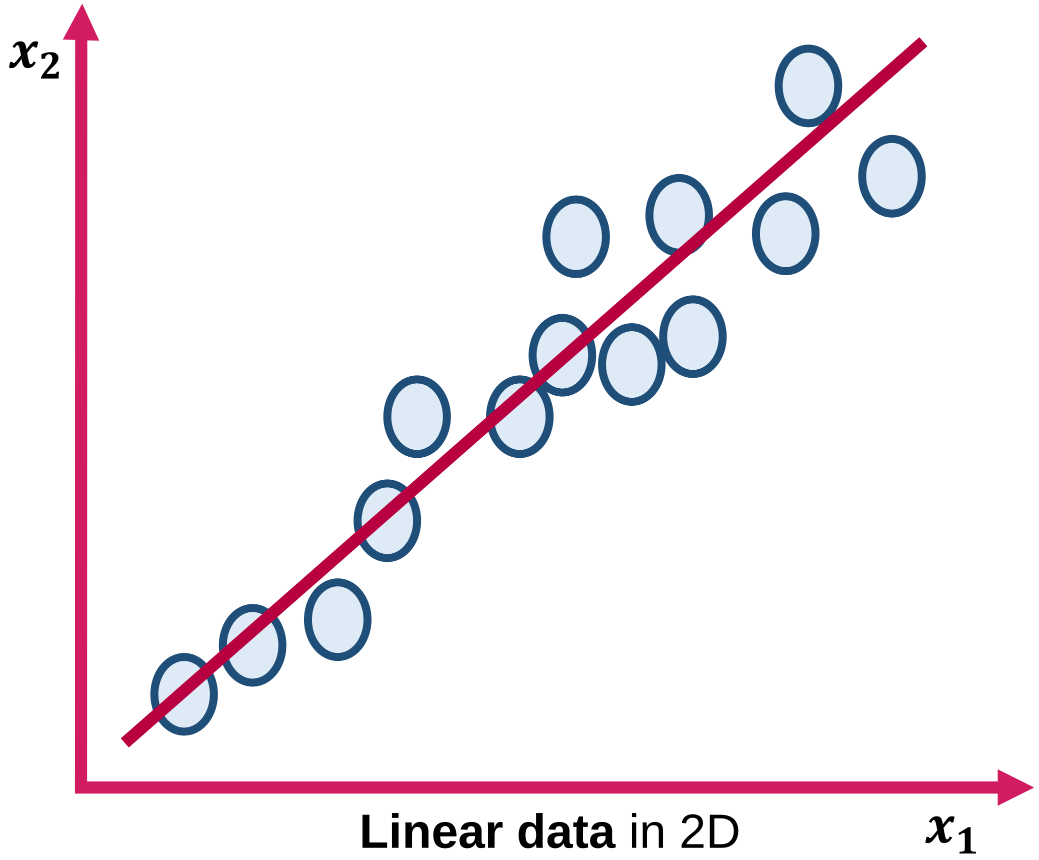
Linear data in 2D
(Discrete Data)

Introduction

Supervised Problems Types

Regression problems

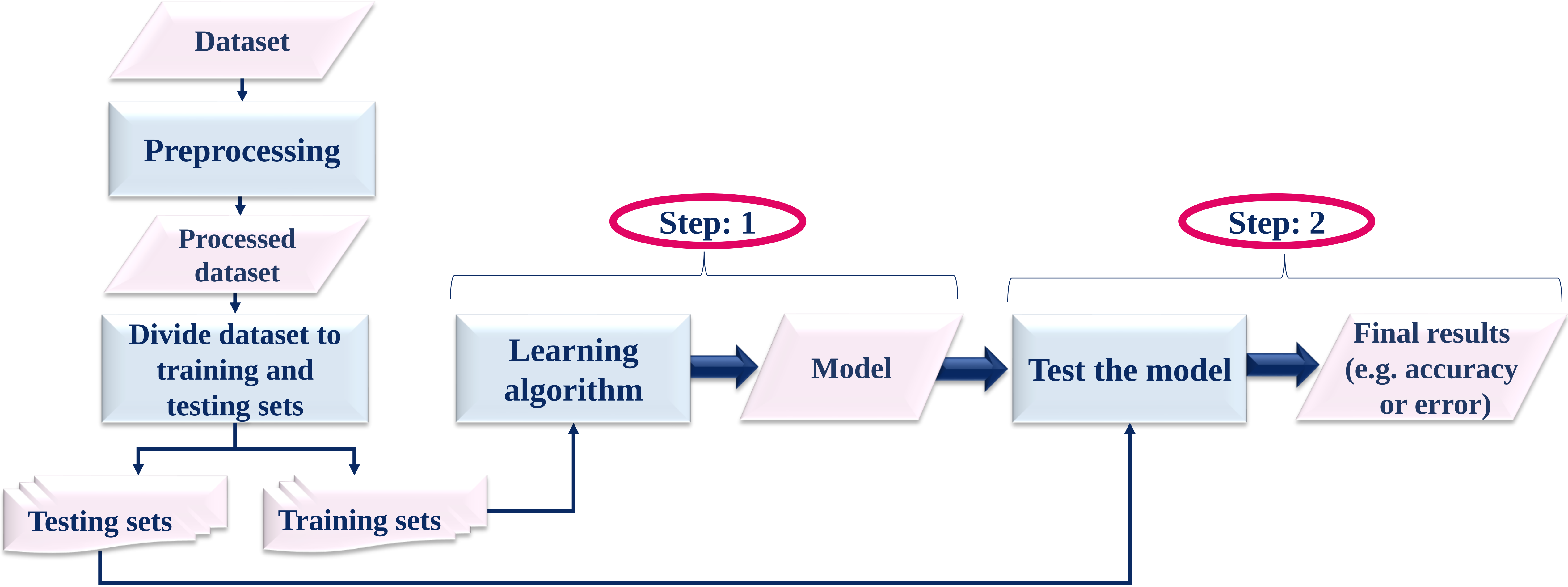
- In **regression problems**, we are given a number of **predictor** (features) variables and a continuous response variable (outcome), and we try to find a **relationship** between those variables that allows us to predict an outcome.



Linear data in 2D
(Continues Data)

Introduction
Supervised Learning

- **Supervised learning process:**



Introduction
Supervised Learning

- **Supervised learning process:**
  - **Step 1: Training**
    - Learn a model using the training dataset.
  - **Step 2: Testing**
    - Test the model using **unseen data** (testing set) to assess the model performance (e.g.g. accuracy or error)

Introduction
Supervised Learning

Example (1.2): Handwritten Digit Recognition

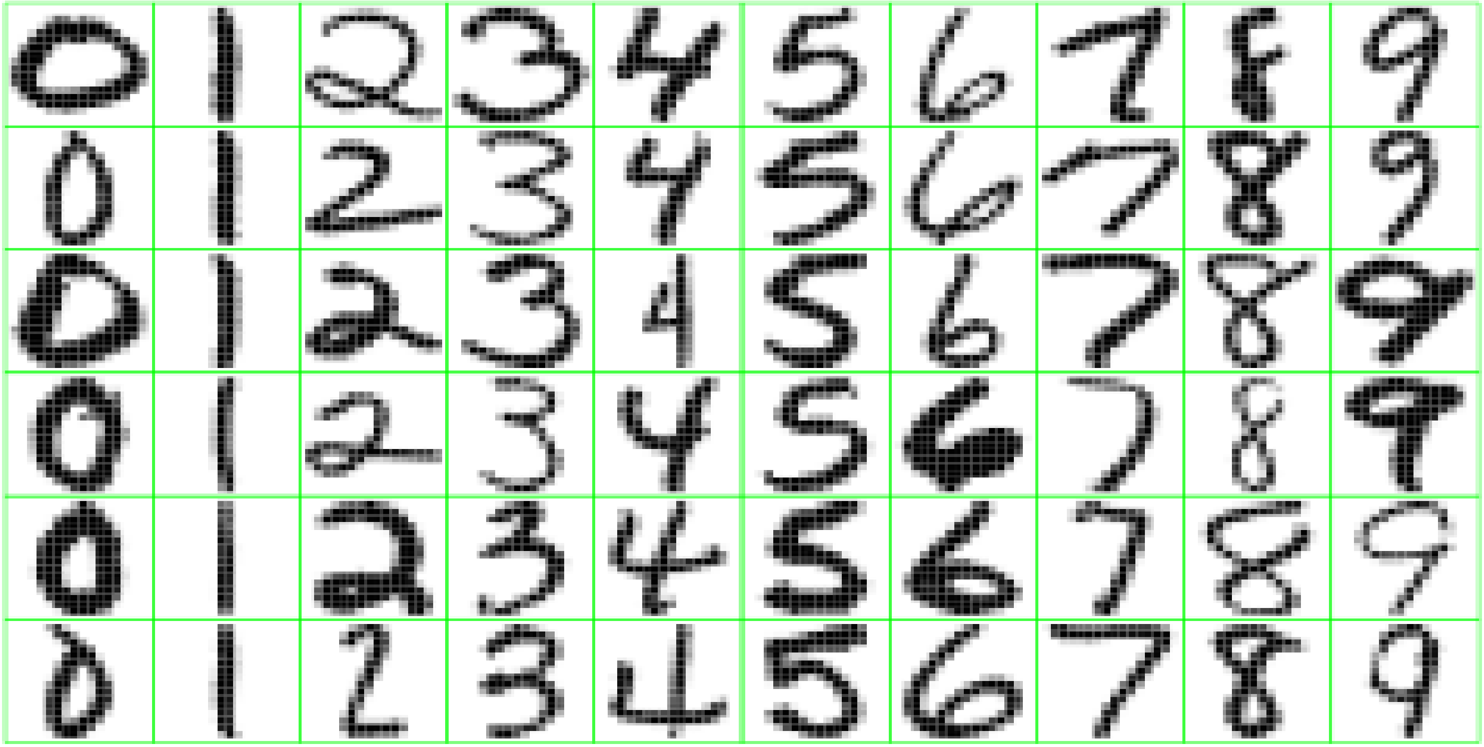


Figure (1.3) Examples of handwritten digits from U.S. postal envelopes.

Introduction
Supervised Learning

Example (1.2): Handwritten Digit Recognition

- The task is to predict, from the 16 × 16 matrix of pixel intensities, the identity of each image (0, 1, . . . , 9) quickly and accurately. If it is accurate enough, the resulting algorithm would be used as part of an automatic sorting procedure for envelopes. This is a classification .
- In order to achieve this low error rate, some objects can be assigned to a “don’t know” category, and sorted instead by hand.

Introduction

Supervised Vs. Unsupervised Learning

Variables types	Machine learning types	
	Supervised learning	Unsupervised learning
Discrete	Classification or categorization	Clustering
Continuous	Regression	Dimensionality reduction

Introduction

A roadmap for building machine learning systems

Raw Dataset

Preprocessing

- Data cleaning
- Data scaling
- Dimensionality reduction
- Divide dataset to training and testing sets

Processed dataset

Learning algorithm

- Model selection
- Cross validation
- Performance metrics
- Meta-parameters optimization

Prediction model

Evaluation

Used to evaluate the final prediction model by using testing sets.

Final results

Prediction

Used the final prediction model to predict a new data by using unseen dataset

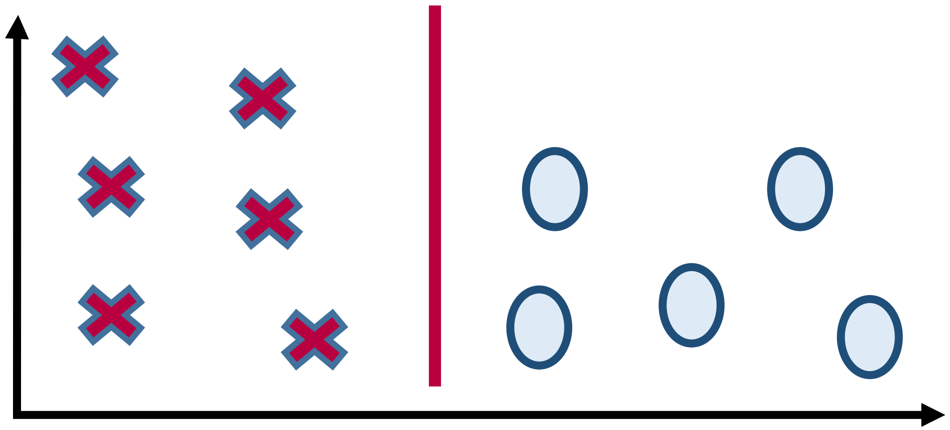
Introduction
Linear separability

Features separation types

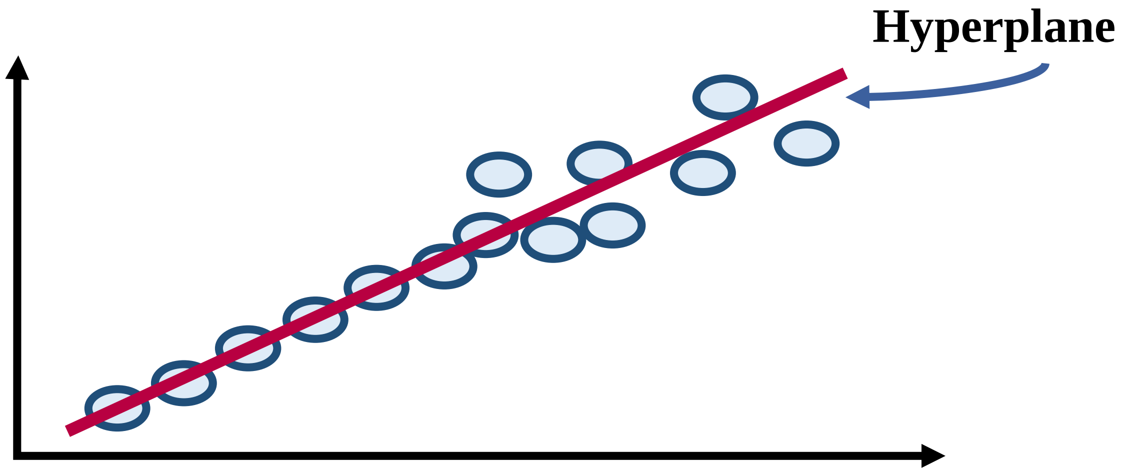
- **Linear separable features**
- **Nonlinear separable features**

Introduction
Linear separability

- **Linear separable features** that can accurately fit/classify by any linear prediction/classification model. Again, These features are can be perfectly separated by a hyperplane.



Linear data in 2D
(Discrete Data)

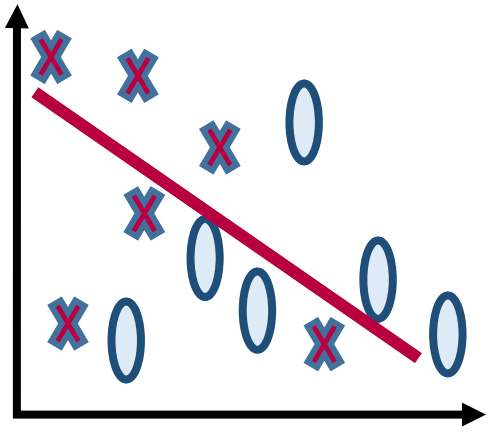


Linear data in 2D
(Continues Data)

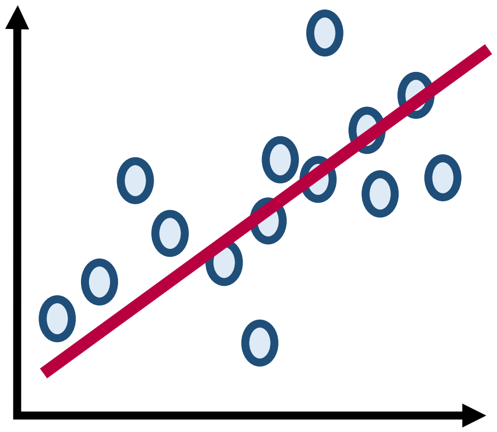
Introduction

Linear separability

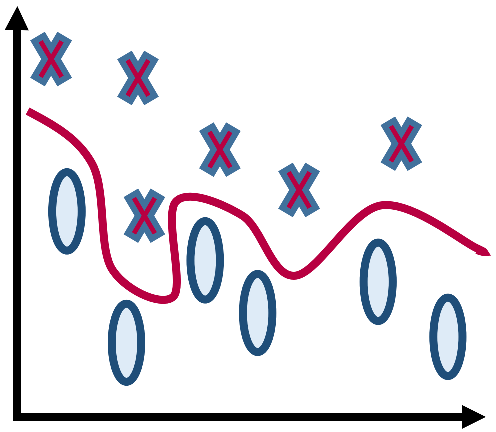
- **Nonlinear separable features** that can not accurately fit/classify by any linear prediction/classification model. Again, These features are can not be perfectly separated by a hyperplane.



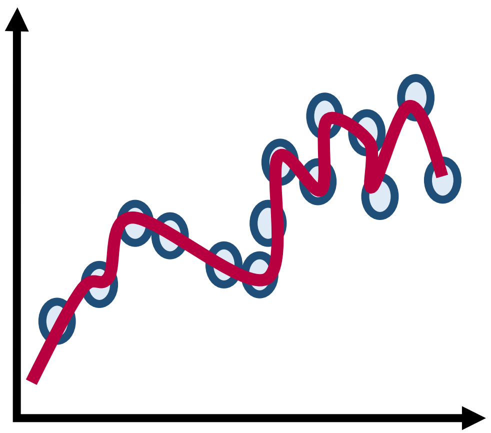
Non-Linear data in 2D
(Discrete Data)



Non-Linear data in 2D
(Continuous Data)



Non-Linear data in 2D
(Discrete Data)



Non-Linear data in 2D
(Continuous Data)

Introduction
Generalization

- **Generalization** means the ability of a machine learning algorithm to perform well on previously unobserved inputs (i.e. unseen inputs—not just those on which our model was trained).

Introduction

Training error vs. generalization error

- **Training error** is defined as the expected value of the error on the training set and we reduce this error.
- **The generalization error (test error)** is defined as the expected value of the error on a new input. We typically estimate the generalization error of a machine learning model by measuring its performance on a test set of examples that were collected separately from the training set.

Introduction

Training error vs. generalization error

- **The factors determining how well a machine learning algorithm will perform are its ability to:**
    1. Make the training error small.
    2. Make the gap between training and test error small.
- These two factors correspond to the two central challenges in machine learning: **underfitting and overfitting**

Introduction
Overtraining vs. underfitting

- **Overfitting** occurs when a machine learning algorithm capture the noise of the data (i.e. fits/classify the data too well), that occurs when the gap between the training error and test error is too large.
- **Underfitting** occurs when the model is not able to obtain a sufficiently low error value on the training set.
- We can control whether a model is more likely to overfit or underfit by altering **its capacity.**

Introduction
Overtraining vs. underfitting

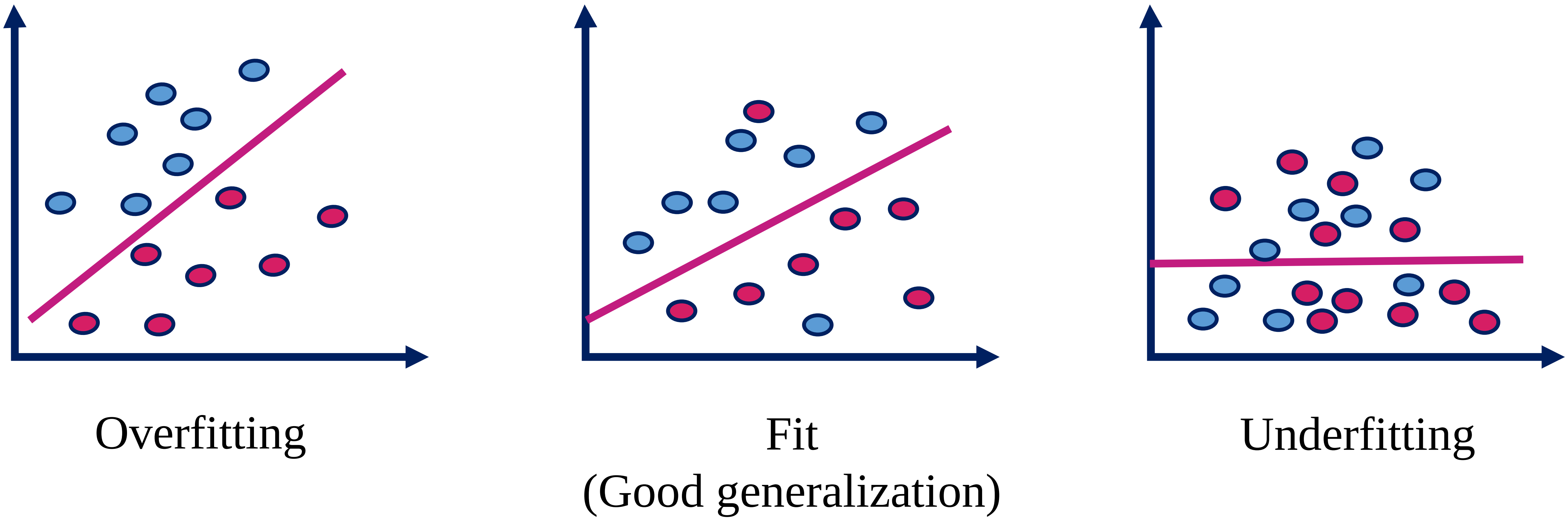


Figure (1.4): Illustration of Overfitting vs. underfitting using discrete samples

Introduction

Overtraining vs. underfitting

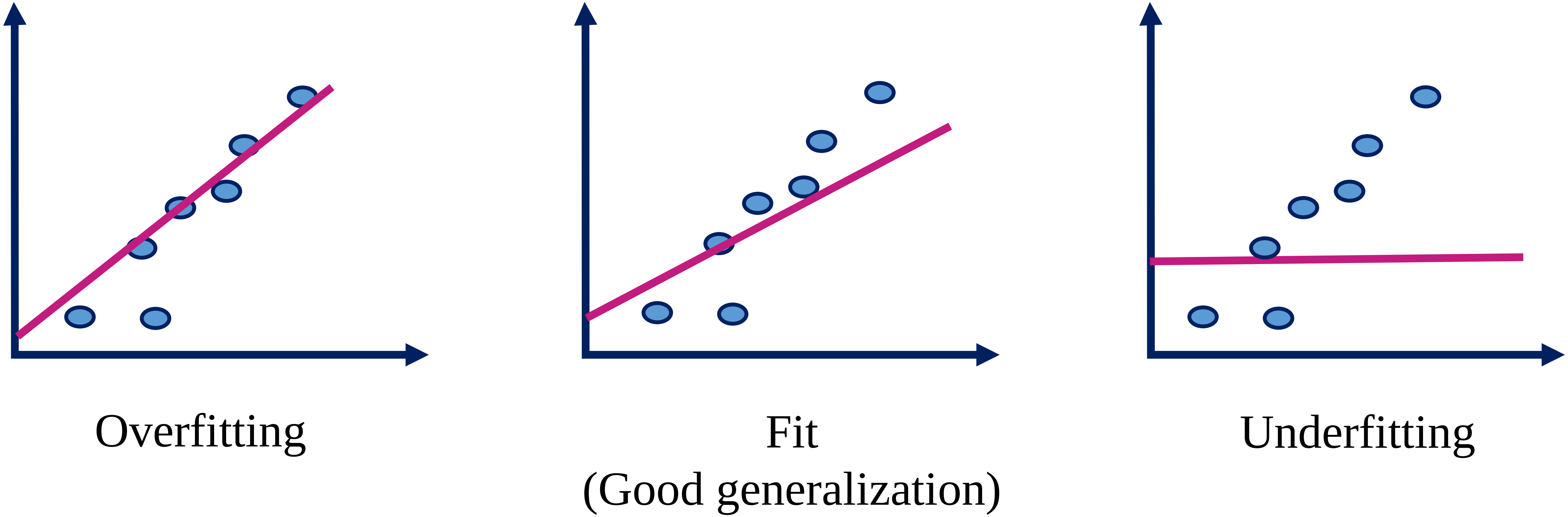


Figure (1.5): Illustration of Overfitting vs. underfitting using continuous samples

Introduction
Regularization

- **Regularization** is any modification we make to a learning algorithm that is intended to reduce its generalization error but not its training error.”

Introduction

Model capacity

- **Model capacity** is its ability to fit a wide variety of functions. Models with low capacity may struggle to fit the training set. Models with high capacity can overfit by memorizing properties of the training set that do not serve them well on the test set.
- One way to control the capacity of a learning algorithm is by choosing its **hypothesis space.**

Introduction

Model capacity and Hypothesis space

- **Hypothesis space is** the set of functions that the learning algorithm is allowed to select as being the solution.
- **For example**, the linear regression algorithm has the set of all linear functions of its input as its **hypothesis space**. We can **generalize linear regression to include polynomials**, rather than just linear functions, in its hypothesis space. **Doing so increases the model's capacity.**

Introduction

Model capacity and Hypothesis space

- Machine learning algorithms will generally **perform best when their capacity is appropriate for the true complexity of the task they need to perform and the amount of training data they are provided with.**
- Models with insufficient capacity are unable to solve complex tasks.
- Models with high capacity can solve complex tasks, but when their capacity is higher than needed to solve the present task they may **overfit.**

Introduction
Changing a model capacity

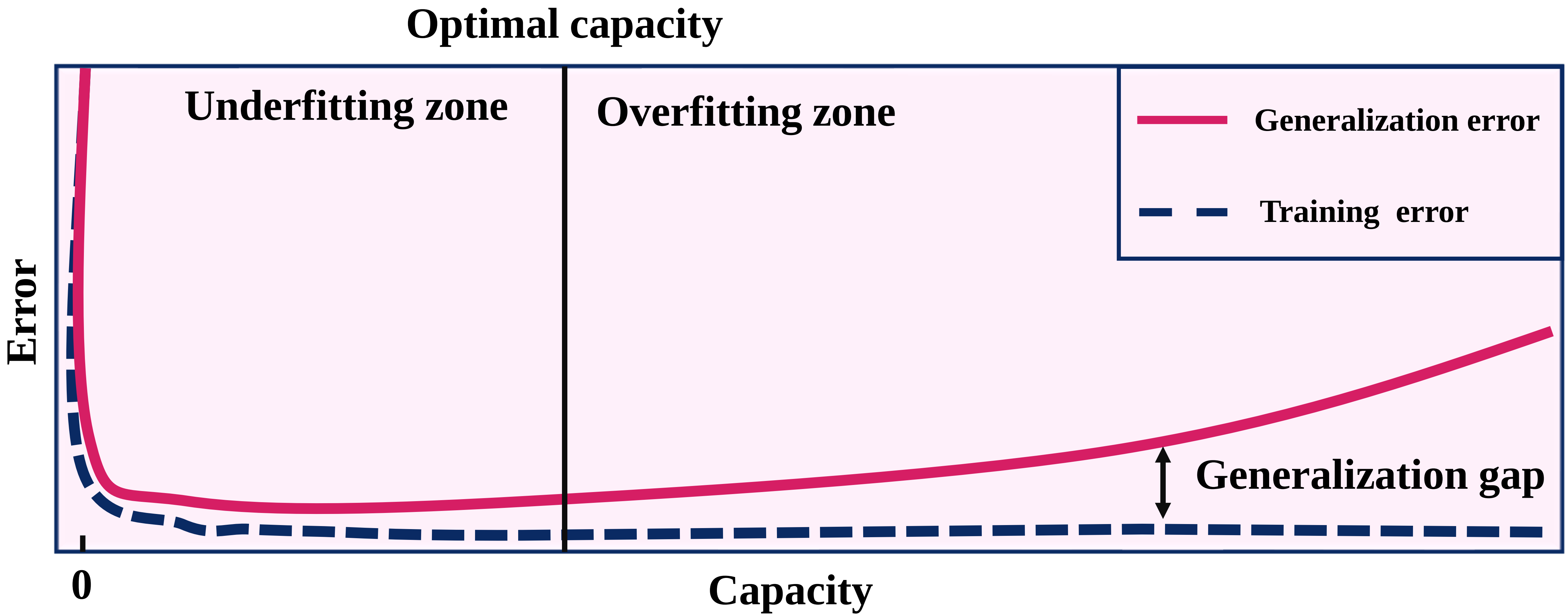
- **There are many ways of changing a model’s capacity one of these ways occurs** by changing the number of input features it has, and simultaneously adding new parameters associated with those features.
- There are in fact many ways of changing a model’s capacity. Capacity is not determined only by the choice of model.
- **Representational capacity of the model means** the model specifies which family of functions the learning algorithm can choose from when varying the parameters in order to reduce a training objective.

Introduction

Changing a model capacity

- In many cases, finding the best function within this family is a very difficult optimization problem.
- In practice, the learning algorithm does not actually find the best function, but merely one that significantly reduces the training error. These additional limitations, such as the imperfection of the **optimization algorithm**, mean that the learning algorithm's effective capacity may be less than the **representational capacity** of the model family.

Introduction
Changing a model capacity



Figure(1.5): Illustrate the relationship between capacity and error.

References

- T. Hastie, R. Tibshirani and J. Fridmn, “The Elements of Statitical Learning”, second edition , Springer
- Raschka, Sebastian. “Python machine learning”. Packt Publishing Ltd, 2015.
- Kriesel, David. "A Brief Introduction to Neural Networks. 2007." URL http://www. dkriesel. com (2007).
- Ian Goodfellow, Yoshua Bengio Aaron Courville, “Deep Learning “
- David H Wolpert and William G Macready. “No free lunch theorems for optimization”. IEEE transactions on evolutionary computation, 1(1):67{82, 1997.

Any Questions!?

Thank you