MLAI 504
NEURAL NETWORKS & DEEP LEARNING

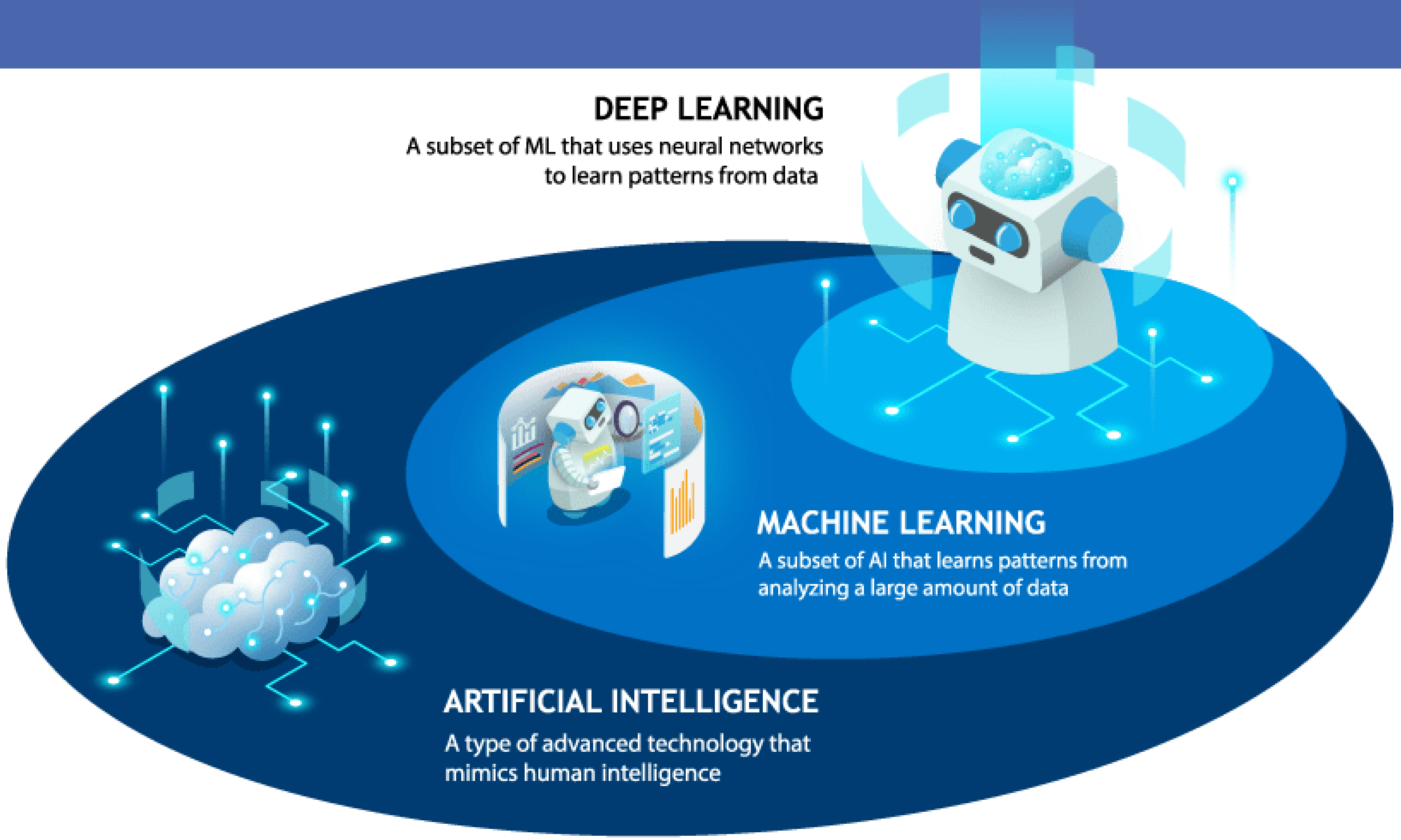
Dr. Zein Al Abidin IBRAHIM

zein.ibrahim@liu.edu.lb

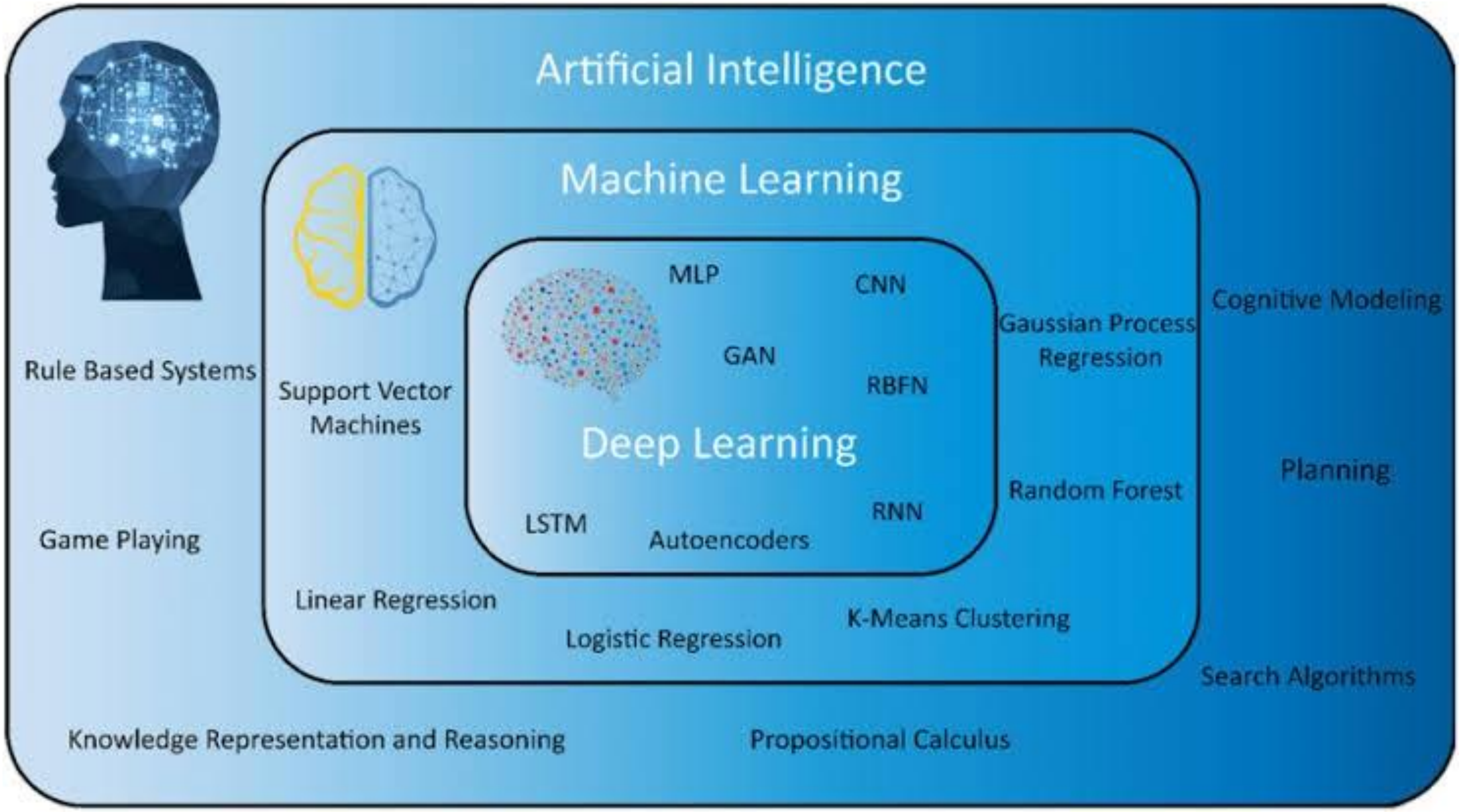
MLAI – 2023/2024

INTRODUCTION

Machine Learning


DEEP LEARNING
A subset of ML that uses neural networks
to learn patterns from data
MACHINE LEARNING
A subset of AI that learns patterns from
analyzing a large amount of data
ARTIFICIAL INTELLIGENCE
A type of advanced technology that
mimics human intelligence

AI, ML & DL


Artificial Intelligence
Machine Learning
Deep Learning
MLP
CNN
GAN
RBFN
LSTM
Autoencoders
RNN
Rule Based Systems
Support Vector Machines
Gaussian Process Regression
Cognitive Modeling
Random Forest
Planning
Game Playing
Linear Regression
Logistic Regression
K-Means Clustering
Search Algorithms
Knowledge Representation and Reasoning
Propositional Calculus

AI, ML & DL

- Machine learning → a part of artificial intelligence
- Ability of computers to learn from data in order to extract the algorithm
- Used to detect some patterns in data
- Ex: learn from several spam emails in order to make an algorithm to classify later emails into spam or no

What is Machine Learning ?

- Machine learning is about learning to do better in the future based on what was experienced in the past.

- Consists of designing efficient and accurate prediction *algorithms*

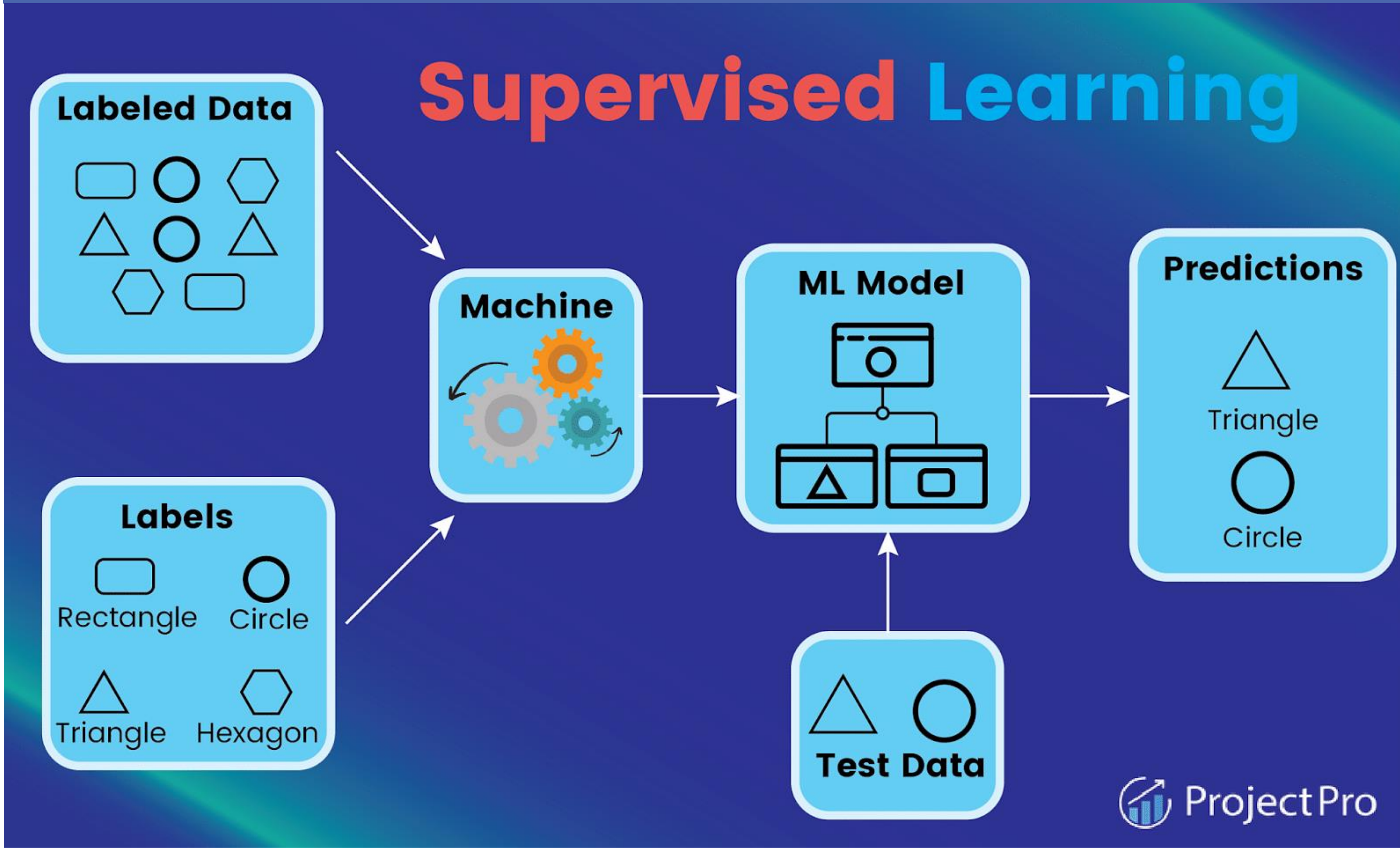
What is Machine Learning ?

- Main types:
  - Supervised Learning or Classification
  - Unsupervised Learning or Clustering
  - Semi-supervised Learning
  - Reinforcement Learning
- Other types used nowadays
  - Deep Learning
  - Transfer Learning
  - Online Learning
  - …

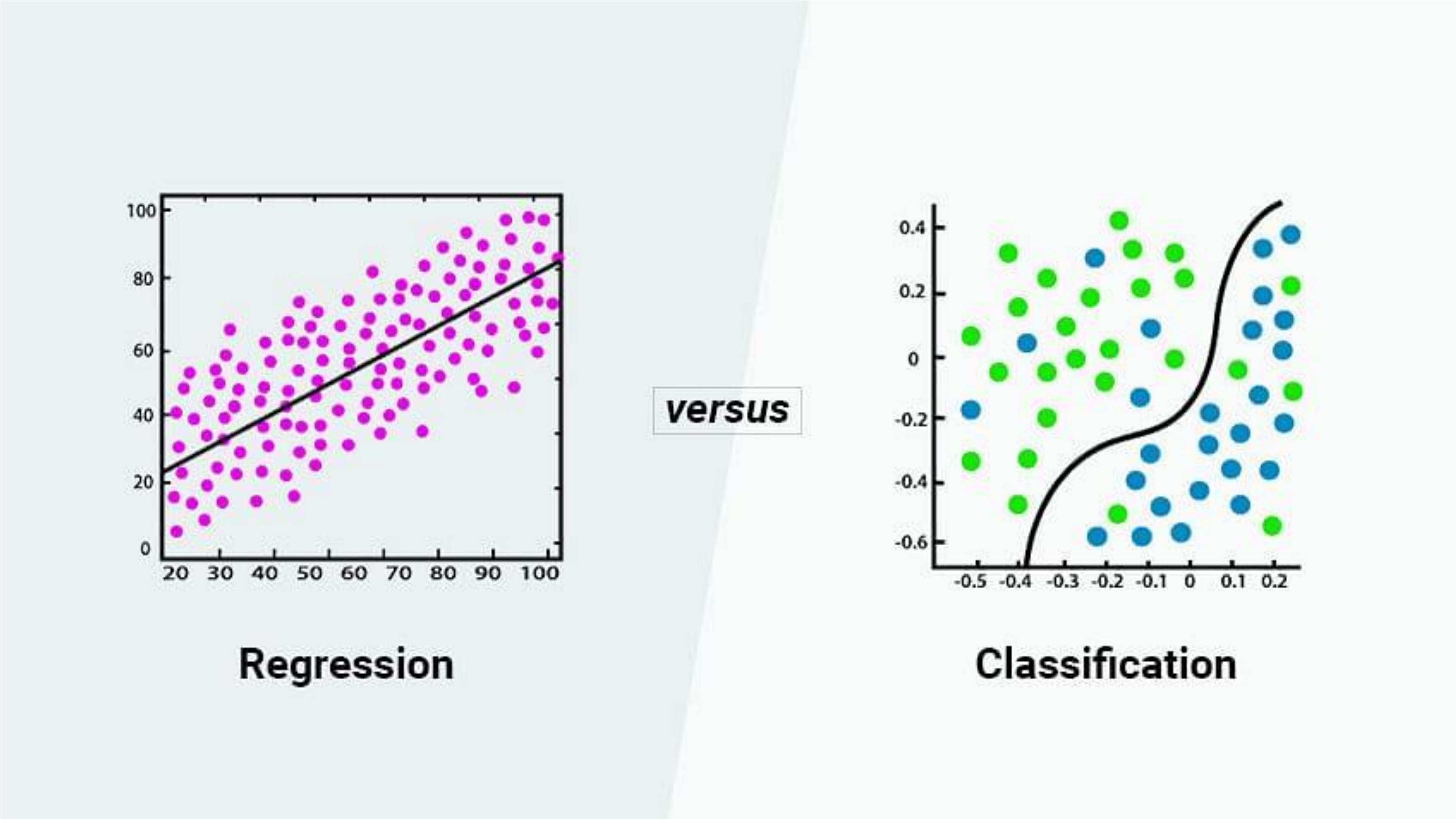
Types of Machine Learning Methods

- In this type of learning, a labeled dataset is used to train the machine learning model.
- The model learns from the labeled data and can make predictions on new, unseen data.
- Two main types here: Classification or regression
- Examples of supervised learning algorithms:
  - Linear regression, logistic regression, decision trees, random forests, neural networks, SVM, ….
- Useful when you have labeled dataset

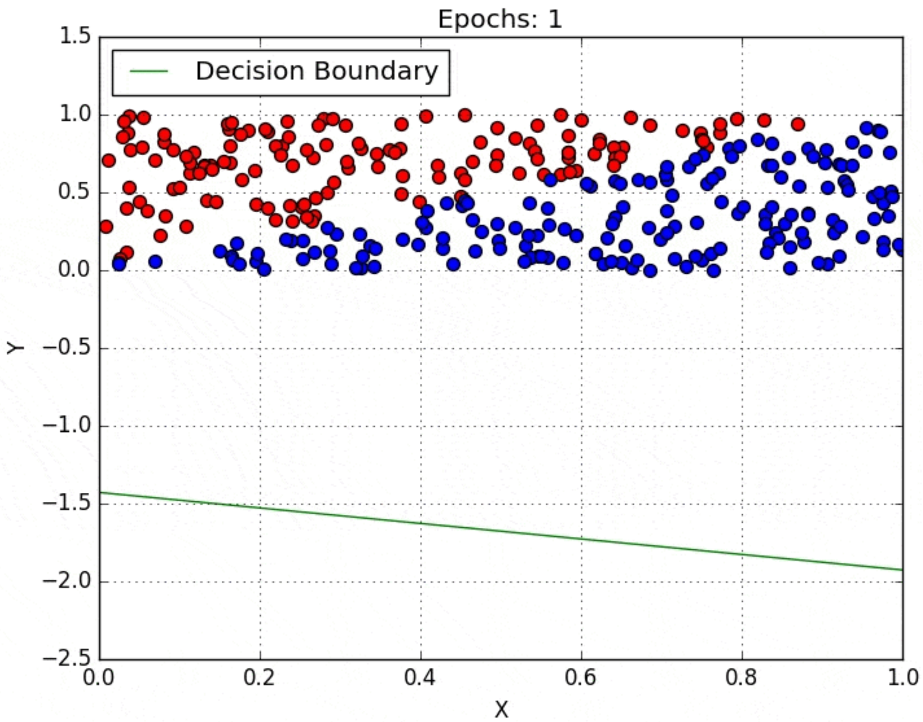
Supervised Learning: Classification

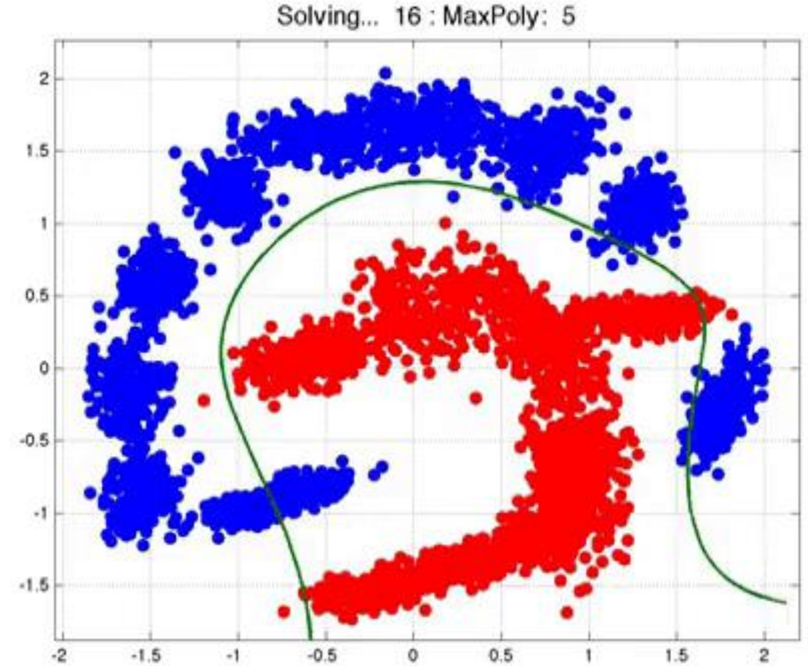


Supervised Learning: Classification

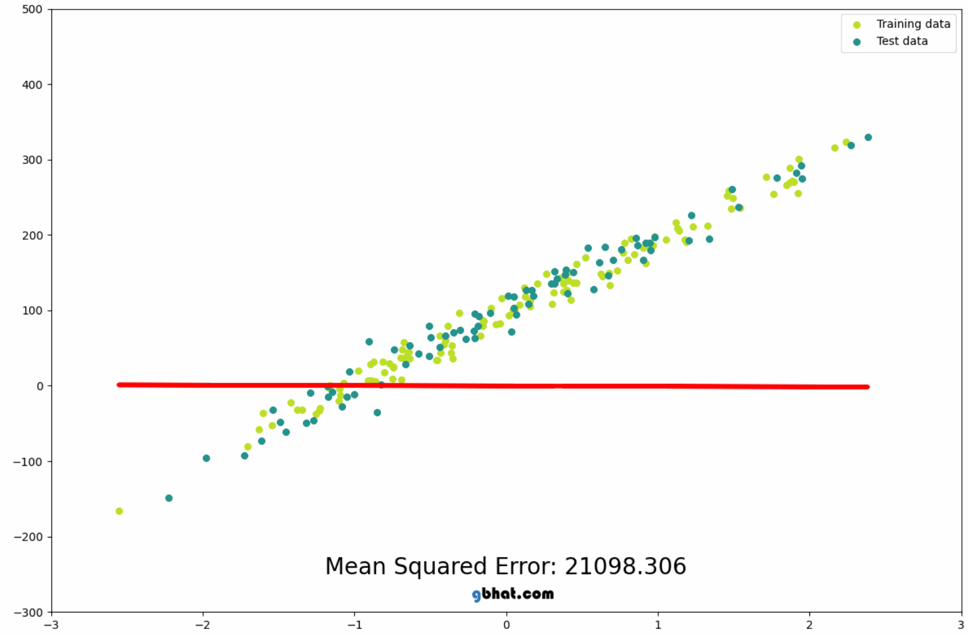


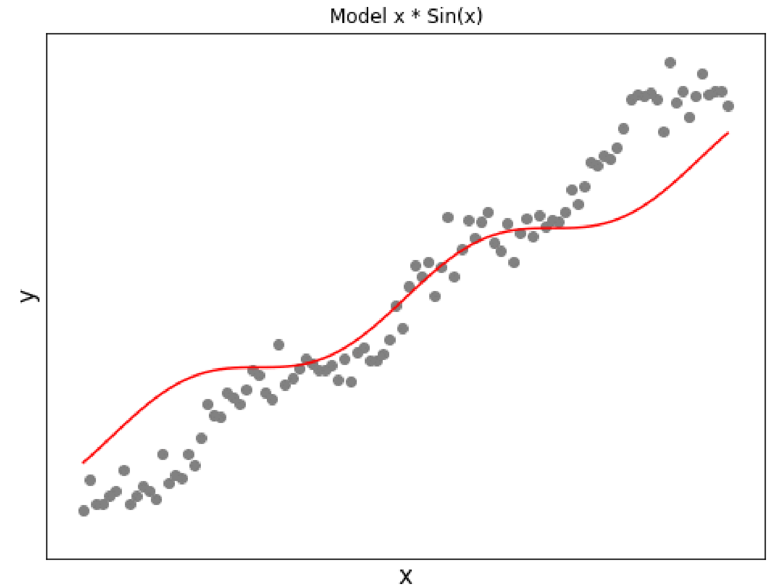
Classification vs Regression

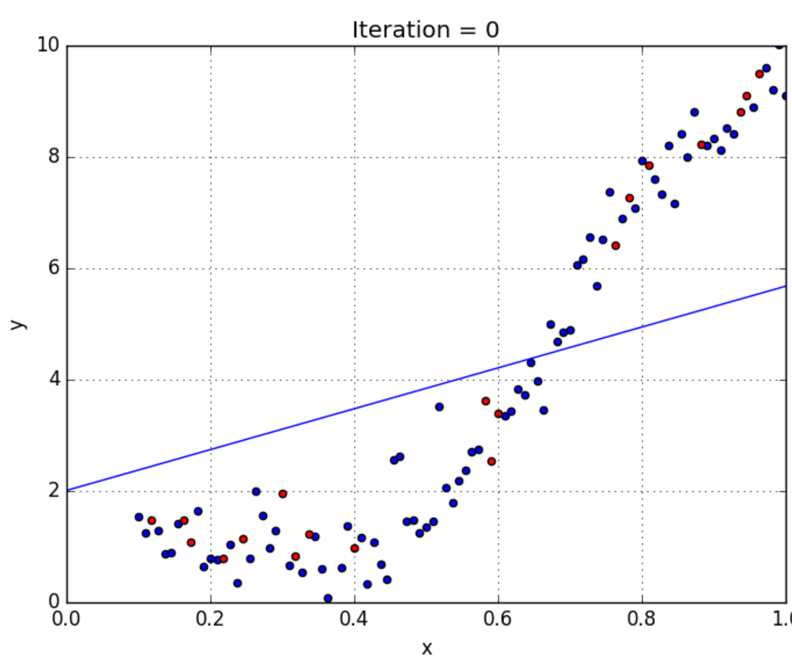




Classification vs Regression



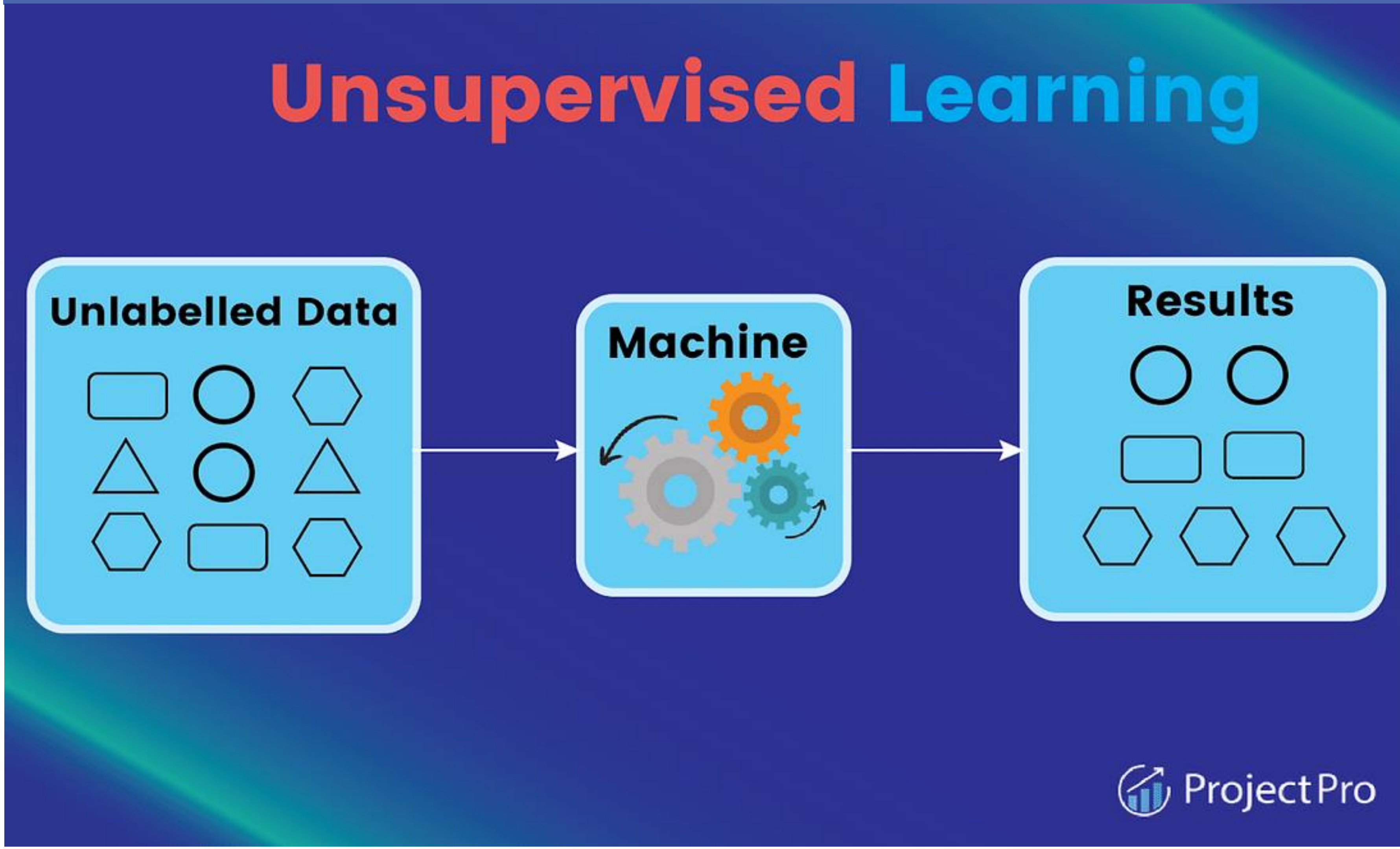




Classifi ssion

- In unsupervised learning, the machine learning model is given an unlabeled dataset and must find patterns or relationships in the data on its own.
- Clustering and dimensionality reduction are common unsupervised learning techniques.
- Examples of unsupervised learning algorithms:
  - K-Means, Hierarchical, DBSCAN, Spectral Clustering, Fuzzy Clustering, ….
- Useful when it is not possible to obtain labeled data to train a model

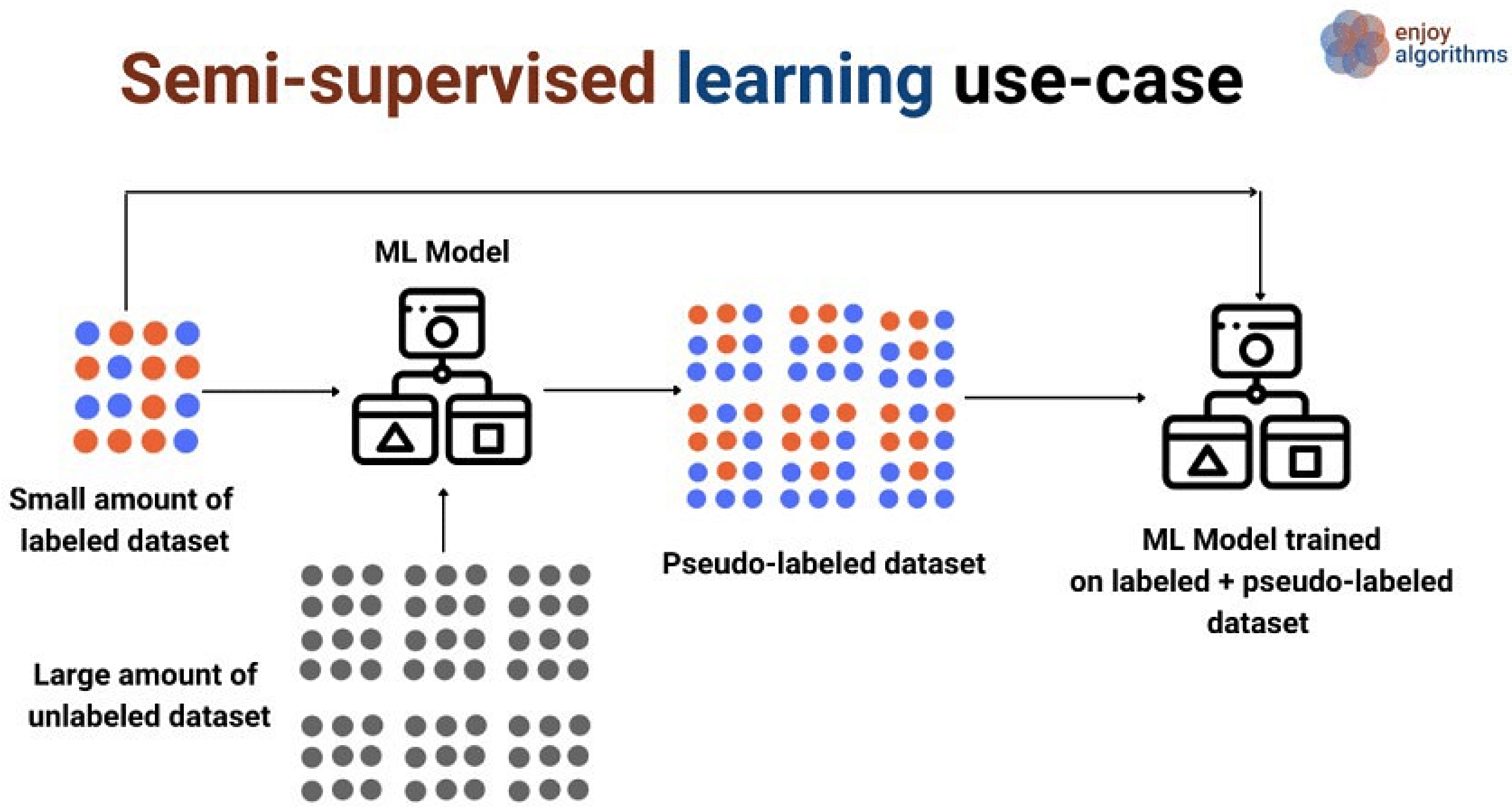
Unsupervised Learning: Clustering



Unsupervised Learning: Clustering

- Semi-supervised learning is a combination of supervised and unsupervised learning.
- In this type of learning, the machine learning model is given a dataset that is partially labeled and partially unlabeled.
- The model learns from both the labeled and unlabeled data to make predictions on new data.
- Useful when labelled data is scarce or expensive to obtain, as they can leverage the large amounts of available unlabelled data to improve the performance of a model

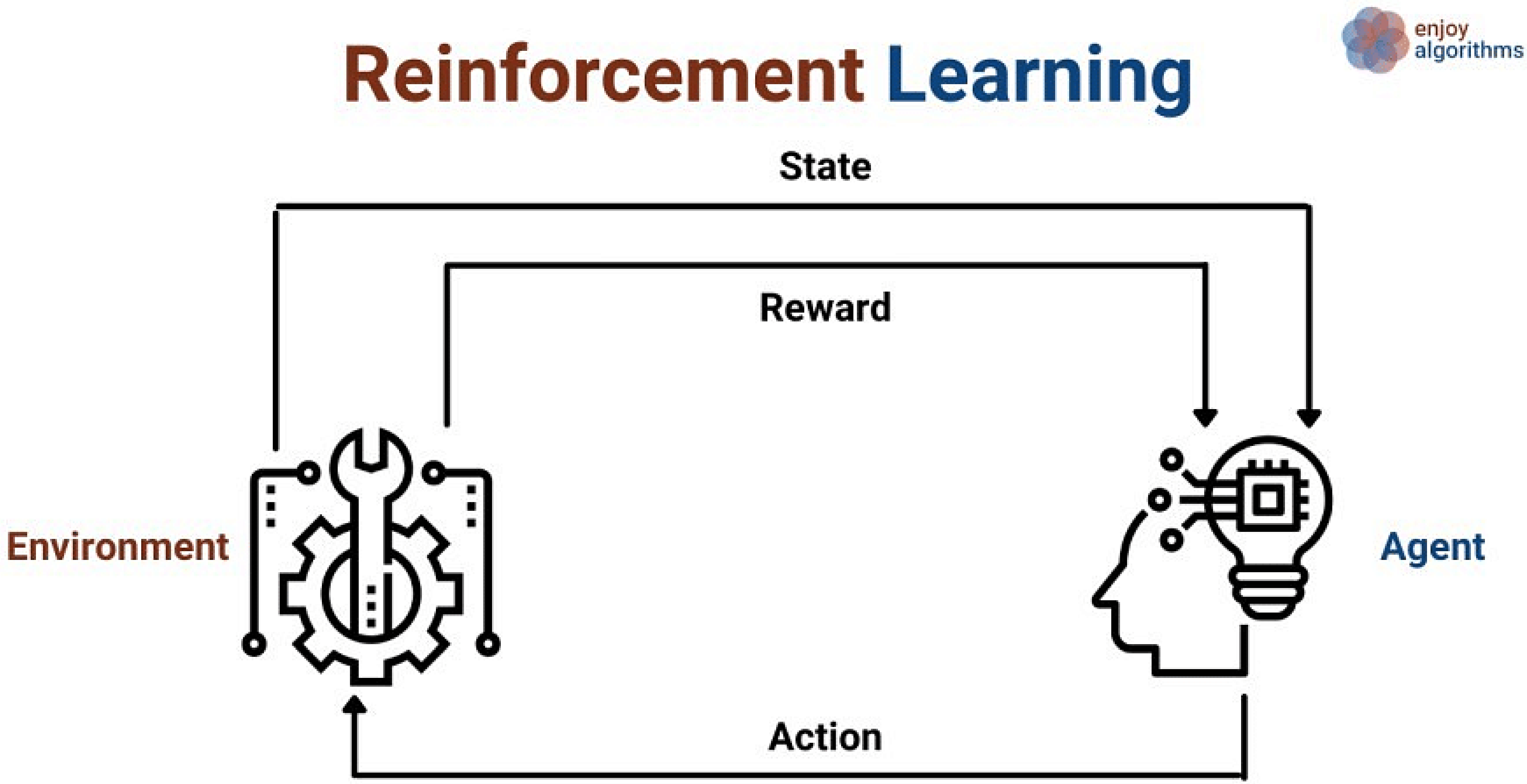
Semi-supervised Learning



Semi-supervised Learning

- Reinforcement learning is a type of machine learning where an agent learns to make decisions by interacting with an environment.
- The agent receives rewards or punishments based on its actions and learns to make better decisions over time.
- Many applications, including game playing, robotics, and autonomous vehicle control use this type of learning.

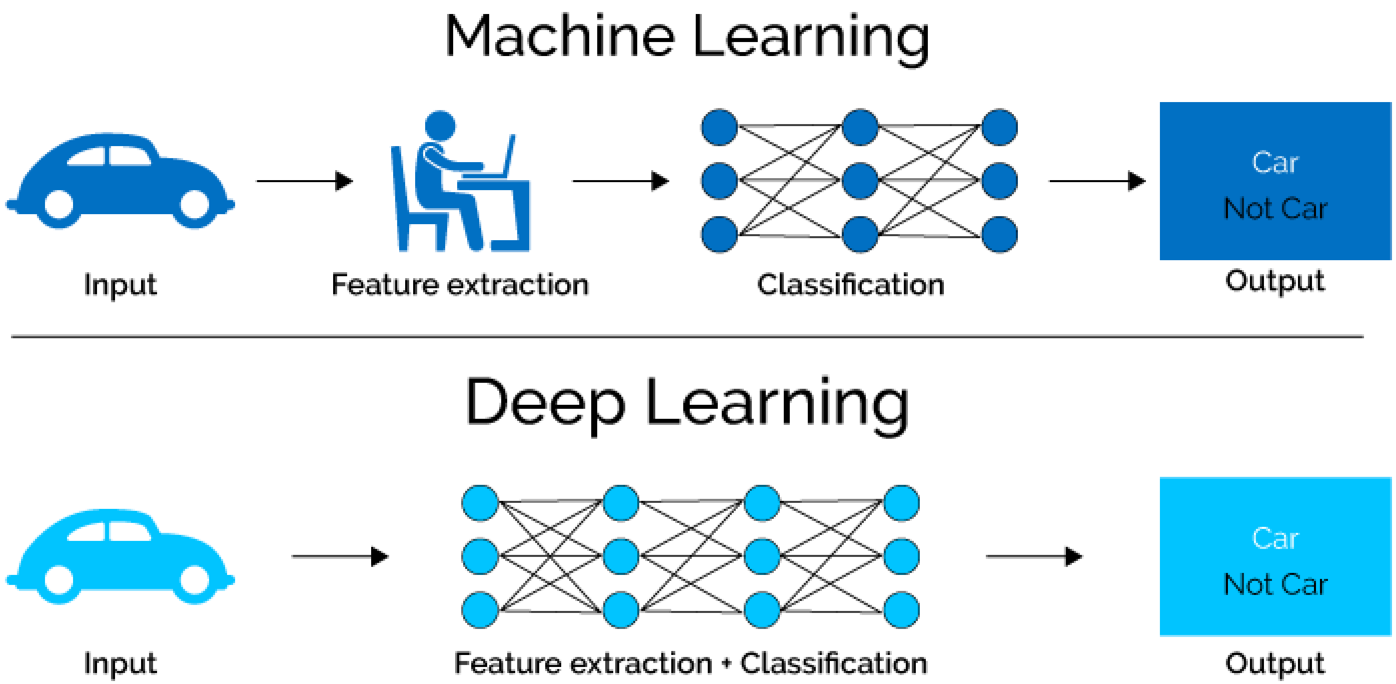
Reinforcement Learning



Reinforcement Learning

- Deep learning is a subset of machine learning that uses neural networks with many layers to model complex relationships in data.
- Deep learning is commonly used in image recognition, natural language processing, and speech recognition.
- Among the well-Known Networks:
  - **Convolutional Neural Networks (CNNs)** → Used for image and video analysis
  - **Recurrent Neural Networks (RNNs)** → used for sequence analysis, such as speech recognition and natural language processing
  - **Long Short-Term Memory (LSTM)** → Like RNNs but has the ability to selectively remember or forget information over time, making them particularly useful for long-term dependencies.
  - **Generative Adversarial Networks (GANs)** → used for generating synthetic data, such as images, video, or text.
  - **Autoencoders** → used for unsupervised feature learning and data compression.

Deep Learning

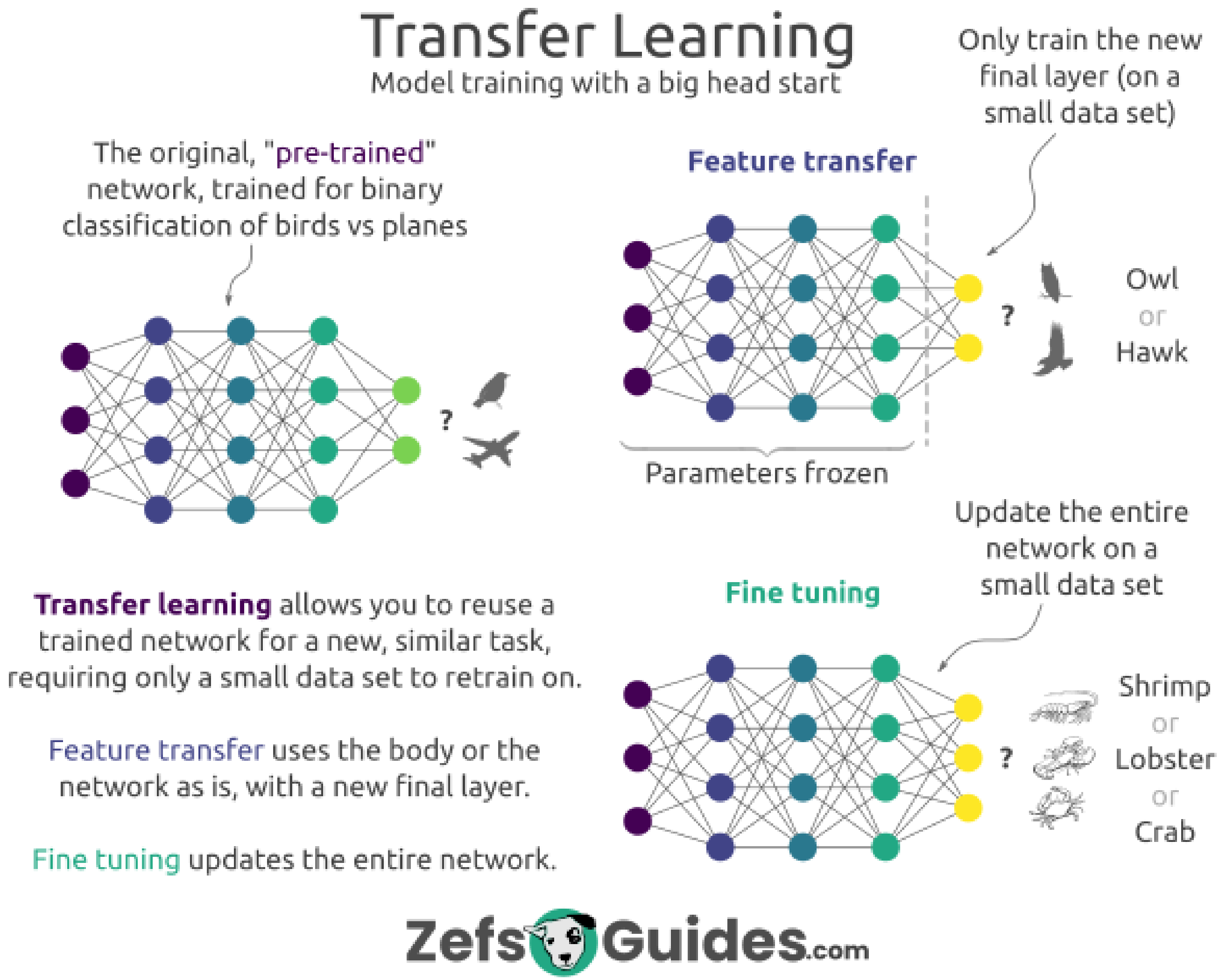


Deep Learning

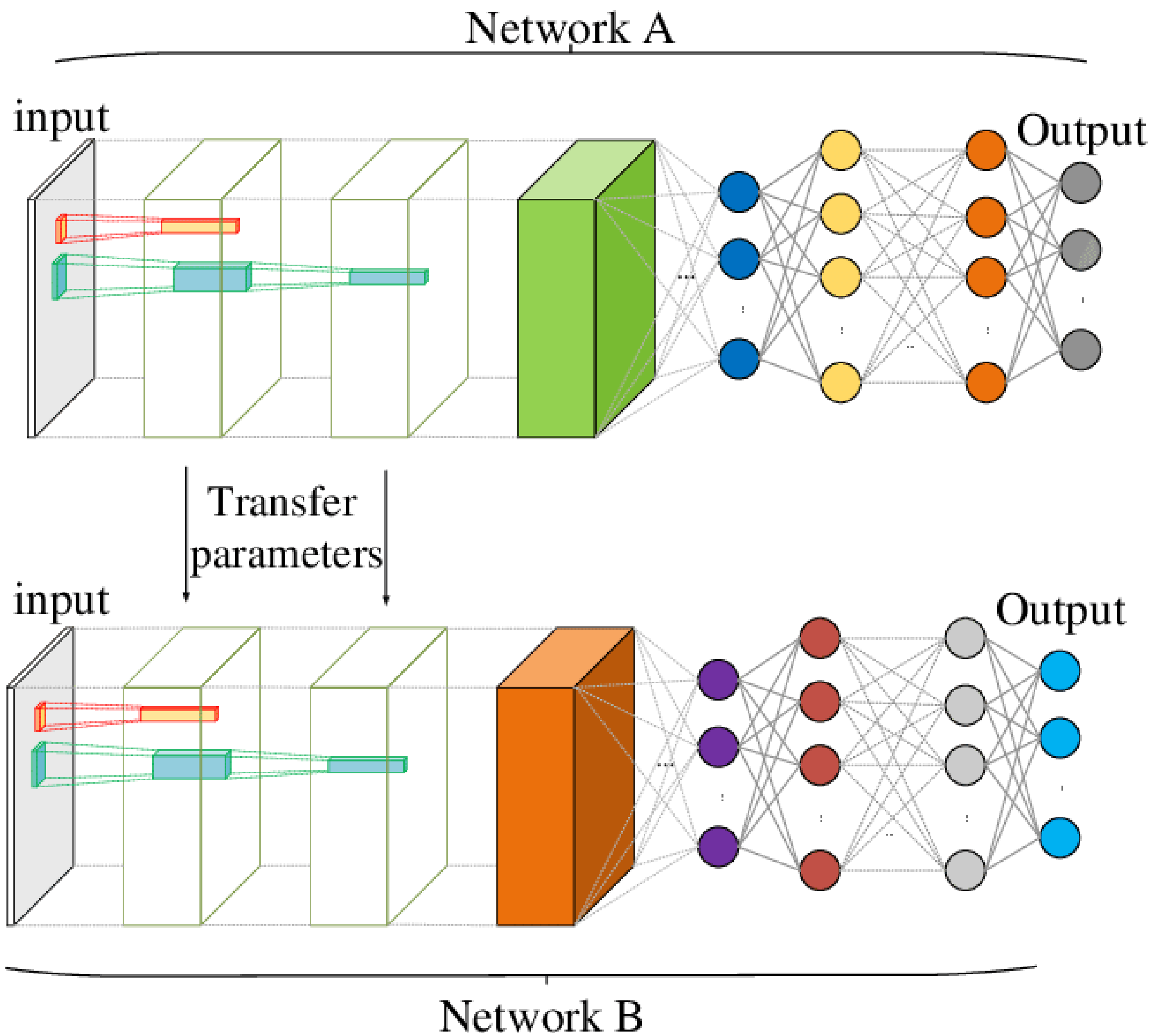
- Transfer learning involves using a pre-trained model for a specific task and fine-tuning it for a new task.
- In transfer learning, the knowledge gained from training on one task is transferred to another task, which can reduce the amount of training data needed and improve the performance of the model
- This approach can save time and resources when building new models.

Transfer Learning

Dr. Zein IBRAHIM

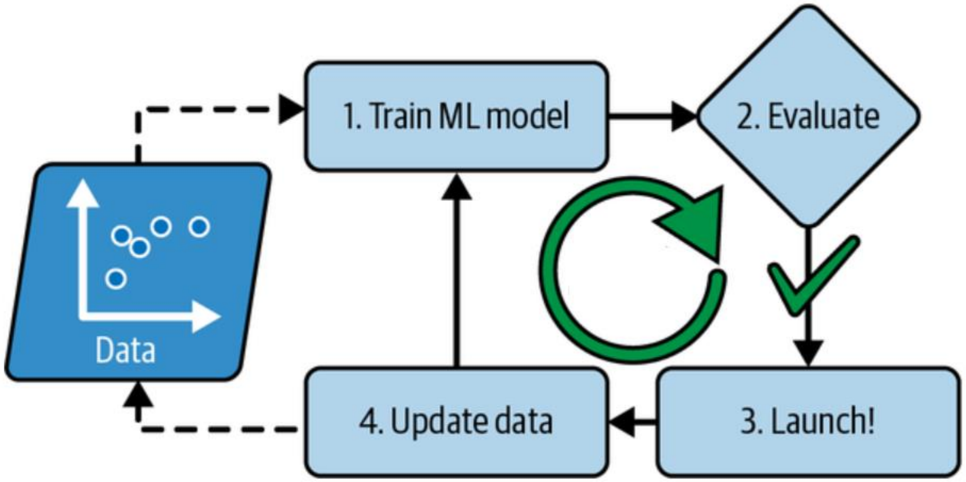


Transfer Learning

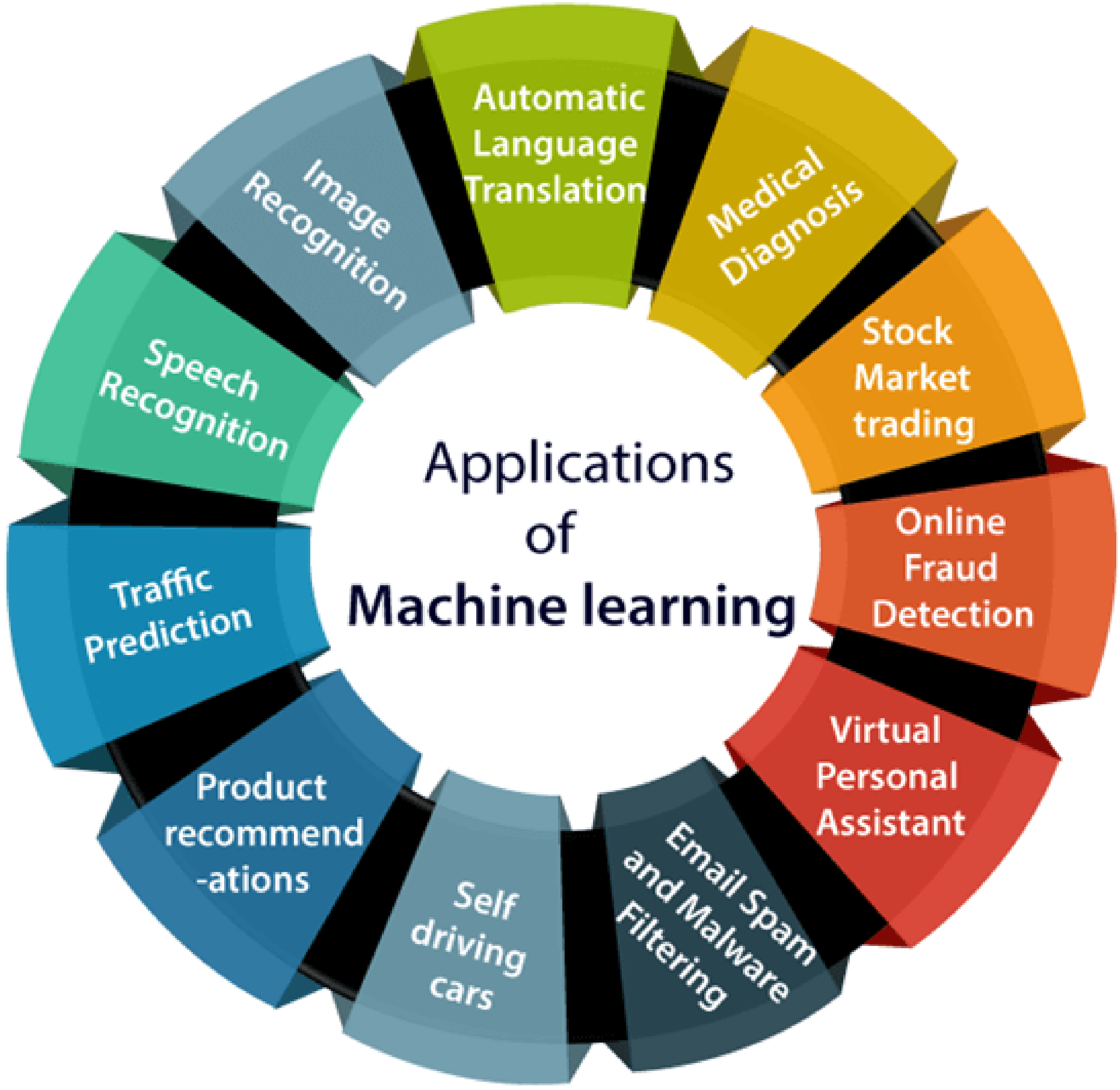


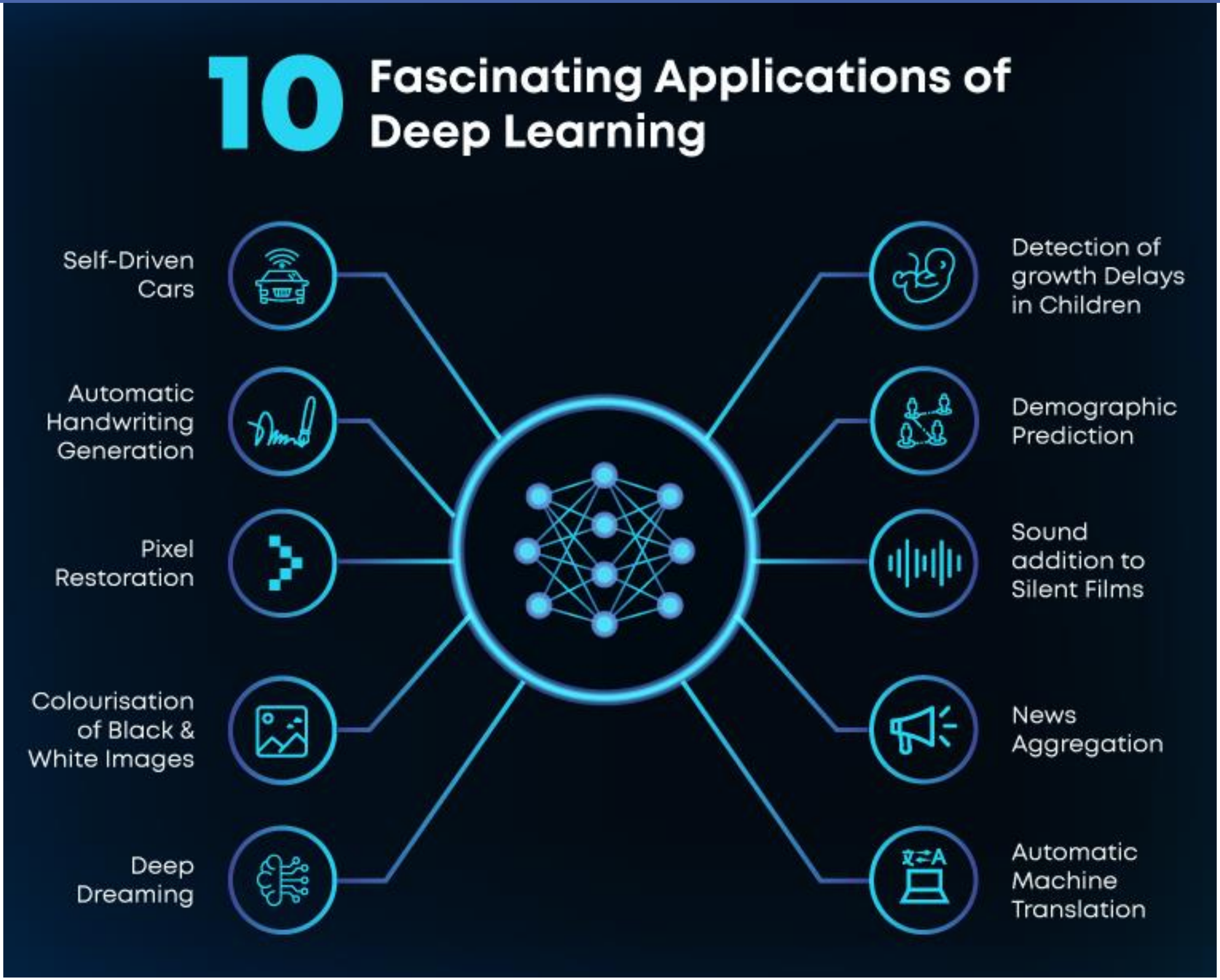
Transfer Learning

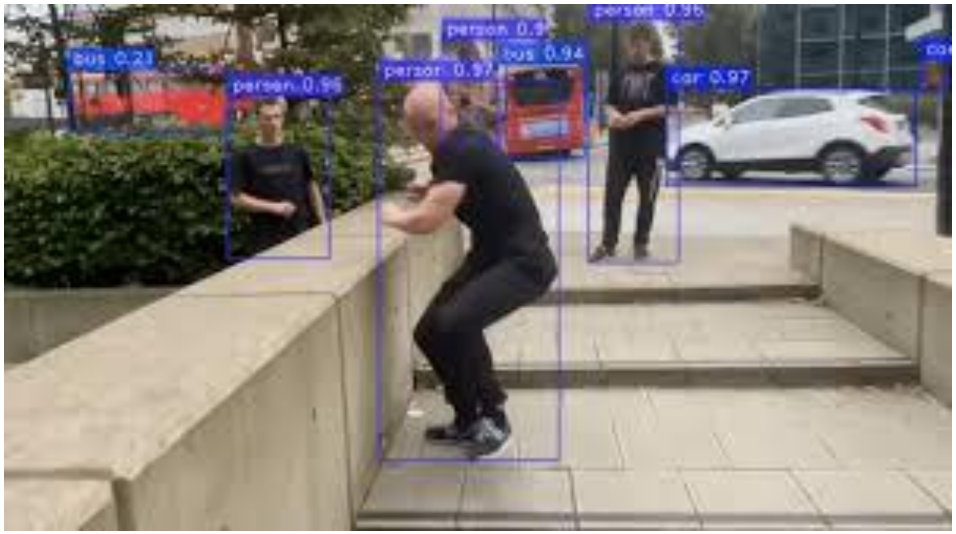
- Training a model and then deploy it is called offline learning
- Online learning involves updating a machine learning model as new data becomes available.
- This approach is commonly used in applications where new data is generated frequently, such as in online advertising or e-commerce.

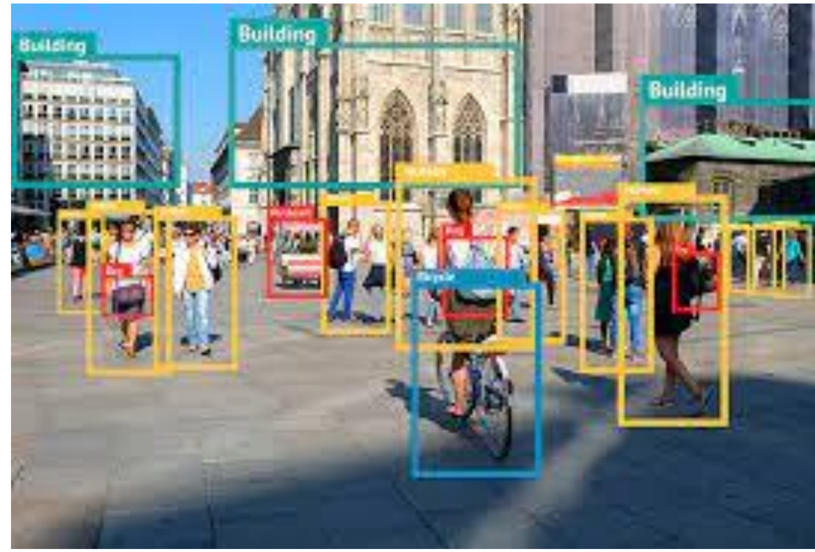


Online Learning


Applications
of
Machine learning
Automatic
Language
Translation
Medical
Diagnosis
Stock
Market
trading
Online
Fraud
Detection
Virtual
Personal
Assistant
Email Spam
and Malware
Filtering
Self
driving
cars
Product
recommend
-ations
Traffic
Prediction
Speech
Recognition
Image
Recognition


10 Fascinating Applications of Deep Learning
Self-Driven Cars
Automatic Handwriting Generation
Pixel Restoration
Colourisation of Black & White Images
Deep Dreaming
Detection of growth Delays in Children
Demographic Prediction
Sound addition to Silent Films
News Aggregation
Automatic Machine Translation





Object Detection

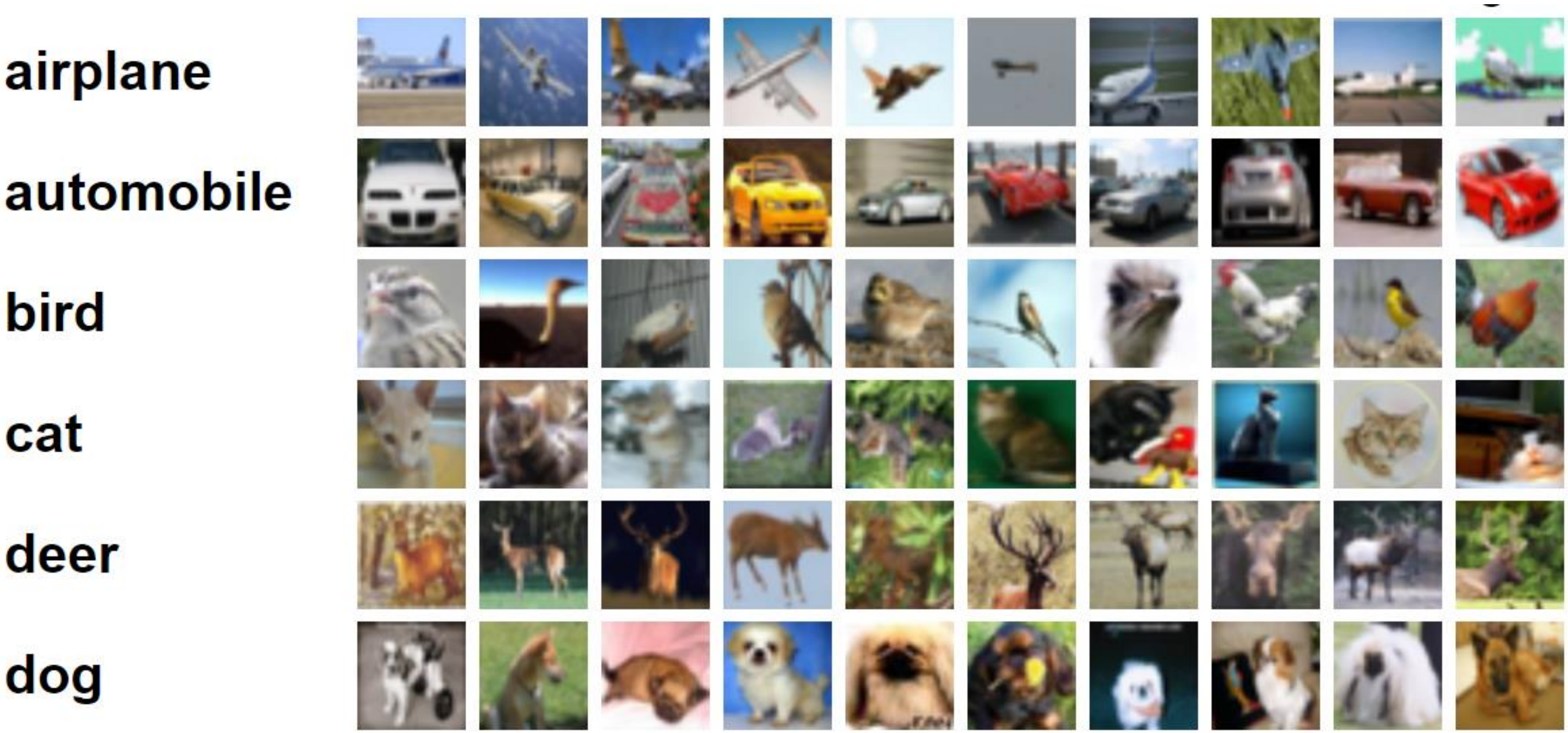
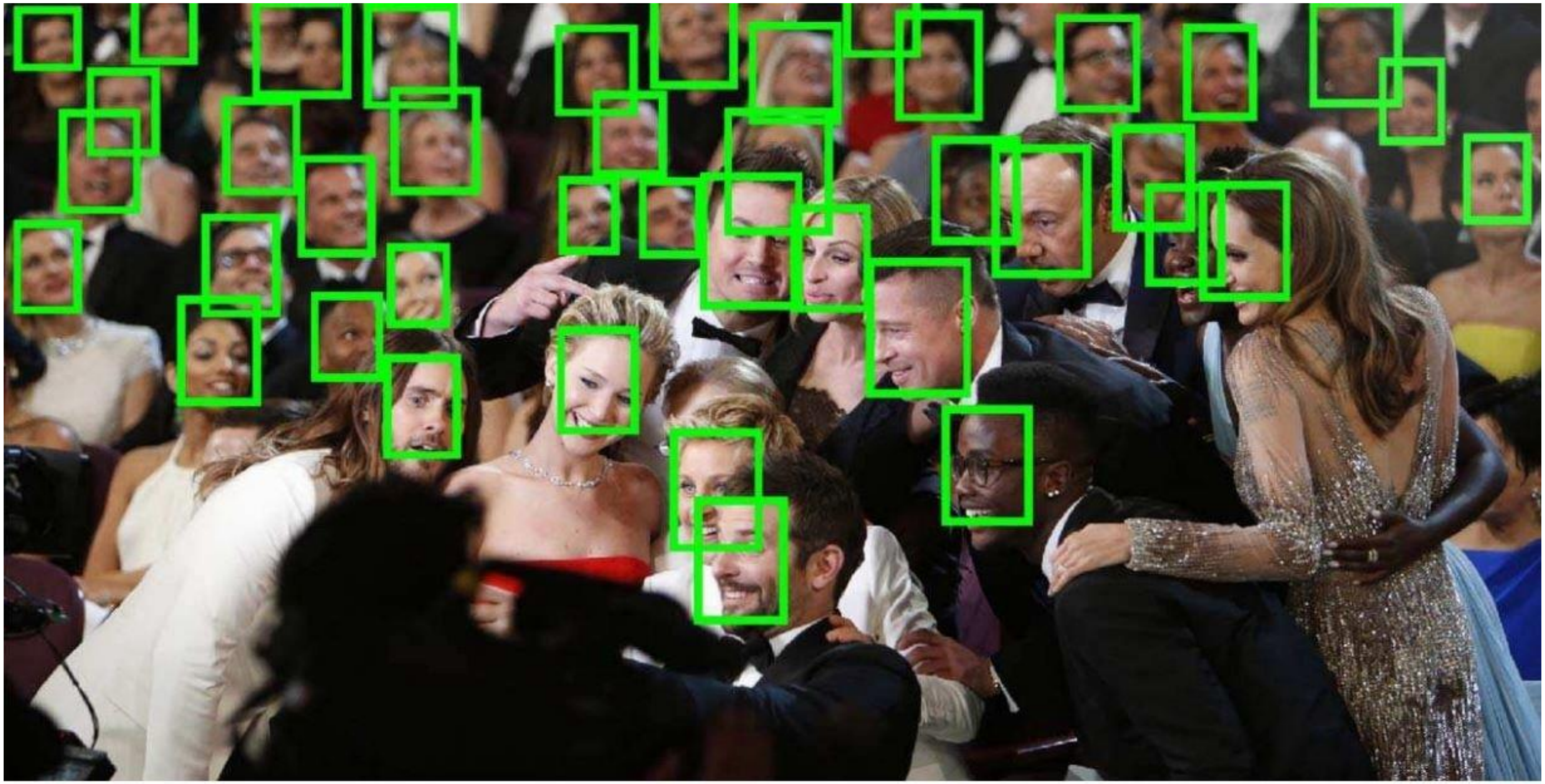


Image Classification



Face recognition

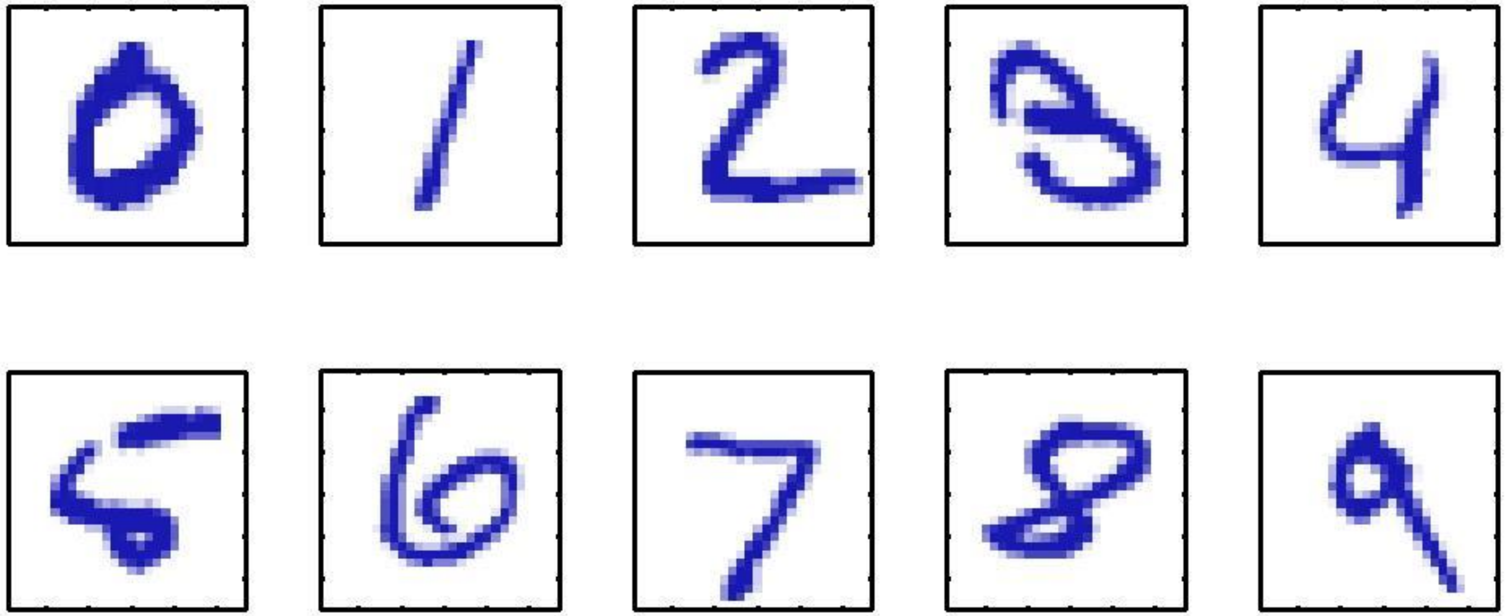
Face recognition

Self-driving Car

Coloring B&W images

Semantic Segmentation

- ***Handwritten Digit Recognition***

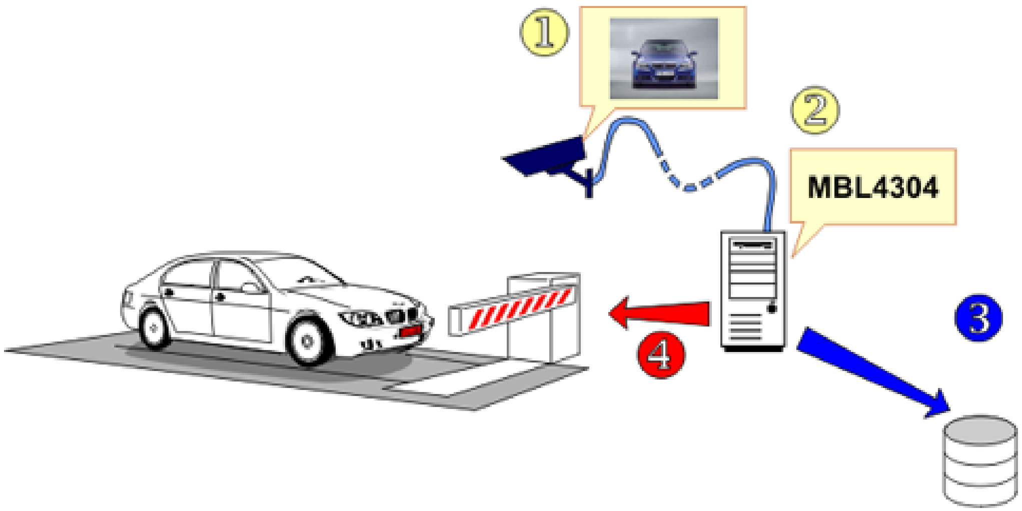


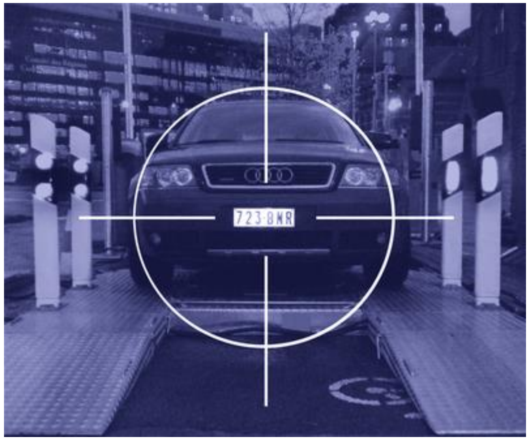
- More general → Handwritten Character Recognition
- Ex. of application → To filter letters at post offices

Applications: Handwritten

Applications: Plate Number

- Automatically detect and read the license plates on cars





- Modules: (i) acquisition, (ii) enhancement, (iii) segmentation, character recognition
- Should work in real time

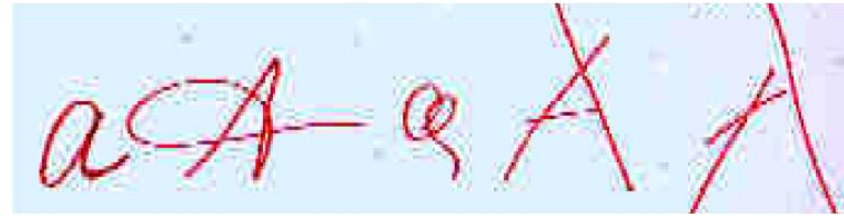
- Highlight conspicuous sections, such as possible diseases
- Helping assisting doctors to make diagnostic decisions
- Computer-aided diagnosis (CAD) used to diagnosis:
  - **breast cancer**,
  - **lung cancer**,
  - **colon cancer**,
  - **prostate cancer**,
  - **bone metastases**,
  - **coronary artery disease**
  - **congenital heart defect**.

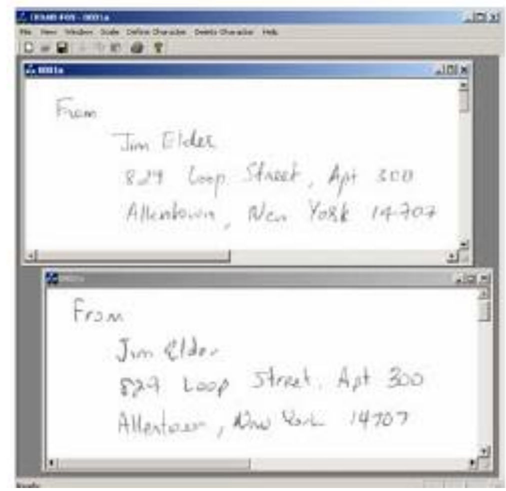
Applications: Diagnosis

- Several applications in the domain of SR
  - Recognition of spoken information → What is the information talked
    - Ex: Make a document (i.e. word doc) by speaking instead of writing.
  - Recognize the person who are speaking.

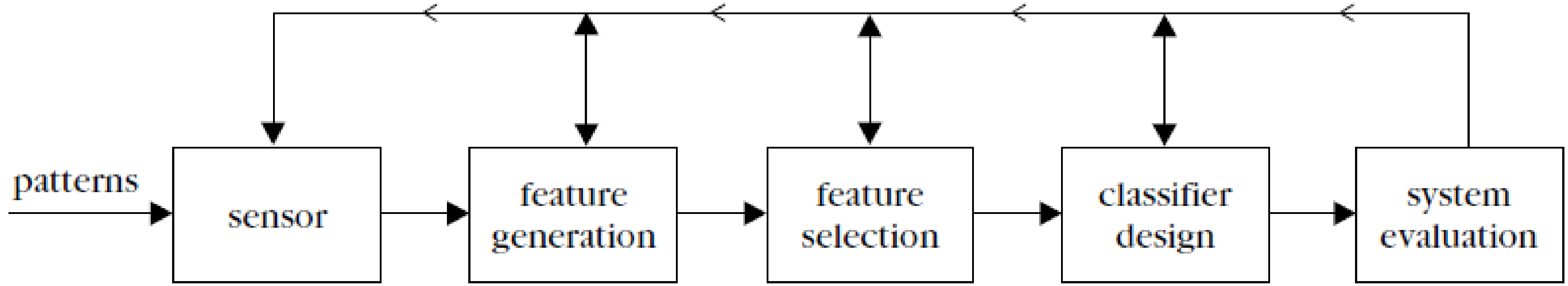
Applications: Speech Recognition

Applications



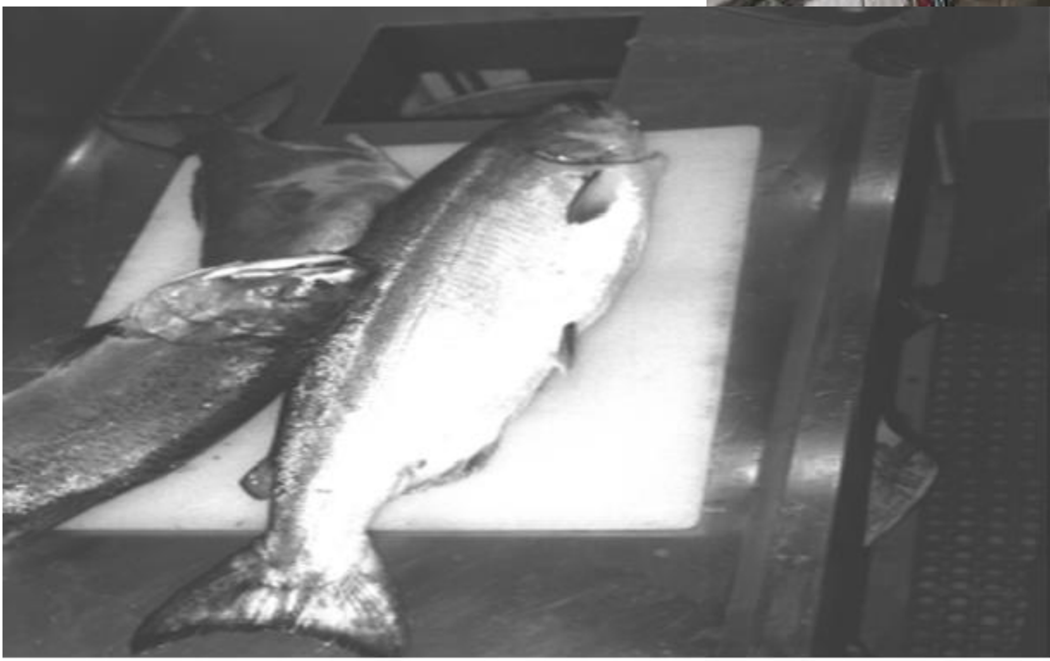


Writer recognition



The basic stages involved in the design of a classification system

Classification System



"Salmon" or "Sea bass" ?

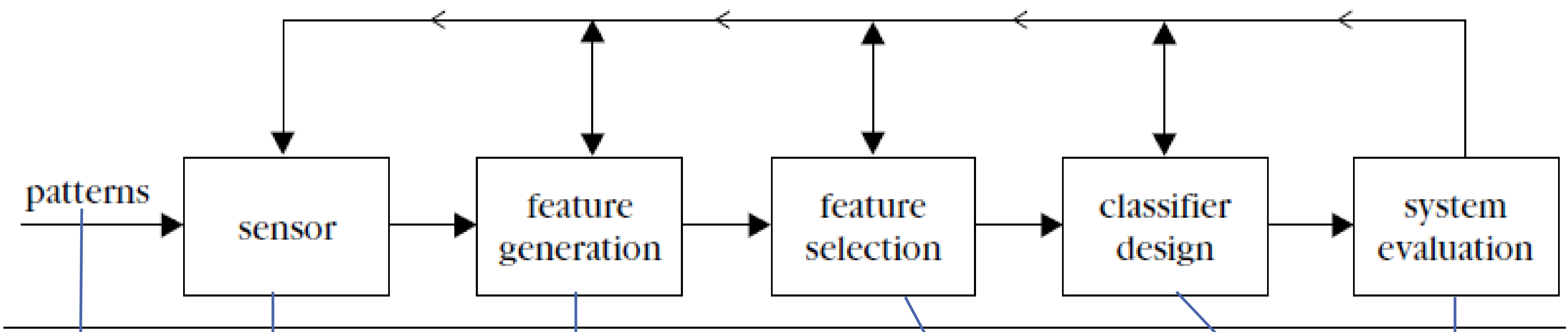
Classification System

1. **Data collection**:
   - Data used to train the machine learning model.
   - can come from various sources such as databases, APIs, web scraping, or sensor data.
2. **Data preprocessing**:
   - Data needs to be preprocessed to ensure that it is clean, formatted, and ready for use in training the machine learning model.
   - This may involve tasks such as cleaning, filtering, feature engineering, and normalization.
   - Feature engineering includes features extraction and selection.
3. **Model selection**:
   - Selecting a suitable machine learning algorithm that is appropriate for the problem at hand.
   - This may involve comparing different algorithms and evaluating their performance on the data.
4. **Training the model**:
   - Training the model using the preprocessed data.
   - This involves feeding the data into the model and adjusting the model's parameters to minimize the error between its predictions and the true labels.

Machine Learning Pipeline

5. **Model evaluation**:
   - Model needs to be evaluated to ensure that it is performing well on unseen data.
   - This involves using a test set to measure the model's accuracy, precision, recall, F1-score, or other performance metrics.
6. **Hyperparameter tuning**:
   - Involves tuning the model's hyperparameters to optimize its performance.
   - This may involve grid search, random search, or other techniques to find the best hyperparameters for the model.
7. **Deployment**:
   - Involves deploying it in a production environment where it can be used to make predictions on new data.
   - This may involve creating an API or integrating the model into an existing system.

Machine Learning Pipeline



The basic stages involved in the design of a classification system

Fishes

Camera, balance ...

Data Collection

Which features?

Ex: Length, width, height, color of the skin?

Are all features valuable? Which are more important ?

Ex: height is useless. Length is the most important

How to separate?

Type of Method

How accurate is?

Evaluate your model

Classification System

- Sometimes Data is ready and available, most times it will be scarce, or have to be collected from scratch.
  - Having large amount of data is very important in Machine Learning. However, attaining large amounts is challenging:
    - Sensors collecting data are subject to noise, thus low error data is hard to attain.
    - Sometimes it is difficult to collecting data.
    - Enough data should be given without excessive computation is impossible to predict.

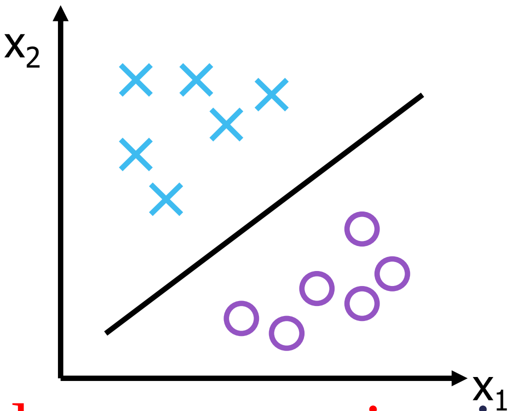
Data Collection

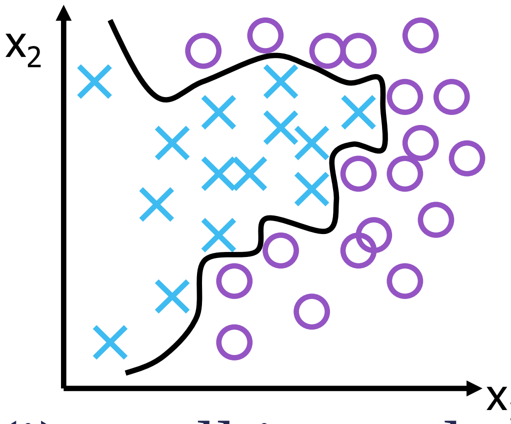
First step:

- What are the measurable quantities that makes these two regions distinct from each other.
- Classification step bases on their values
- Ex: Mean value of the intensity in each region + standard deviation.
- These are called ***features***.
- Feature vector $F = [\, f_1, f_2, \ldots, f_l \,]$ of dimension l.
- Feature vectors are treated as random vectors
  - Dependent on each other or independent.

Data Preprocessing: Feature generation

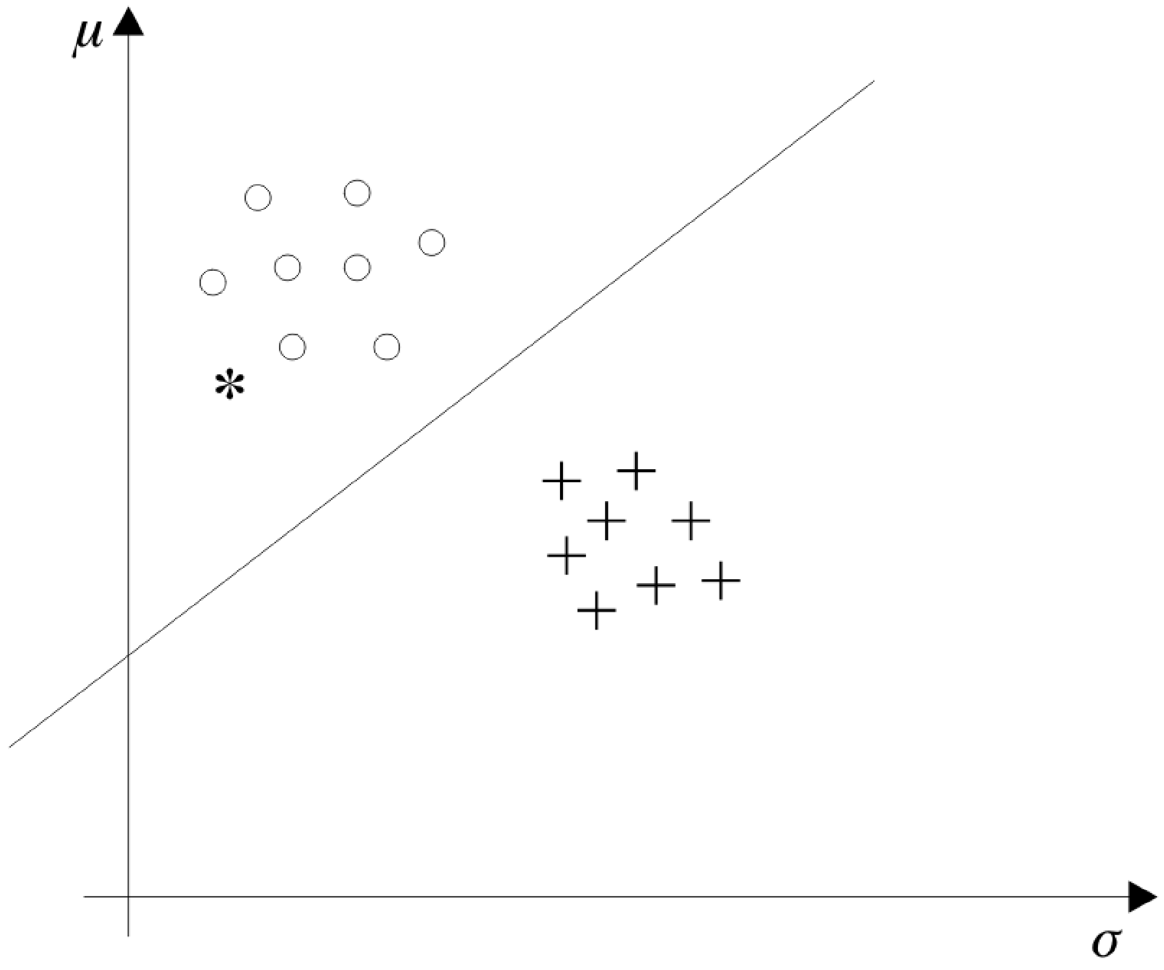
- Each pattern is represented as a point in *d*-dimensional feature space
- Choice of features and their desired invariance properties are domain-specific





- Good representation implies (i) small intra-class variation, (ii) large interclass separation and (iii) simple decision boundary

Feature representation



Feature generation

Second step:

- Question: Is it simple to find the feature vector F → NO
  - F may contain usefulness features
  - F may contain redundant features

1. What is the best number ***i*** of features to use for the problem ($l < i$)?
   - Called ***feature selection*** stage
2. We can transformation the feature vector from the l-dimensional space to a lower space
   - Called ***feature extraction*** stage

Feature Engineering

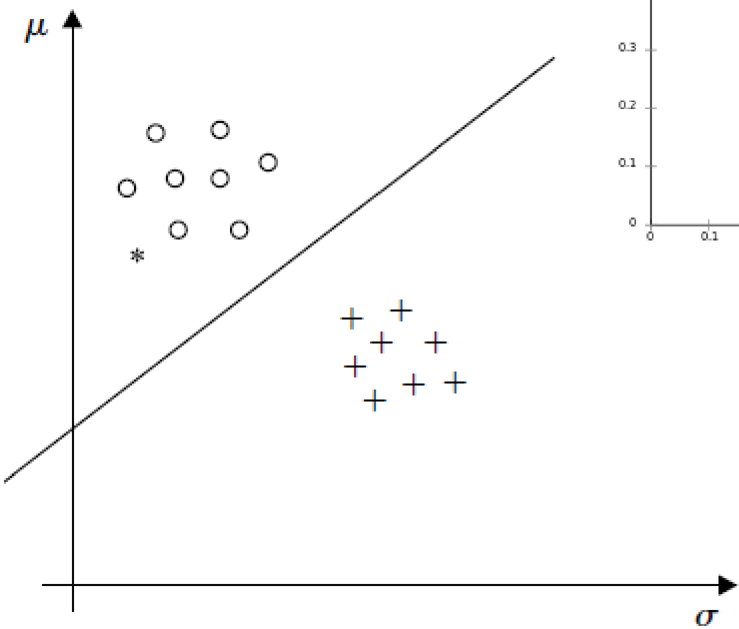
- Used to give the same importance for all features:
  - **Ex**: $X_1 = (2, 120)$; $X_2 = (8, 533)$; $X_3 = (1, 987)$;

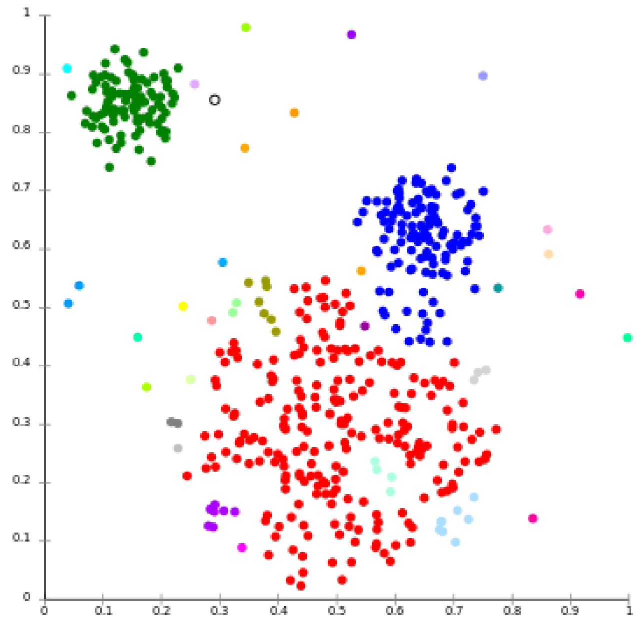
    $X_4 = (15, 1121)$; $X_5 = (18, 1023)$;
  - **Problem** → feature one has values below 18 while feature two has values below 1121
  - $\Rightarrow$ the value of the distance will be more affected by the second feature than the two at the same time.
  - Normalization → divide each value of a feature by the maximum value of that feature.
    - Ex: f_1 by 18 and f_2 by 1121

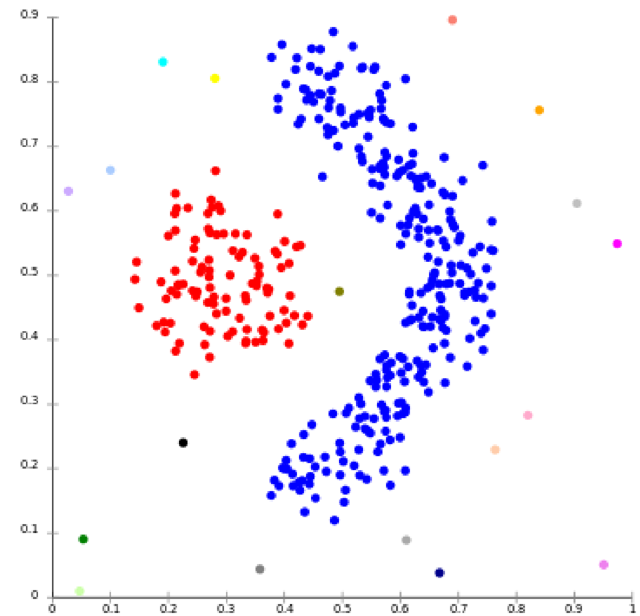
Normalization

- Three main types of classifiers
  1. Classification or supervised classification
  2. Clustering or unsupervised classification
  3. Semi-supervised classification
- In classification :
  – Assign a pattern to one of predefined classes
  – Classifier is represented by a model generated from a set of training data.
- In clustering:
  – classes are unknown
  – we want to discover classes from the data (cluster analysis)

Model Selection







Model Selection

- For some classifiers → classify a pattern ⇒ compare it against other patterns or against the models representing the classes.
- Several similarity and dissimilarity measures have been proposed in the literature.
- Distance measure used to find dissimilarity
  - Lower ⇒ more similar
  - Higher ⇒ more dissimilar

Proximity Measures

- Distance measures are (1) metric or (2) non-metric
- The metric distances have the following properties:
  - $d(x, x) = 0;$
  - $d(x, y) \geq 0;$
  - $d(x, y) = d(y, x);$
  - $d(x, y) \leq d(x, z) + d(z, y);$

Distance measure

- Some of used distances:
- Minkowski: $d^m(X,Y) = \left(\sum_{k=1}^{d} |x_k - y_k|^m\right)^{\frac{1}{m}}$
- Manhattan or L_1: m = 1; $d(X,Y) = \left(\sum_{k=1}^{d} |x_k - y_k|\right)$
- Euclidean or L_2: m = 2; $d(X,Y) = \sqrt{\sum_{k=1}^{d} (x_k - y_k)^2}$
- L_∞: $d(X,Y) = \max_{k=1..d} |x_k - y_k|$
- Mahalanobis: $d(X,Y)^2 = (X-Y)^T \Sigma^{-1} (X-Y)$

Metric Distance measure

- ***Example***: Suppose that I have a pattern p
- $\boldsymbol{F}_1$ = (4,1,3) the features vector of values.
- Suppose that I have two other patterns p and p' such that ***F*** = (2,5,1) and ***F***'=(4,6,2).
  - $d_2(p, p_1) = \sqrt{(4-2)^2 + (1-5)^2 + (3-1)^2} = 4.899$
  - $d_2(p, p_2) = \sqrt{(4-4)^2 + (1-6)^2 + (3-2)^2} = 5.099$
- p is closer to p than to p_2

Metric Distance measure

Weighted Distance Measure

- Used to give more importance to a feature than the other

- $d(X,Y) = \left(\sum_{k=1}^{d} w_k \times (x_k - y_k)^m\right)^{\frac{1}{m}}$

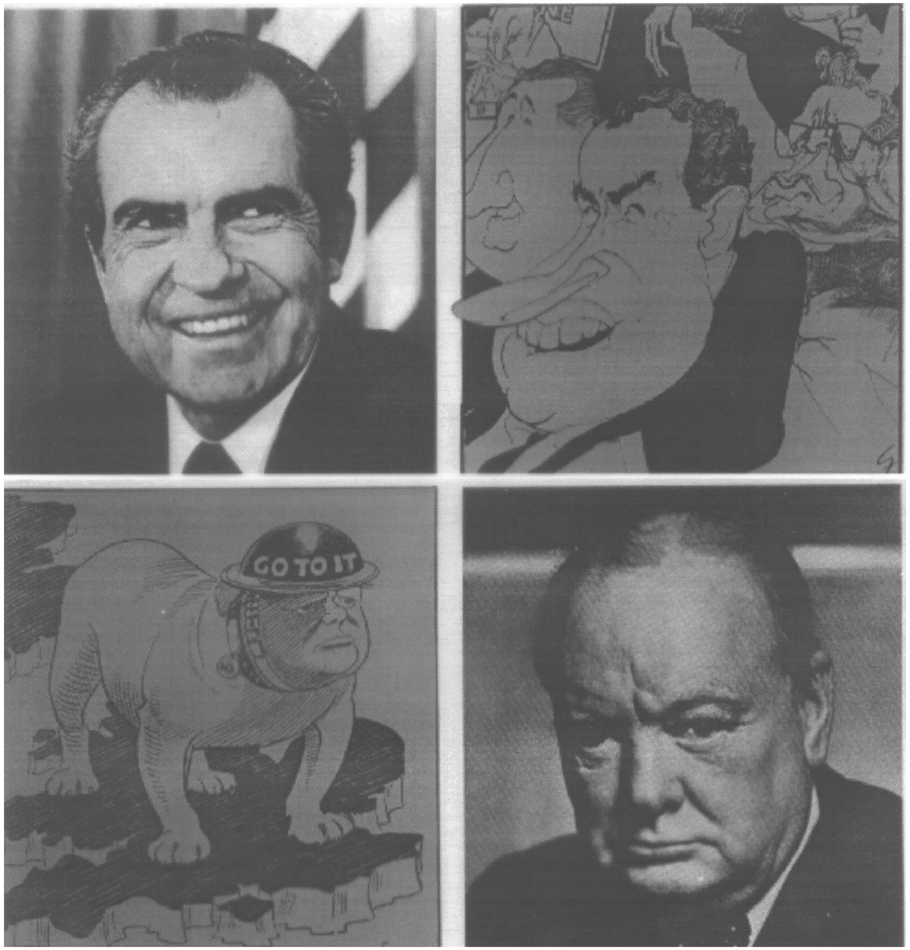
- Ex: X=(4, 2, 3), Y=(2, 5, 1) / w_1=0.3, w_2=0.6, w_3=0.1
  – Feature 2 (***f_2***) is more important than feature 1 (***f_1***) which is more important than feature 3 (***f_3***).
  – $L_2 = \sqrt{0.3 \times (4-2)^2 + 0.6 \times (1-5)^2 + 0.1 \times (3-1)^2} = 3.35$

- Edit distance or called also Levenshtein
- Measure the distance between two strings
- = number of operation to change a string s_1 to string s_2
  - Operations involved:
    - Update, insert, or delete
- Computed by recurrence as follows:
  - d(“ “, “ “) = 0, & d(s, “ “) = d(“ “,s) = ||s||
- Ex: s_1 = “TRAIN” & s_2 = “BRAIN” → $d(s_1, s_2) = 1$;
- Ex: s_1 = “TRAIN” & s_2 = “CRANE” → $d(s_1, s_2) = 3$;

Edit distance

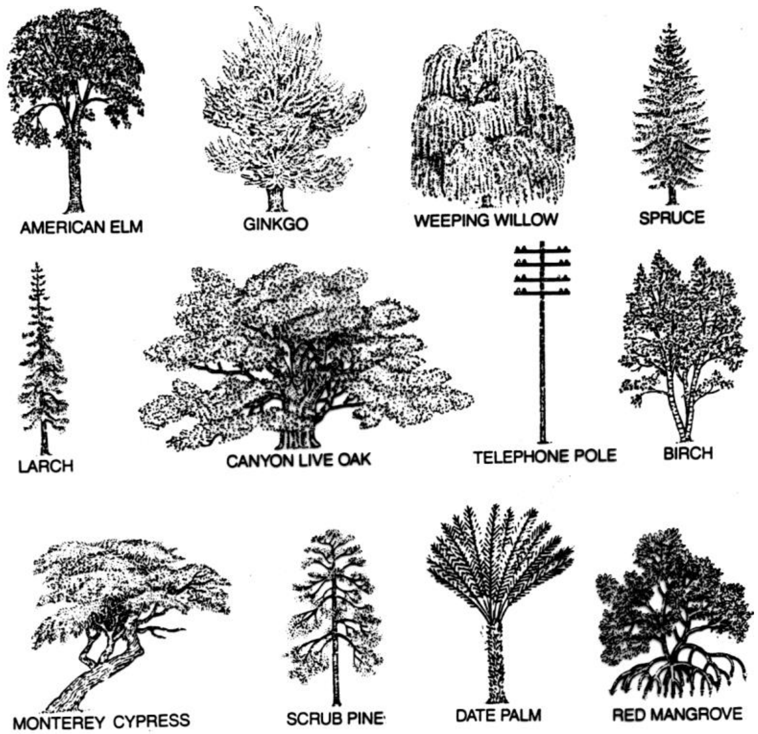
- Several challenges
  - Representation of patterns → features
  - Curse of dimensionality
  - Which classification method
  - Matching problem
  - Evaluation
  - ….

Challenges



What facial features (description) or model should we use to account for the large intra-class variability?

Difficulties of Representation



ARE ALL THESE OBJECTS TREES? Even a young child can answer correctly; a conventional computer, however, has enormous difficulty in doing so. Although there is a fair amount of regularity among the trees shown (each has a trunk and branches, for example), there is also a major component of arboreal irregularity among them. A generalized definition of a tree based on the underlying regularity could lead to erroneous identifications (such as mistaking a telephone pole, which has a "trunk" and "branches," for a tree). Hence any effective program designed to recognize trees would essentially have to be a list of all types of trees, which cannot be done in a few lines of computer code.

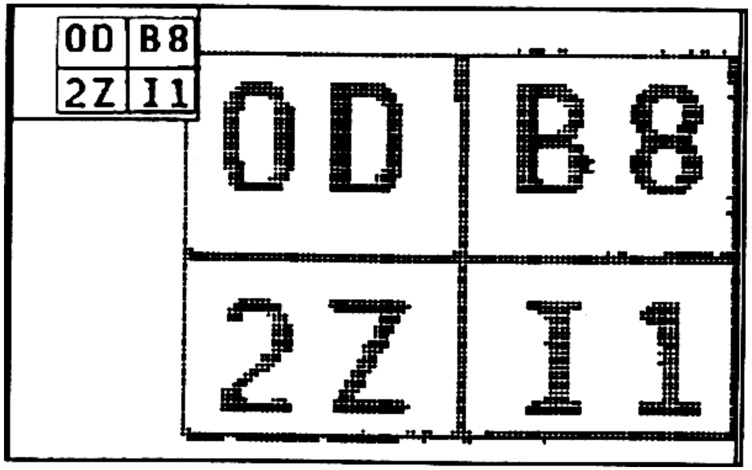
Difficulties of Representation

The letter “T” in different typefaces

Same face under different expression, pose, illumination

Intra-class variability

Identical twins



Characters that look similar

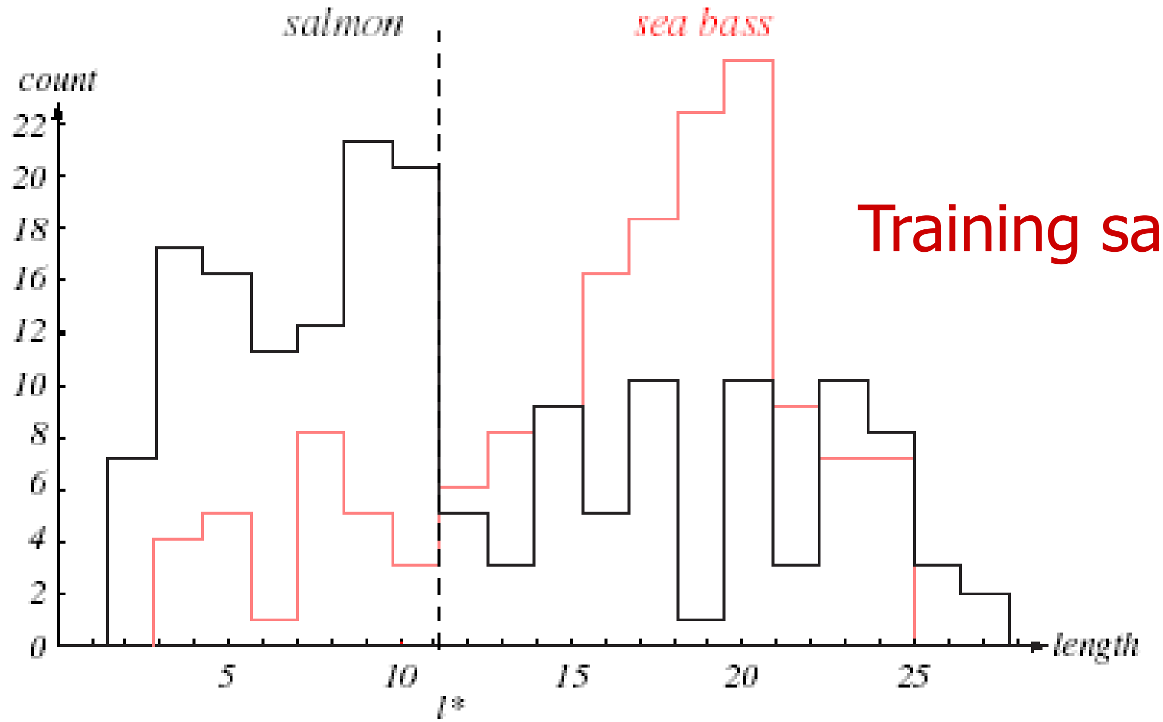
Inter-class variability

- Should have some invariant properties (e.g., w.r.t. rotation, translation, scale…)
- Account for intra-class variations
- Ability to discriminate pattern classes of interest; low inter-class similarity;
- Robustness to noise, occlusion,..
- Lead to simple matching or decision-making strategies (e.g., linear decision boundary)
- Low measurement cost; real-time

Good Representation


"Salmon" or "Sea bass" ?

Classification System



Training samples

FIGURE 1.2. Histograms for the length feature for the two categories. No single threshold value of the length will serve to unambiguously discriminate between the two categories; using length alone, we will have some errors. The value marked l^* will lead to the smallest number of errors, on average. From: Richard O. Duda, Peter E. Hart, and David G. Stork, *Pattern Classification*. Copyright © 2001 by John Wiley & Sons, Inc.

Length feature Only ?

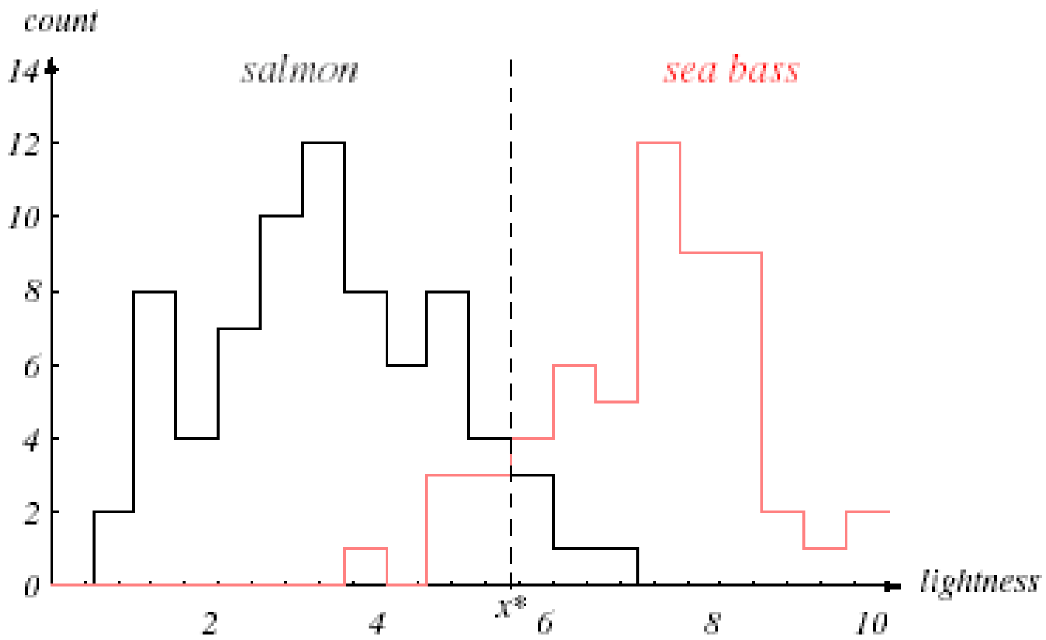


FIGURE 1.3. Histograms for the lightness feature for the two categories. No single threshold value x^* (decision boundary) will serve to unambiguously discriminate between the two categories; using lightness alone, we will have some errors. The value x^* marked will lead to the smallest number of errors, on average. From: Richard O. Duda, Peter E. Hart, and David G. Stork, *Pattern Classification*. Copyright © 2001 by John Wiley & Sons, Inc.

Overlap of these histograms is small compared to length feature

Lightness Feature Only ?

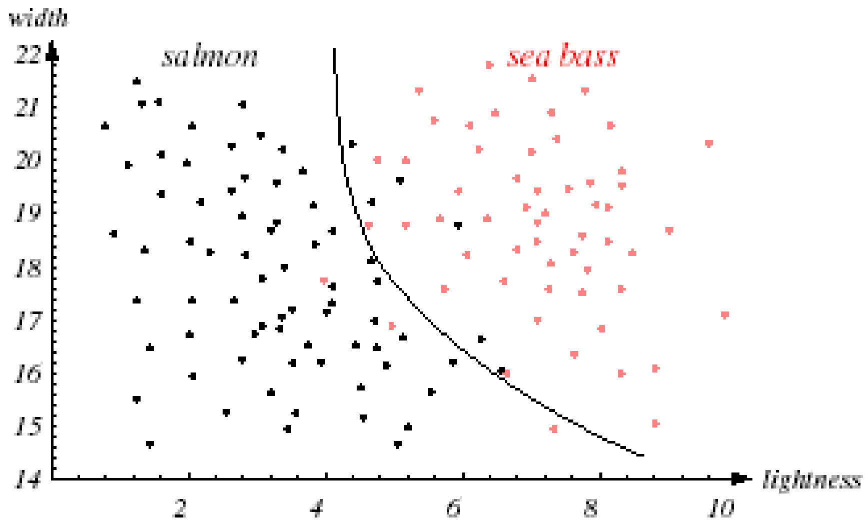


FIGURE 1.6. The decision boundary shown might represent the optimal tradeoff between performance on the training set and simplicity of classifier, thereby giving the highest accuracy on new patterns. From: Richard O. Duda, Peter E. Hart, and David G. Stork, *Pattern Classification*. Copyright © 2001 by John Wiley & Sons, Inc.

Simple decision boundaries are preferred

Boundary With Good Generalization

- **The Curse of Dimensionality**
  - The essence of this curse apply to machine learning algorithms because as the number of input dimension increases, we need more data to enable the algorithm to generalize sufficiently well.
  - The algorithm will try to Separate out data in to classes based on the features.
    - As the number of features increases so will be the needed Data points.

- **Testing Machine learning algorithms.**
  - We need a training set to train our algorithm based on targets (supervised)
  - Another set us needed (test set) to test how well things are going
    - The only problem is that is reduces the amount of data to be used for training.

Machine Learning Issues

Linear (simple) decision boundary; Cost of misclassification?

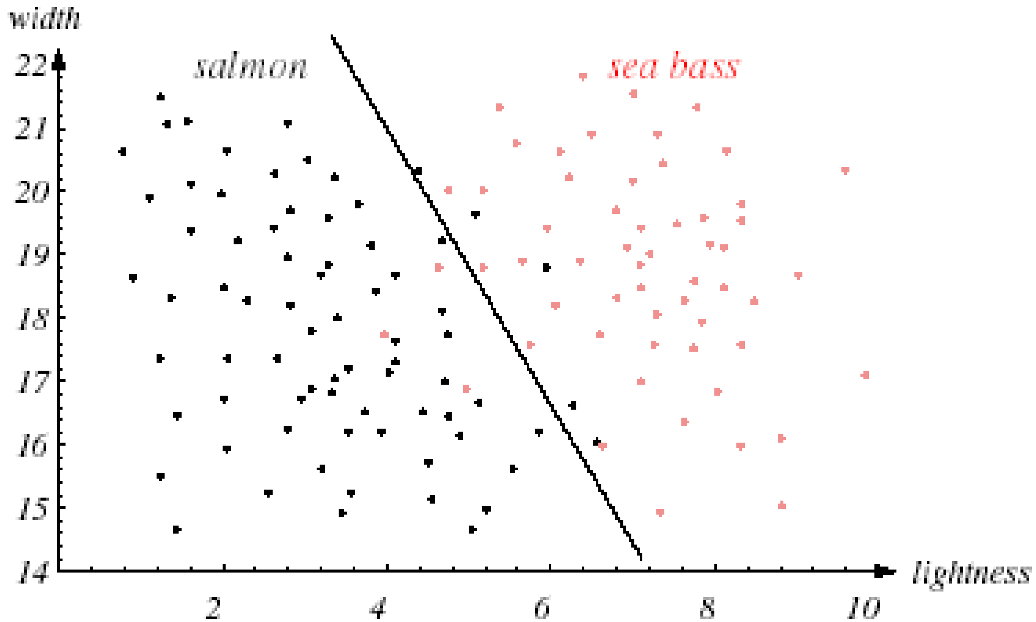


FIGURE 1.4. The two features of lightness and width for sea bass and salmon. The dark line could serve as a decision boundary of our classifier. Overall classification error on the data shown is lower than if we use only one feature as in Fig. 1.3, but there will still be some errors. From: Richard O. Duda, Peter E. Hart, and David G. Stork, *Pattern Classification*. Copyright © 2001 by John Wiley & Sons, Inc.

Simple Decision Boundary

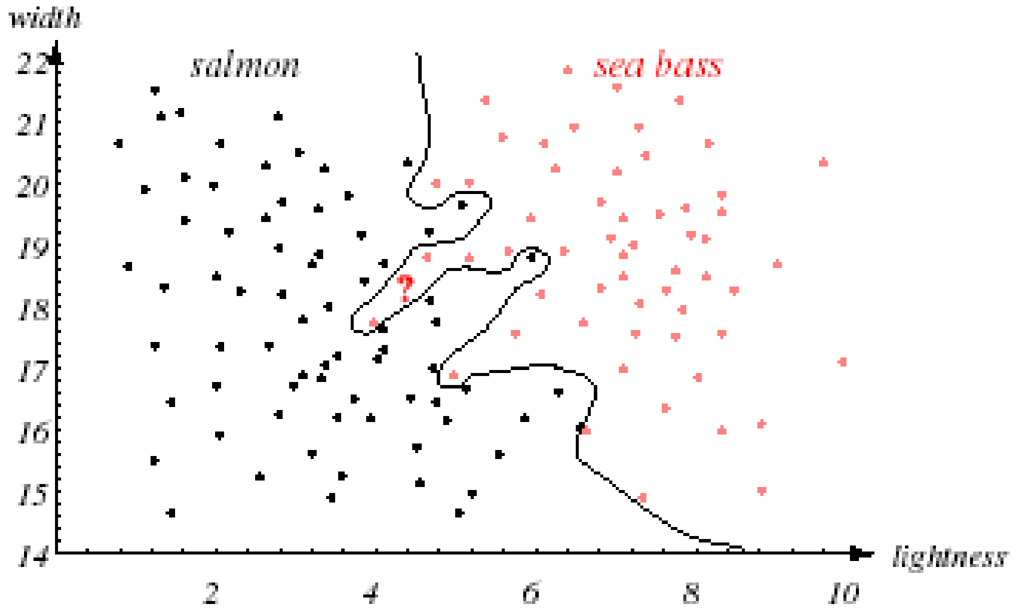
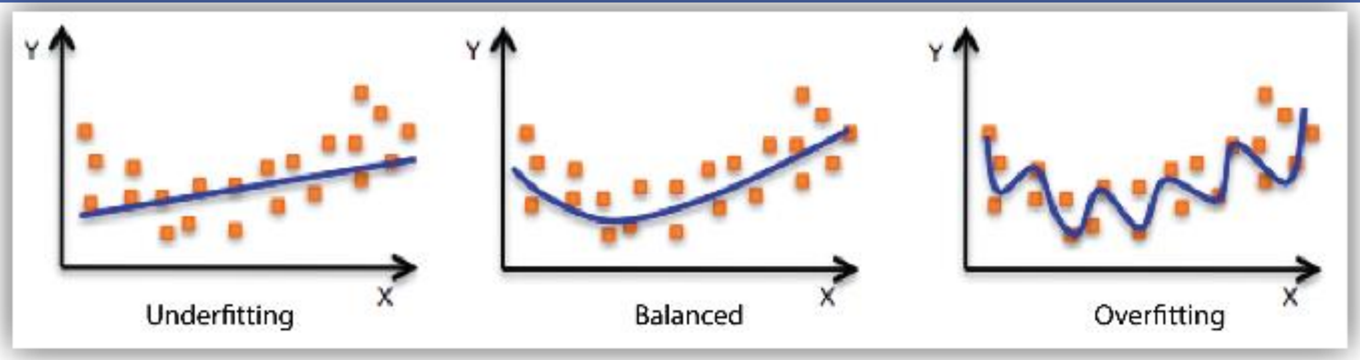
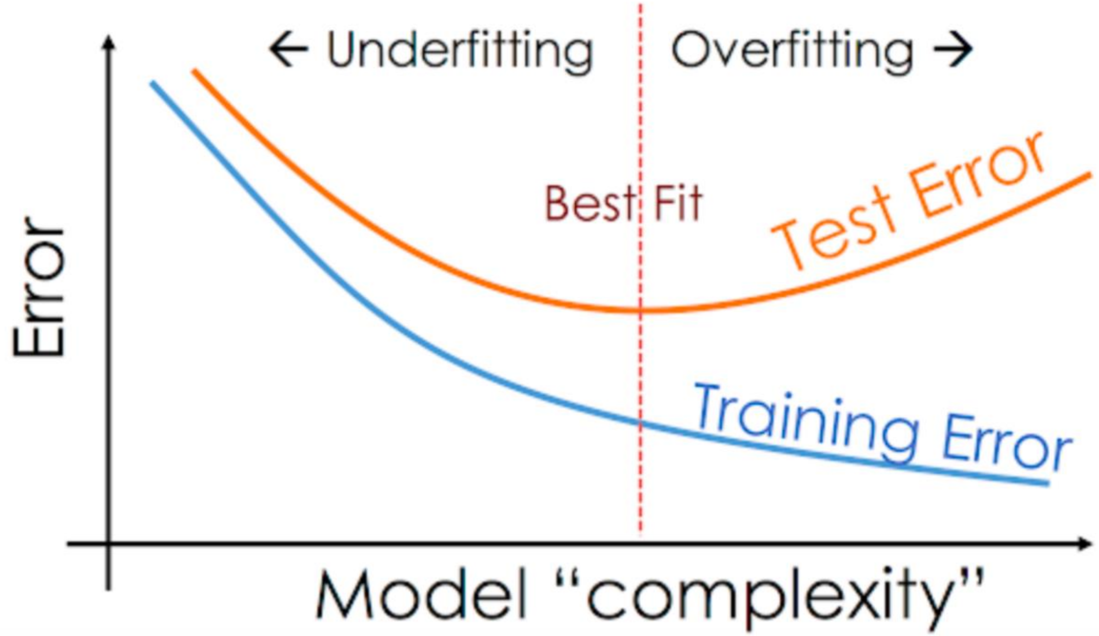


FIGURE 1.5. Overly complex models for the fish will lead to decision boundaries that are complicated. While such a decision may lead to perfect classification of our training samples, it would lead to poor performance on future patterns. The novel test point marked ? is evidently most likely a salmon, whereas the complex decision boundary shown leads it to be classified as a sea bass. From: Richard O. Duda, Peter E. Hart, and David G. Stork, *Pattern Classification*. Copyright © 2001 by John Wiley & Sons, Inc.

Complex Decision Boundary

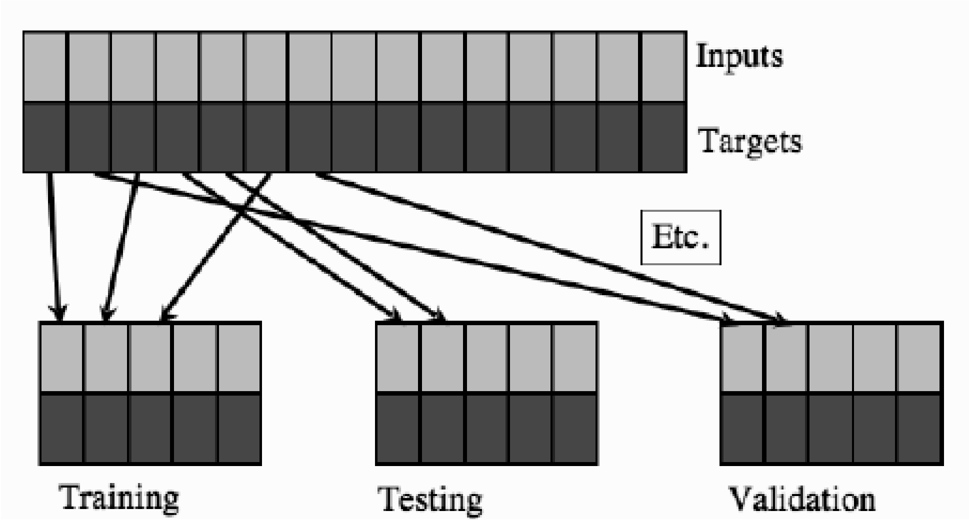




Machine Learning Issues (Overfitting)

- Given some data sets a typical percentage of data usage is below:

Type	Large Data Set	Small Data Set
Testing	25%	20%
Validation	25%	20%
Training	50%	60%



Machine Learning Issues (Training Testing and Validation)

This is the end of the lecture!

Do you have any question ?

End of Lecture !