Machine Learning and Computational Intelligence
Lecture 1

Sanjeeb Prasad Panday, PhD

Associate Professor

Dept. of Electronics and Computer Engineering

Director (ICTC)

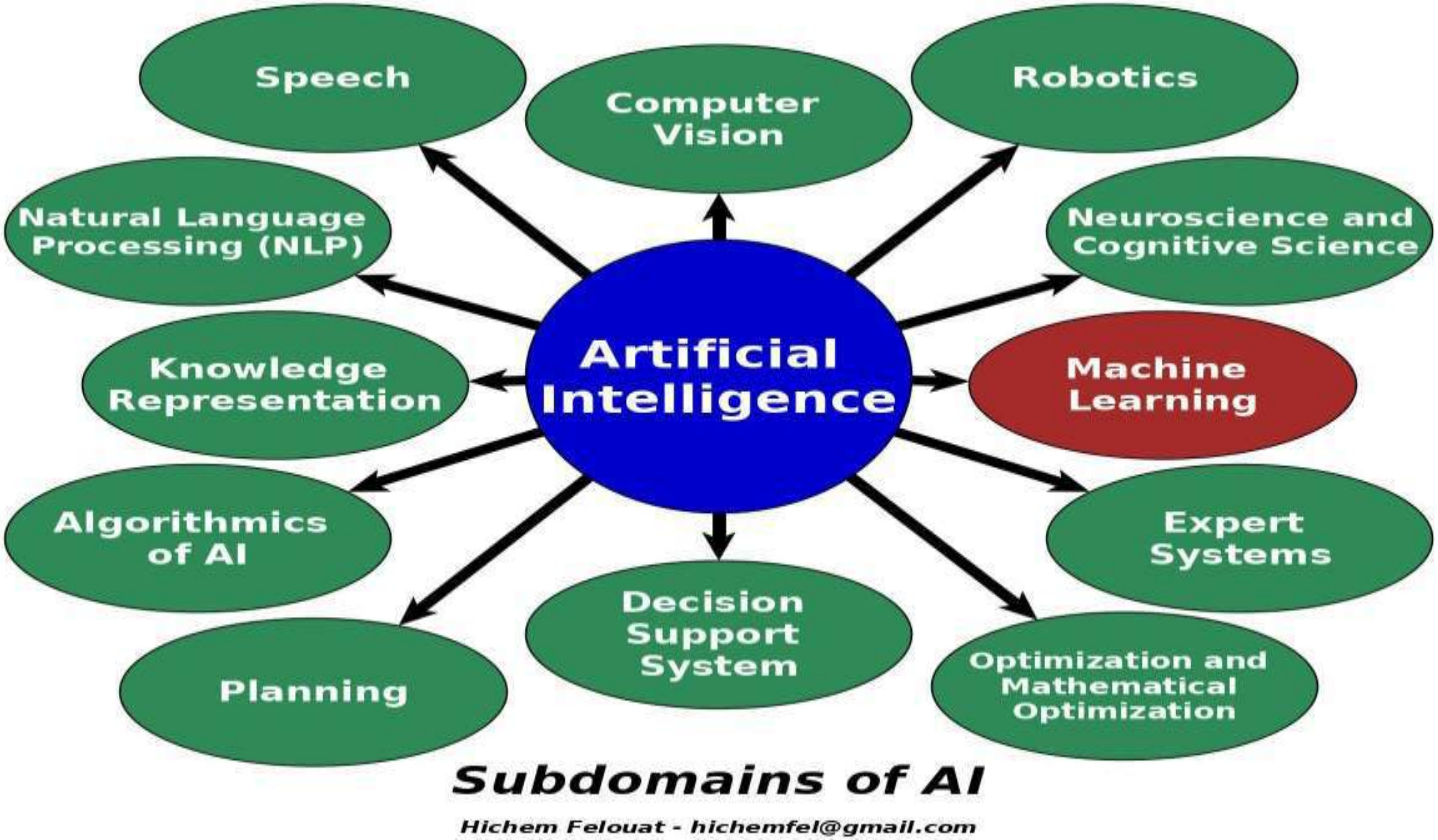
IOE, TU

What Is Artificial Intelligence?

Artificial intelligence (AI) is an area of computer science that emphasizes the creation of intelligent machines that work and react like humans.

- **AI is an interdisciplinary science with multiple approaches.**
- **AI has become an essential part of the technology industry.**

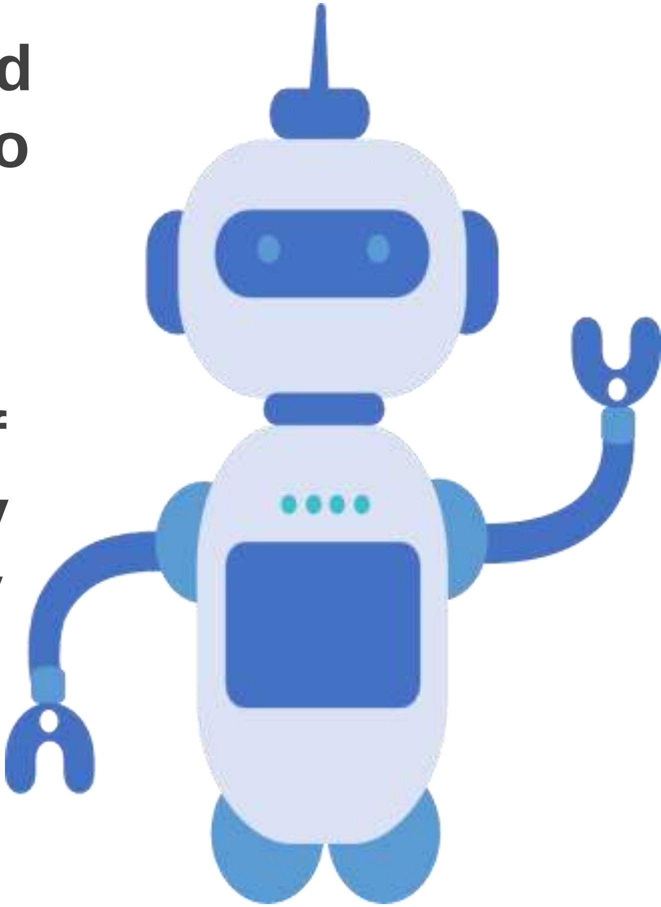
Subdomains of Artificial Intelligence



What Is Machine Learning?

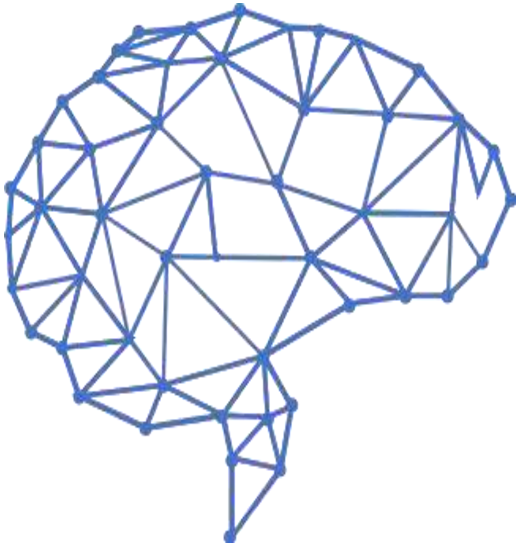
- Machine Learning is the science (and art) of programming computers so they can learn from data.

- Machine Learning is the field of study that gives computers the ability to learn without being explicitly programmed. —Arthur Samuel, 1959

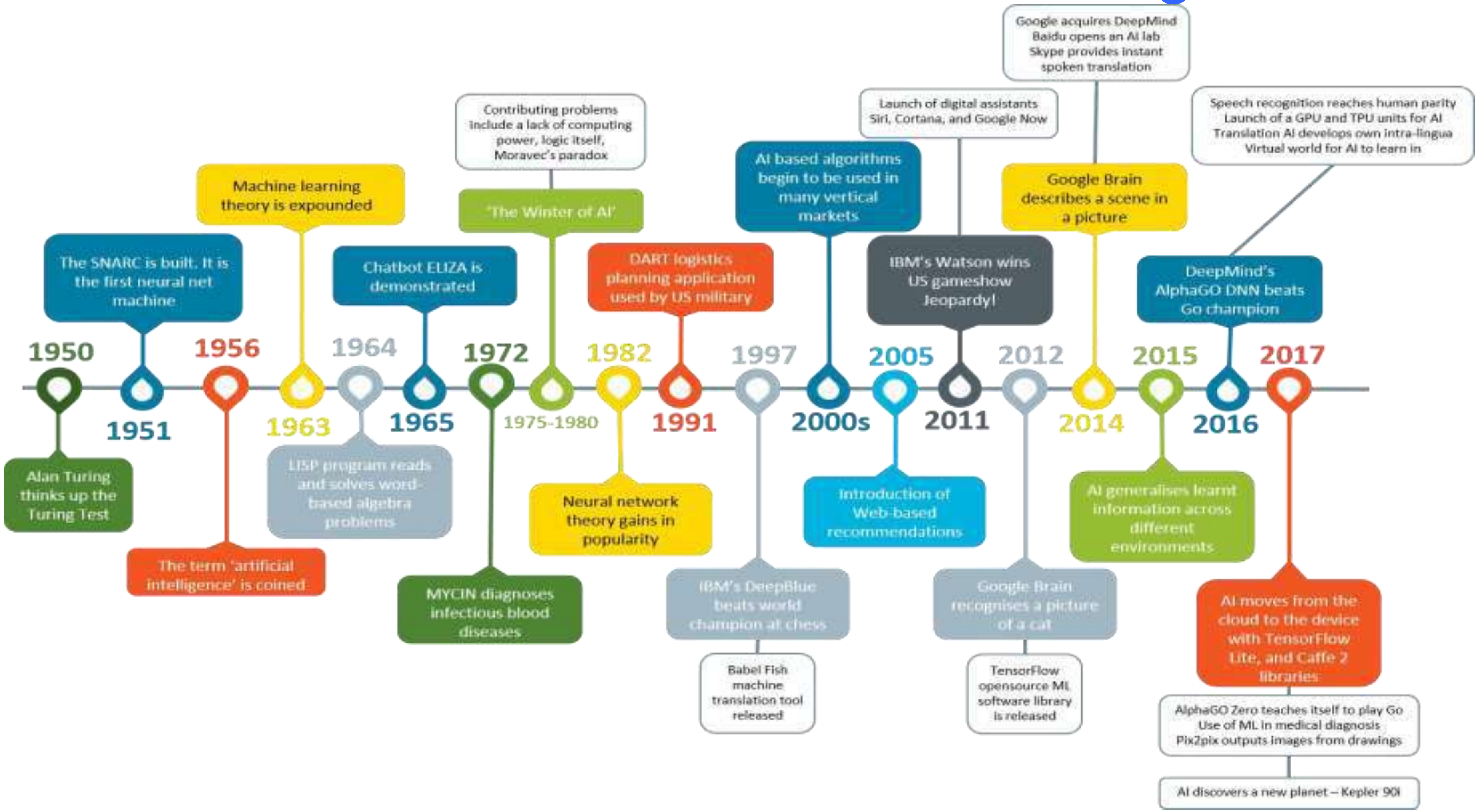


What Does Learning Mean?

A computer program is said to learn from experience E with respect to some task T and some performance measure P, if its performance on T, as measured by P, improves with experience E. — Tom Mitchell, 1997



Timeline of Machine Learning



1950 Alan Turing thinks up the Turing Test

1951 The SNARC is built. It is the first neural net machine

1956 The term 'artificial intelligence' is coined

1963 Machine learning theory is expounded

1964 LISP program reads and solves word-based algebra problems

1965 Chatbot ELIZA is demonstrated

1972 MYCIN diagnoses infectious blood diseases

1975-1980 'The Winter of AI'

Contributing problems include a lack of computing power, logic itself, Moravec's paradox

1982 Neural network theory gains in popularity

1991 DART logistics planning application used by US military

1997 IBM's DeepBlue beats world champion at chess

Babel Fish machine translation tool released

2000s AI based algorithms begin to be used in many vertical markets

2005 Introduction of Web-based recommendations

2011 IBM's Watson wins US gameshow Jeopardy!

Launch of digital assistants Siri, Cortana, and Google Now

2012 Google Brain recognises a picture of a cat

TensorFlow opensource ML software library is released

2014 Google Brain describes a scene in a picture

Google acquires DeepMind
Baidu opens an AI lab
Skype provides instant spoken translation

2015 AI generalises learnt information across different environments

2016 DeepMind's AlphaGO DNN beats Go champion

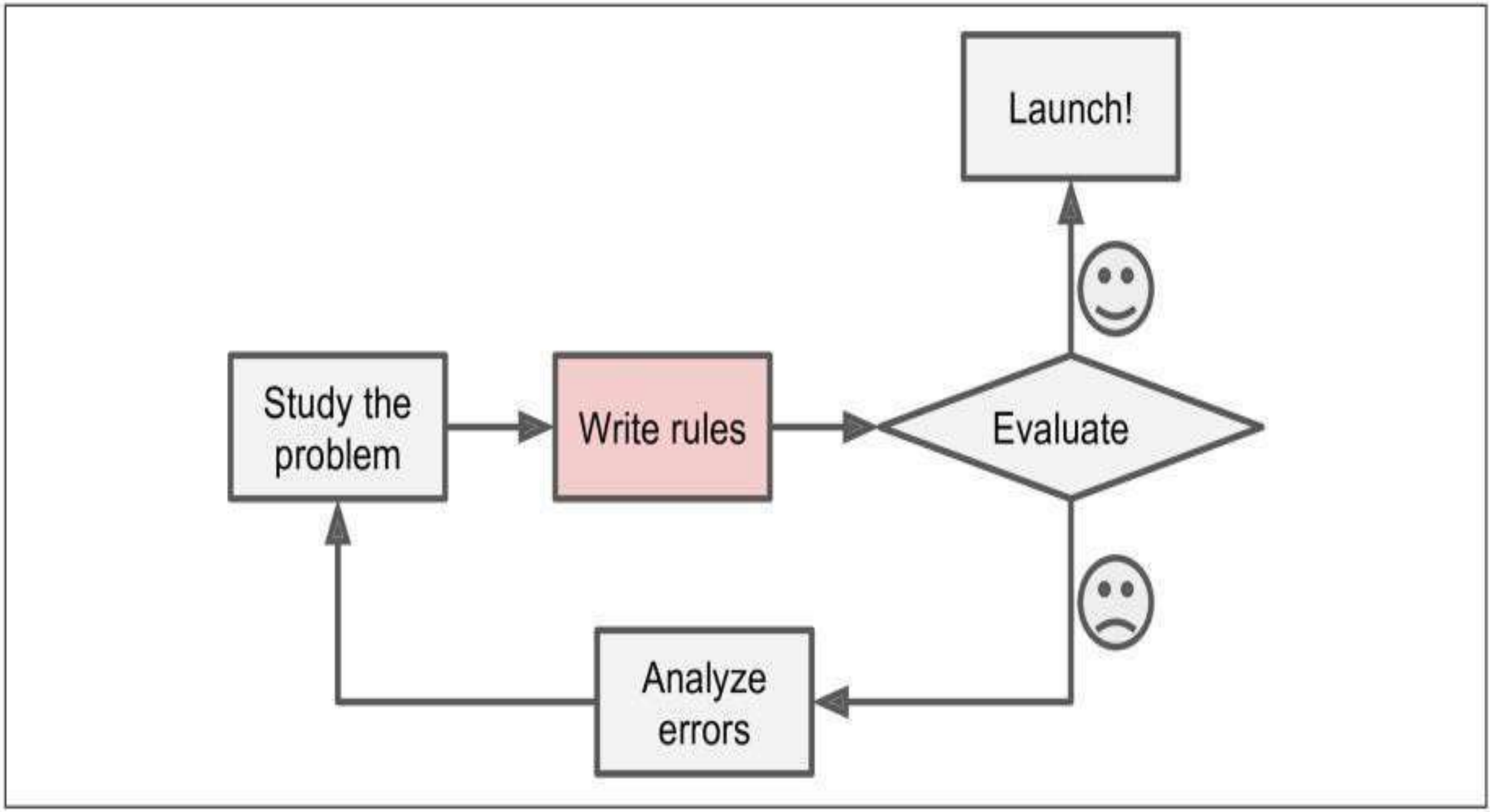
Speech recognition reaches human parity
Launch of a GPU and TPU units for AI
Translation AI develops own intra-lingua
Virtual world for AI to learn in

2017 AI moves from the cloud to the device with TensorFlow Lite, and Caffe 2 libraries

AlphaGO Zero teaches itself to play Go
Use of ML in medical diagnosis
Pix2pix outputs images from drawings

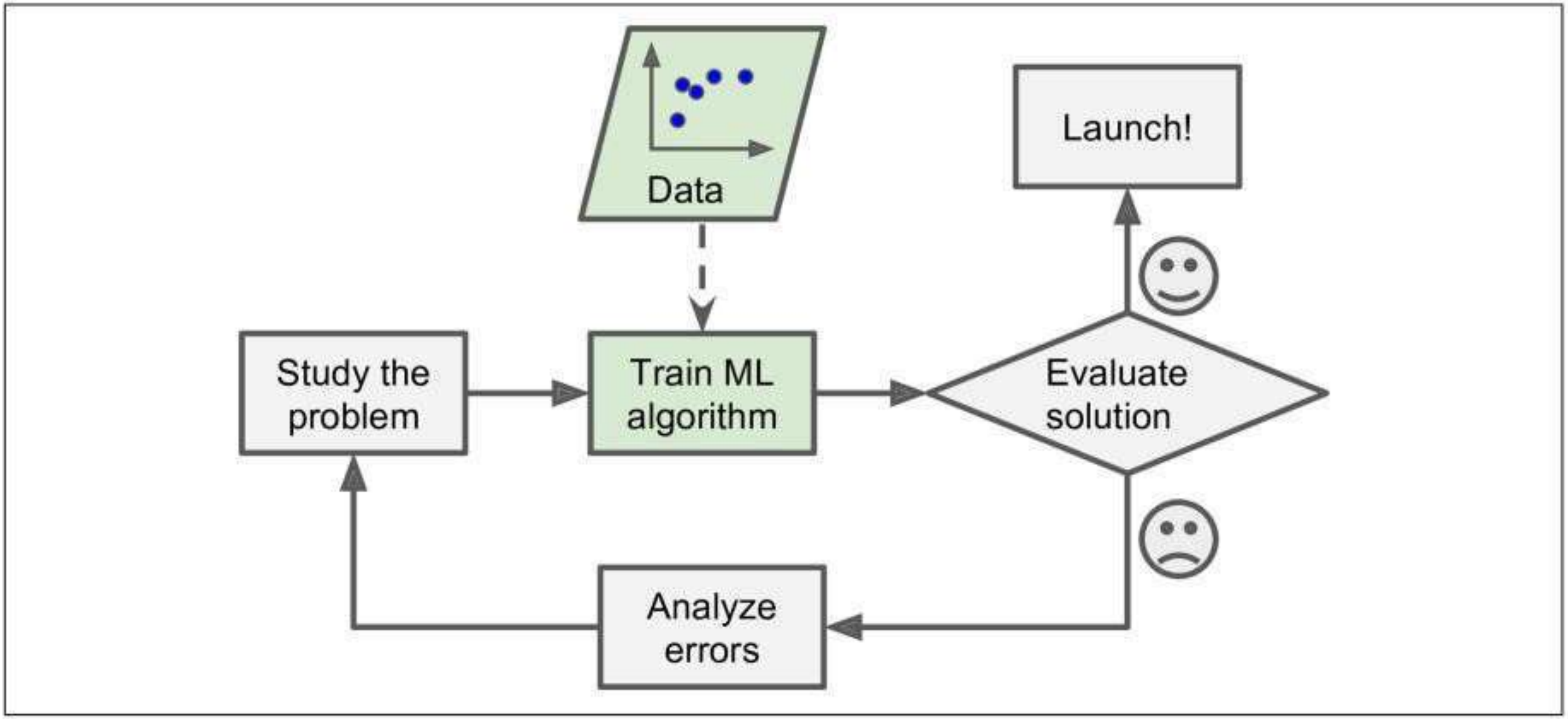
AI discovers a new planet – Kepler 90i

Why Use Machine Learning?



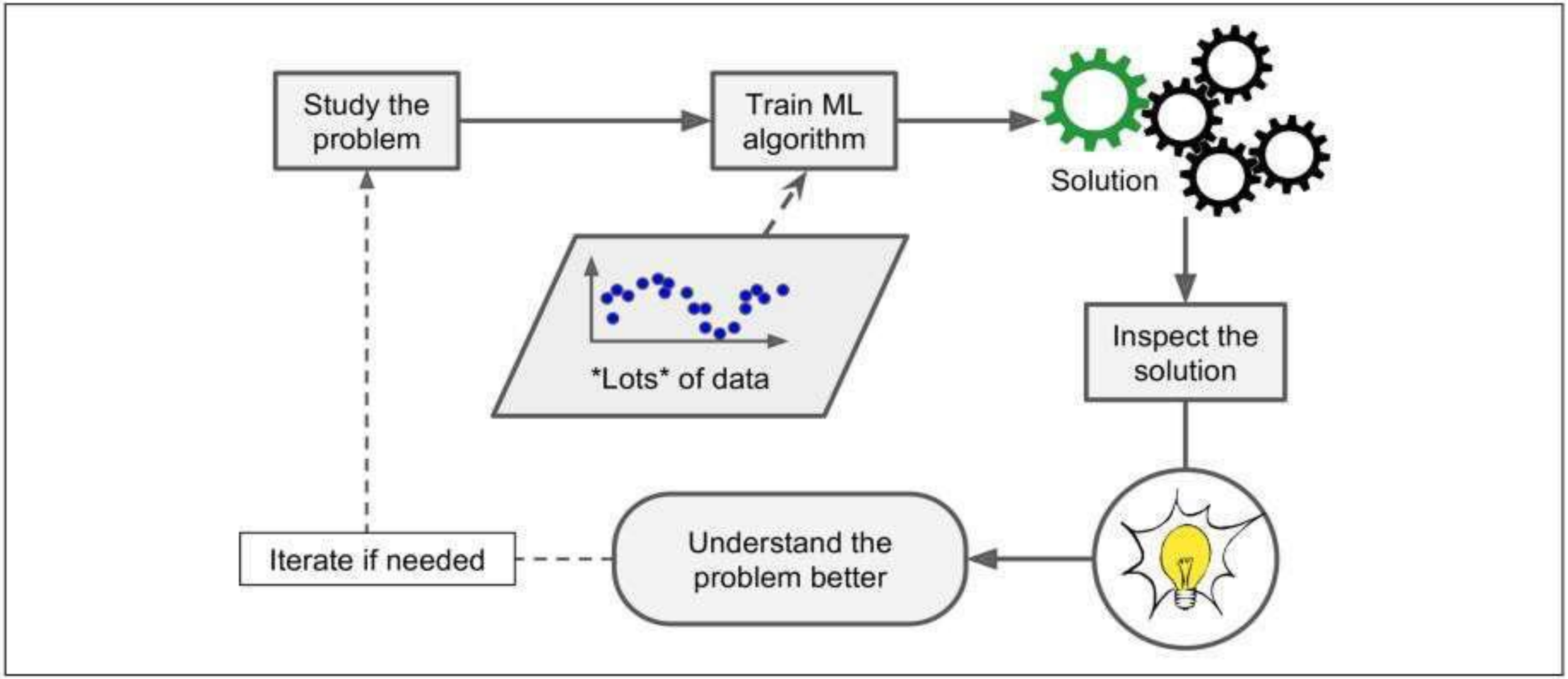
The traditional approach. ***If the problem is not trivial, your program will likely become a long list of complex rules pretty hard to maintain.***

Why Use Machine Learning?



***Machine Learning approach**. The program is much shorter, easier to maintain, and most likely more accurate.*

Why Use Machine Learning?

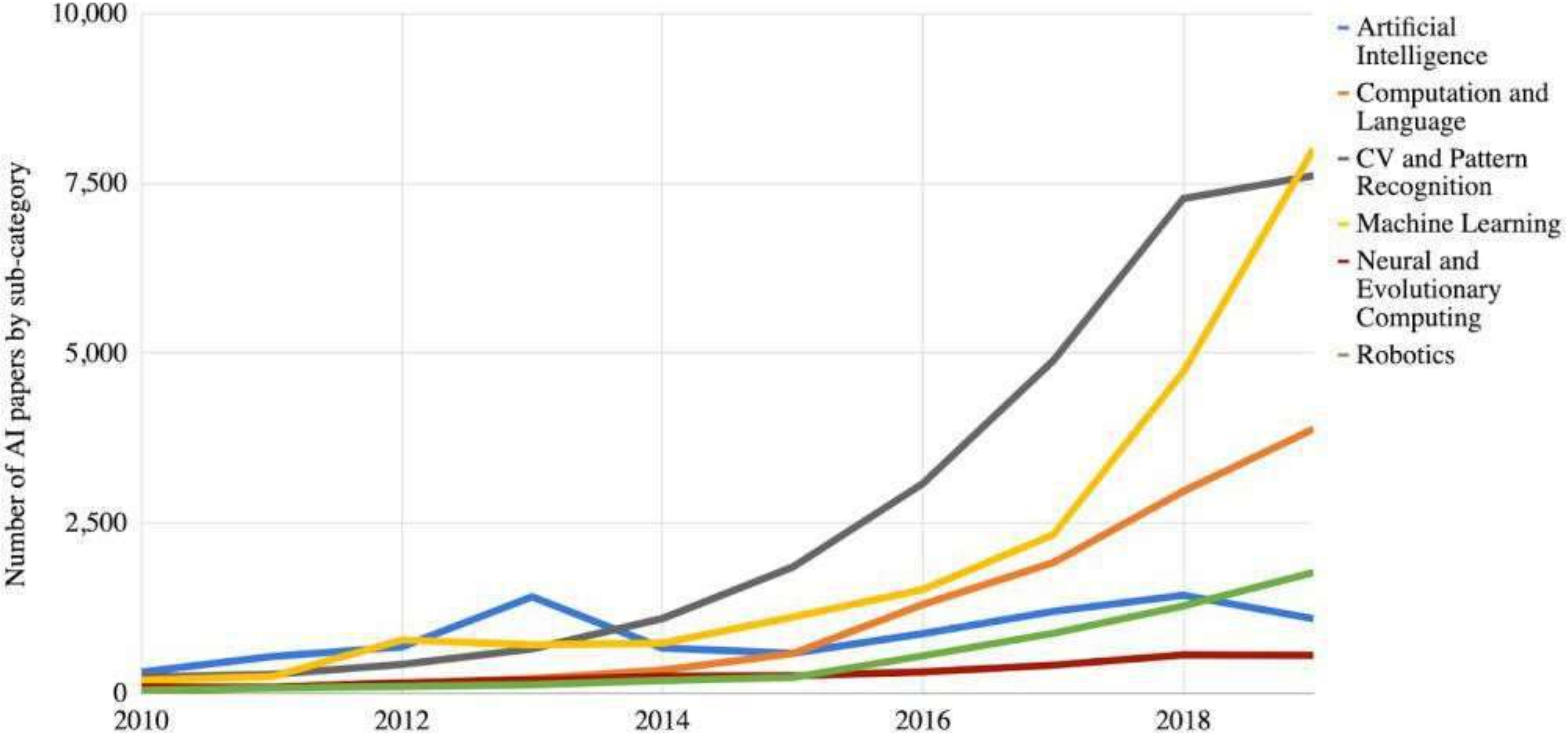


Machine Learning can help humans learn.

Why Use Machine Learning?

Number of AI papers on arXiv, 2010-2019

Source: arXiv, 2019.



AI Index 2019 Annual Report.

Applications of Machine Learning

- **Machine learning is currently the preferred approach in the following domains:**

1) **Speech analysis:** e.g., speech recognition, synthesis.
2) **Computer vision:** e.g., object recognition/detection.
3) **Robotics:** e.g., position/map estimation.
4) **Bio-informatics:** e.g., sequence alignment, genetic analysis.
5) **E-commerce:** e.g., automatic trading, fraud detection.
6) **Financial analysis:** e.g., portfolio allocation, credits.
7) **Medicine:** e.g., diagnosis, therapy conception.
8) **Web:** e.g., Content management, social networks, etc.

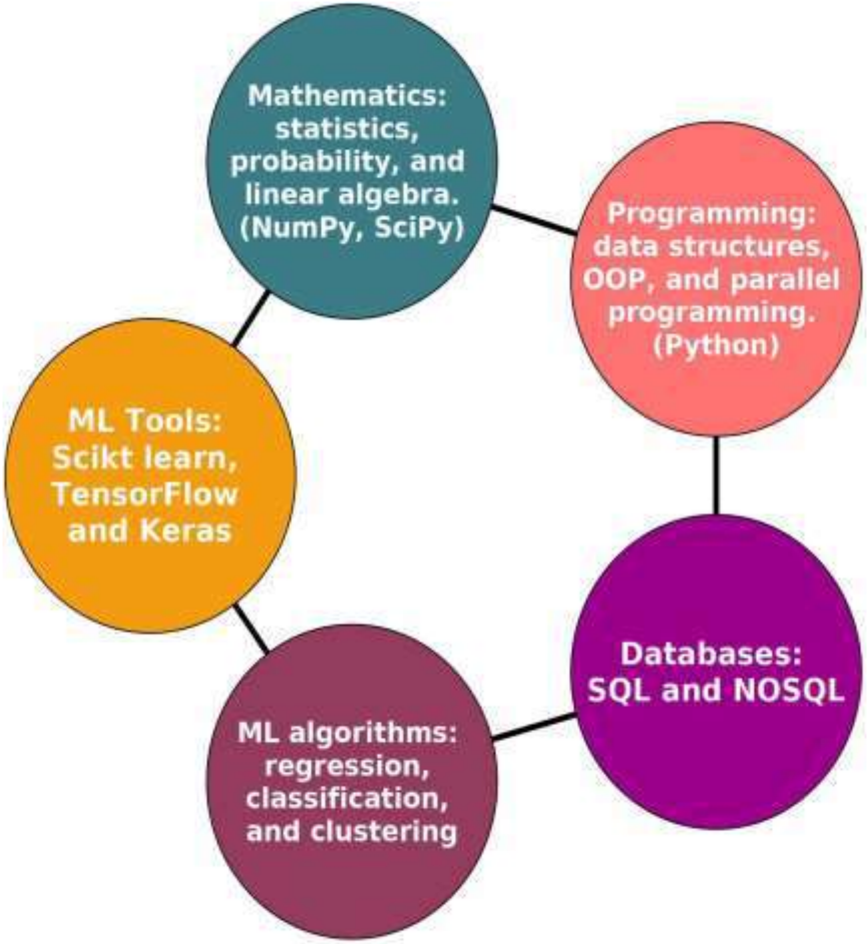
Applications of Machine Learning

To summarize, Machine Learning is great for:

- **Problems for which existing solutions require a lot of hand-tuning or long lists of rules: one Machine Learning algorithm can often simplify code and perform better.**

- **Complex problems for which there is no good solution at all using a traditional approach: the best Machine Learning techniques can find a solution.**

How to get started with ML


Mathematics: statistics, probability, and linear algebra. (NumPy, SciPy)
Programming: data structures, OOP, and parallel programming. (Python)
Databases: SQL and NOSQL
ML algorithms: regression, classification, and clustering
ML Tools: Scikt learn, TensorFlow and Keras

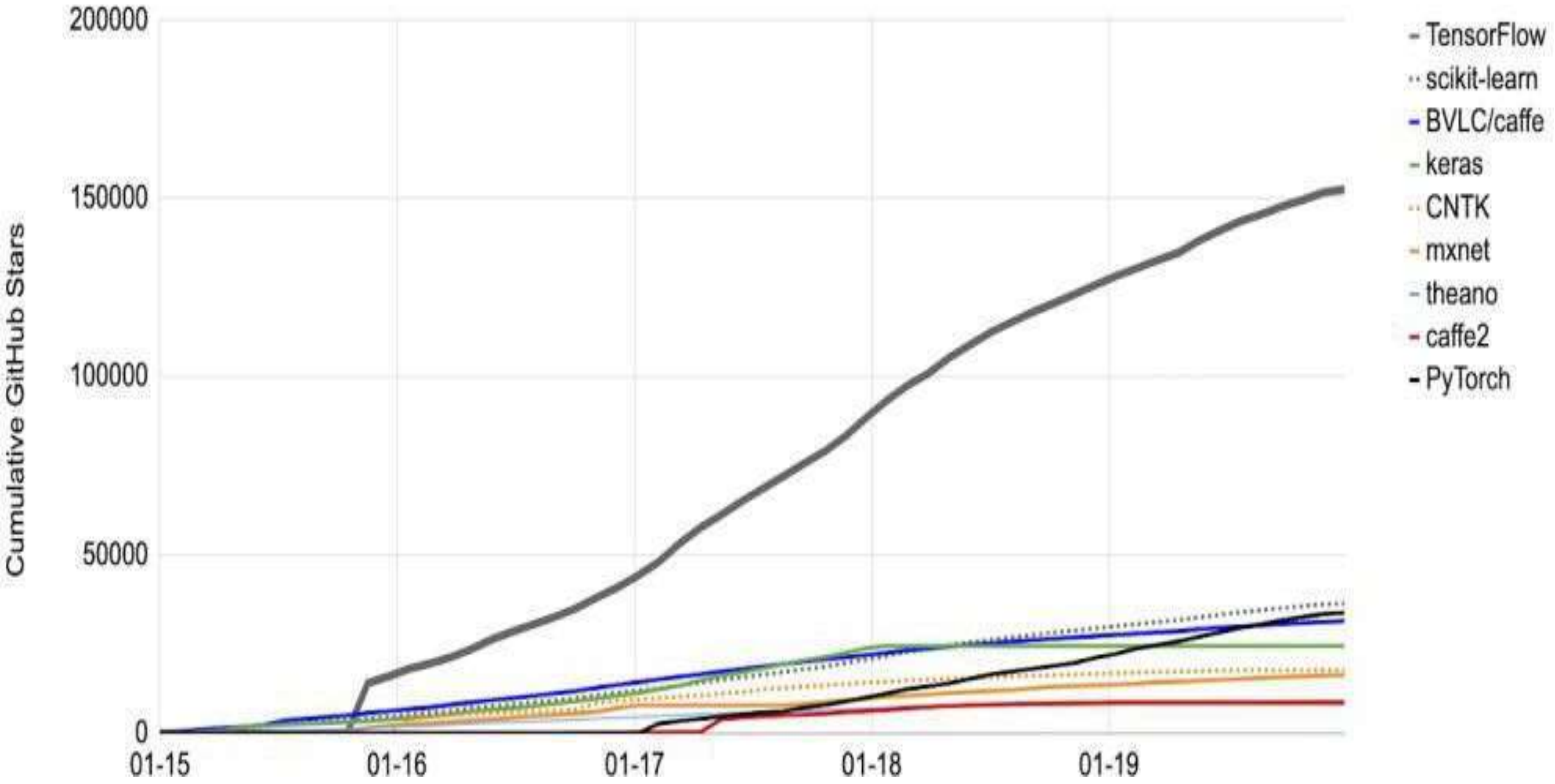
How to get started with Machine Learning

1) **Mathematics: statistics, probability, and linear algebra.(NumPy, SciPy)**
2) **Programming: data structures, OOP, and parallel programming. (Python)**
3) **Databases: SQL and NOSQL.**
4) **ML algorithms: regression, classification, and clustering.**
5) **ML Tools: Scikt learn, TensorFlow and Keras.**

How to get started with ML

Cumulative GitHub stars by AI library (2015—2019)

Source: Github, 2019.



Machine Learning Vocabulary 1

1) **Examples:** Items or instances of data used for learning or evaluation. In our spam problem, these examples correspond to the collection of email messages we will use for learning and testing.

2) **Training sample:** Examples used to train a learning algorithm. In our spam problem, the training sample consists of a set of email examples along with their associated labels.

3) **Labels:** Values or categories assigned to examples. In classification problems, examples are assigned specific categories, for instance, the spam and non-spam categories in our binary classification problem. In regression, items are assigned real-valued labels.

Machine Learning Vocabulary 2

4) **Features:** The set of attributes, often represented as a vector, associated to an example. In the case of email messages, some relevant features may include the length of the message, the name of the sender, various characteristics of the header, the presence of certain keywords in the body of the message, and so on.

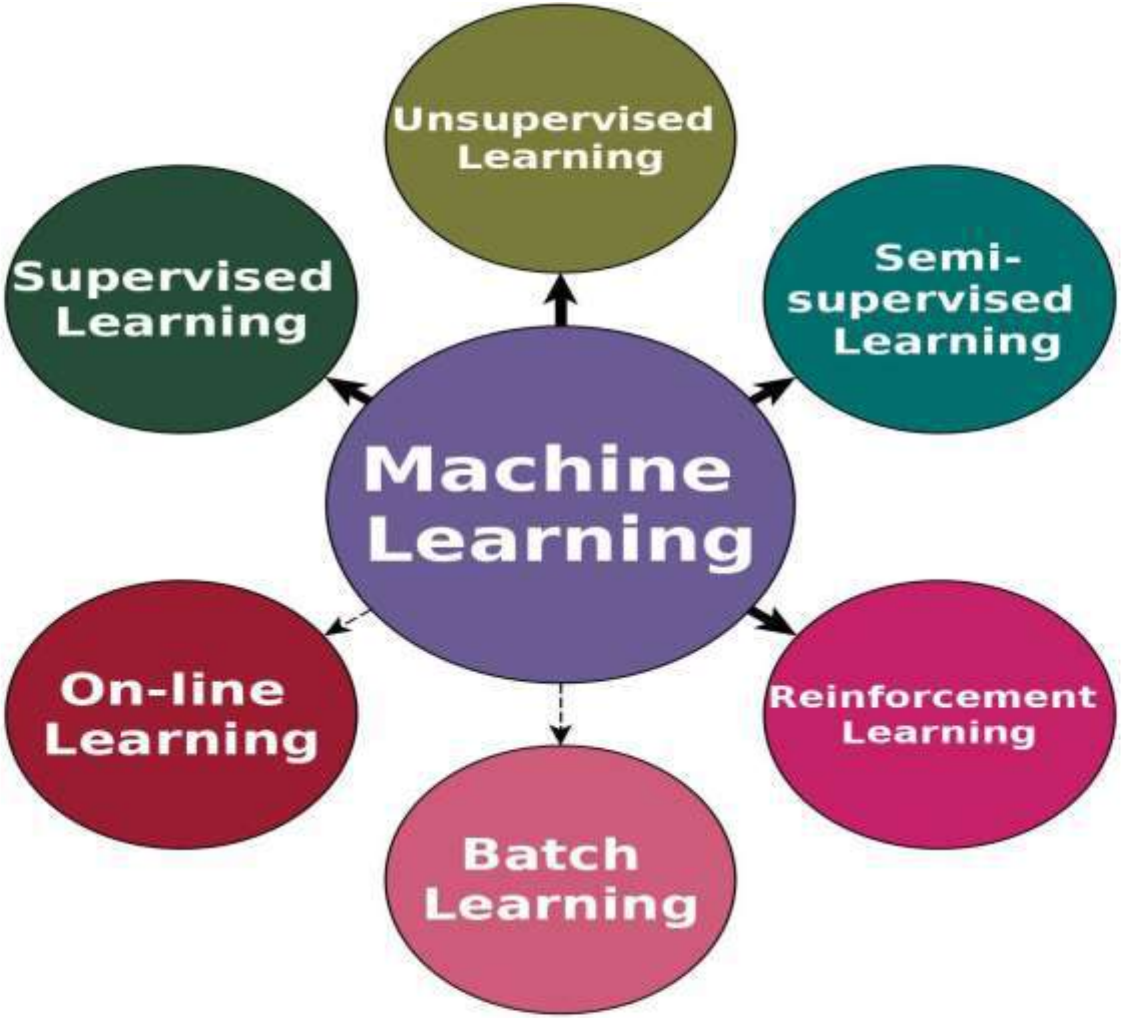
5) **Test sample:** Examples used to evaluate the performance of a learning algorithm. The test sample is separate from the training and validation data and is not made available in the learning stage. In the spam problem, the test sample consists of a collection of email examples for which the learning algorithm must predict labels based on features. These predictions are then compared with the labels of the test sample to measure the performance of the algorithm.

6) **Loss function:** A function that measures the difference, or loss, between a predicted label and a true label.

Types of Machine Learning Systems

- There are so many different types of Machine Learning systems that it is useful to classify them in broad categories based on:
- Whether or not they are trained with **human supervision** (**supervised**, **unsupervised**, **semisupervised**, and **Reinforcement Learning**).
- Whether or not they can learn **incrementally on the fly** (**online** versus **batch learning**).
- Whether they work by simply **comparing new data points** to known data points, or instead detect patterns in the training data and build a **predictive model**, much like scientists do (instance-based versus model-based learning).

Types of Machine Learning Systems



The types of machine learning

Hichem Felouat

Types of Machine Learning Systems

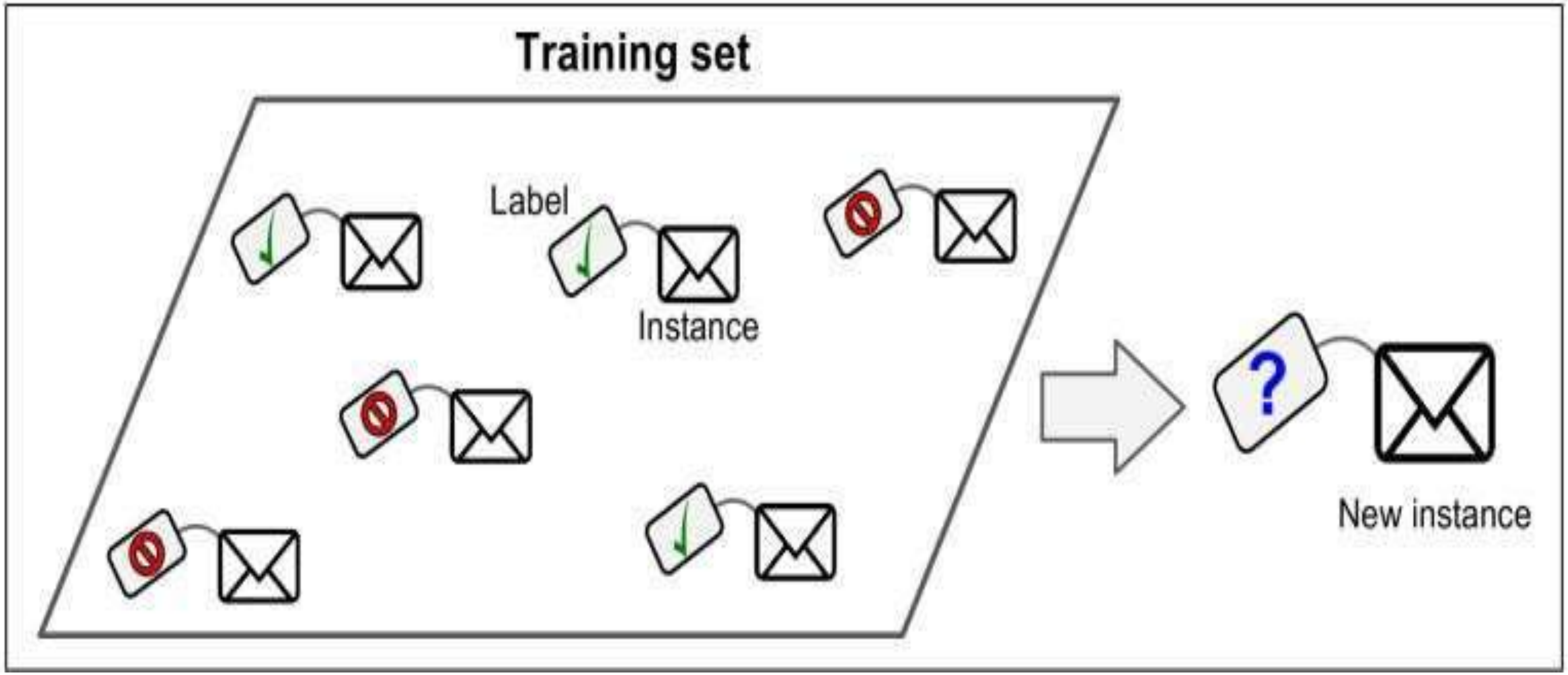
Supervised learning :

In supervised learning, the training data you feed to the algorithm includes the **desired solutions**, called **labels**.

$$\mathcal{D} = \{(\mathbf{x}^{(1)}, y^{(1)}), (\mathbf{x}^{(2)}, y^{(2)}), \cdots, (\mathbf{x}^{(n)}, y^{(n)})\}$$

- When ***y*** is real, we talk about ***regression***.
- When ***y*** is discrete, we talk about ***classification***.

Types of Machine Learning Systems



A labeled training set for supervised learning.

Types of Machine Learning Systems

Here are some of the most important supervised learning algorithms:

- k-Nearest Neighbors
- Linear Regression
- Logistic Regression
- Support Vector Machines (SVMs)
- Decision Trees and Random Forests
- Neural networks*

Types of Machine Learning Systems

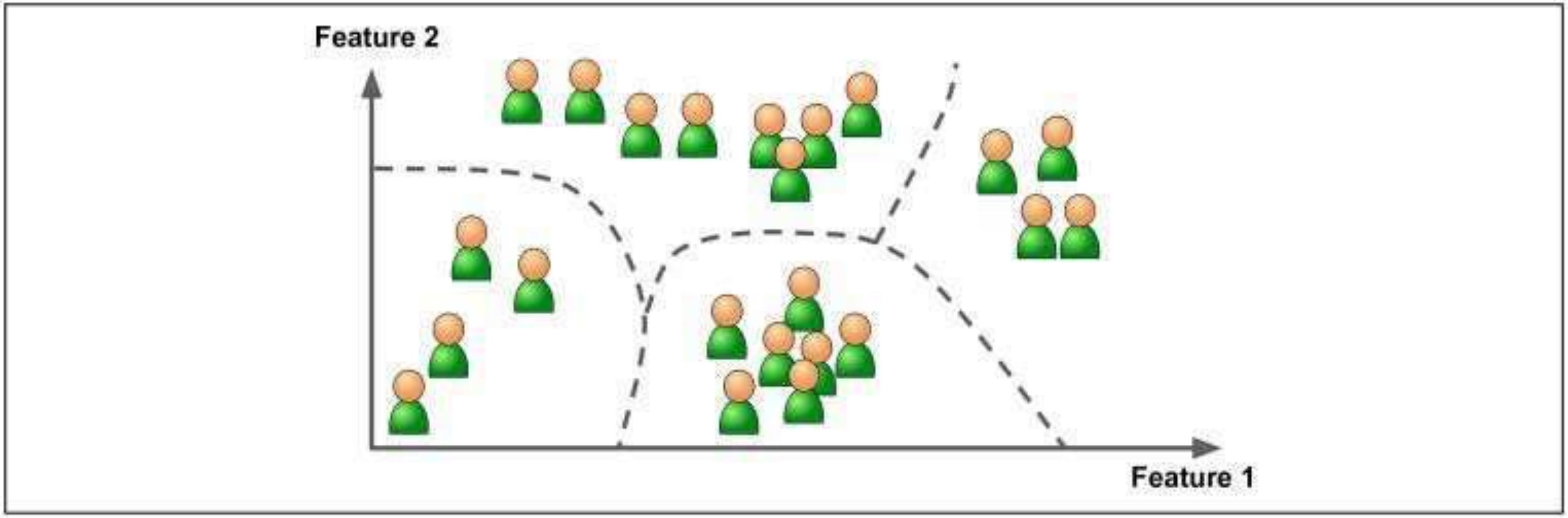
Unsupervised Learning:

In unsupervised learning, as you might guess, **the training data is unlabeled**. The system tries to learn without a teacher.

$$\mathcal{D} = \left\{x^{(1)}, x^{(2)}, \cdots, x^{(n)}\right\}$$

No labels are given to the learning algorithm, leaving it on its own to explore or find structure in the data.

Types of Machine Learning Systems



An unlabeled training set for unsupervised learning.

Types of Machine Learning Systems

Here are some of the most important unsupervised learning algorithms:

- Clustering
- Visualization and dimensionality reduction

Types of Machine Learning Systems

Semi-Supervised Learning :

Some algorithms can deal with partially labeled training data, usually a lot of unlabeled data and a little bit of labeled data. This is called semi-supervised learning.

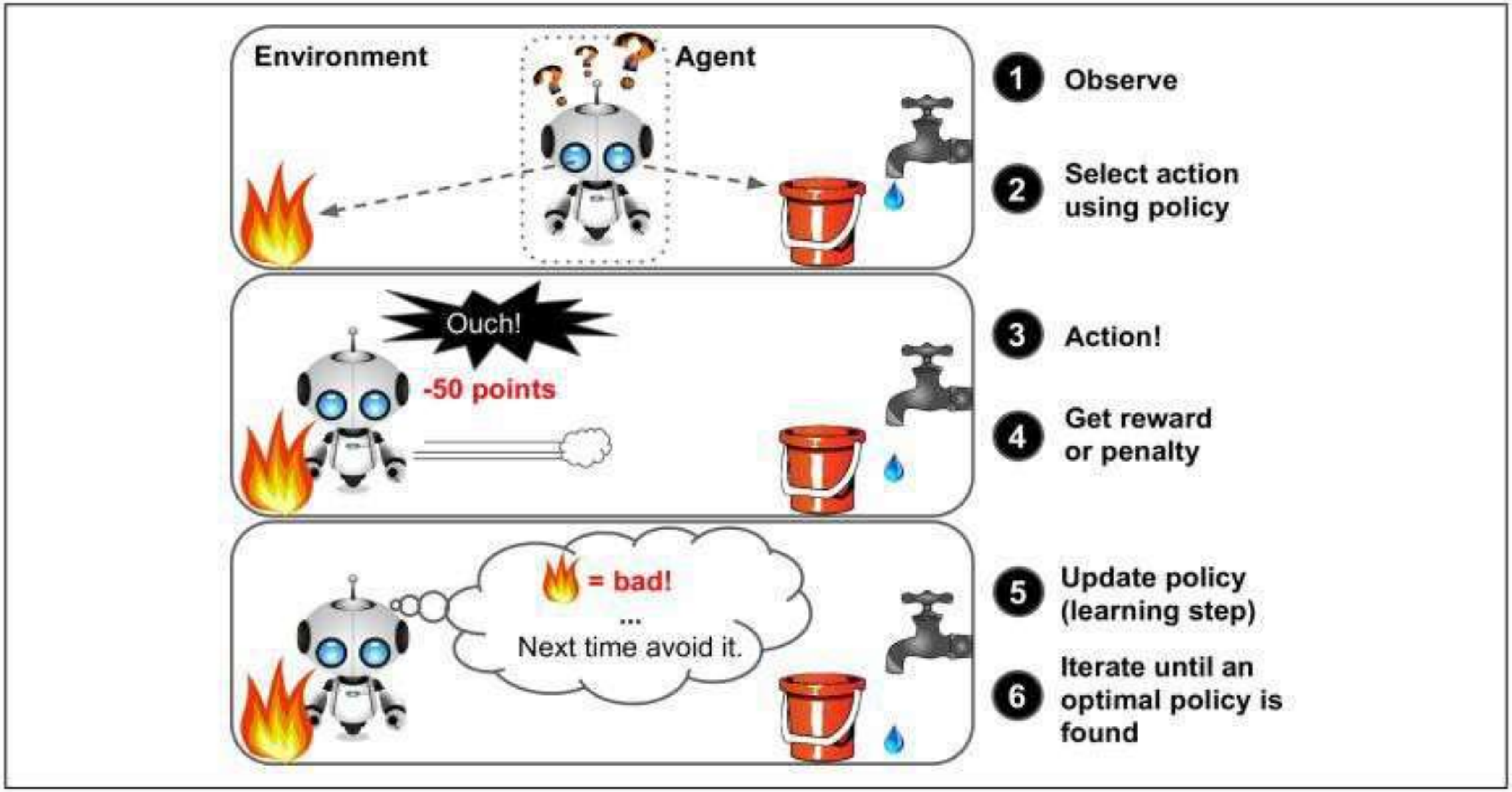
Most semi-supervised learning algorithms are combinations of unsupervised and supervised algorithms.

Types of Machine Learning Systems

Reinforcement Learning :

- The learning system called an **agent** in this context.
- Can **observe** the environment, **select and perform actions**, and **get rewards** in return (or penalties in the form of negative rewards).
- It must then learn by itself what is **the best strategy**, called a **policy**, to get the most reward over time.
- A policy defines what action the agent should choose when it is in a given situation.

Types of Machine Learning Systems



1. Observe
2. Select action using policy
3. Action!
4. Get reward or penalty
5. Update policy (learning step)
6. Iterate until an optimal policy is found

Reinforcement Learning

Types of Machine Learning Systems

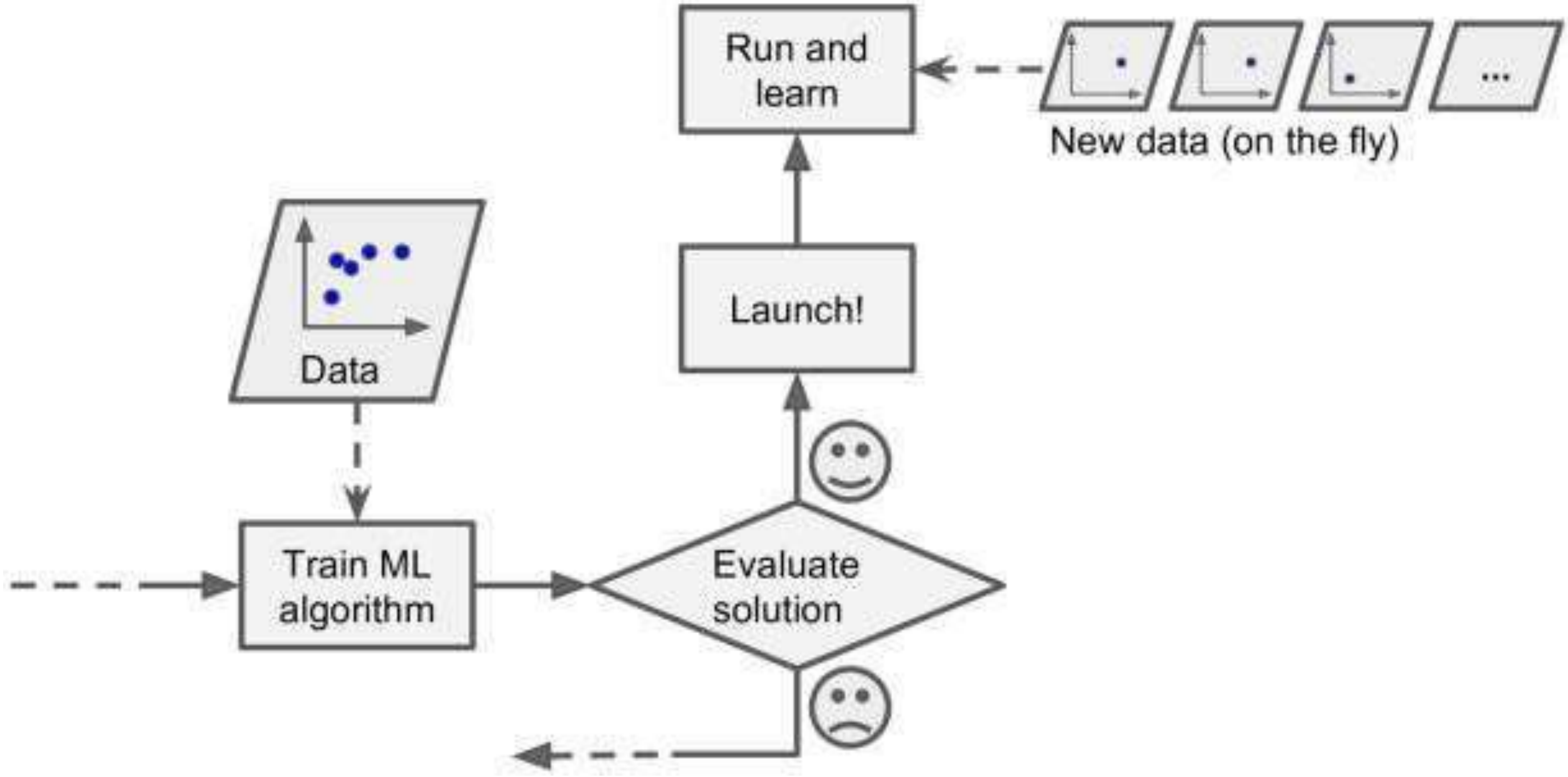
- **Batch learning:**

- ***In batch learning, the system is incapable of learning incrementally:*** **it must be trained using all the available data.** This will generally take a lot of time and computing resources, so it is typically **done offline**. First, the system is trained, and then it is launched into production and runs without learning anymore; it just applies what it has learned. This is called offline learning.

Types of Machine Learning Systems

- **On-line learning:**
- In online learning, **you train the system incrementally by feeding it data instances sequentially**, either individually or by small groups called mini batches. Each learning step is fast and cheap, so **the system can learn about new data on the fly**, as it arrives.

Types of Machine Learning Systems



Online learning

Instance-Based VS Model-Based Learning

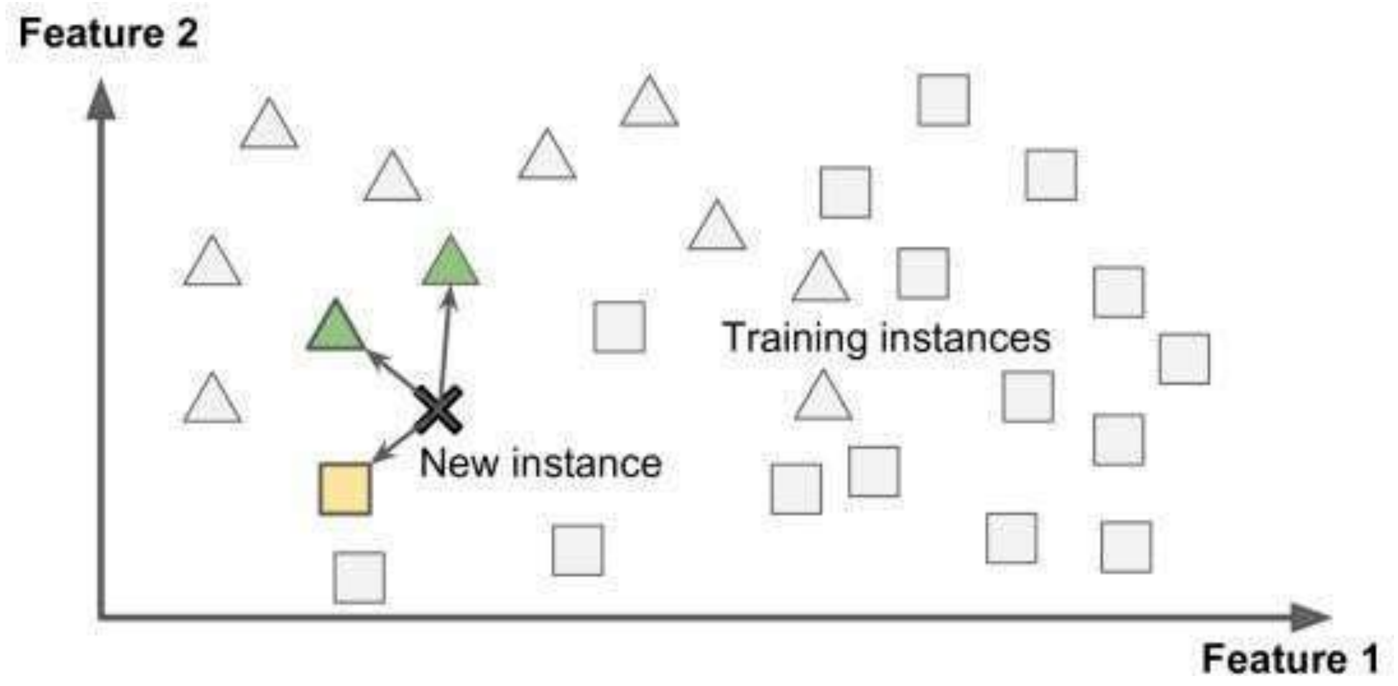
One more way to categorize Machine Learning systems is by how they generalize. **Most Machine Learning tasks are about making predictions.** This means that given a number of training examples, the system needs to be able to generalize to examples it has never seen before.

Having a good performance measure on the training data is good, but insufficient; **the true goal is to perform well on new instances**. There are two main approaches to generalization: **instance-based learning** and **model-based learning**.

Instance-Based VS Model-Based Learning

Instance-based learning:

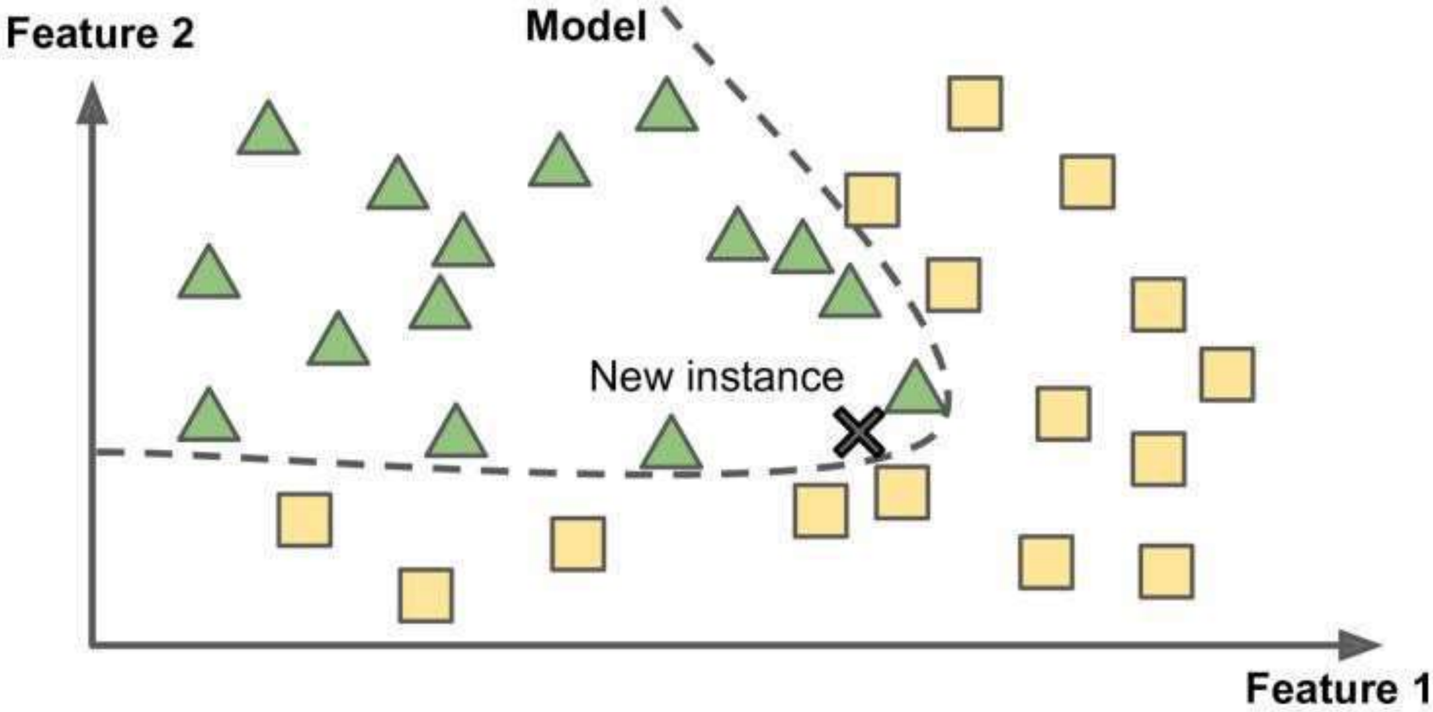
The system learns the examples by heart, then generalizes to new cases using a similarity measure.



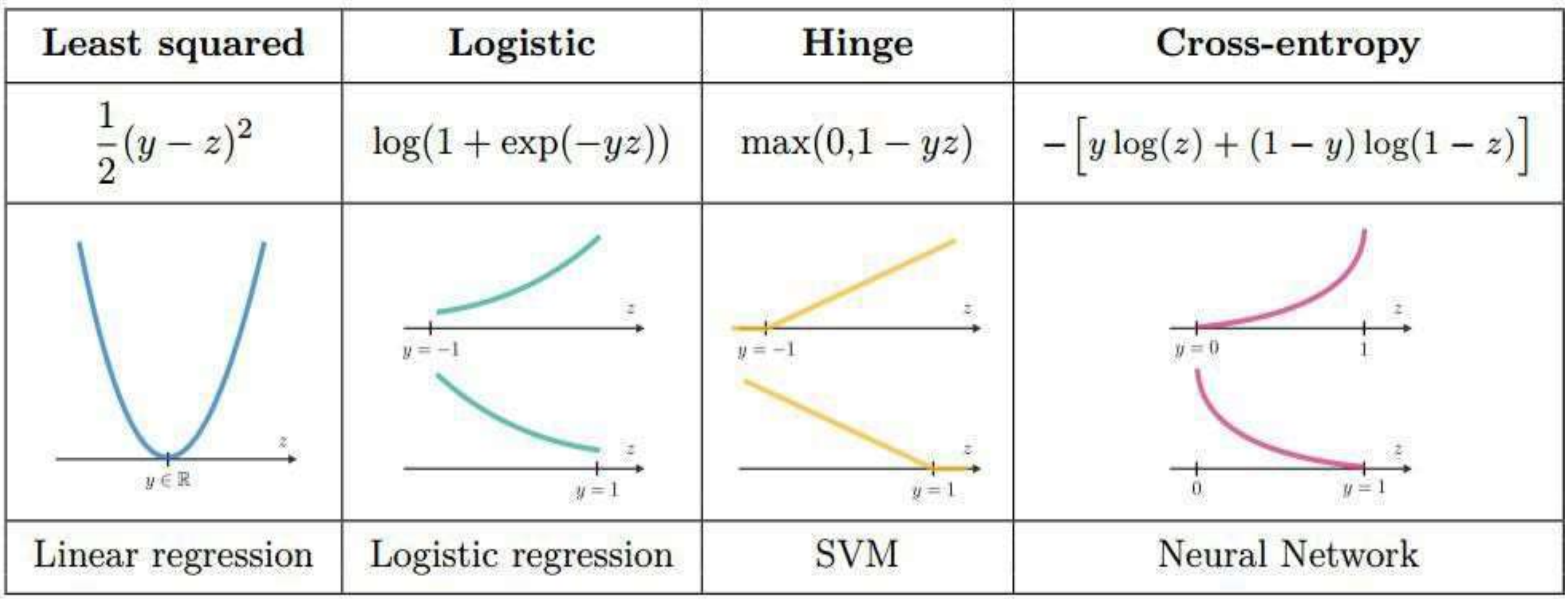
Instance-Based VS Model-Based Learning

Model-based learning:

Build a model of these examples, then use that model to make predictions.



Loss Function



Least squared	Logistic	Hinge	Cross-entropy
$\frac{1}{2}(y-z)^2$	$\log(1+\exp(-yz))$	$\max(0,1-yz)$	$-\left[y\log(z)+(1-y)\log(1-z)\right]$
Linear regression	Logistic regression	SVM	Neural Network

z
$y \in \mathbb{R}$
z
$y=-1$
z
$y=1$
z
$y=-1$
z
$y=1$
z
$y=0$
1
z
0
$y=1$

The loss function computes the error for a single training example, while the cost function is the average of the loss functions of the entire training set.

Machine Learning Vocabulary 3

- **Hyperparameters :** are configuration variables that are external to the model and whose values cannot be estimated from data. That is to say, they can not be learned directly from the data in standard model training. They are almost always specified by the machine learning engineer prior to training.

- **Regression:** this is the problem of predicting a real value for each item. Examples of regression include prediction of stock values or that of variations of economic variables.

- **Classification:** this is the problem of assigning a category to each item.

- **Clustering:** this is the problem of partitioning a set of i tems into homogeneous subsets.

In Summary

1) You studied the data.

2) You selected a model.

3) You trained it on the training data.

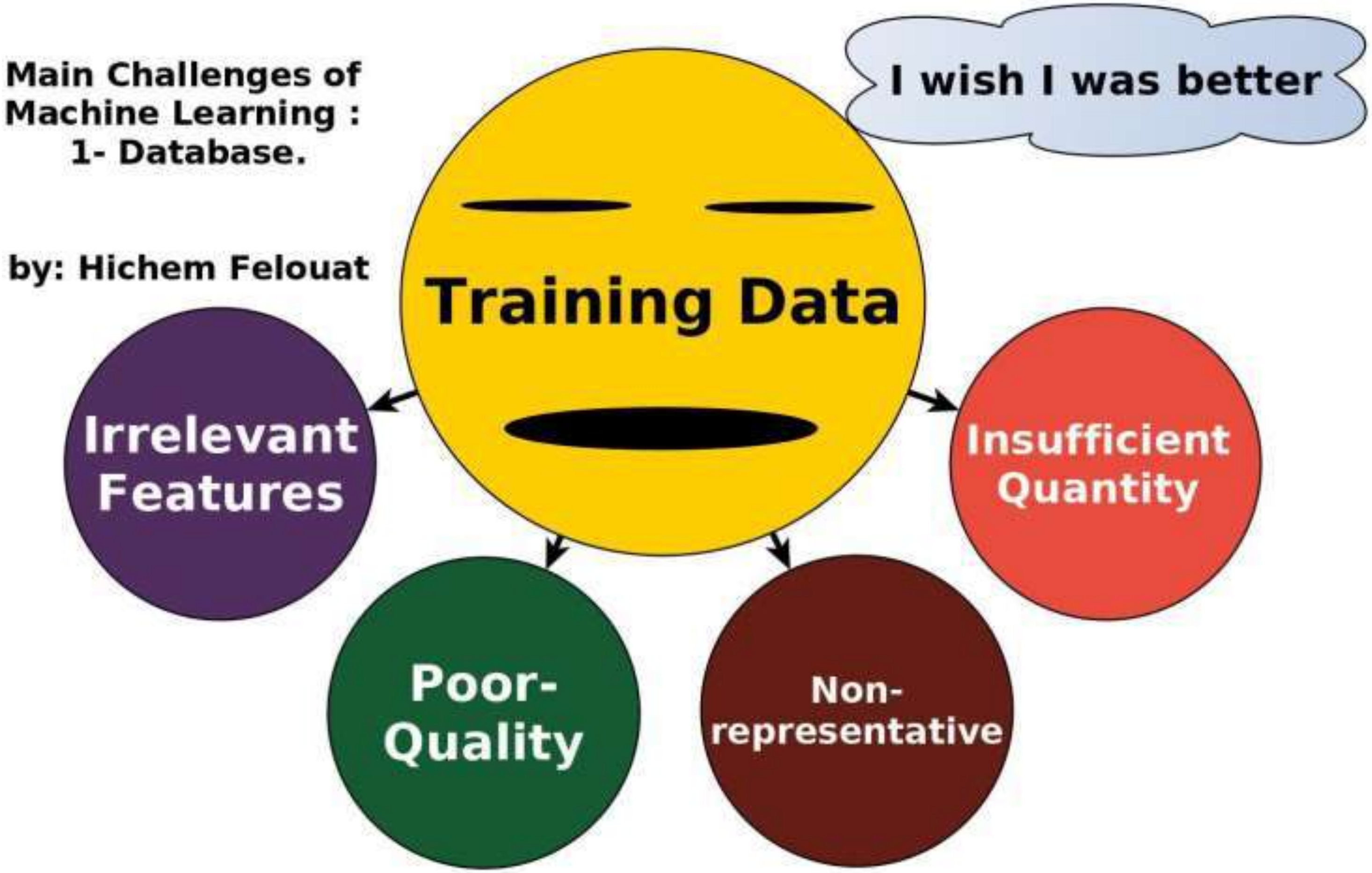
4) Finally, you applied the model to make predictions on new cases.

Main Challenges of Machine Learning

In short, since your main task is to select a learning algorithm and train it on some data, the two things that can go wrong are **“bad data”** and **“bad algorithm”**.

Main Challenges of Machine Learning

1- Database



Main Challenges of Machine Learning 1- Database

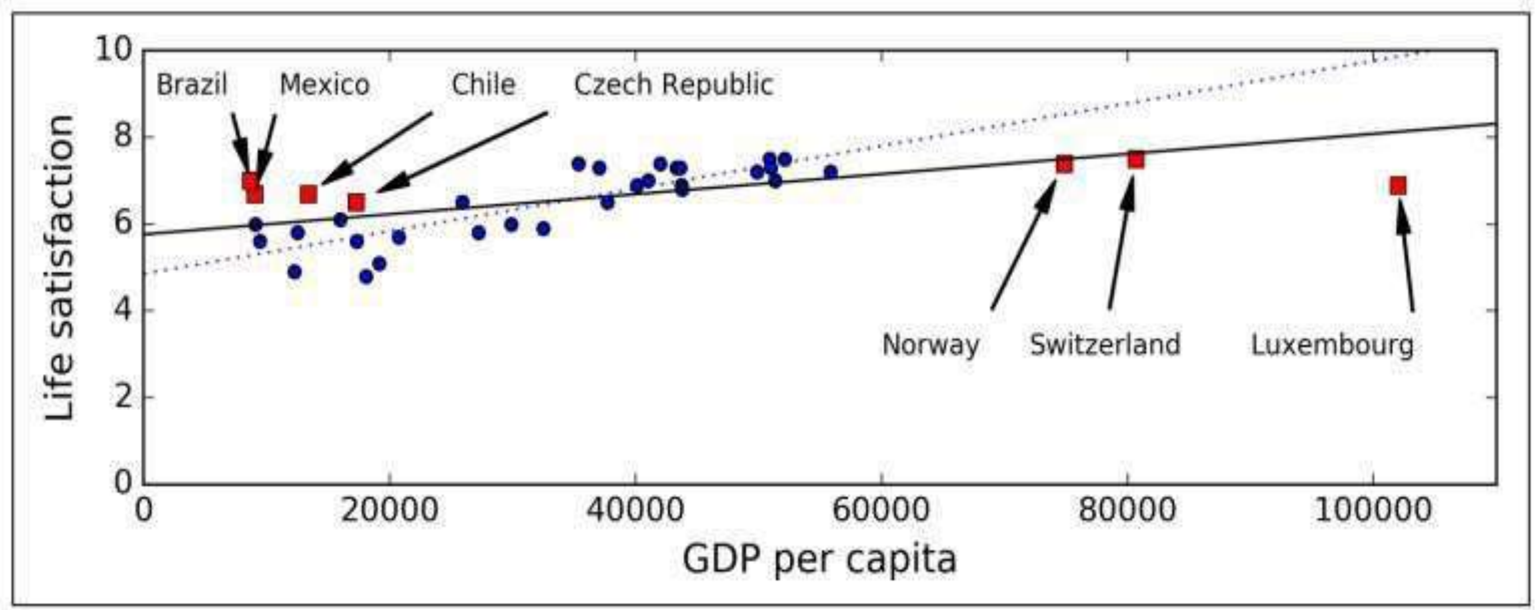
1- Insufficient Quantity of Training Data :

Machine Learning takes a lot of data for most Machine Learning algorithms to work properly. Even for very simple problems you typically need thousands of examples, and for complex problems such as image or speech recognition you may need millions of examples (unless you can reuse parts of an existing model).

Main Challenges of Machine Learning 1- Database

2) Non-representative Training Data:

In order to generalize well, it is crucial that your training data be representative of the new cases you want to generalize to. This is true whether you use instance-based learning or model-based learning.



Main Challenges of Machine Learning 1- Database

3) Poor-Quality Data:

If your training data is full of errors, outliers, and noise (e.g., **due to poor quality measurements**), it will make it harder for the system to detect the underlying patterns, so your system is less likely to perform well. **It is often well worth the effort to spend time cleaning up your training data. The truth is, most data scientists spend a significant part of their time doing just that.** For example:

1) **If some instances are clearly outliers, it may help to simply discard them or try to fix the errors manually.**
2) **If some instances are missing a few features (e.g., 5% of your customers did not specify their age), you must decide whether you want to ignore this attribute altogether, ignore these instances, fill in the missing values (e.g., with the median age), or train one model with the feature and one model without it, and so on.**

Main Challenges of Machine Learning 1- Database

4) Irrelevant Features:

Your system will only be capable of learning if the training data contains enough relevant features and not too many irrelevant ones. A critical part of the success of a Machine Learning project is coming up with a good set of features to train on. This process, called ***feature engineering***, involves:

1) **Feature selection:** selecting the most useful features to train on among existing features.
2) **Feature extraction:** combining existing features to produce a more useful one (dimensionality reduction algorithms can help).
3) **Creating new features** by gathering new data.

Main Challenges of Machine Learning

2- Algorithm

1) Overfitting the Training Data:

Overfitting happens when a model **learns the detail and noise in the training data** to the extent that it negatively impacts the performance of the model on new data. This means that the noise or random fluctuations in the training data is picked up and learned as concepts by the model. The problem is that these concepts do not apply to new data and negatively impact the models ability to generalize.

The model performs well on the training data, but it does not generalize well.

Main Challenges of Machine Learning

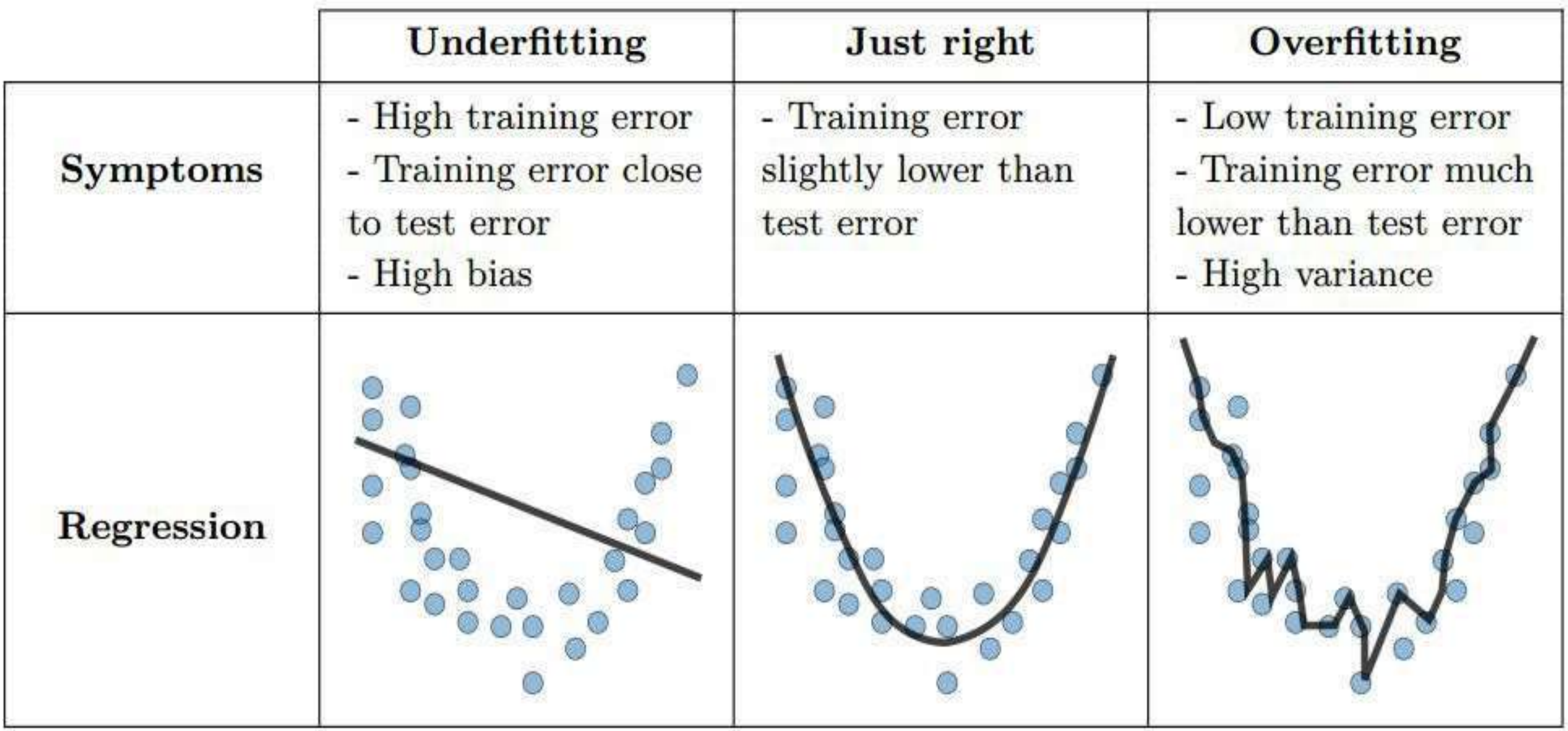
2- Algorithm

2) Underfitting the Training Data:

Underfitting is the opposite of overfitting: it occurs when **your model is too simple** to learn the underlying structure of the data.

Main Challenges of Machine Learning

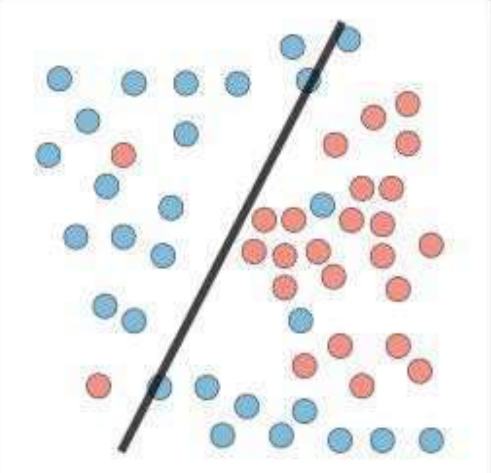
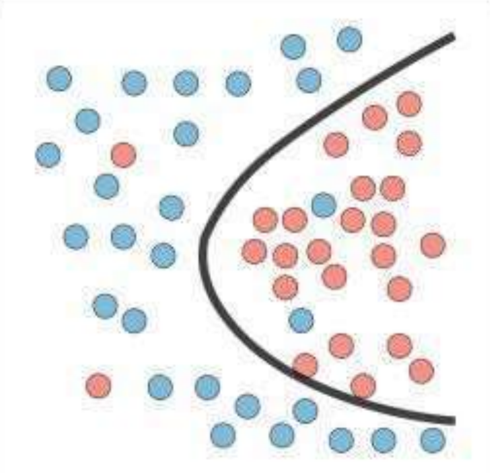
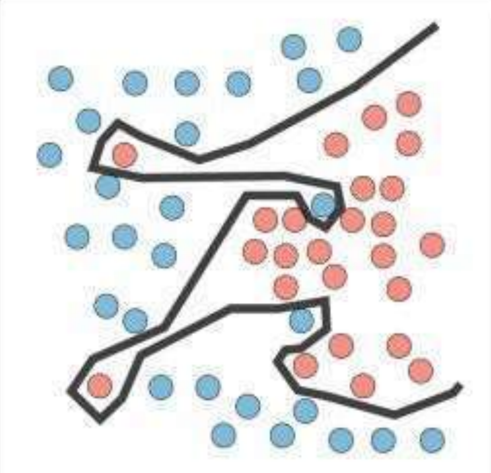
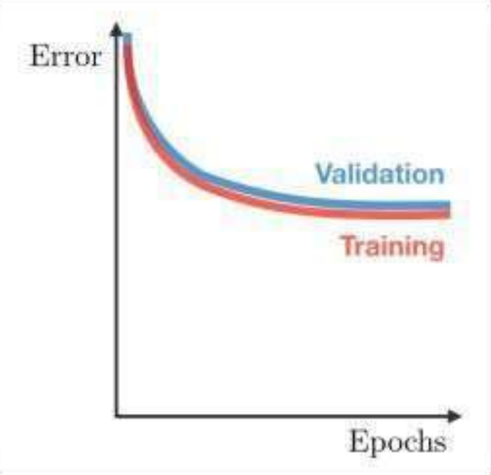
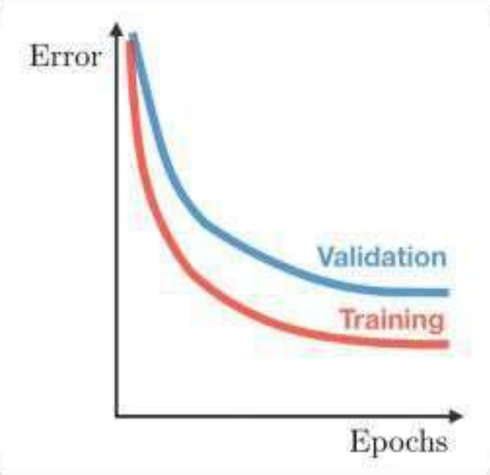
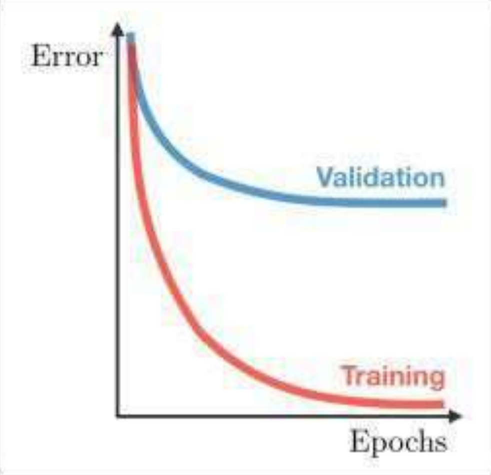
2- Algorithm



	Underfitting	Just right	Overfitting
Symptoms	- High training error - Training error close to test error - High bias	- Training error slightly lower than test error	- Low training error - Training error much lower than test error - High variance
Regression			

Main Challenges of Machine Learning

2- Algorithm

Classification			
Deep learning			

How to Avoid Underfitting and Overfitting

Underfitting :

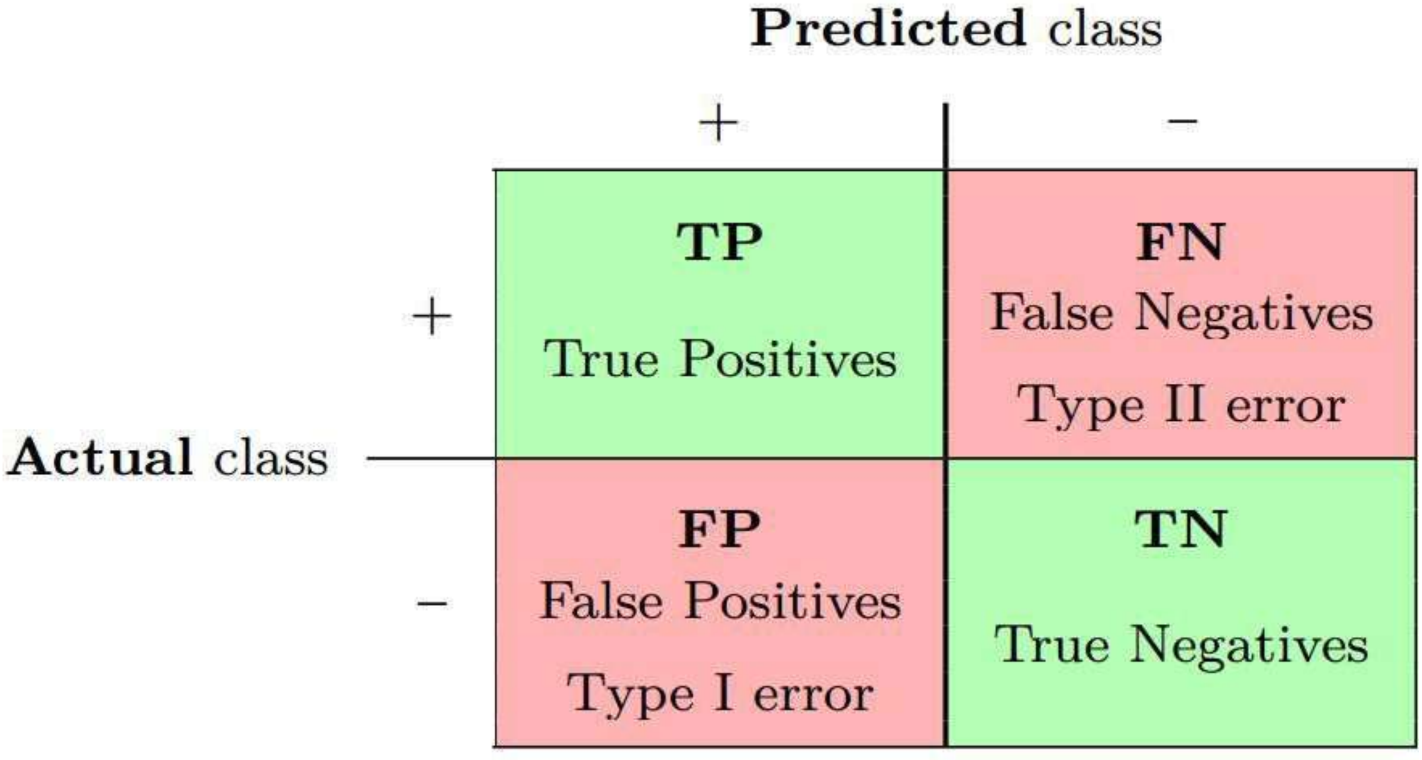
- Complexify model
- Add more features
- Train longer

Overfitting :

- validation
- Perform regularization
- Get more data
- Remove/Add some features

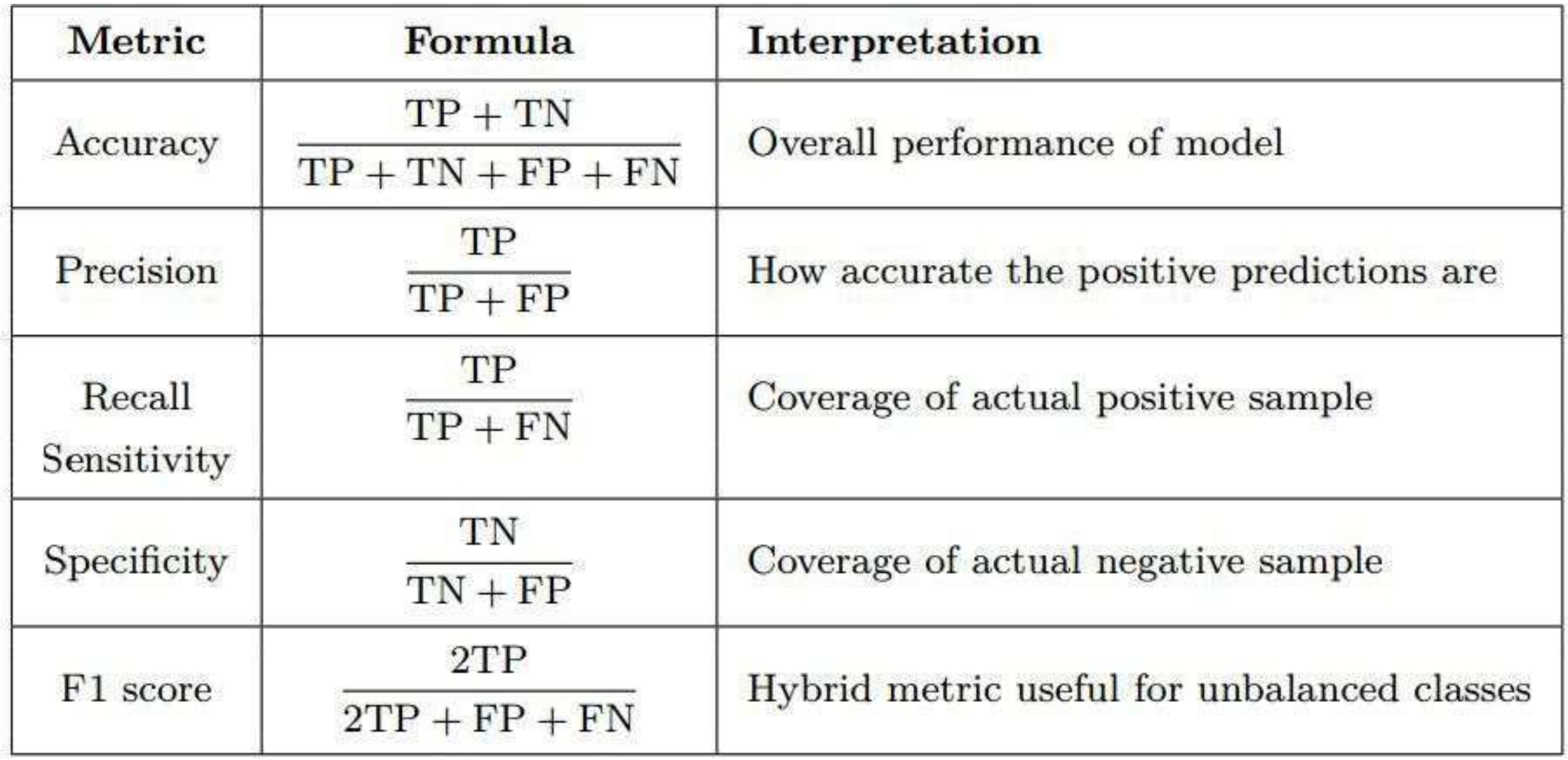
Common Classification Model Evaluation Metrics : Confusion Matrix

The confusion matrix is used to describe the performance of a classification model on a set of test data for which true values are known.



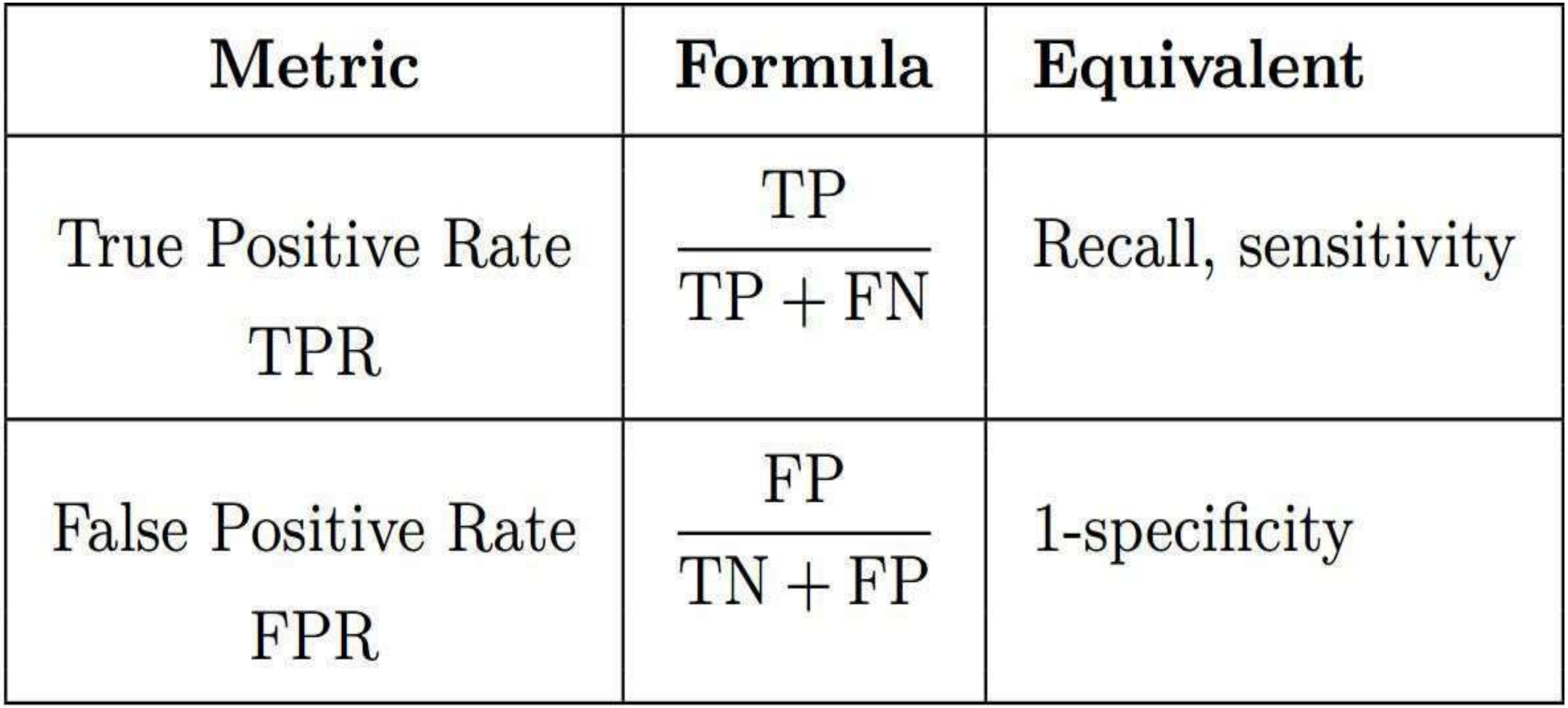
		Predicted class	
		+	–
Actual class	+	**TP** True Positives	**FN** False Negatives Type II error
	–	**FP** False Positives Type I error	**TN** True Negatives

Common Classification Model Evaluation metrics : Main Metrics



Metric	Formula	Interpretation
Accuracy	$\frac{TP + TN}{TP + TN + FP + FN}$	Overall performance of model
Precision	$\frac{TP}{TP + FP}$	How accurate the positive predictions are
Recall Sensitivity	$\frac{TP}{TP + FN}$	Coverage of actual positive sample
Specificity	$\frac{TN}{TN + FP}$	Coverage of actual negative sample
F1 score	$\frac{2TP}{2TP + FP + FN}$	Hybrid metric useful for unbalanced classes

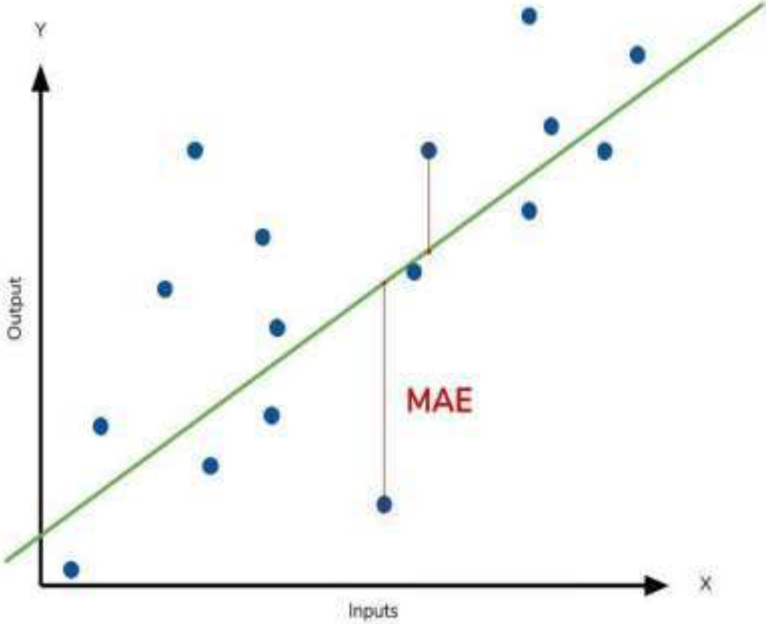
Common Classification Model Evaluation metrics : Main Metrics



Metric	Formula	Equivalent
True Positive Rate TPR	$\frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}}$	Recall, sensitivity
False Positive Rate FPR	$\frac{\mathrm{FP}}{\mathrm{TN}+\mathrm{FP}}$	1-specificity

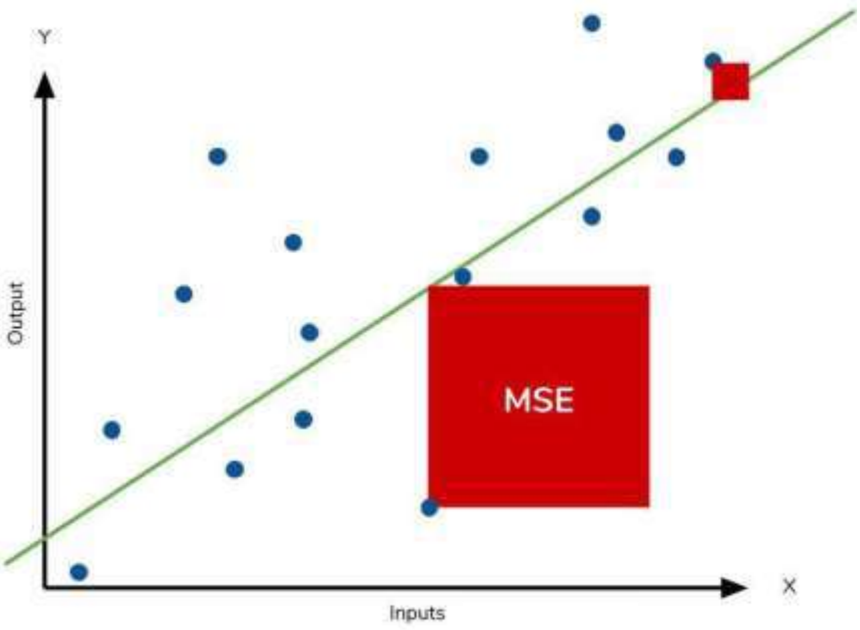
Common Regression Model Evaluation metrics : Mean Absolute Error

$$MAE = \overset{\text{Divide by the total number of data points}}{\boxed{\frac{1}{n}}} \underset{\text{Sum of}}{\sum} \underbrace{\left|\overset{\text{Actual output value}}{\boxed{y}} - \overset{\text{Predicted output value}}{\boxed{\hat{y}}}\right|}_{\text{The absolute value of the residual}}$$



Common Regression Model Evaluation metrics : Mean Square Error

$$MSE = \frac{1}{n}\Sigma\underbrace{\left(y - \widehat{y}\right)}_{\text{The square of the difference between actual and predicted}}^{2}$$

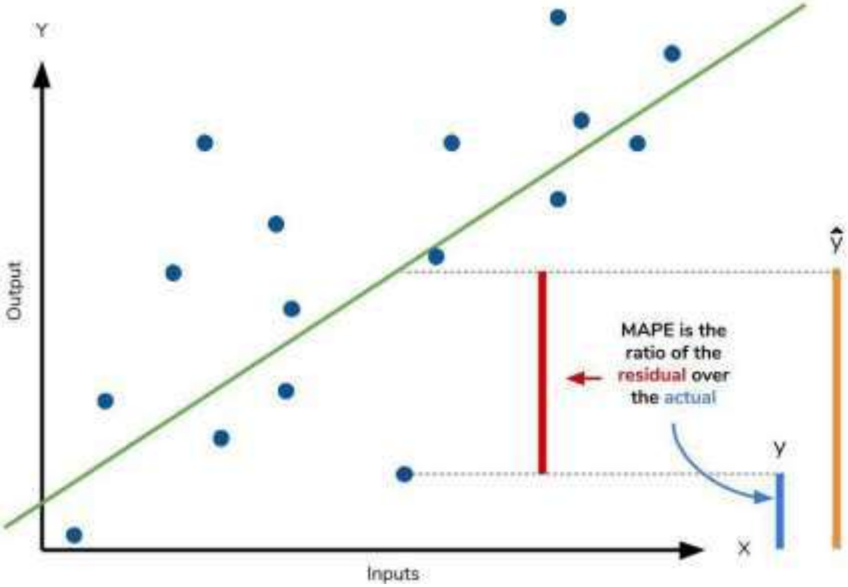


Common Regression Model Evaluation metrics : Mean Absolute Percentage Error

Multiplying by 100% converts to percentage

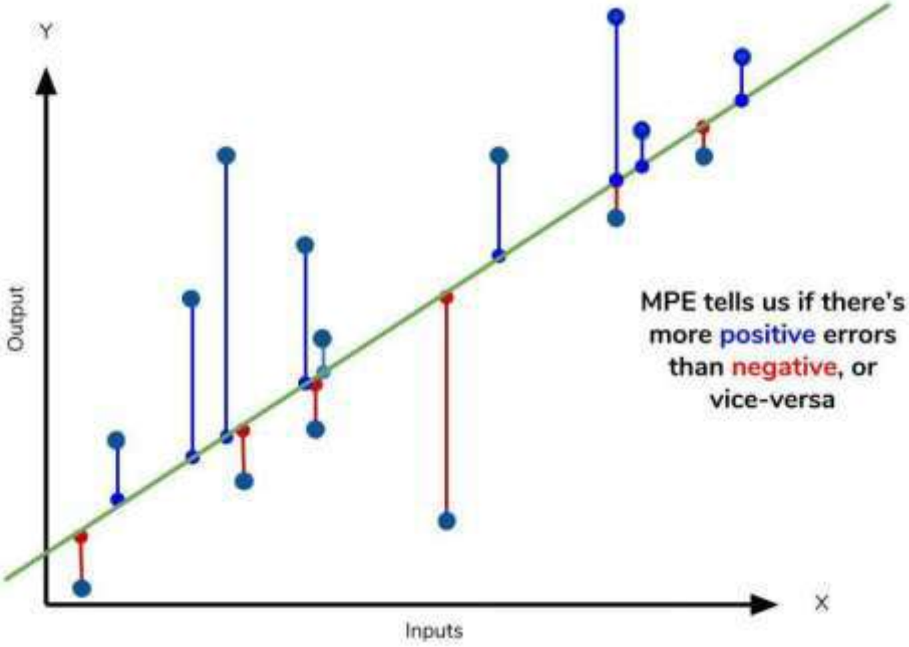
$$MAPE = \frac{\boxed{100\%}}{n} \sum \left| \frac{\overbrace{y - \hat{y}}^{\text{The residual}}}{\boxed{y}} \right|$$

Each residual is scaled against the actual value



Common Regression Model Evaluation metrics : Mean Percentage Error

$$MPE = \frac{100\%}{n} \Sigma \left(\frac{y - \hat{y}}{y} \right)$$



MPE tells us if there's more positive errors than negative, or vice-versa

Testing and Validating

It is common to use 80% of the data for training and hold out 20% for testing.

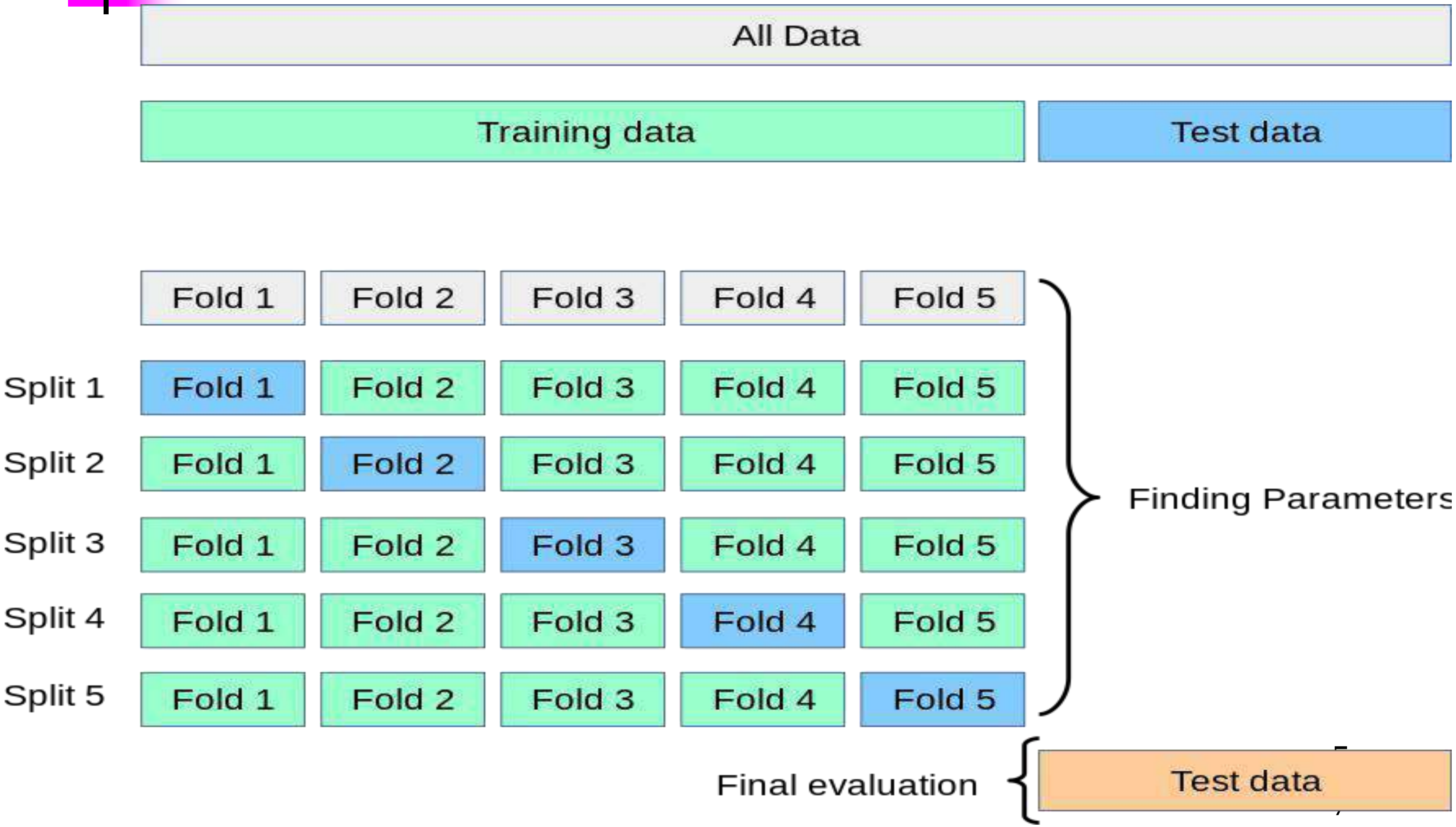
If the training error is low (i.e., your model makes few mistakes on the training set) but the generalization error is high, it means that your model is overfitting the training data.

A common solution to this problem is to have a second holdout set called the validation set. You train multiple models with various hyperparameters using the training set, you select the model and hyperparameters that perform best on the validation set, and when you’re happy with your model you run a single final test against the test set to get an estimate of the generalization error.

Testing and Validating : Cross-Validation

Cross-Validation (CV) : the training set is split into complementary subsets, and each model is trained against a different combination of these subsets and validated against the remaining parts. Once the model type and hyperparameters have been selected, a final model is trained using these hyperparameters on the full training set, and the generalized error is measured on the test set.

Testing and Validating : Cross-Validation



All Data

Training data

Test data

	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
Split 1	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
Split 2	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
Split 3	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
Split 4	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
Split 5	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5

Finding Parameters

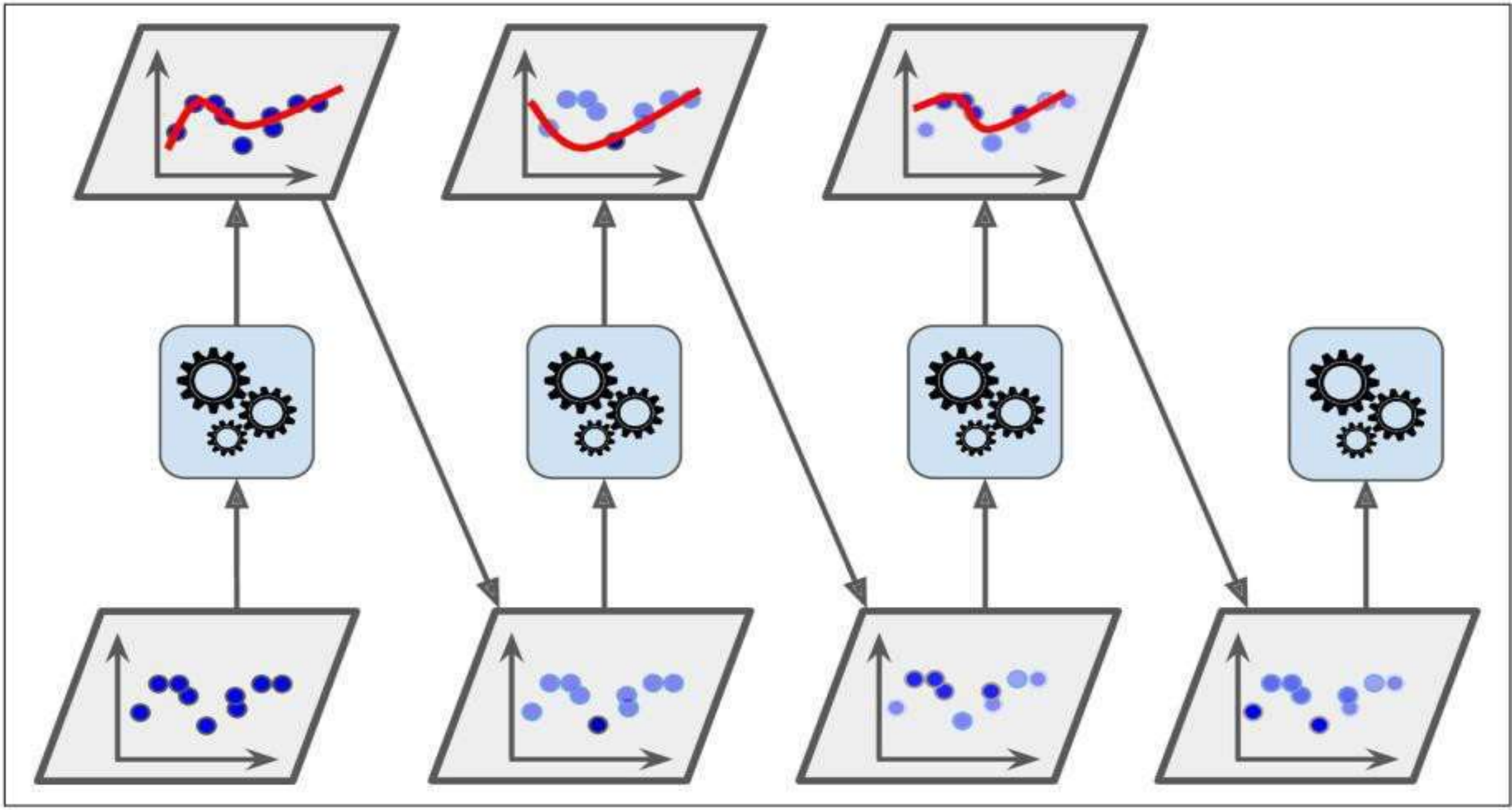
Final evaluation

Test data

Boosting

Boosting refers to any Ensemble method that can combine **several weak learners into a strong learner**. The general idea of most boosting methods is to train predictors sequentially, each trying to correct its predecessor. There are many boosting methods available, but by far the most popular are **AdaBoost** (Adaptive Boosting) and **Gradient Boosting**.

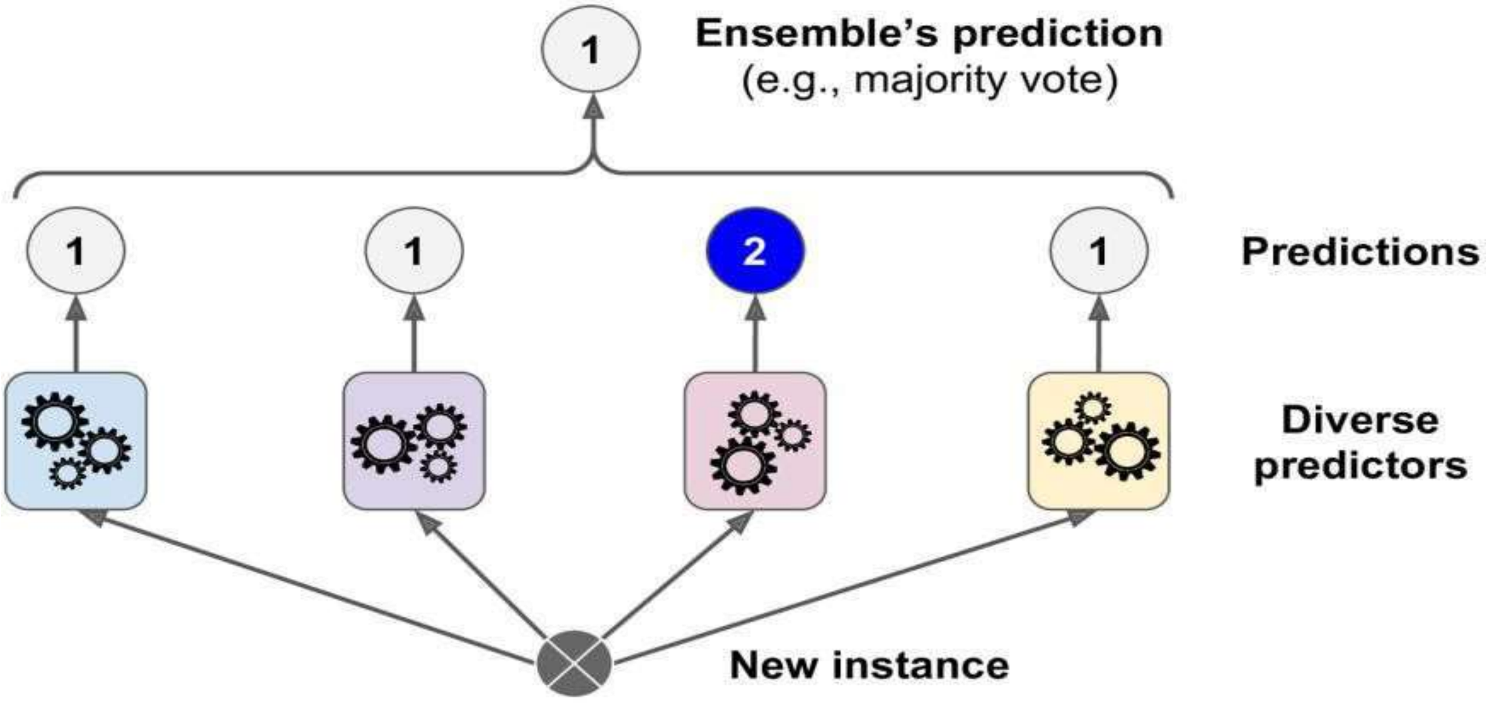
Boosting



AdaBoost sequential training with instance weight updates

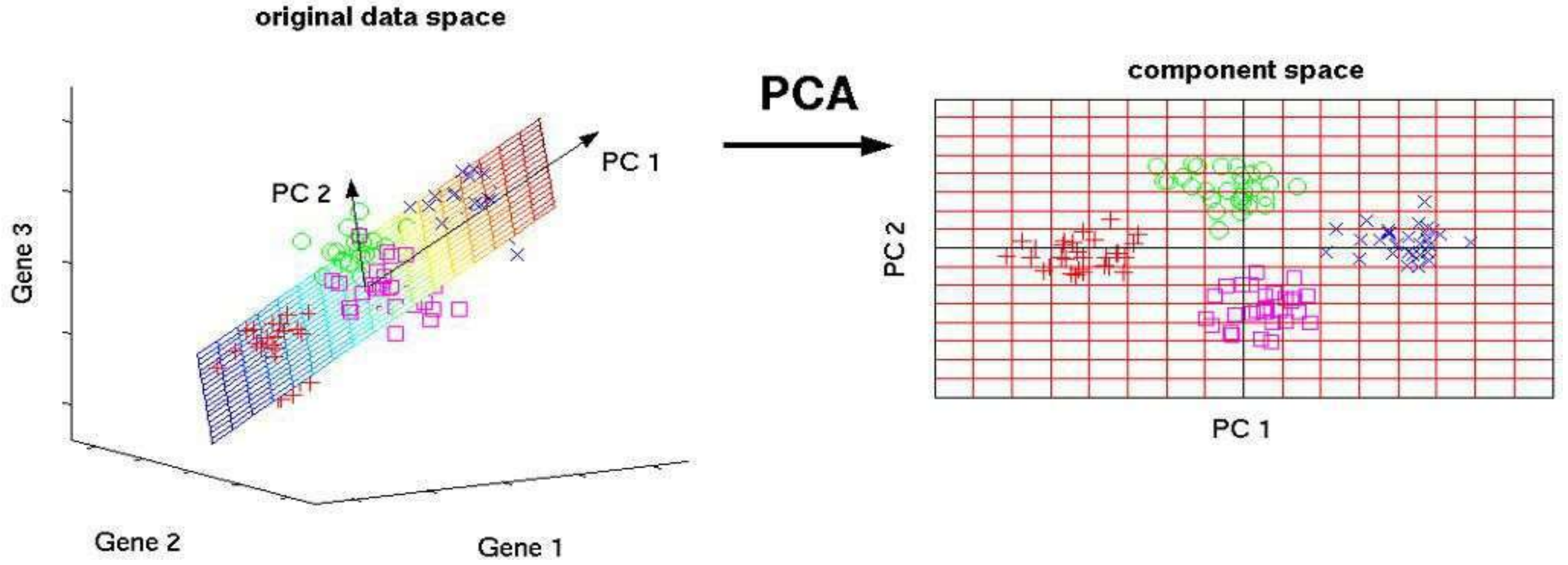
Voting Classifiers

The Voting Classifier: is a meta-classifier for combining similar or conceptually different machine learning classifiers for classification via majority or plurality voting. (For simplicity, we will refer to both majority and plurality voting as majority voting.)



Dimensionality Reduction

- Many Machine Learning problems involve thousands or even millions of features for each training instance. Not only does this make training extremely slow, but it can also make it much harder to find a good solution. This problem is often referred to as the curse of dimensionality.



PrincipalComponent Analysis

Hyperparameter Tuning

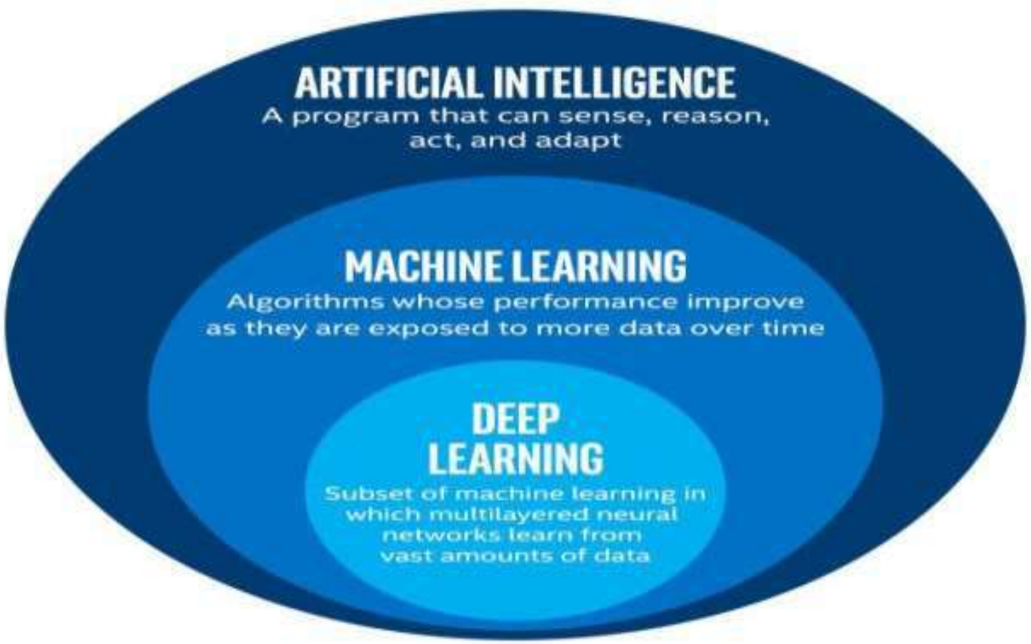
Hyperparameter Tuning : works by running multiple trials in a single training job. Each trial is a complete execution of your training application with values for your chosen hyperparameters, set within limits you specify. The AI Platform training service keeps track of the results of each trial and makes adjustments for subsequent trials. When the job is finished, you can get a summary of all the trials along with the most effective configuration of values according to the criteria you specify.

Steps to Build a Machine Learning System

1. **Data collection.**
2. **Improving data quality (data preprocessing).**
3. **Feature engineering (feature extraction and selection, dimensionality reduction).**
4. **Splitting data into training and evaluation sets.**
5. **Algorithm selection.**
6. **Training.**
7. **Evaluation + Hyperparameter tuning.**
8. **Testing.**
9. **Deployment**

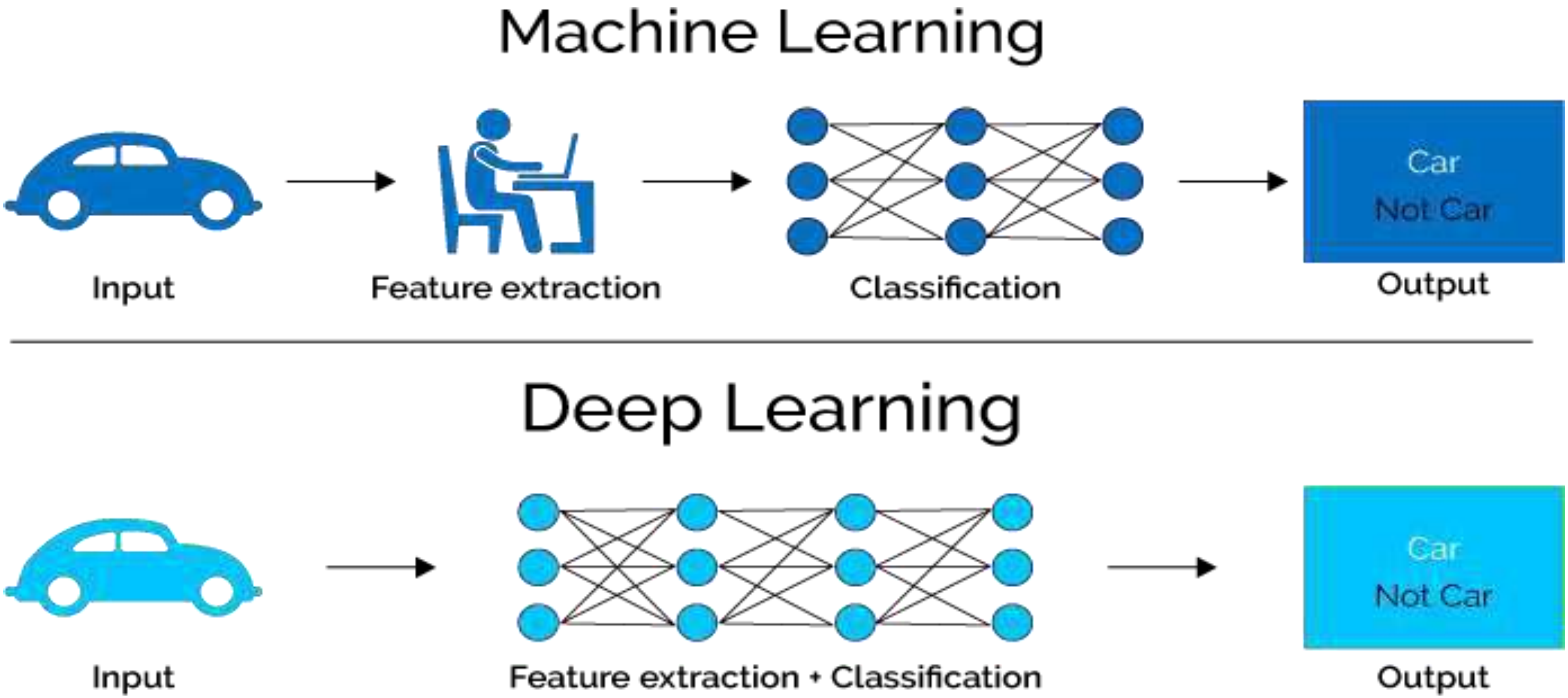
Deep Learning

Deep Learning is a subfield of machine learning concerned with algorithms inspired by the structure and function of the brain called artificial neural networks.



ARTIFICIAL INTELLIGENCE
A program that can sense, reason, act, and adapt
MACHINE LEARNING
Algorithms whose performance improve as they are exposed to more data over time
DEEP LEARNING
Subset of machine learning in which multilayered neural networks learn from vast amounts of data

Deep Learning VS Machine Learning



Feature extraction

Engineering of features is , however, a tedious process for several reasons: **Takes a lot of time** and **Requires expert knowledge**.

For learning-based applications, a lot of time is spent to adjust the features.

Extracted features often lack a **structural representation** reflecting **abstraction** levels in the problem at hand.

Representation learning

Deep Learning aims at learning automatically representations from large sets of labeled data:

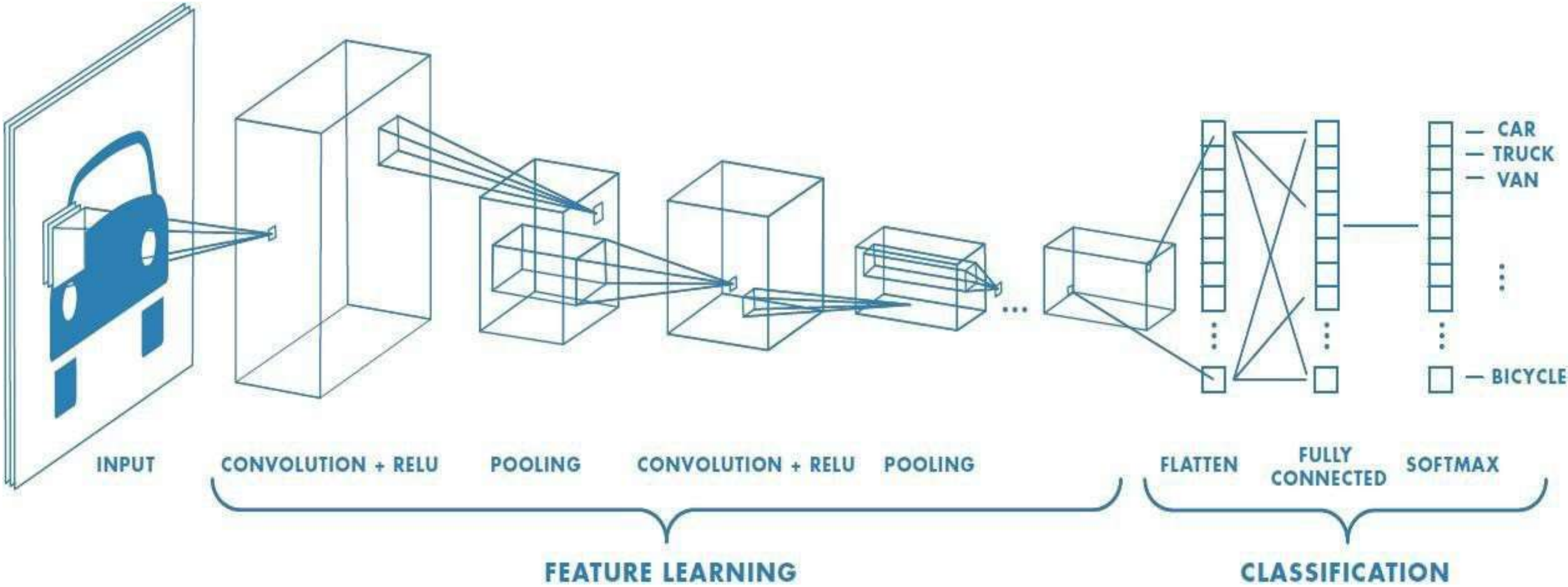
- The machine is powered with raw data.
- Automatic discovery of representations.

Deep learning models

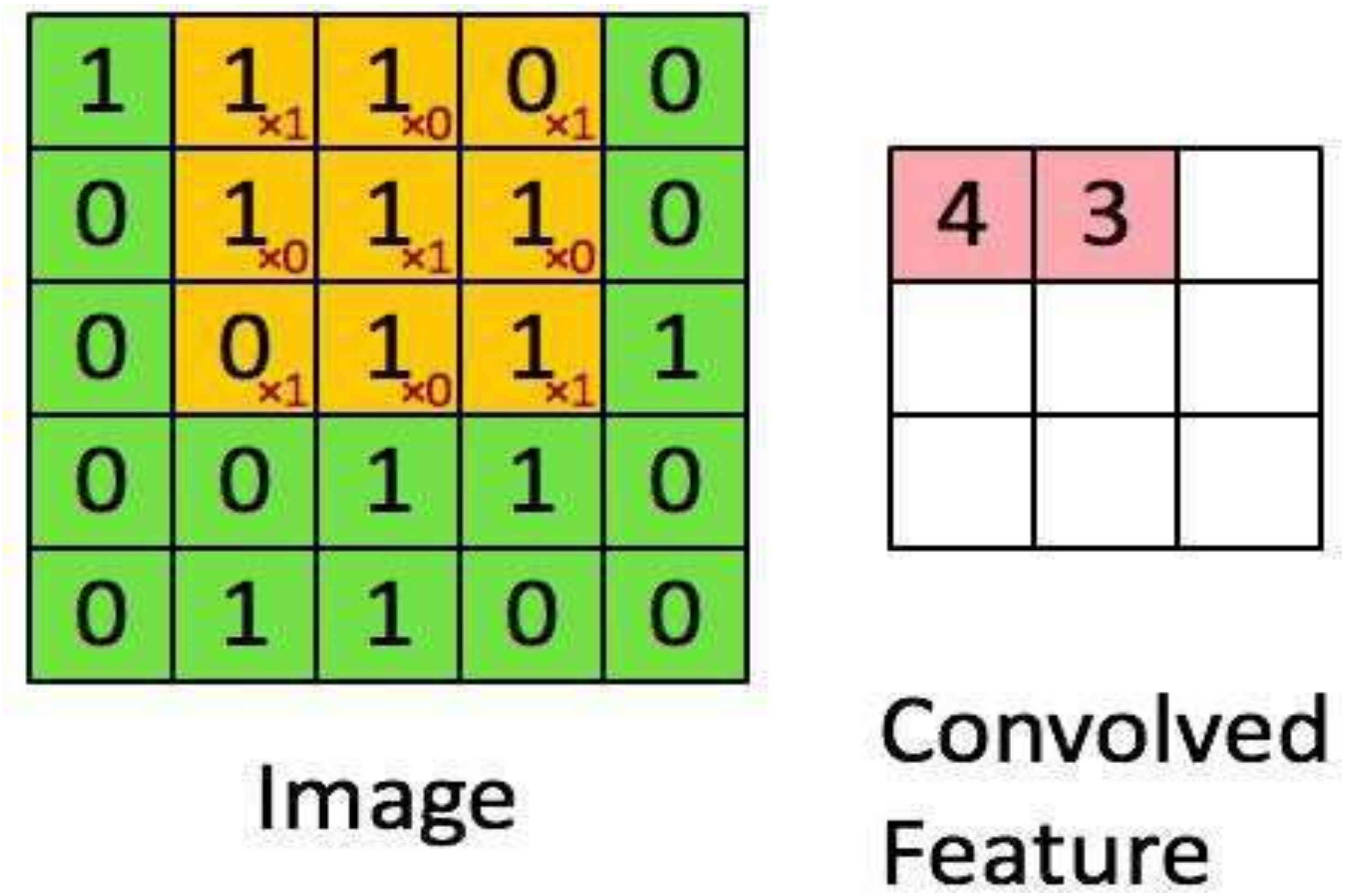
Several DL models have been proposed :

- Autoencoders (Aes)
- Deep belief networks (DBNs)
- Convolutional neural networks (CNNs).
- Recurrent neural networks (RNNs).
- Generative adversial networks (GANs), etc.

Convolutional neural networks (CNNs)



Convolutional neural networks (CNNs)



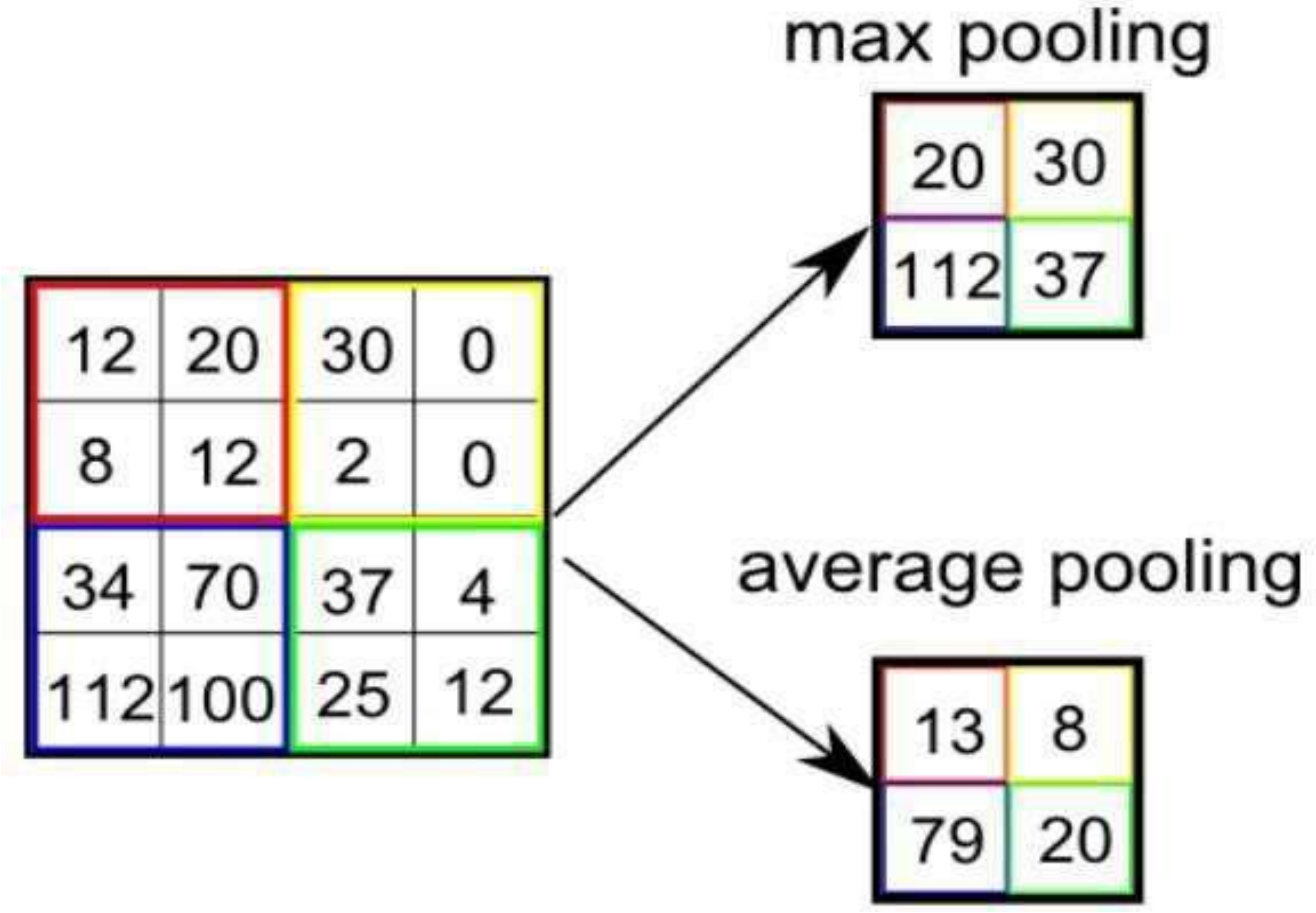
1	1 ×1	1 ×0	0 ×1	0
0	1 ×0	1 ×1	1 ×0	0
0	0 ×1	1 ×0	1 ×1	1
0	0	1	1	0
0	1	1	0	0

Image

4	3	

Convolved Feature

Convolutional neural networks (CNNs)



12	20	30	0
8	12	2	0
34	70	37	4
112	100	25	12

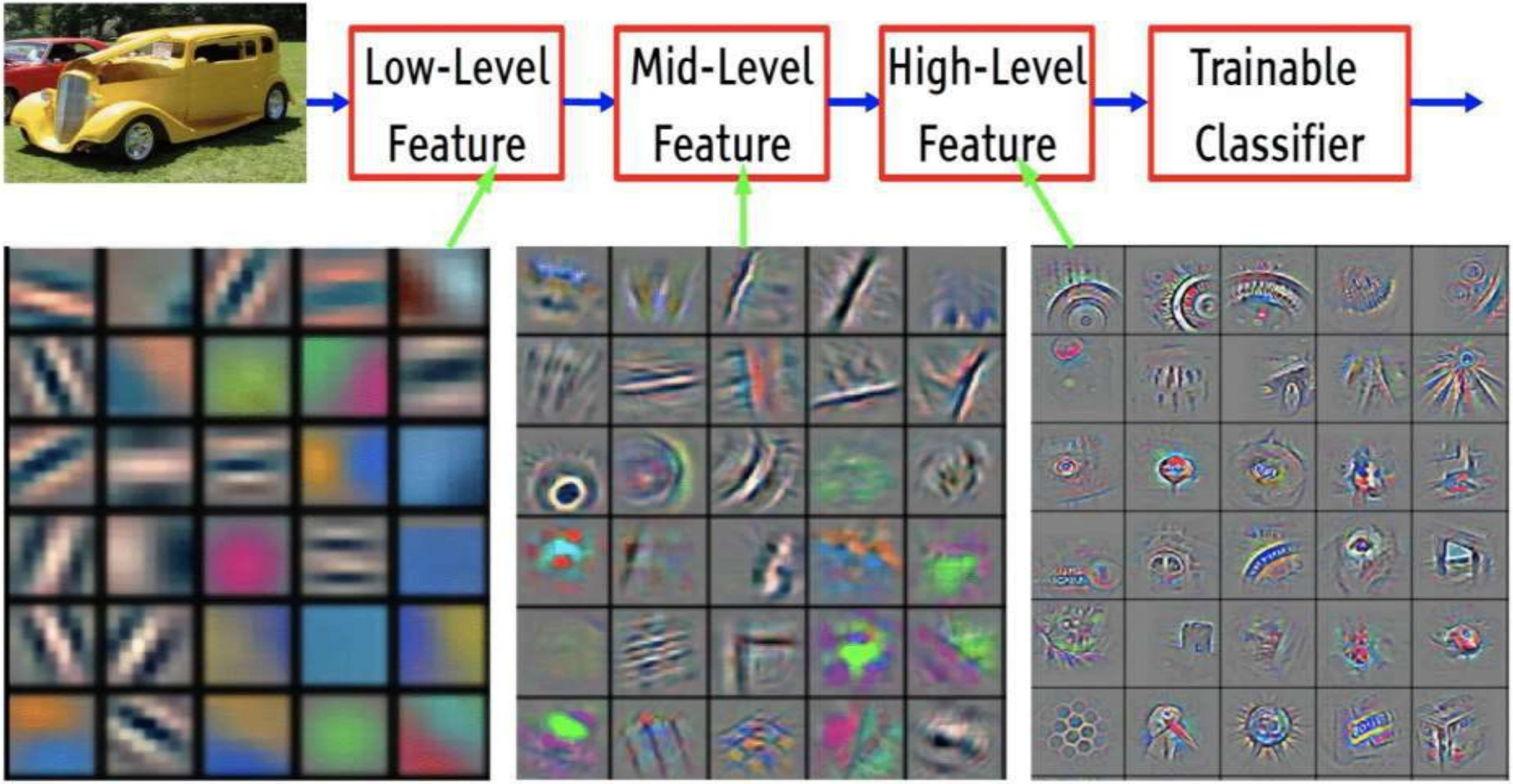
max pooling

20	30
112	37

average pooling

13	8
79	20

Convolutional neural networks (CNNs)



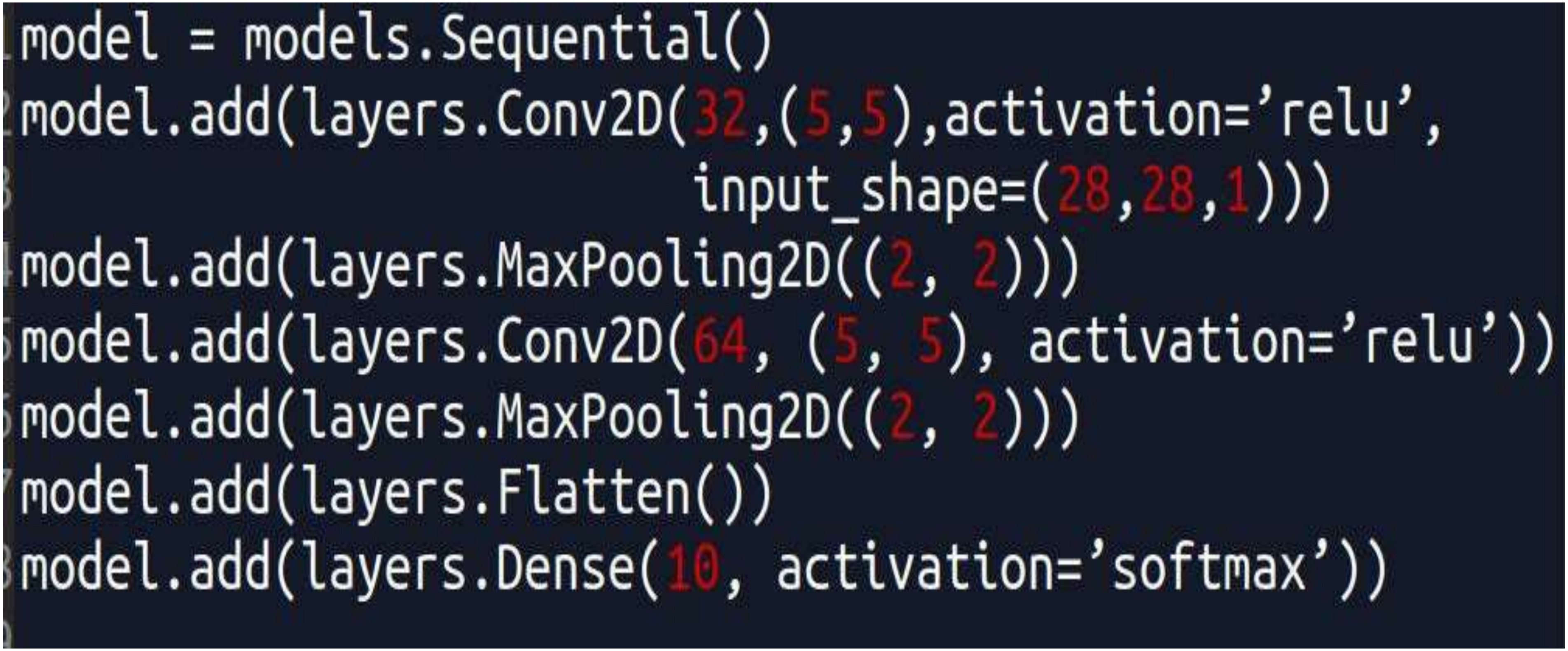
Low-Level Feature

Mid-Level Feature

High-Level Feature

Trainable Classifier

Convolutional neural networks (CNNs)



```
model = models.Sequential()
model.add(layers.Conv2D(32,(5,5),activation=’relu’,
                                 input_shape=(28,28,1)))
model.add(layers.MaxPooling2D((2, 2)))
model.add(layers.Conv2D(64, (5, 5), activation=’relu’))
model.add(layers.MaxPooling2D((2, 2)))
model.add(layers.Flatten())
model.add(layers.Dense(10, activation=’softmax’))
```

Computational Intelligence (CI)

- "Computational intelligence" and "machine learning" are related fields within the broader discipline of artificial intelligence, but they have some distinct differences.
- "Computational intelligence" refers to a set of nature-inspired computational methodologies and approaches to address complex real-world problems. It encompasses various techniques such as neural networks, fuzzy systems, evolutionary computation, and swarm intelligence. These methods are often used to develop systems that can learn from and adapt to their environments.

Computational Intelligence (CI)

- On the other hand, "machine learning" is a specific subset of computational intelligence that focuses on the development of algorithms and statistical models that enable computers to improve their performance on a specific task through experience. Machine learning algorithms can be categorized into supervised, unsupervised, and reinforcement learning, and they are widely used in applications such as data analysis, pattern recognition, and predictive modeling.

Computational Intelligence (CI)

- In summary, computational intelligence is a broader field that encompasses various nature-inspired computational methodologies, while machine learning is a specific subset of computational intelligence focused on developing algorithms that enable computers to learn from data.

Computational Intelligence (CI)

- Machine learning refers, more or less, to the ability of a computer program to learn from a set of inputs either in a supervised (by being actively trained), or unsupervised (by exploring the characteristics of raw data on its own) fashion, in order to provide answers to questions that it wasn't specifically designed to know the answer to.

- Computational Intelligence refers to the ability of natural (and potentially artificial) agents to *behave* intelligently

Computational Intelligence (CI)

- In the quest for artificial intelligence we're exploring machine learning as a means to give artificial agents (computer programs, and computers by extension) the ability to behave intelligently. And in some respects machine learning attempts to mimic natural occurring processes associated with computational intelligence (neural networks, genetic optimization algorithms are examples) but there are also machine learning methodologies that would not necessarily be characterized as computational intelligence.

Computational Intelligence (CI)

- Intelligence built in computer programs
- Covers
  - Evolutionary computing
  - Fuzzy computing
  - Neuro-computing
- Also known as
  - Soft computing

CI Techniques

- Artificial Intelligence (AI)
  - Artificial Neural Networks (ANNs)
  - Fuzzy Logic (FL)
  - Support Vector Machines (SVM)
  - Self Organizing Maps (SOM)- unsupervised
- Genetic Algorithm (GA)
- Genetic Programming (GP)
- Swarm Intelligence/Particle Swarm Optimization (PSO)

CI Techniques (contd.)

- ANNs
  - Multi-layer Perceptron (MLP)
  - Radial Basis Function (RBF)
  - Probabilistic Neural Network (PNN)
- Fuzzy Logic + ANN
  - Adaptive neuro-fuzzy inference system (ANFIS)

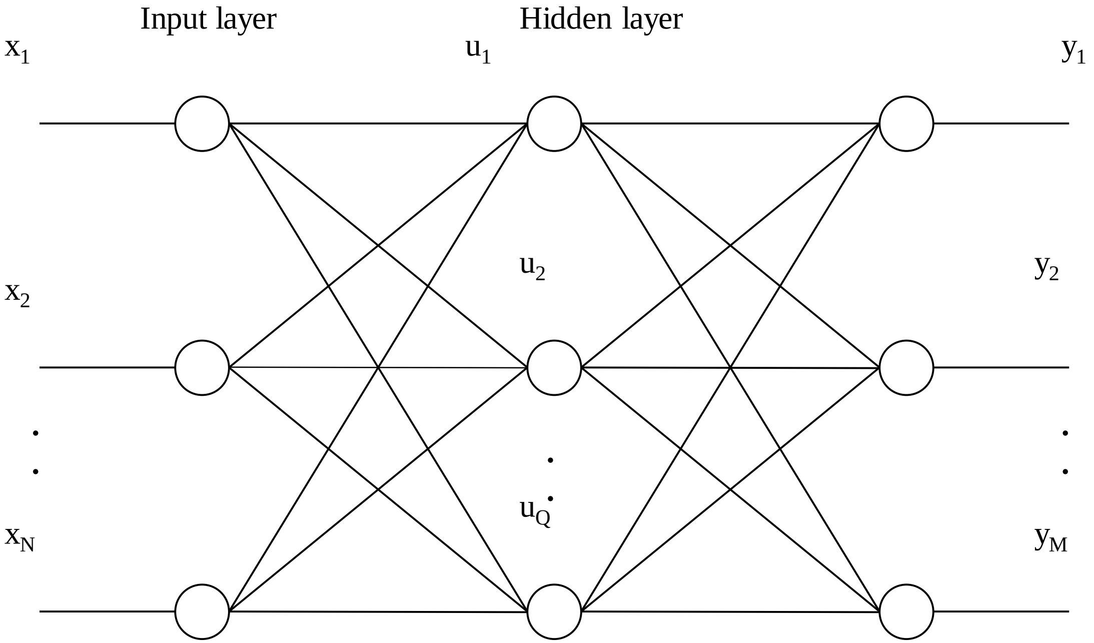
CI Techniques (contd.)

ANN structure

- Input layer
- Hidden Layer (s)
- Output layer
- Number of nodes in each layer
- Functions and their parameters

Mostly decided on trial and error basis

ANN- a typical example



Fuzzy Logic

Steps involved

- Fuzzification using membership functions (MFs)-input
- Generation of rule base
- Aggregation
- Defuzzification using MFs -output

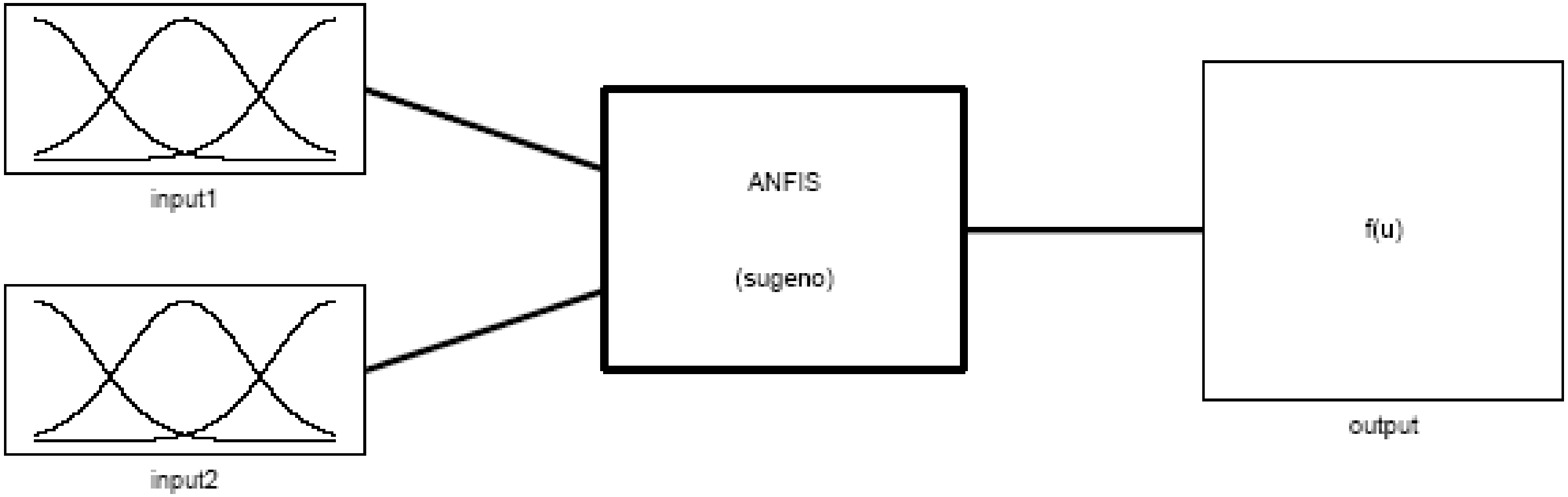
Fuzzy Logic (contd.)

- Input and output MFs
  - Number
  - Type
  - Parameters
- Rule base (experience guided)

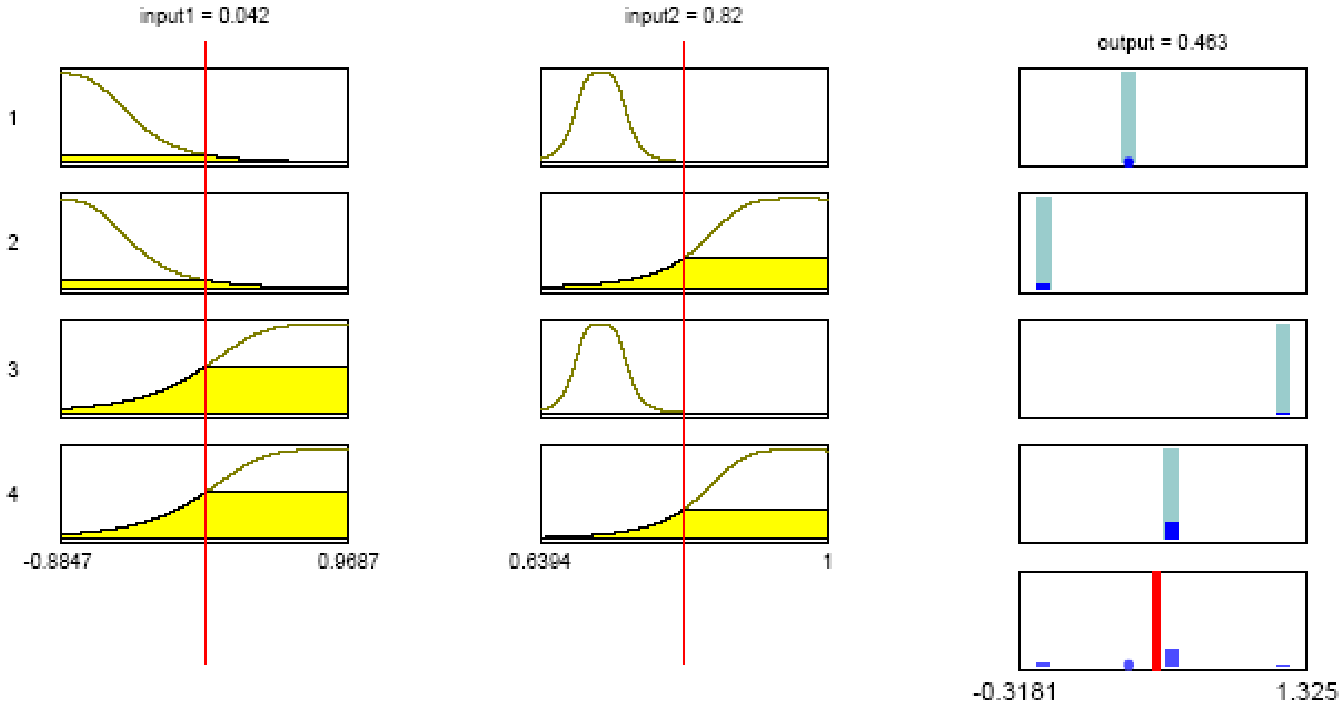
Neuro-Fuzzy System

- Combines the advantages of fuzzy logic (FL) and ANNs
- Starts with an initial FL structure
- Uses ANN for adapting the FL (MF) parameters and the rule base to the training data

Fuzzy Logic – An Example



ANFIS structure for an example system with 2 inputs and 1 output.



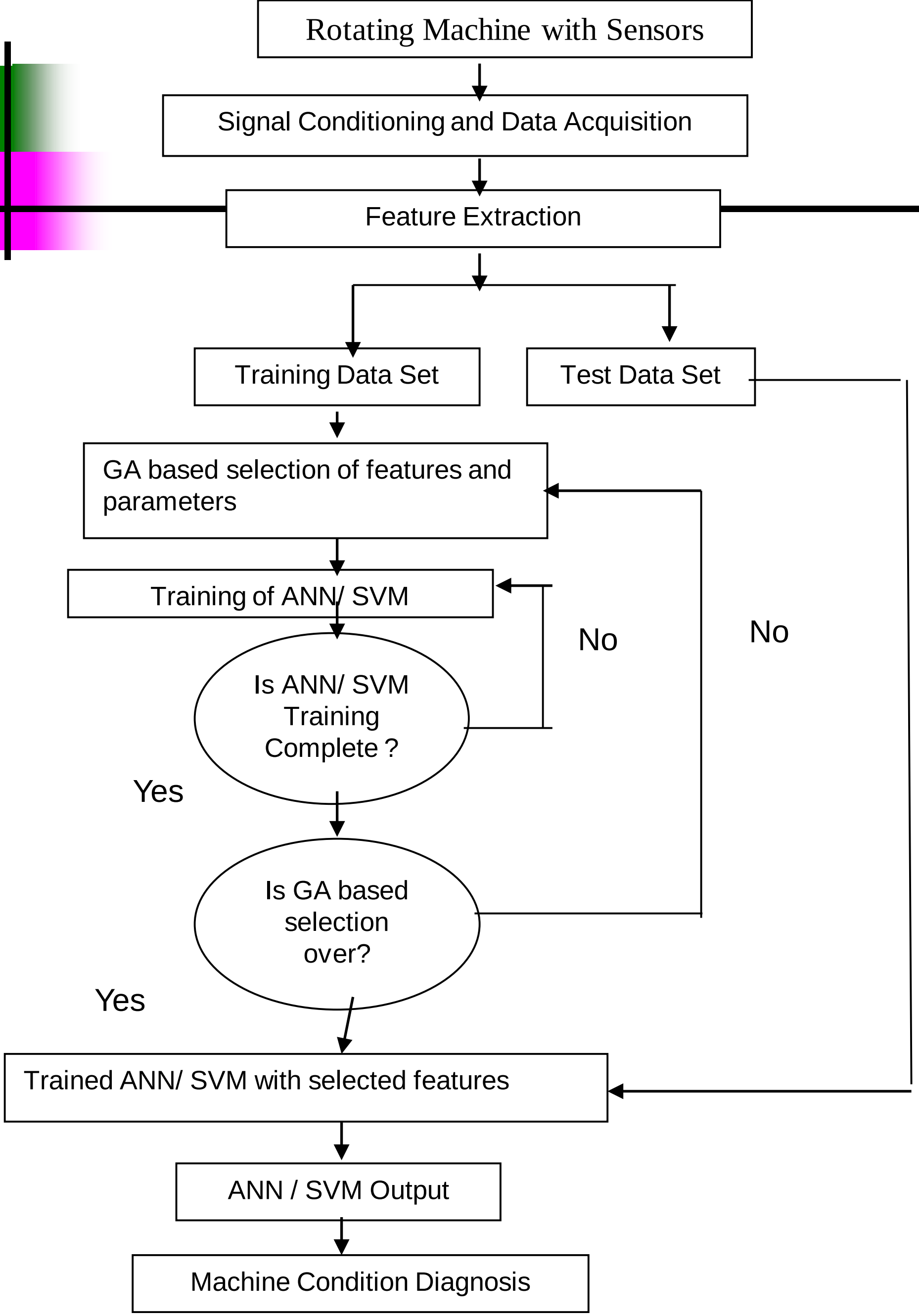
Snapshot of rule base for an example system with 2 inputs and 1 output.

Genetic Algorithms

- Construction of genome (individual)
- Generation of initial population (group of individuals)
- Evaluation of individuals
- Selection of individuals based on criteria
- Generation of new individuals
  - Mutation
  - Crossover
- Repetition of the process - generation, evaluation, selection
- Termination of the process based on max generation no. and/or performance criteria

Combinations

- Combine advantages of GA and other classifiers
- GA and ANN
- GA and ANFIS
- GA and SVM
- for automatic selection of classifier structure and parameters
  - ANNs -Number of neurons in hidden layer
  - ANFIS - Number of MFs and their parameters
  - SVM – SVM parameters
- Selection of most important system features from a pool
- Selection of most important sensors (in the context of on-line condition monitoring and diagnostics)- sensor fusion.



Rotating Machine with Sensors

Signal Conditioning and Data Acquisition

Feature Extraction

Training Data Set

Test Data Set

GA based selection of features and parameters

Training of ANN/ SVM

Is ANN/ SVM Training Complete ?

No

Yes

Is GA based selection over?

No

Yes

Trained ANN/ SVM with selected features

ANN / SVM Output

Machine Condition Diagnosis

Fig. 1. Flow chart of diagnostic procedure

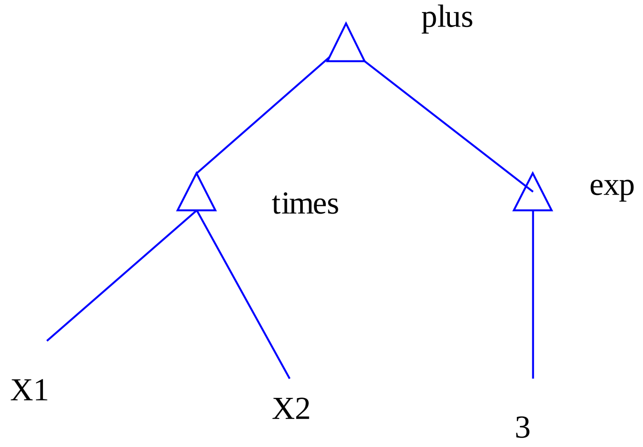
Genetic Programming (GP)

- GP – a branch of GA with a lot of similarities.
- Main difference of GP and GA is in the representation of the solution.
- In GA, the output is in form of a string of numbers representing the solution.
- GP produces a computer program in form of a tree-based structure relating
  - the inputs (leaves)
  - the mathematical functions (nodes) and
  - the output (root node).

GP output –An Example

- Terminals (leaves): inputs x1, x2 and constant 3
- Nodes: Math functions *,+, exp
- Output: x1*x2+exp(3)

(+ (* (X1 X2))(exp(3))



Applications

- Computer Science
  - Pattern Recognition (PR)
  - Data Mining
  - Knowledge Discovery/ Machine Learning
  - Feature Extraction and Selection
- Mechanical Systems
  - Condition monitoring and diagnostics
  - Multiobjective optimization in design
  - Control System Design
- Manufacturing Systems
  - Development of data-driven models
  - Multiobjective optimization of machining parameters

Applications (contd.)

- Engineering Management/IE
  - Inventory management
  - Project selection
  - Facility layout design
  - Scheduling
- Medicine
  - Patient condition monitoring and diagnosis
- Social Science
- Business
  - Market analysis and forecasting
  - Credit rating

Recent Work

- Machine Condition Monitoring and Diagnostics using
  - ANNs-MLP, RBF, PNN
  - SVM
  - ANFIS
  - GA-ANN
  - GA-ANFIS
  - GA-SVM
  - GP
- Involving signal processing, feature extraction, selection and sensor fusion

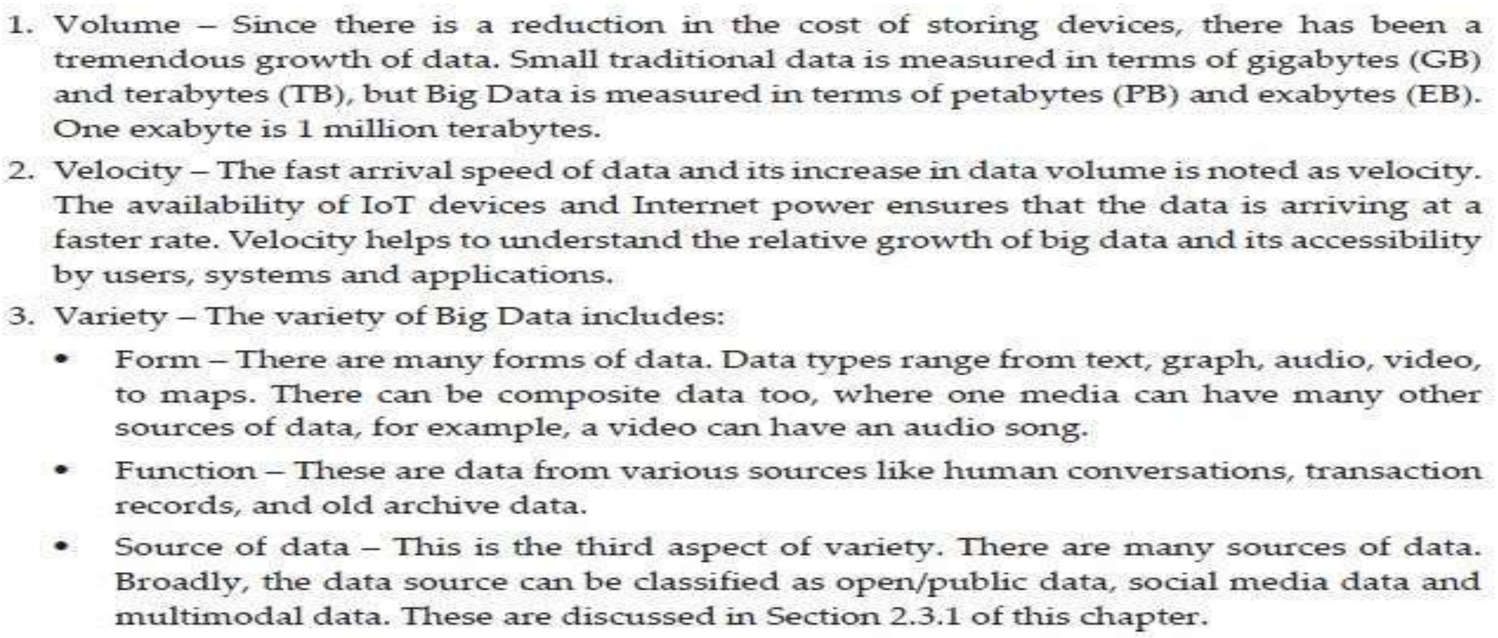
Recent work (contd.)

- Materials
  - ANN based estimation of fatigue life
  - Modeling of material properties in terms of heat treatment parameters
- Rotordynamics
- Control System Design

What is Data?

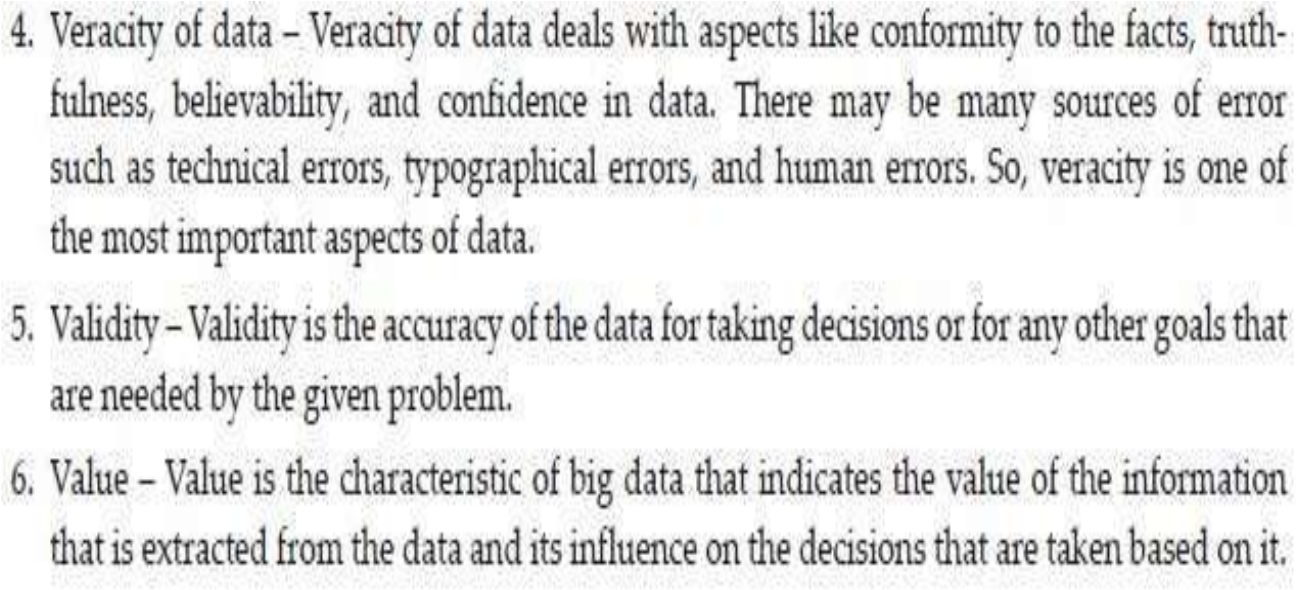
- DATA ARE FACTS
- FACTS ARE IN THE FORM OF NUMBERS, AUDIO, VIDEO, AND IMAGE
- NEED TO ANALYZE DATA FOR TAKING DECISIONS

Characteristics of Big Data



1. Volume – Since there is a reduction in the cost of storing devices, there has been a tremendous growth of data. Small traditional data is measured in terms of gigabytes (GB) and terabytes (TB), but Big Data is measured in terms of petabytes (PB) and exabytes (EB). One exabyte is 1 million terabytes.
2. Velocity – The fast arrival speed of data and its increase in data volume is noted as velocity. The availability of IoT devices and Internet power ensures that the data is arriving at a faster rate. Velocity helps to understand the relative growth of big data and its accessibility by users, systems and applications.
3. Variety – The variety of Big Data includes:
   - Form – There are many forms of data. Data types range from text, graph, audio, video, to maps. There can be composite data too, where one media can have many other sources of data, for example, a video can have an audio song.
   - Function – These are data from various sources like human conversations, transaction records, and old archive data.
   - Source of data – This is the third aspect of variety. There are many sources of data. Broadly, the data source can be classified as open/public data, social media data and multimodal data. These are discussed in Section 2.3.1 of this chapter.

Characteristics of Data



4. Veracity of data – Veracity of data deals with aspects like conformity to the facts, truthfulness, believability, and confidence in data. There may be many sources of error such as technical errors, typographical errors, and human errors. So, veracity is one of the most important aspects of data.

5. Validity – Validity is the accuracy of the data for taking decisions or for any other goals that are needed by the given problem.

6. Value – Value is the characteristic of big data that indicates the value of the information that is extracted from the data and its influence on the decisions that are taken based on it.

Data Sources

A DATA SOURCE CAN BE ANYTHING –

- STRUCTURED DATA
- SEMI-STRUCTURED DATA
- UNSTRUCTURED DATA

Structured Data

- A STRUCTURED DATA CAN BE ANY ONE OF THE FOLLOWING –

- RECORD DATA
- GRAPHICS DATA
- DATA MATRIX
- ORDERED DATA – SEQUENCE DATA, TIME SERIES DATA, TEMPORAL DATA

Unstructured Data

AN UNSTRUCTURED DATA CAN BE ANY ONE OF THE FOLLOWING –

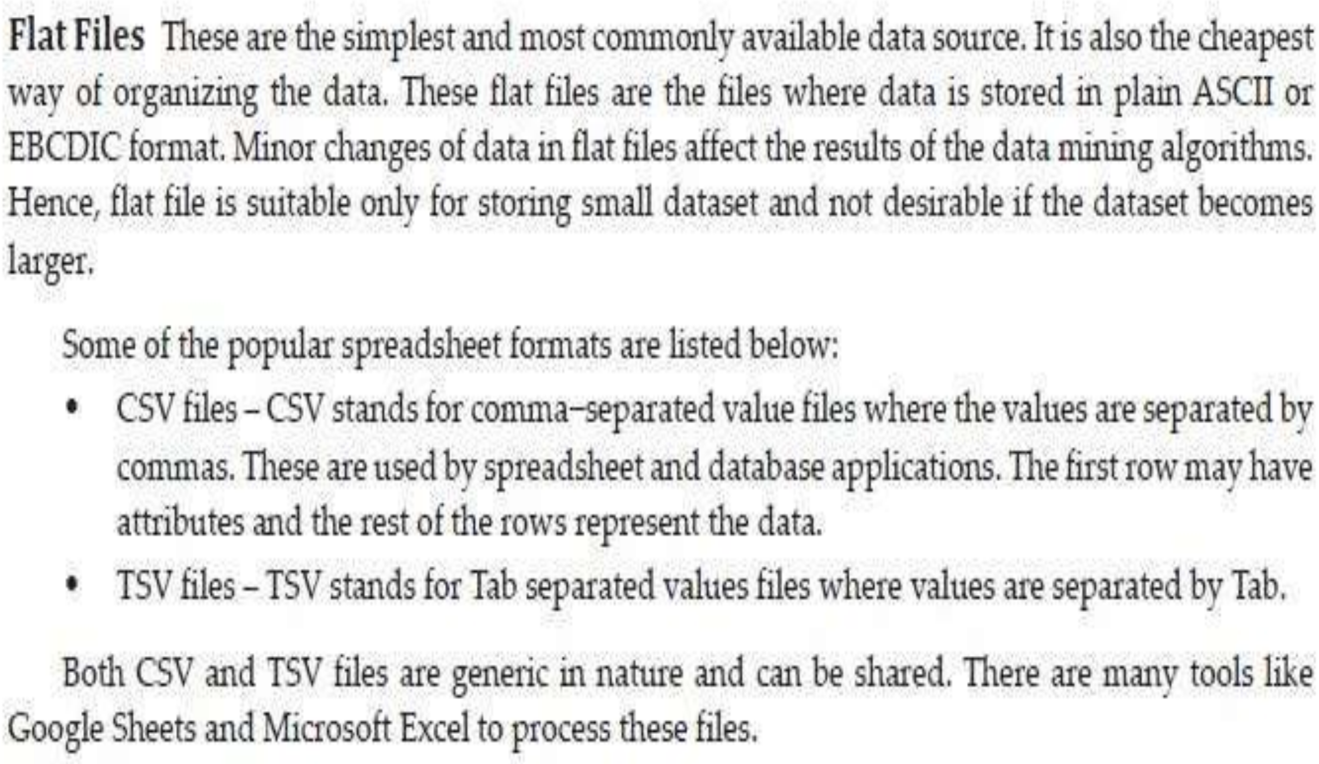
- VIDEO, IMAGE, PROGRAMS
- BLOG DATA
- 80% OF ORGANIZATION DATA

Semi-Structured Data

A SEMI-STRUCTURED DATA CAN BE ANY ONE OF THE FOLLOWING –

- XML/JSON OBJECTS
- RSS FEEDS
- HIERARCHICAL RECORDS

Data Storage



Flat Files These are the simplest and most commonly available data source. It is also the cheapest way of organizing the data. These flat files are the files where data is stored in plain ASCII or EBCDIC format. Minor changes of data in flat files affect the results of the data mining algorithms. Hence, flat file is suitable only for storing small dataset and not desirable if the dataset becomes larger.

Some of the popular spreadsheet formats are listed below:

- CSV files – CSV stands for comma–separated value files where the values are separated by commas. These are used by spreadsheet and database applications. The first row may have attributes and the rest of the rows represent the data.
- TSV files – TSV stands for Tab separated values files where values are separated by Tab.

Both CSV and TSV files are generic in nature and can be shared. There are many tools like Google Sheets and Microsoft Excel to process these files.

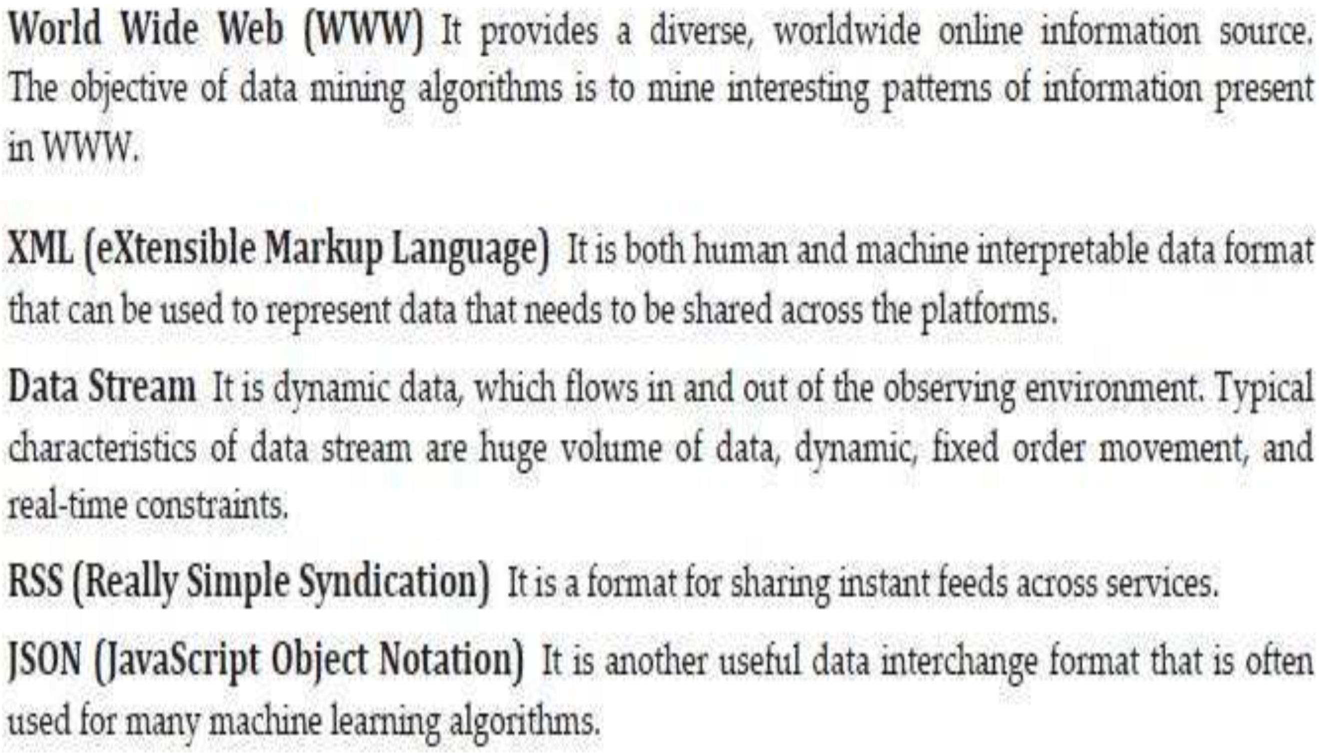
Data Storage

- DATABASE SYSTEMS
- TYPES ARE

1. TRANSACTIONAL DATABASE
2. TIME SERIES DATABASE
3. TEMPORAL DATABASE

Data Storage

- OTHER TYPES



World Wide Web (WWW) It provides a diverse, worldwide online information source. The objective of data mining algorithms is to mine interesting patterns of information present in WWW.

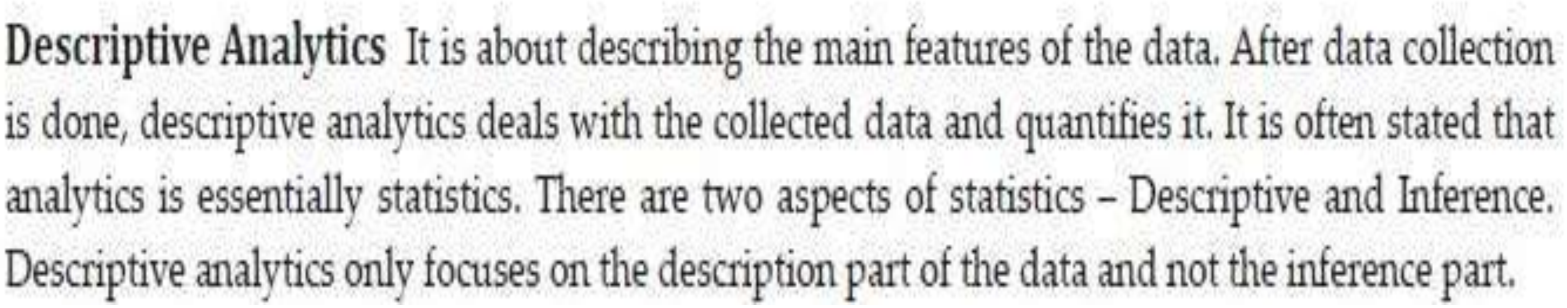
XML (eXtensible Markup Language) It is both human and machine interpretable data format that can be used to represent data that needs to be shared across the platforms.

Data Stream It is dynamic data, which flows in and out of the observing environment. Typical characteristics of data stream are huge volume of data, dynamic, fixed order movement, and real-time constraints.

RSS (Really Simple Syndication) It is a format for sharing instant feeds across services.

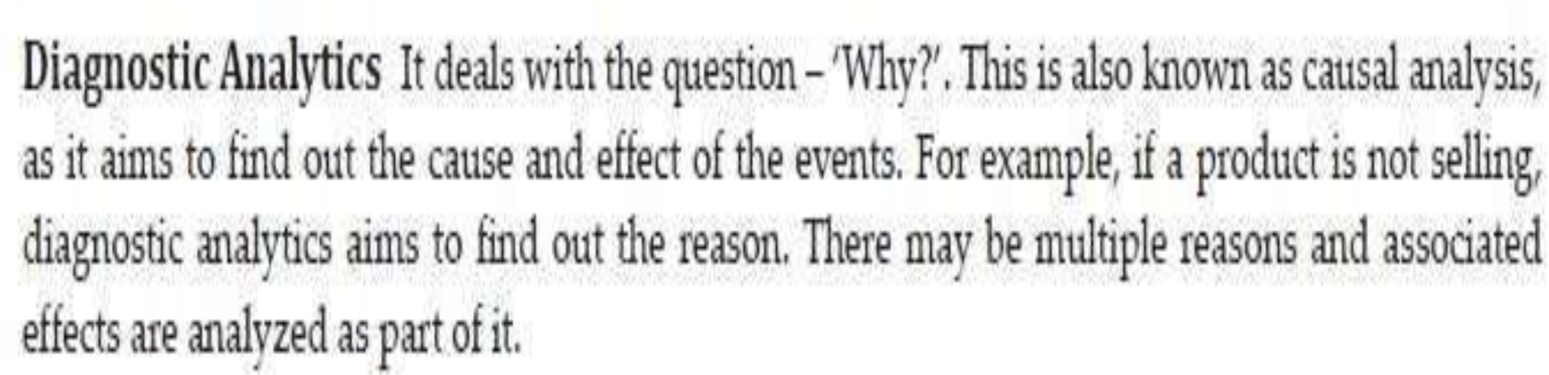
JSON (JavaScript Object Notation) It is another useful data interchange format that is often used for many machine learning algorithms.

Descriptive Analytics



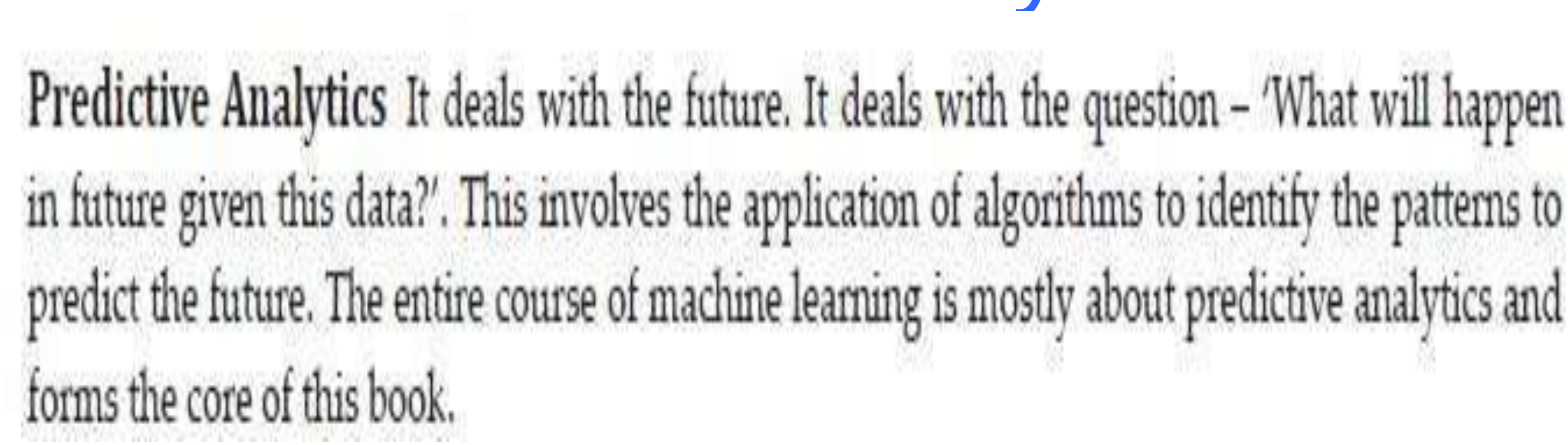
Descriptive Analytics It is about describing the main features of the data. After data collection is done, descriptive analytics deals with the collected data and quantifies it. It is often stated that analytics is essentially statistics. There are two aspects of statistics – Descriptive and Inference. Descriptive analytics only focuses on the description part of the data and not the inference part.

Diagnostic Analytics



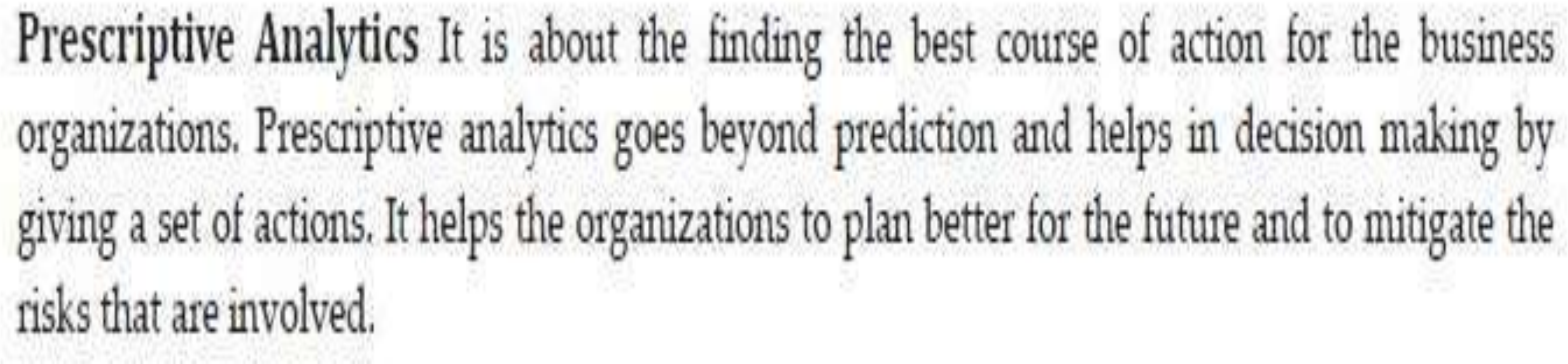
Diagnostic Analytics It deals with the question – 'Why?'. This is also known as causal analysis, as it aims to find out the cause and effect of the events. For example, if a product is not selling, diagnostic analytics aims to find out the reason. There may be multiple reasons and associated effects are analyzed as part of it.

Predictive Analytics



Predictive Analytics It deals with the future. It deals with the question – 'What will happen in future given this data?'. This involves the application of algorithms to identify the patterns to predict the future. The entire course of machine learning is mostly about predictive analytics and forms the core of this book.

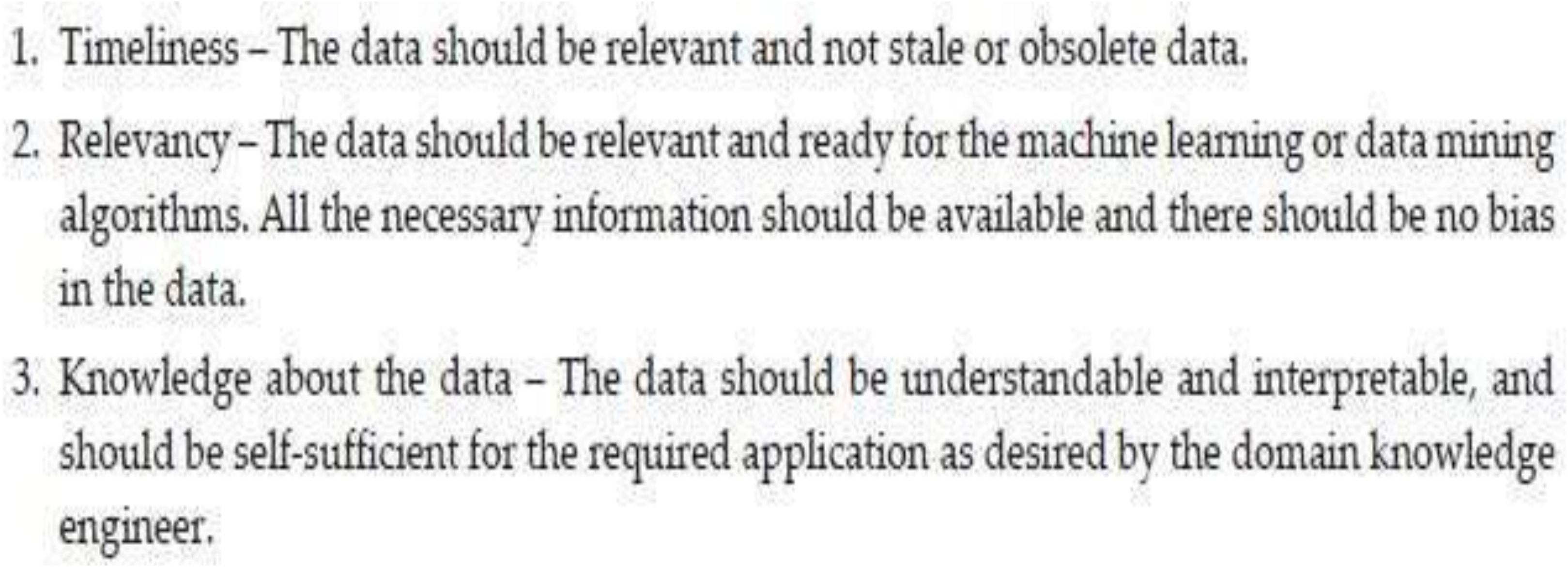
Prescriptive Analytics



Prescriptive Analytics It is about the finding the best course of action for the business organizations. Prescriptive analytics goes beyond prediction and helps in decision making by giving a set of actions. It helps the organizations to plan better for the future and to mitigate the risks that are involved.

Good Data Characteristics

- GOOD DATA SHOULD HAVE THESE CHARACTERISTICS



1. Timeliness – The data should be relevant and not stale or obsolete data.
2. Relevancy – The data should be relevant and ready for the machine learning or data mining algorithms. All the necessary information should be available and there should be no bias in the data.
3. Knowledge about the data – The data should be understandable and interpretable, and should be self-sufficient for the required application as desired by the domain knowledge engineer.

Open-Source Data

1. DIGITAL LIBRARIES
2. EXPERIMENTAL DATA LIKE GENOMIC AND BIOLOGICAL DATA
3. HEALTHCARE SYSTEMS LIKE PATIENT INSURANCE DATA

Social-Media Data

1. TWITTER DATA
2. FACEBOOK DATA
3. YOUTUBE VIDEOS
4. INSTAGRAM DATA

Multimodal Data

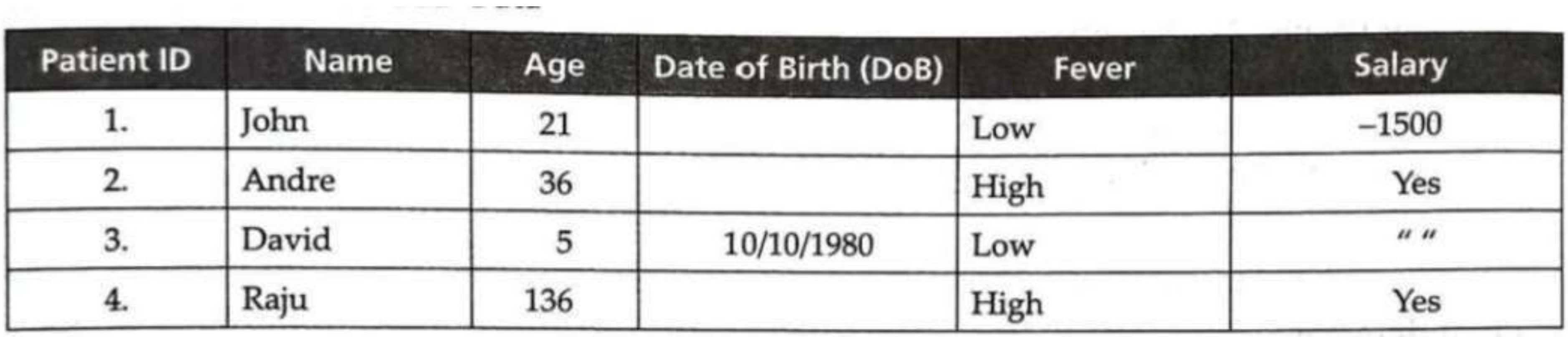
- IMAGE ARCHIVES WITH TEXT AND NUMERIC DATA
- WWW

Data Preprocessing

DATA THAT CAN CAUSE PROBLEMS

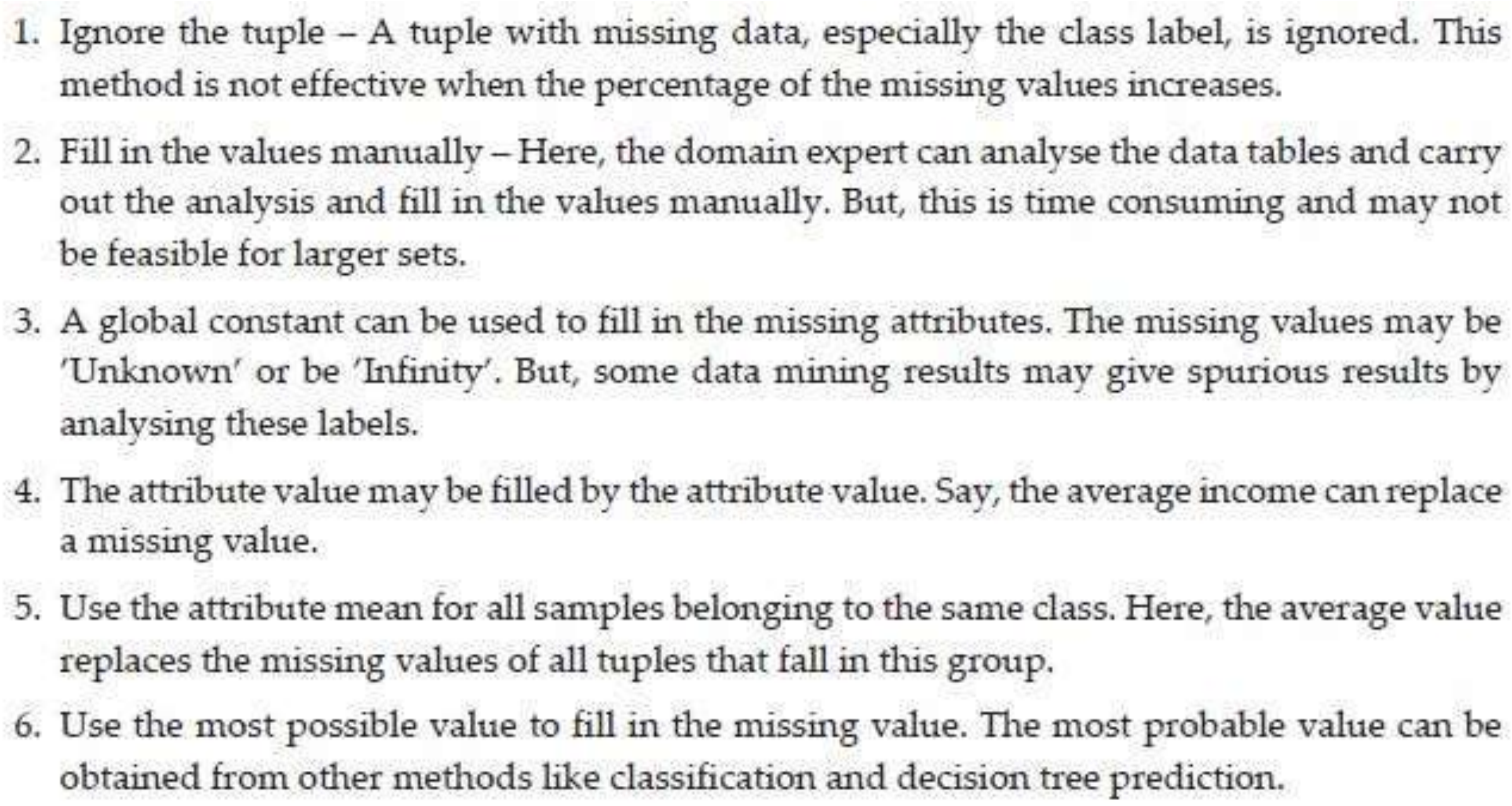
- INCOMPLETE DATA
- OUTLIER DATA
- INCONSISTENT DATA
- INACCURATE DATA
- MISSING VALUES
- DUPLICATE DATA

Data Preprocessing



Patient ID	Name	Age	Date of Birth (DoB)	Fever	Salary
1.	John	21		Low	–1500
2.	Andre	36		High	Yes
3.	David	5	10/10/1980	Low	“ ”
4.	Raju	136		High	Yes

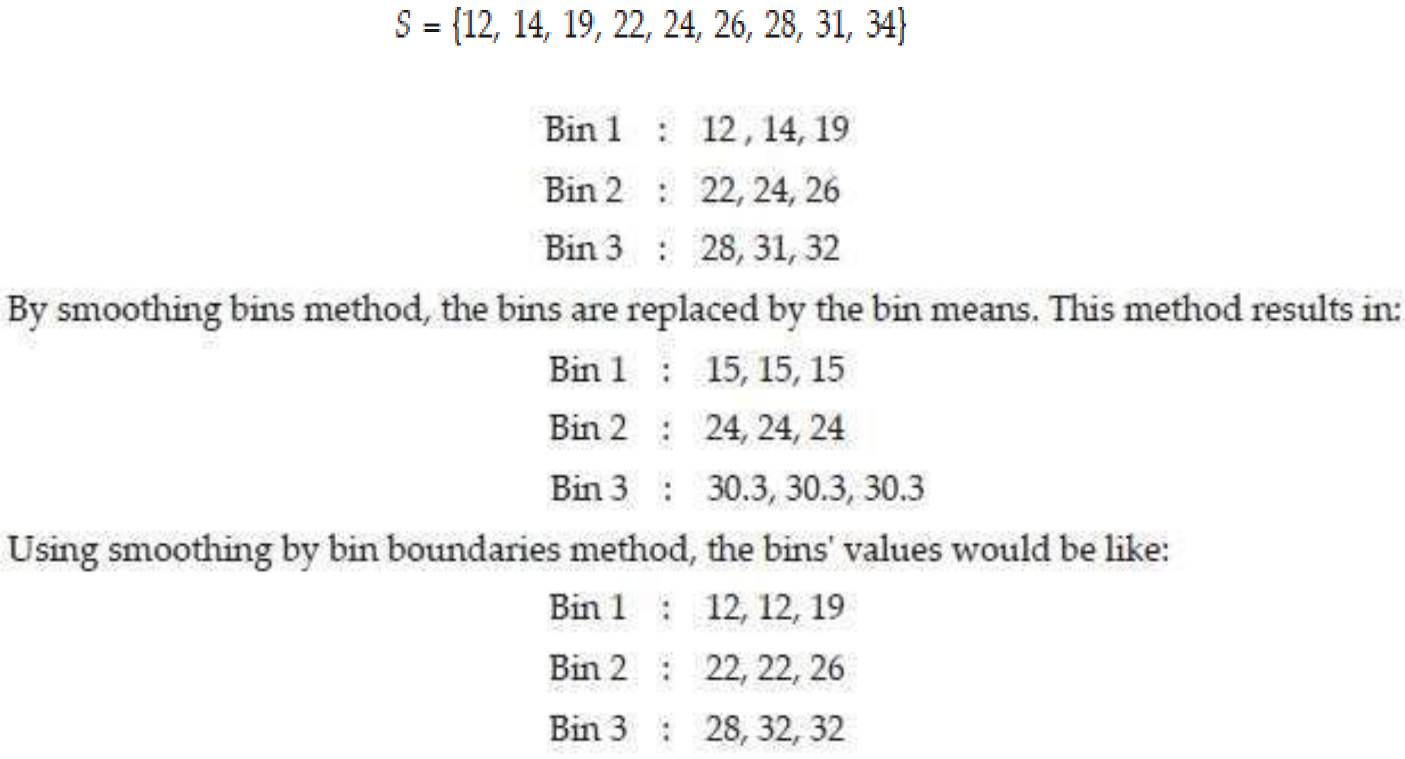
Missing Data



1. Ignore the tuple – A tuple with missing data, especially the class label, is ignored. This method is not effective when the percentage of the missing values increases.
2. Fill in the values manually – Here, the domain expert can analyse the data tables and carry out the analysis and fill in the values manually. But, this is time consuming and may not be feasible for larger sets.
3. A global constant can be used to fill in the missing attributes. The missing values may be 'Unknown' or be 'Infinity'. But, some data mining results may give spurious results by analysing these labels.
4. The attribute value may be filled by the attribute value. Say, the average income can replace a missing value.
5. Use the attribute mean for all samples belonging to the same class. Here, the average value replaces the missing values of all tuples that fall in this group.
6. Use the most possible value to fill in the missing value. The most probable value can be obtained from other methods like classification and decision tree prediction.

Noisy Data

BINNING TECHNIQUE



S = {12, 14, 19, 22, 24, 26, 28, 31, 34}

Bin 1	:	12 , 14, 19
Bin 2	:	22, 24, 26
Bin 3	:	28, 31, 32

By smoothing bins method, the bins are replaced by the bin means. This method results in:

Bin 1	:	15, 15, 15
Bin 2	:	24, 24, 24
Bin 3	:	30.3, 30.3, 30.3

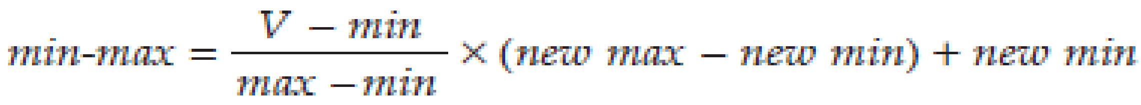
Using smoothing by bin boundaries method, the bins' values would be like:

Bin 1	:	12, 12, 19
Bin 2	:	22, 22, 26
Bin 3	:	28, 32, 32

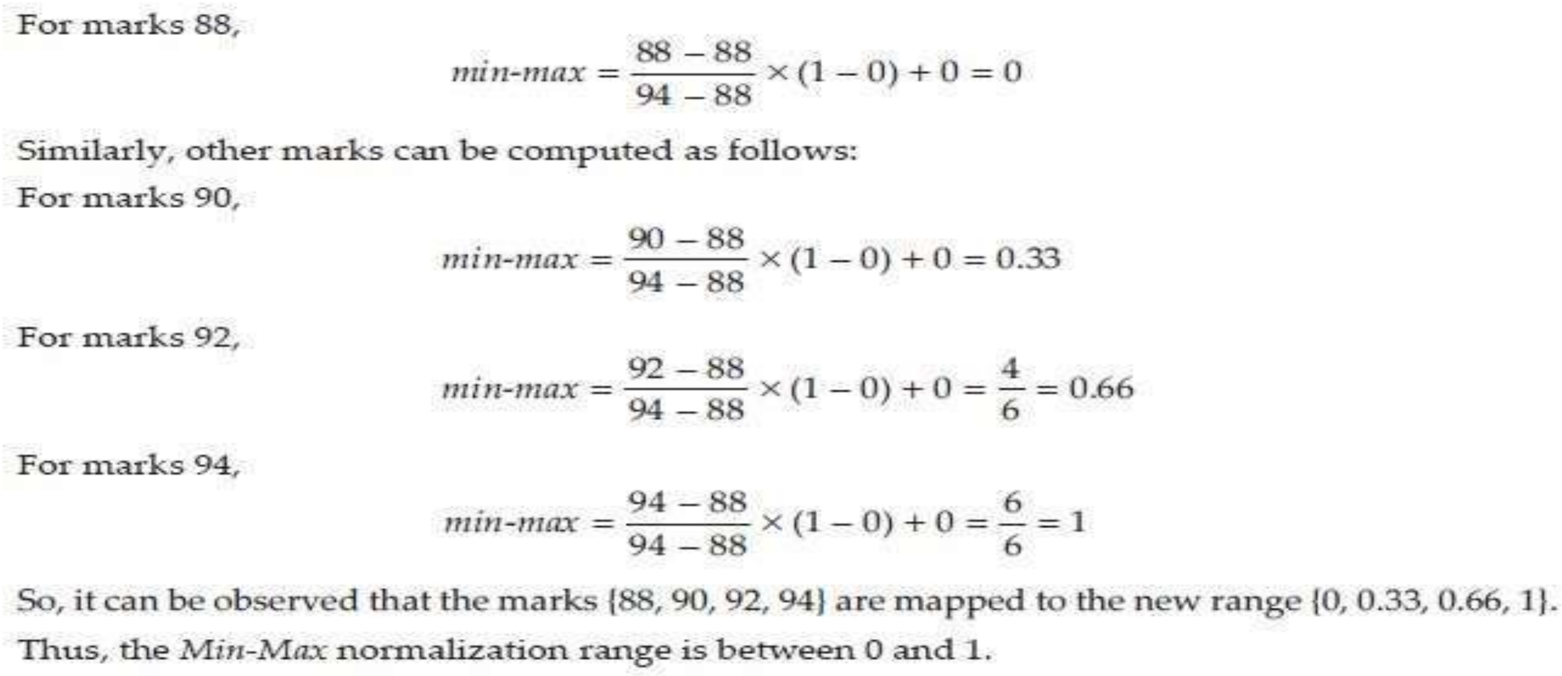
Data Normalization

MIN-MAX PROCEDURE

TRANSFORMS DATA TO THE RANGE 0-1



$$\text{min-max} = \frac{V - \text{min}}{\text{max} - \text{min}} \times (\text{new max} - \text{new min}) + \text{new min}$$



For marks 88,

$$\text{min-max} = \frac{88 - 88}{94 - 88} \times (1 - 0) + 0 = 0$$

Similarly, other marks can be computed as follows:

For marks 90,

$$\text{min-max} = \frac{90 - 88}{94 - 88} \times (1 - 0) + 0 = 0.33$$

For marks 92,

$$\text{min-max} = \frac{92 - 88}{94 - 88} \times (1 - 0) + 0 = \frac{4}{6} = 0.66$$

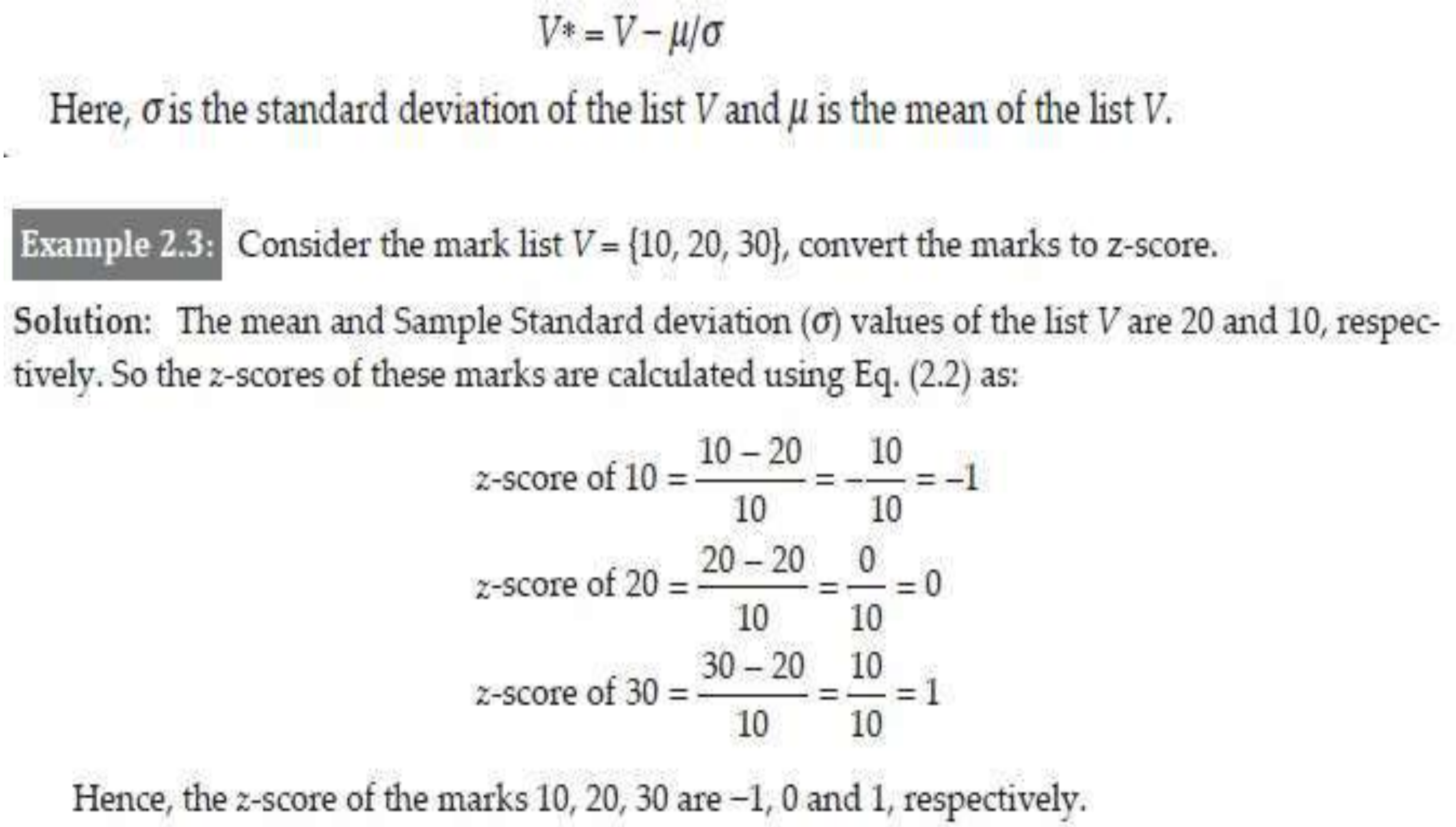
For marks 94,

$$\text{min-max} = \frac{94 - 88}{94 - 88} \times (1 - 0) + 0 = \frac{6}{6} = 1$$

So, it can be observed that the marks {88, 90, 92, 94} are mapped to the new range {0, 0.33, 0.66, 1}. Thus, the *Min-Max* normalization range is between 0 and 1.

Data Normalization

Z-SCORE



$$V* = V - \mu/\sigma$$

Here, σ is the standard deviation of the list V and μ is the mean of the list V.

Example 2.3: Consider the mark list $V = \{10, 20, 30\}$, convert the marks to z-score.

Solution: The mean and Sample Standard deviation (σ) values of the list V are 20 and 10, respectively. So the z-scores of these marks are calculated using Eq. (2.2) as:

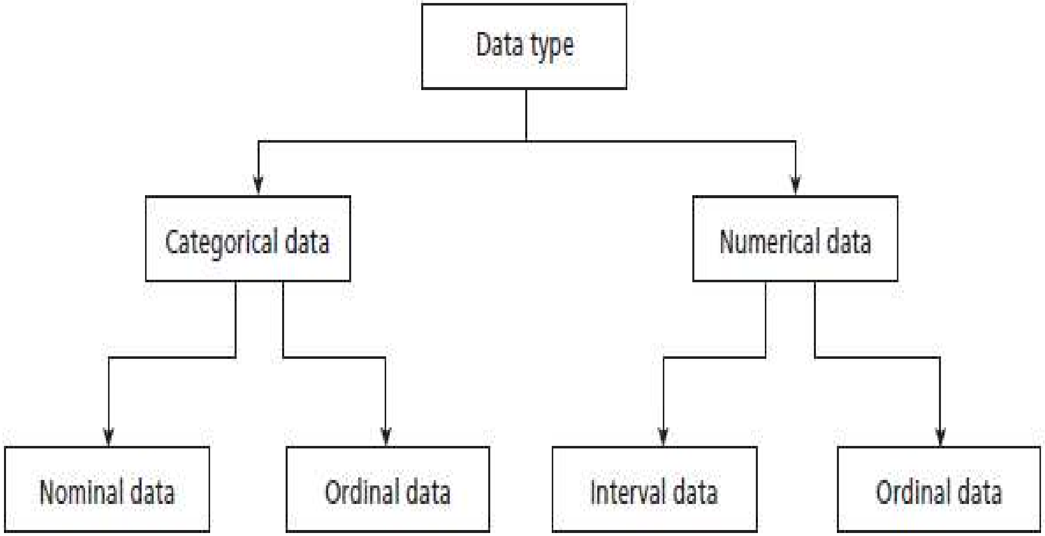
$$z\text{-score of } 10 = \frac{10-20}{10} = -\frac{10}{10} = -1$$

$$z\text{-score of } 20 = \frac{20-20}{10} = \frac{0}{10} = 0$$

$$z\text{-score of } 30 = \frac{30-20}{10} = \frac{10}{10} = 1$$

Hence, the z-score of the marks 10, 20, 30 are –1, 0 and 1, respectively.

Types of Data



Data type

Categorical data

Numerical data

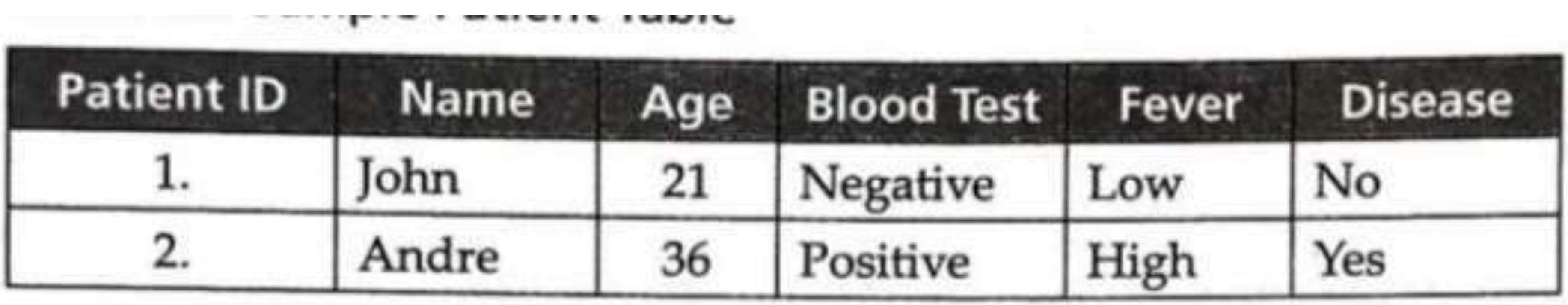
Nominal data

Ordinal data

Interval data

Ordinal data

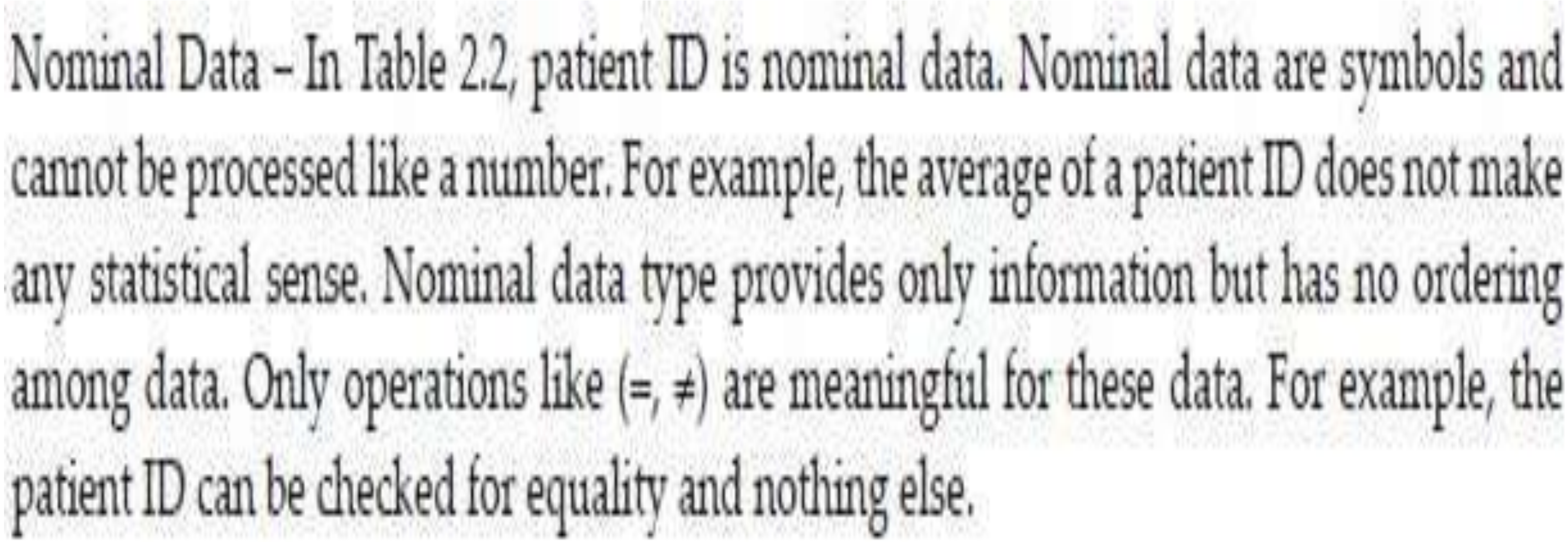
Figure 2.1: Types of Data



Sample Patient Table

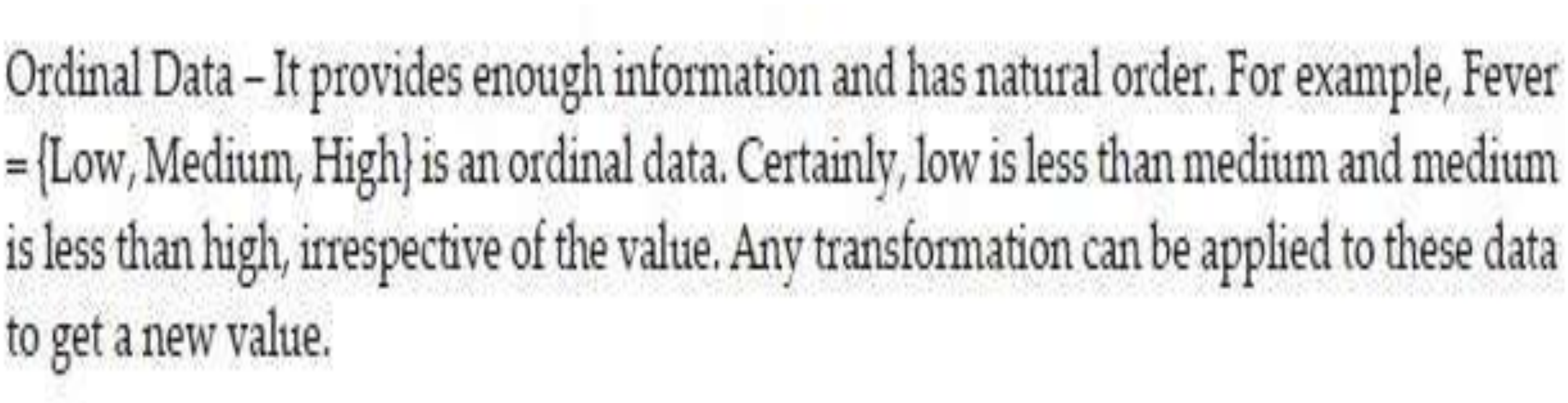
Patient ID	Name	Age	Blood Test	Fever	Disease
1.	John	21	Negative	Low	No
2.	Andre	36	Positive	High	Yes

Nominal Data



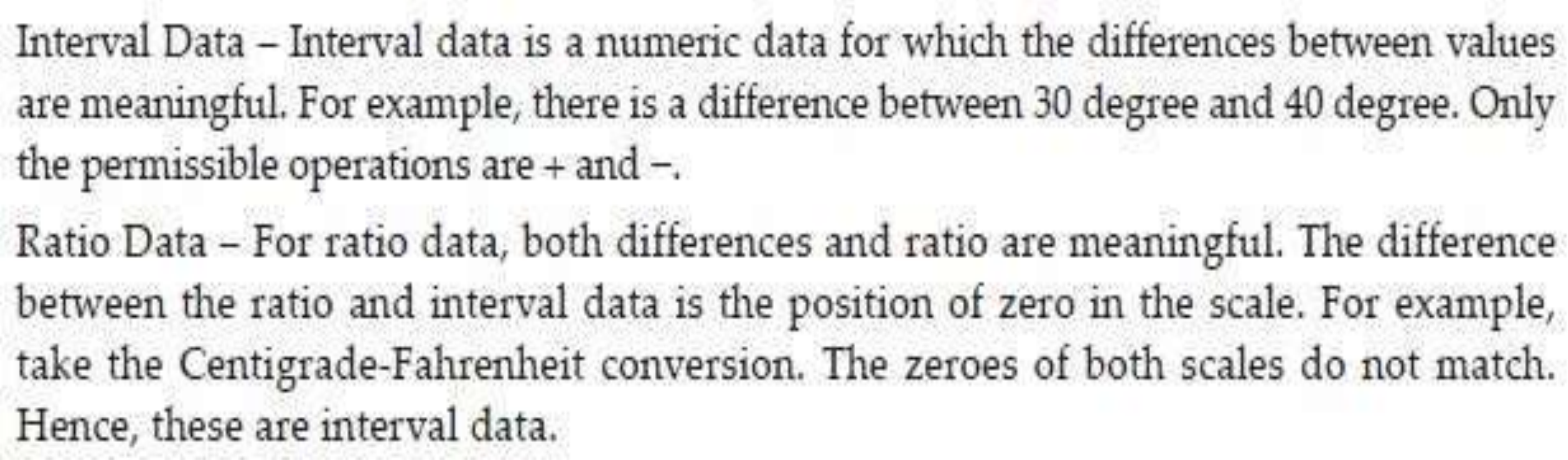
Nominal Data – In Table 2.2, patient ID is nominal data. Nominal data are symbols and cannot be processed like a number. For example, the average of a patient ID does not make any statistical sense. Nominal data type provides only information but has no ordering among data. Only operations like (=, ≠) are meaningful for these data. For example, the patient ID can be checked for equality and nothing else.

Ordinal Data



Ordinal Data – It provides enough information and has natural order. For example, Fever = {Low, Medium, High} is an ordinal data. Certainly, low is less than medium and medium is less than high, irrespective of the value. Any transformation can be applied to these data to get a new value.

Numerical Data



Interval Data – Interval data is a numeric data for which the differences between values are meaningful. For example, there is a difference between 30 degree and 40 degree. Only the permissible operations are + and –.

Ratio Data – For ratio data, both differences and ratio are meaningful. The difference between the ratio and interval data is the position of zero in the scale. For example, take the Centigrade-Fahrenheit conversion. The zeroes of both scales do not match. Hence, these are interval data.

Types of Data

BASED ON VARIABLES

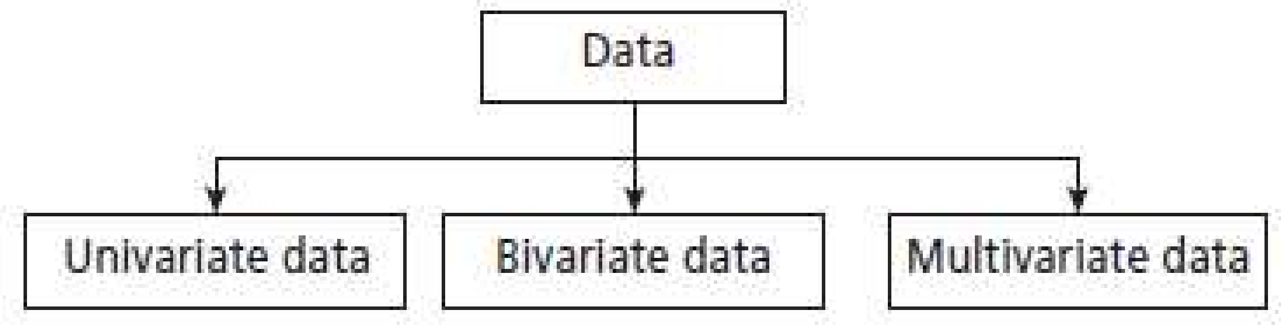
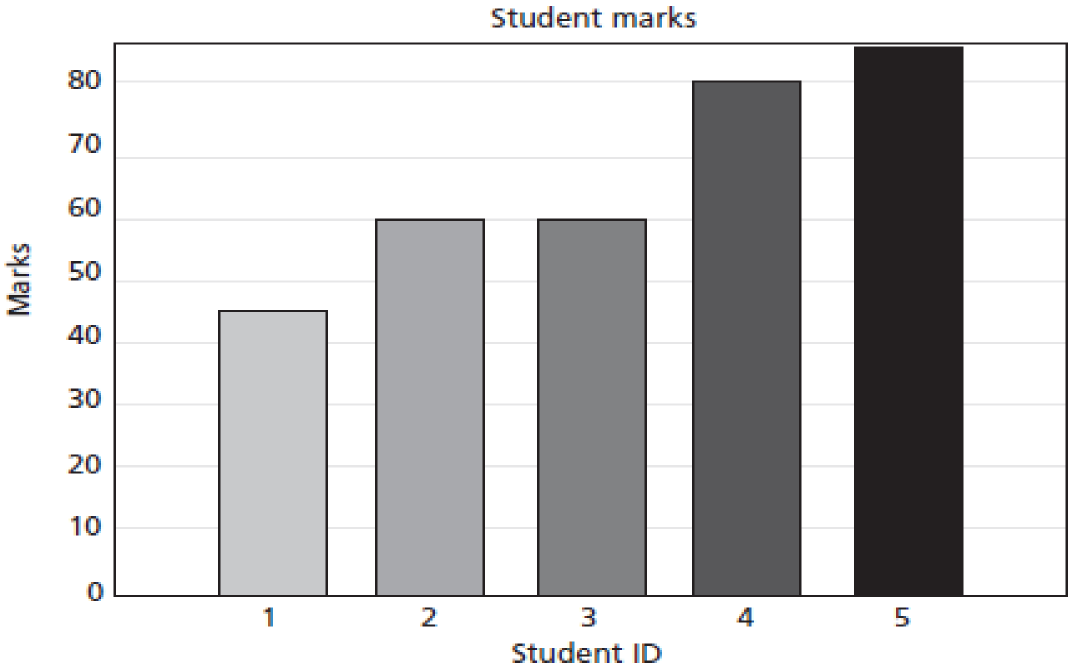


Figure 2.2: Types of Data Based on Variables

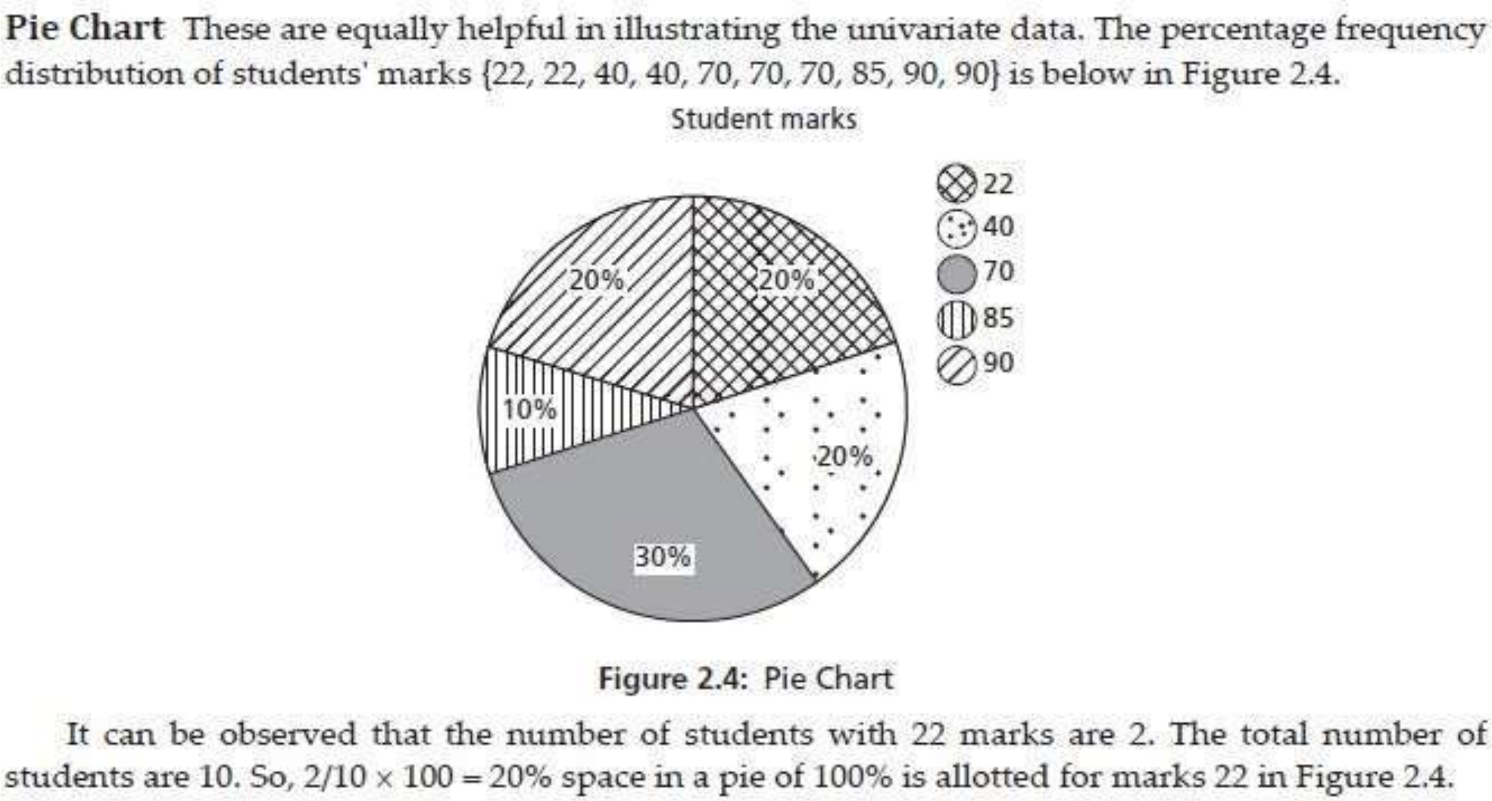
Data Visualization

Bar Chart A Bar chart (or Bar graph) is used to display the frequency distribution for variables. Bar charts are used to illustrate discrete data. The charts can also help to explain the counts of nominal data. It also helps in comparing the frequency of different groups.

The bar chart for students' marks {45, 60, 60, 80, 85} with Student ID = {1, 2, 3, 4, 5} is shown below in Figure 2.3.



Data Visualization



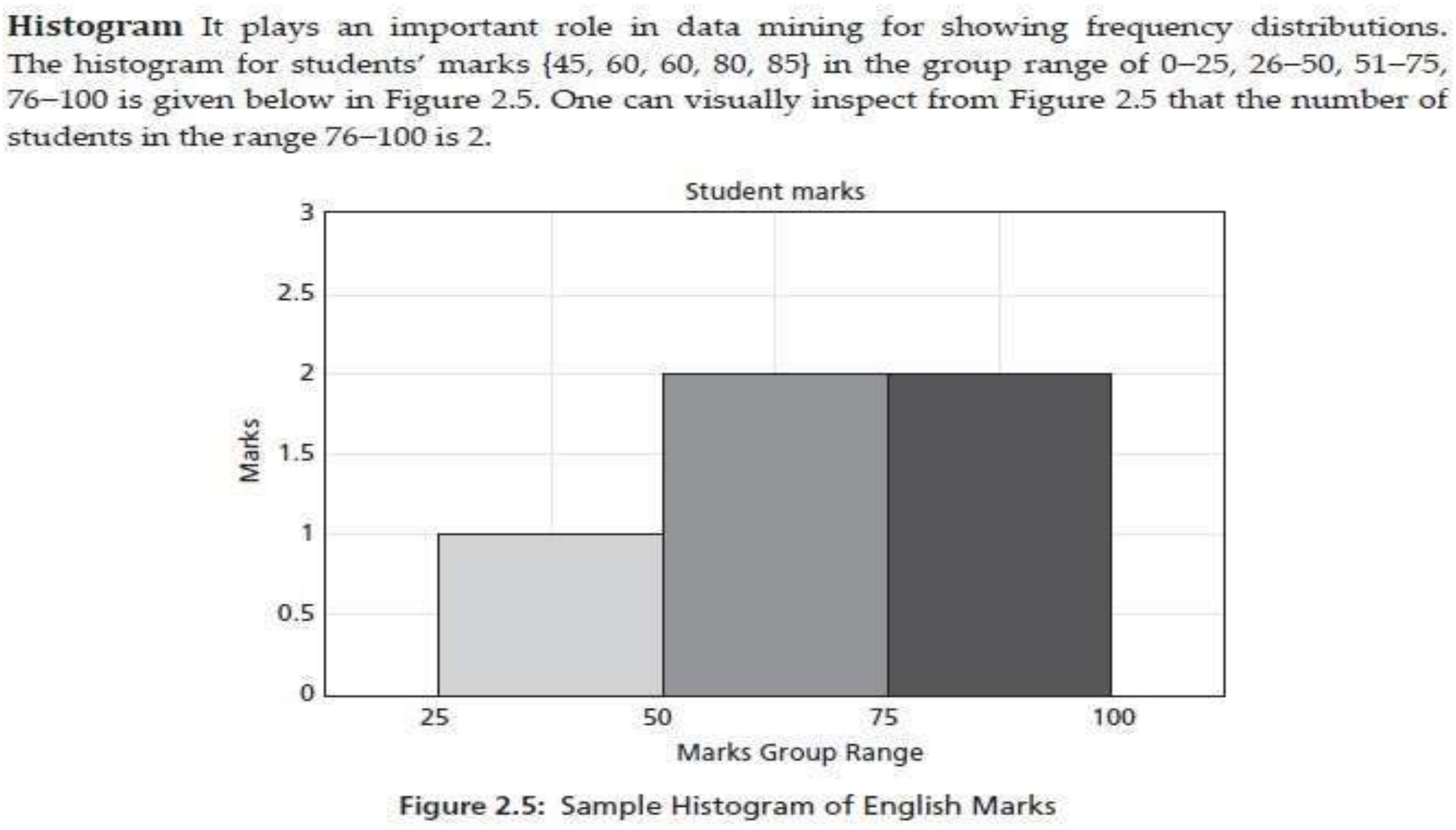
Pie Chart These are equally helpful in illustrating the univariate data. The percentage frequency distribution of students' marks {22, 22, 40, 40, 70, 70, 70, 85, 90, 90} is below in Figure 2.4.

Student marks
20%
20%
10%
20%
30%
▩ 22
⁘ 40
● 70
▥ 85
▨ 90

Figure 2.4: Pie Chart

It can be observed that the number of students with 22 marks are 2. The total number of students are 10. So, 2/10 × 100 = 20% space in a pie of 100% is allotted for marks 22 in Figure 2.4.

Data Visualization



Histogram It plays an important role in data mining for showing frequency distributions. The histogram for students' marks {45, 60, 60, 80, 85} in the group range of 0–25, 26–50, 51–75, 76–100 is given below in Figure 2.5. One can visually inspect from Figure 2.5 that the number of students in the range 76–100 is 2.

Figure 2.5: Sample Histogram of English Marks

Data Visualization

Dot Plots These are similar to bar charts. They are less clustered as compared to bar charts, as they illustrate the bars only with single points. The dot plot of English marks for five students with ID as {1, 2, 3, 4, 5} and marks {45, 60, 60, 80, 85} is given in Figure 2.6. The advantage is that by visual inspection one can find out who got more marks.

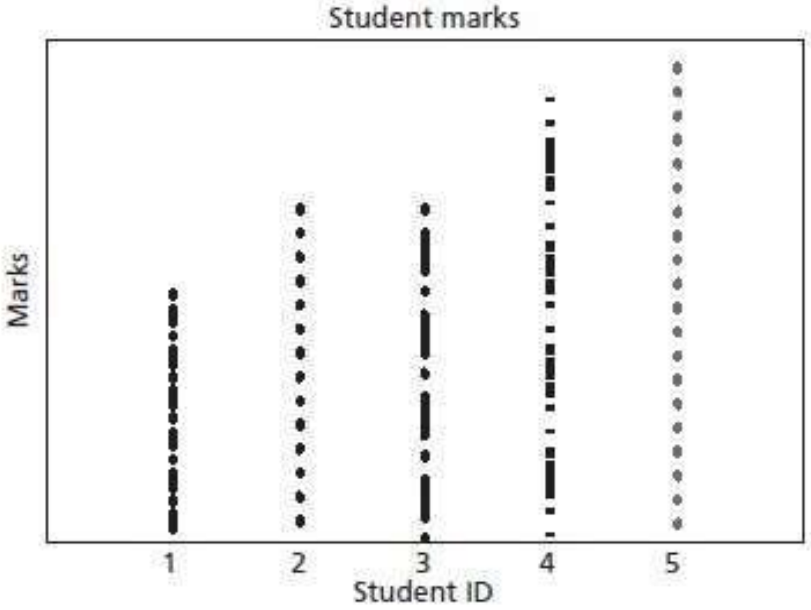


Figure 2.6: Dot Plots

Central Tendency

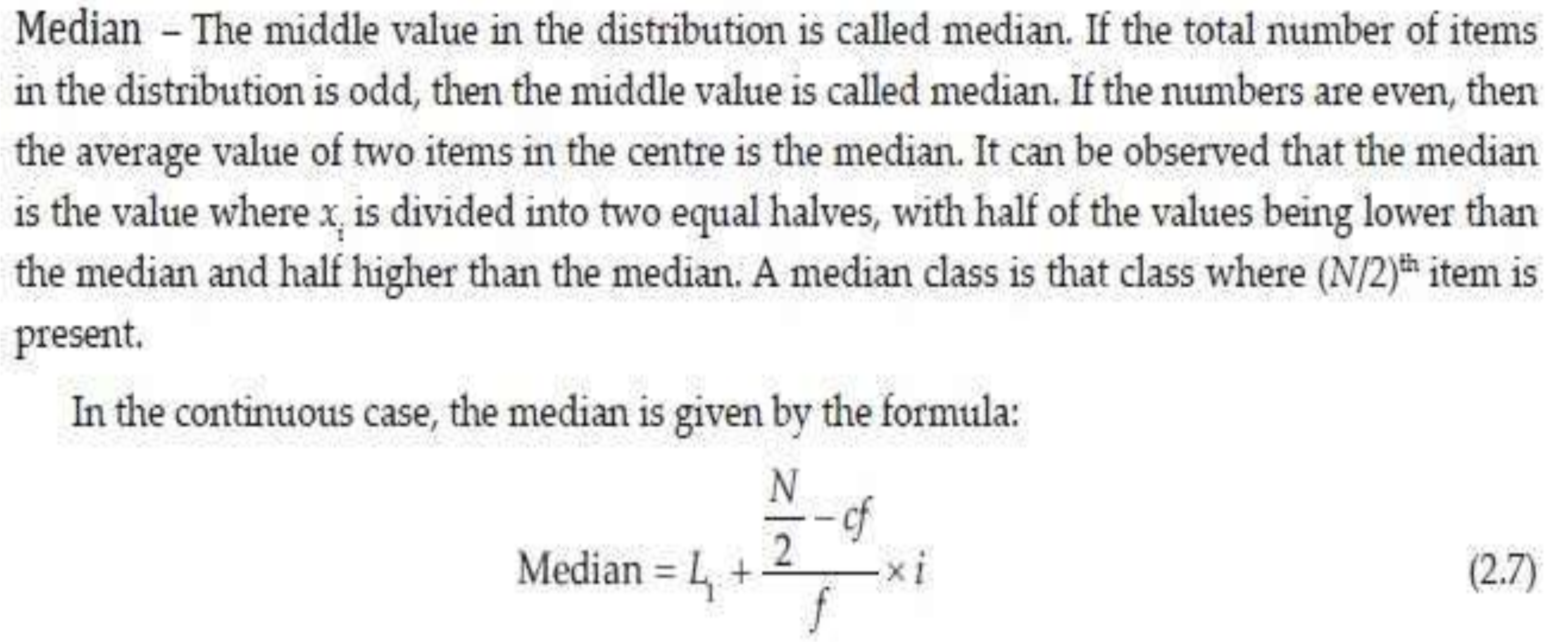
MEAN OF DATA

$$\bar{x} = \frac{x_1 + x_2 + \cdots + x_N}{N} = \frac{1}{N}\sum_{i=1}^{N} x_i$$

$$\text{Geometric mean} = \left(\prod_{i=1}^{n} x_i\right)^{\frac{1}{N}} = \sqrt[N]{x_1 \times x_2 \times \cdots \times x_N}$$

Central Tendency

MEDIAN OF DATA



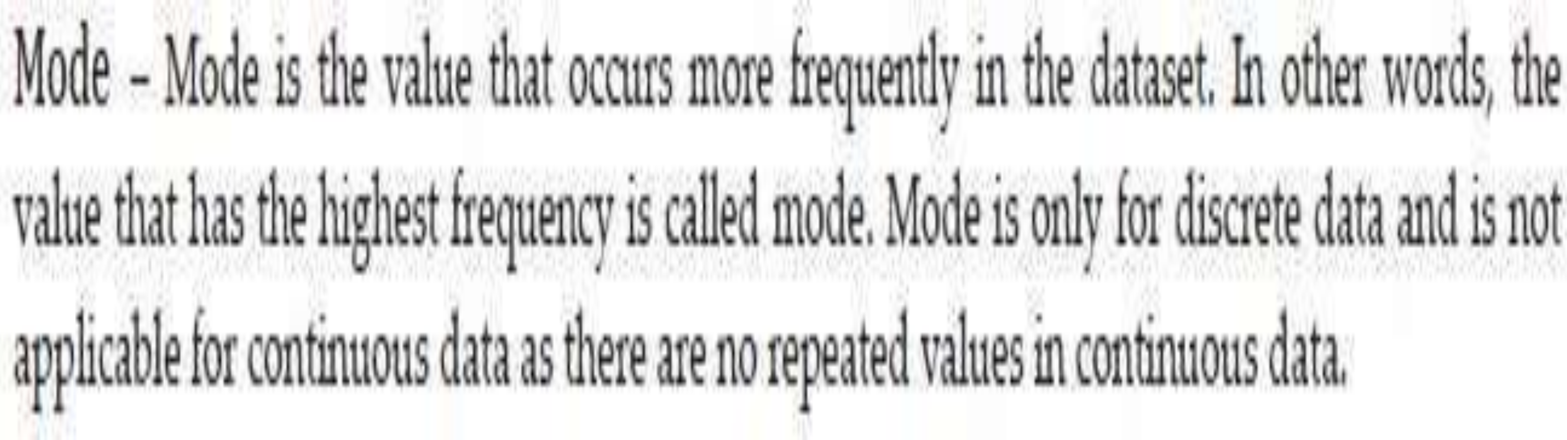
Median – The middle value in the distribution is called median. If the total number of items in the distribution is odd, then the middle value is called median. If the numbers are even, then the average value of two items in the centre is the median. It can be observed that the median is the value where x_i is divided into two equal halves, with half of the values being lower than the median and half higher than the median. A median class is that class where $(N/2)^{th}$ item is present.

In the continuous case, the median is given by the formula:

$$\text{Median} = L_1 + \frac{\frac{N}{2} - cf}{f} \times i \qquad (2.7)$$

Central Tendency

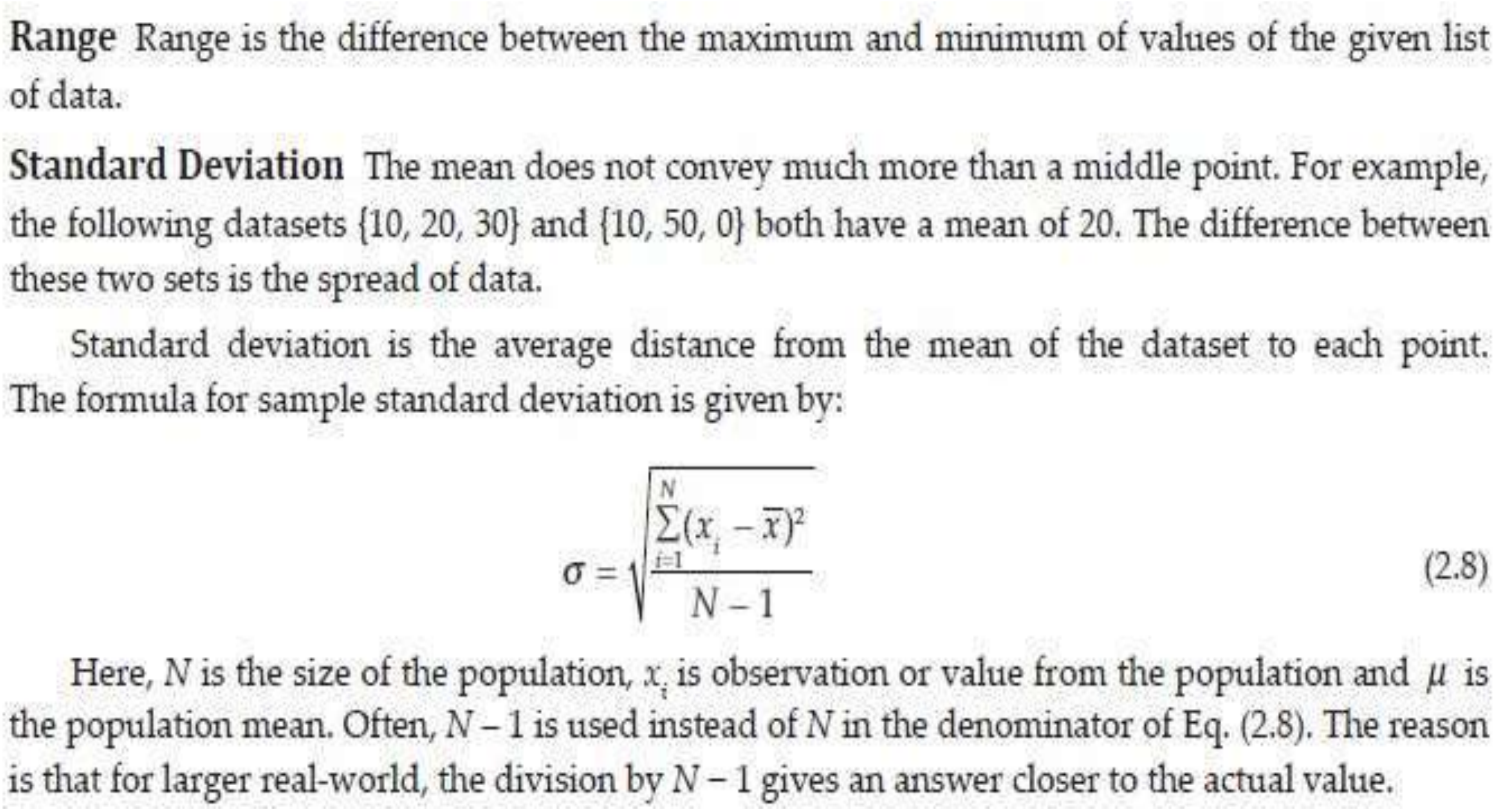
MODE OF DATA



Mode – Mode is the value that occurs more frequently in the dataset. In other words, the value that has the highest frequency is called mode. Mode is only for discrete data and is not applicable for continuous data as there are no repeated values in continuous data.

DISPERSION

RANGE AND STANDARD DEVIATION



Range Range is the difference between the maximum and minimum of values of the given list of data.

Standard Deviation The mean does not convey much more than a middle point. For example, the following datasets {10, 20, 30} and {10, 50, 0} both have a mean of 20. The difference between these two sets is the spread of data.

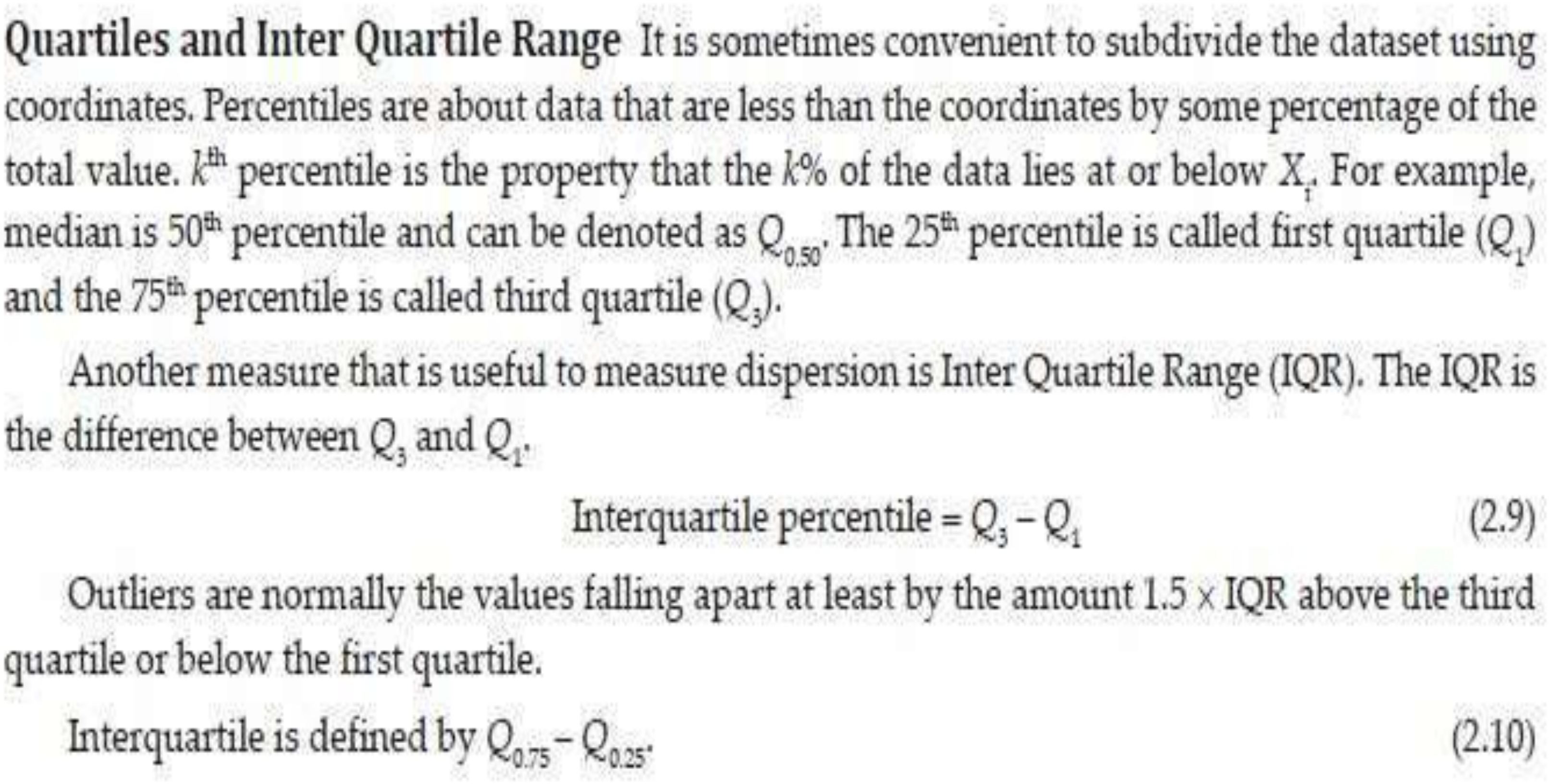
Standard deviation is the average distance from the mean of the dataset to each point. The formula for sample standard deviation is given by:

$$\sigma = \sqrt{\frac{\sum_{i=1}^{N}(x_i - \overline{x})^2}{N-1}} \tag{2.8}$$

Here, N is the size of the population, x_i is observation or value from the population and μ is the population mean. Often, $N - 1$ is used instead of N in the denominator of Eq. (2.8). The reason is that for larger real-world, the division by $N - 1$ gives an answer closer to the actual value.

DISPERSION

QUARTILES AND IQR



Quartiles and Inter Quartile Range It is sometimes convenient to subdivide the dataset using coordinates. Percentiles are about data that are less than the coordinates by some percentage of the total value. k^{th} percentile is the property that the k% of the data lies at or below X_i. For example, median is 50th percentile and can be denoted as $Q_{0.50}$. The 25th percentile is called first quartile (Q_1) and the 75th percentile is called third quartile (Q_3).

Another measure that is useful to measure dispersion is Inter Quartile Range (IQR). The IQR is the difference between Q_3 and Q_1.

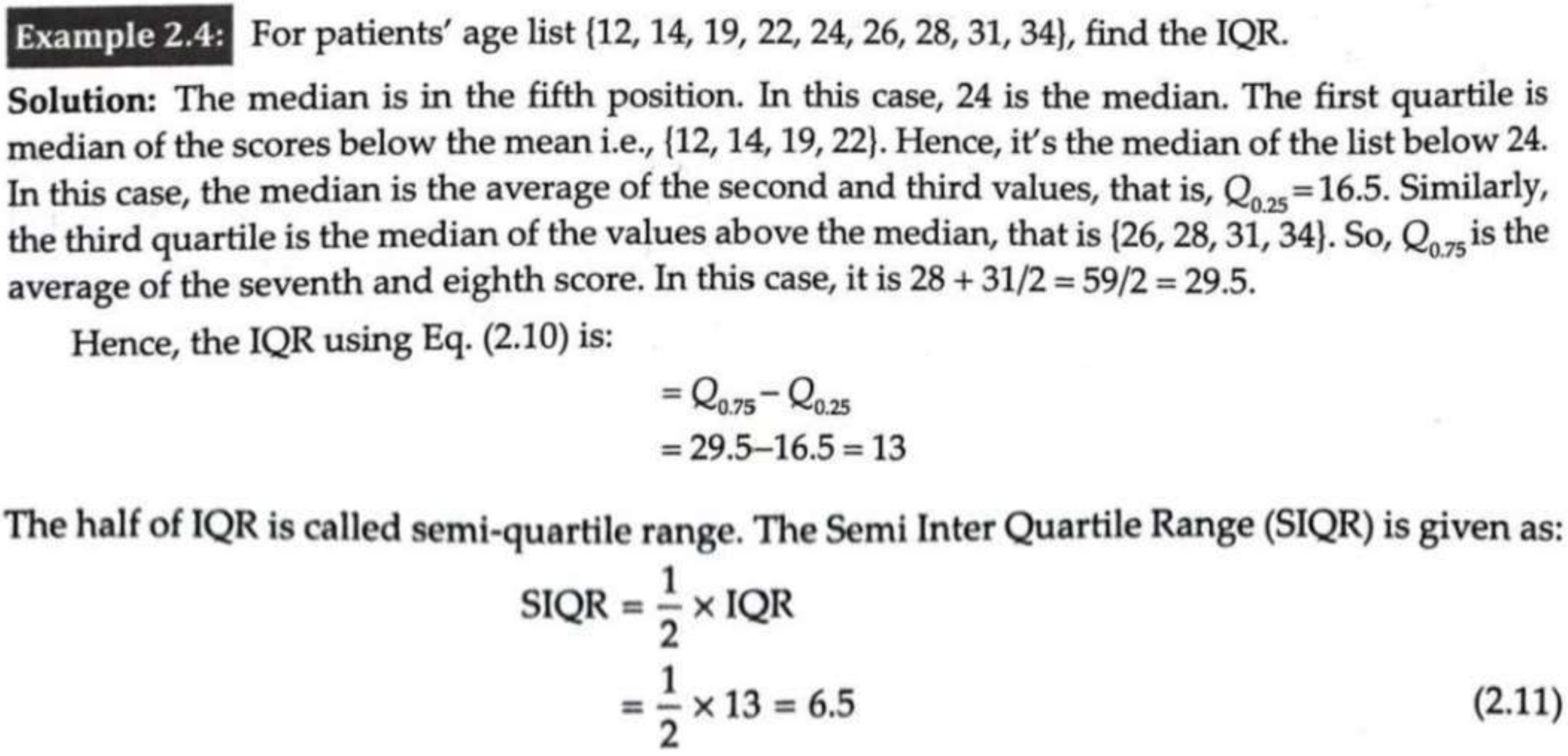
$$\text{Interquartile percentile} = Q_3 - Q_1 \qquad (2.9)$$

Outliers are normally the values falling apart at least by the amount 1.5 × IQR above the third quartile or below the first quartile.

Interquartile is defined by $Q_{0.75} - Q_{0.25}$. (2.10)

DISPERSION

QUARTILES AND IQR



Example 2.4: For patients' age list {12, 14, 19, 22, 24, 26, 28, 31, 34}, find the IQR.

Solution: The median is in the fifth position. In this case, 24 is the median. The first quartile is median of the scores below the mean i.e., {12, 14, 19, 22}. Hence, it's the median of the list below 24. In this case, the median is the average of the second and third values, that is, $Q_{0.25} = 16.5$. Similarly, the third quartile is the median of the values above the median, that is {26, 28, 31, 34}. So, $Q_{0.75}$ is the average of the seventh and eighth score. In this case, it is $28 + 31/2 = 59/2 = 29.5$.

Hence, the IQR using Eq. (2.10) is:

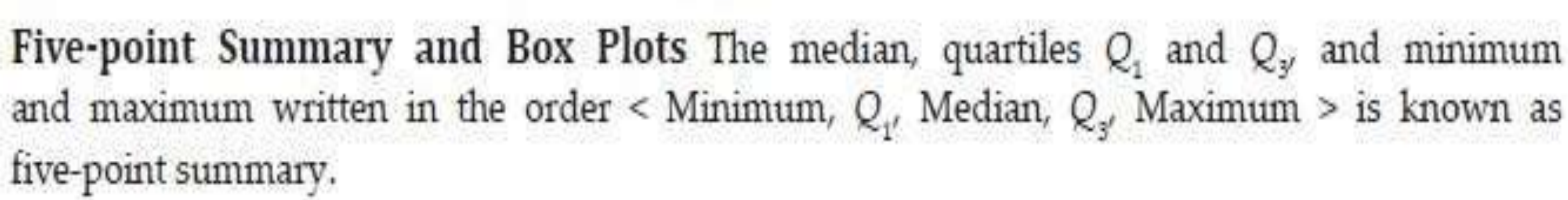
$$\begin{aligned} &= Q_{0.75} - Q_{0.25} \\ &= 29.5 - 16.5 = 13 \end{aligned}$$

The half of IQR is called semi-quartile range. The Semi Inter Quartile Range (SIQR) is given as:

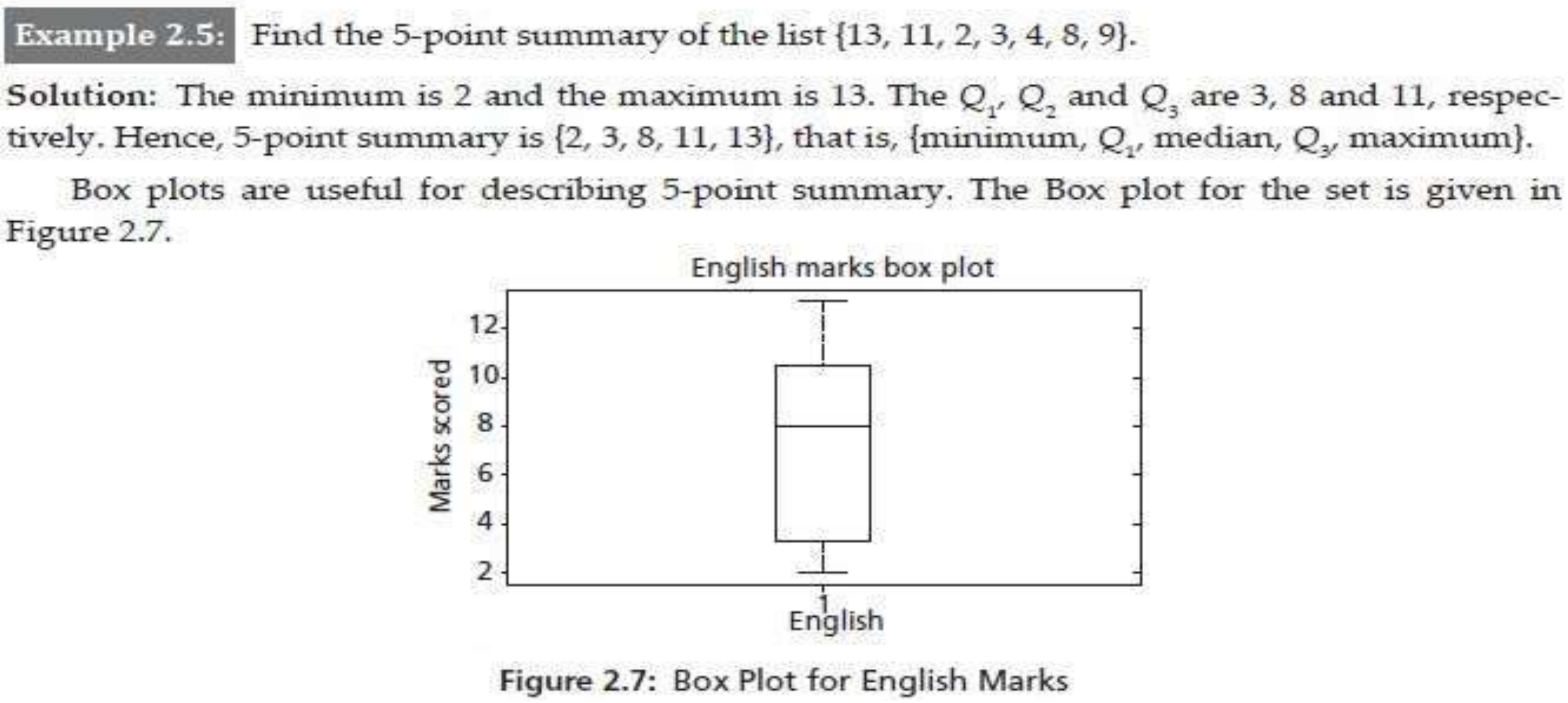
$$\begin{aligned} \text{SIQR} &= \frac{1}{2} \times \text{IQR} \\ &= \frac{1}{2} \times 13 = 6.5 \qquad (2.11) \end{aligned}$$

Five-point summary

5-POINT SUMMARY



Five-point Summary and Box Plots The median, quartiles Q_1 and Q_3, and minimum and maximum written in the order < Minimum, Q_1, Median, Q_3, Maximum > is known as five-point summary.



Example 2.5: Find the 5-point summary of the list {13, 11, 2, 3, 4, 8, 9}.

Solution: The minimum is 2 and the maximum is 13. The Q_1, Q_2 and Q_3 are 3, 8 and 11, respectively. Hence, 5-point summary is {2, 3, 8, 11, 13}, that is, {minimum, Q_1, median, Q_3, maximum}.

Box plots are useful for describing 5-point summary. The Box plot for the set is given in Figure 2.7.

English marks box plot
Marks scored
12
10
8
6
4
2
1
English

Figure 2.7: Box Plot for English Marks

Shape of Data

SKEWNESS AND KURTOSIS

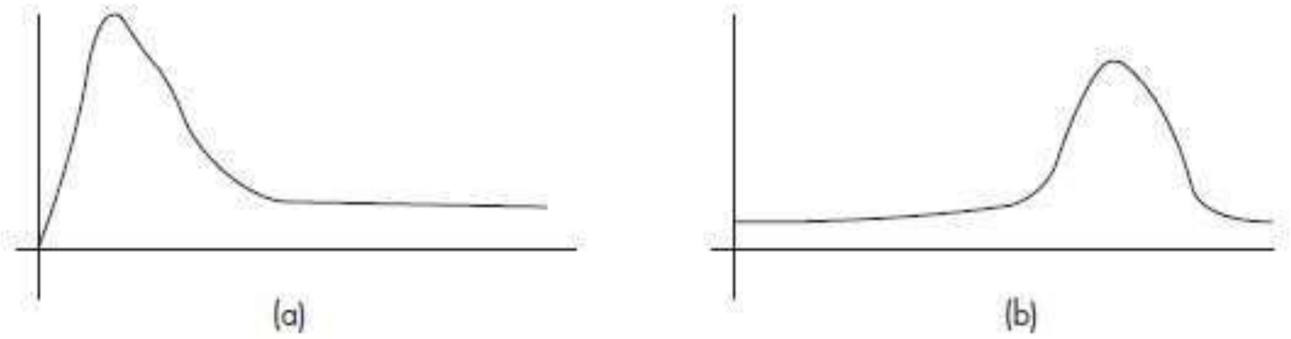


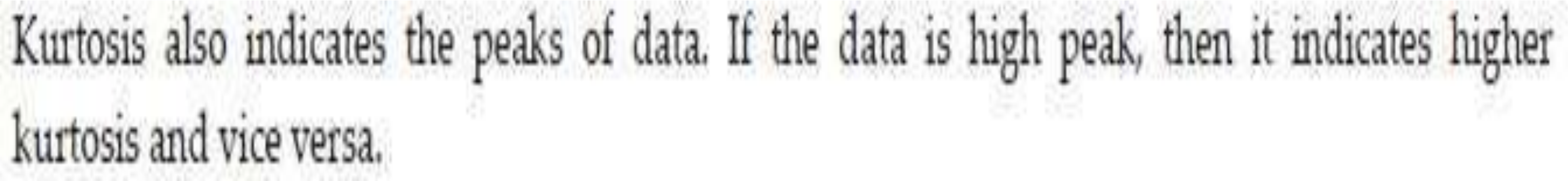
Figure 2.8: (a) Positive Skewed and (b) Negative Skewed Data

Also, the following measure is more commonly used to measure skewness. Let X_1, X_2, $\cdots$, X_N be a set of 'N' values or observations then the skewness can be given as:

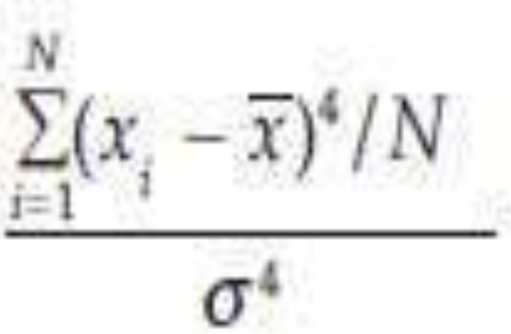
$$\frac{1}{N} \times \sum_{i=1}^{N} \frac{(x_i - \mu)^3}{\sigma^3} \tag{2.13}$$

Shape of Data

KURTOSIS



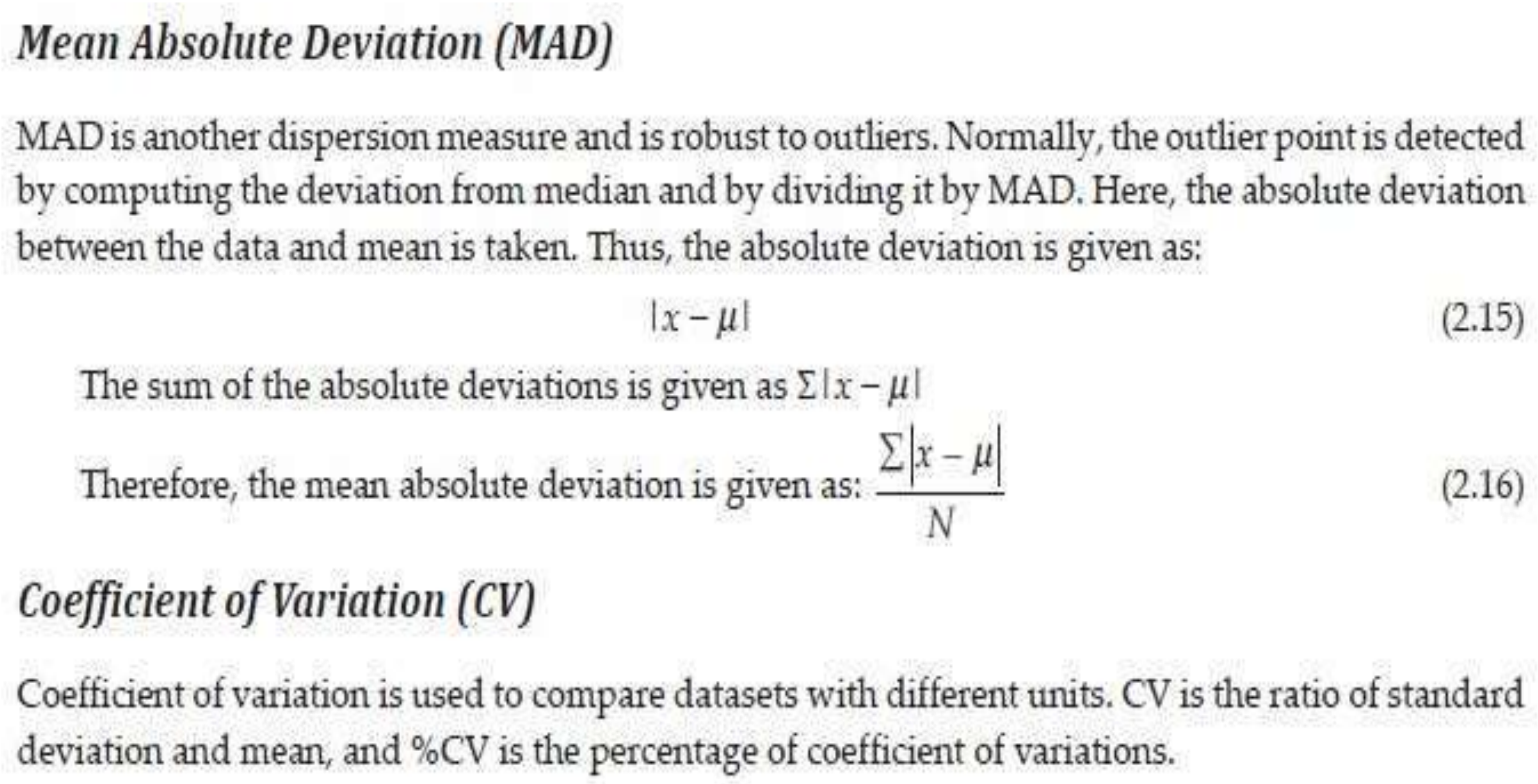
Kurtosis also indicates the peaks of data. If the data is high peak, then it indicates higher kurtosis and vice versa.



$$\frac{\sum_{i=1}^{N}(x_i - \overline{x})^4 / N}{\sigma^4}$$

Shape of Data

MEAN ABSOLUTE DEVIATION AND COEFFICIENT OF VARIATION



Mean Absolute Deviation (MAD)

MAD is another dispersion measure and is robust to outliers. Normally, the outlier point is detected by computing the deviation from median and by dividing it by MAD. Here, the absolute deviation between the data and mean is taken. Thus, the absolute deviation is given as:

$$|x - \mu| \tag{2.15}$$

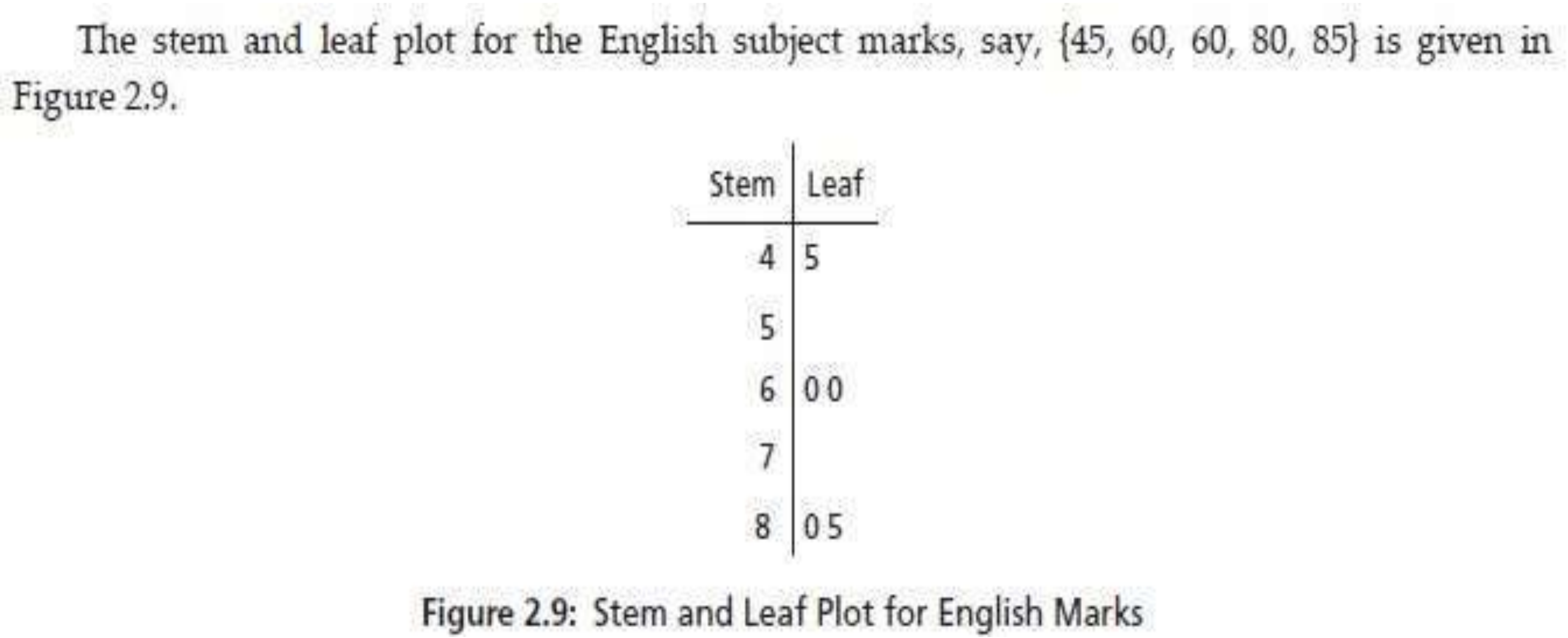
The sum of the absolute deviations is given as $\Sigma |x - \mu|$

Therefore, the mean absolute deviation is given as: $\dfrac{\Sigma|x - \mu|}{N}$ (2.16)

Coefficient of Variation (CV)

Coefficient of variation is used to compare datasets with different units. CV is the ratio of standard deviation and mean, and %CV is the percentage of coefficient of variations.

Stem-Leaf Plot



The stem and leaf plot for the English subject marks, say, {45, 60, 60, 80, 85} is given in Figure 2.9.

Stem	Leaf
4	5
5	
6	0 0
7	
8	0 5

Figure 2.9: Stem and Leaf Plot for English Marks

Bivariate Data

INVOLVES TWO VARIABLES

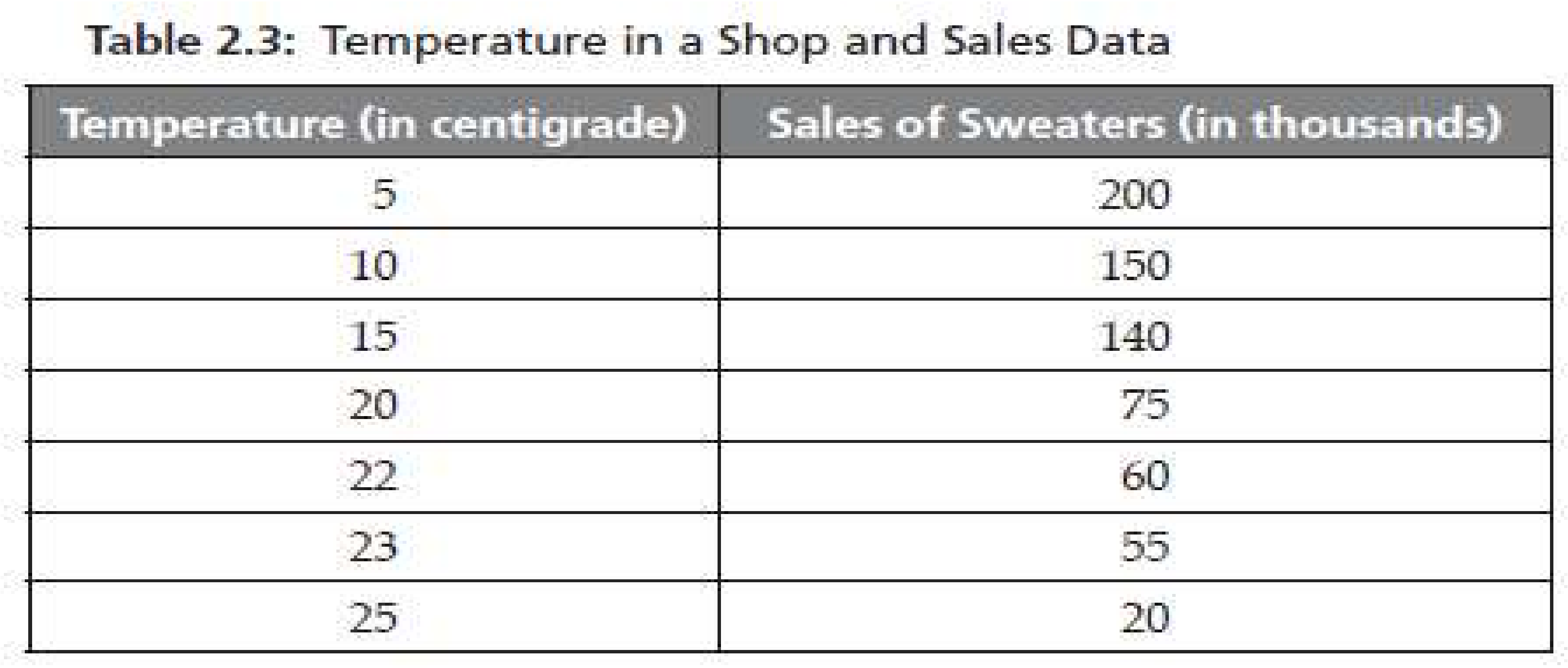


Table 2.3: Temperature in a Shop and Sales Data

Temperature (in centigrade)	Sales of Sweaters (in thousands)
5	200
10	150
15	140
20	75
22	60
23	55
25	20

Bivariate Data Visualization

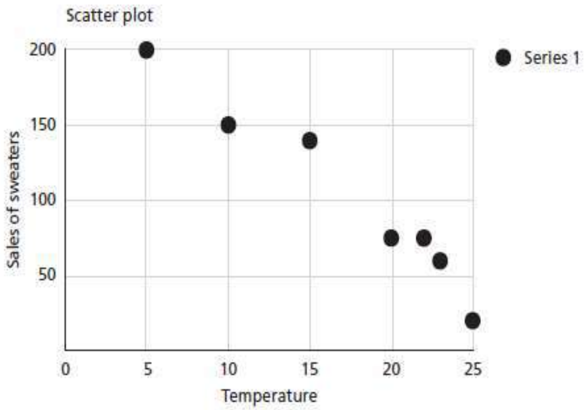


Figure 2.11: Scatter Plot

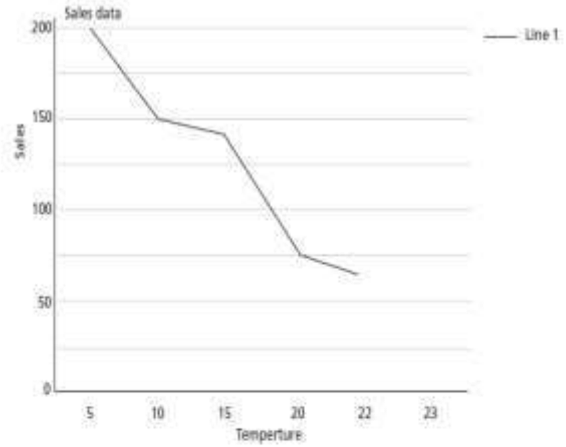
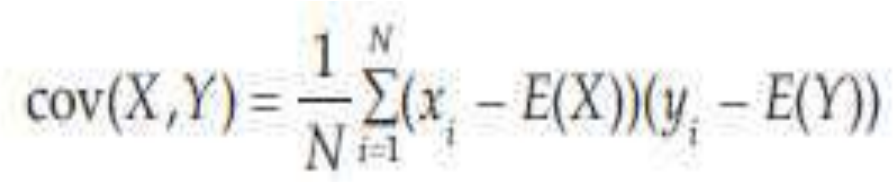
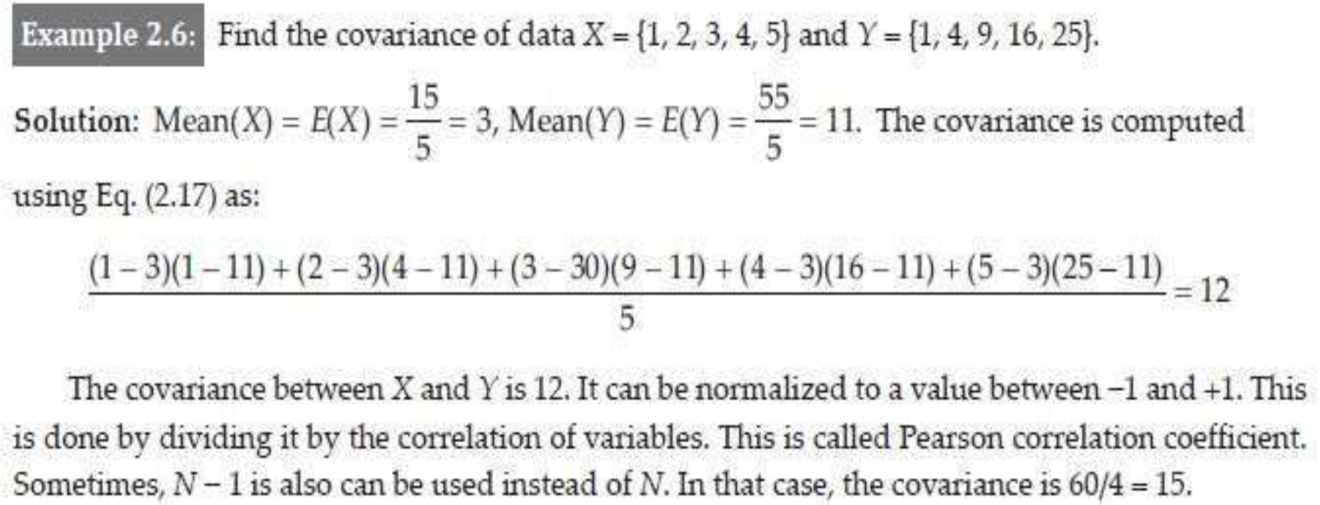


Figure 2.12: Line Chart

Bivariate Data – Covariance



$$\text{cov}(X,Y) = \frac{1}{N}\sum_{i=1}^{N}(x_i - E(X))(y_i - E(Y))$$



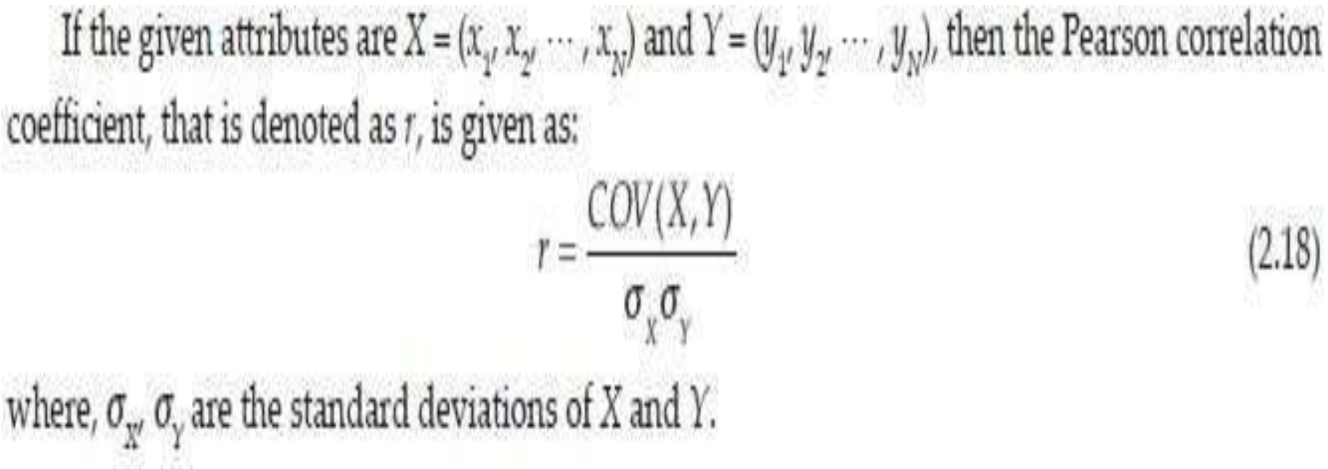
Example 2.6: Find the covariance of data X = {1, 2, 3, 4, 5} and Y = {1, 4, 9, 16, 25}.

Solution: Mean(X) = $E(X) = \frac{15}{5} = 3$, Mean(Y) = $E(Y) = \frac{55}{5} = 11$. The covariance is computed using Eq. (2.17) as:

$$\frac{(1-3)(1-11)+(2-3)(4-11)+(3-30)(9-11)+(4-3)(16-11)+(5-3)(25-11)}{5} = 12$$

The covariance between X and Y is 12. It can be normalized to a value between −1 and +1. This is done by dividing it by the correlation of variables. This is called Pearson correlation coefficient. Sometimes, $N - 1$ is also can be used instead of N. In that case, the covariance is 60/4 = 15.

Bivariate Data – Correlation

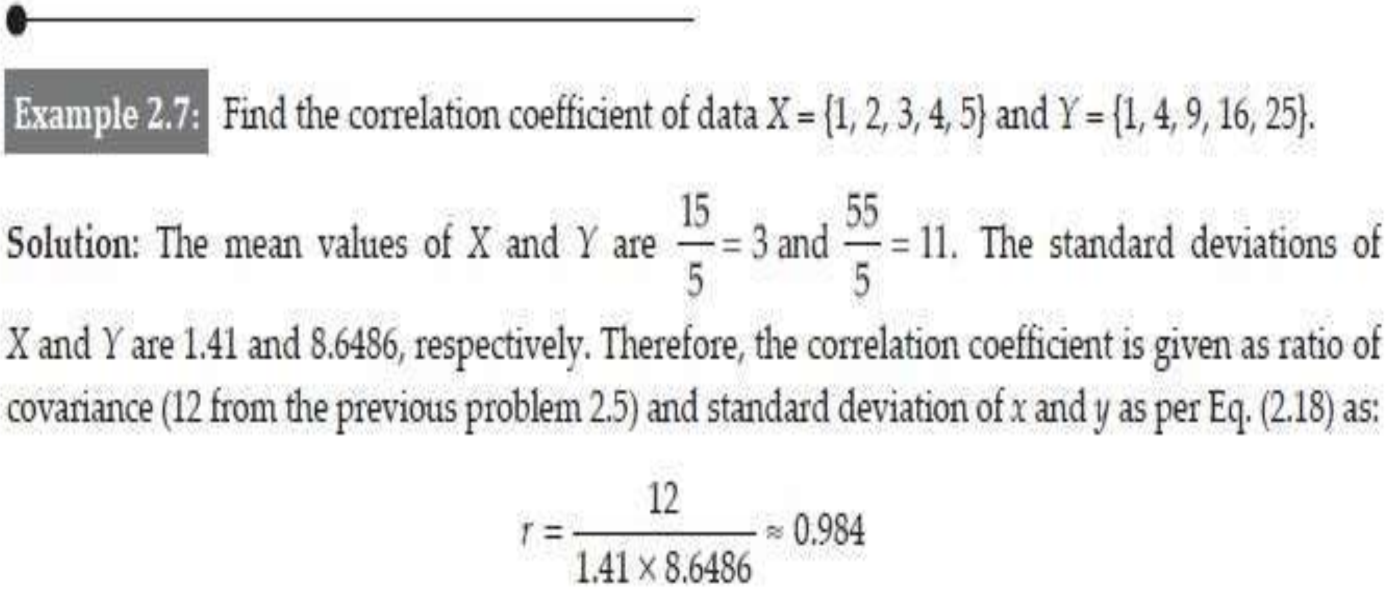


If the given attributes are $X = (x_1, x_2, \cdots, x_N)$ and $Y = (y_1, y_2, \cdots, y_N)$, then the Pearson correlation coefficient, that is denoted as r, is given as:

$$r = \frac{COV(X,Y)}{\sigma_X \sigma_Y} \tag{2.18}$$

where, σ_X, σ_Y are the standard deviations of X and Y.

Bivariate Data – Correlation



Example 2.7: Find the correlation coefficient of data $X = \{1, 2, 3, 4, 5\}$ and $Y = \{1, 4, 9, 16, 25\}$.

Solution: The mean values of X and Y are $\frac{15}{5} = 3$ and $\frac{55}{5} = 11$. The standard deviations of X and Y are 1.41 and 8.6486, respectively. Therefore, the correlation coefficient is given as ratio of covariance (12 from the previous problem 2.5) and standard deviation of x and y as per Eq. (2.18) as:

$$r = \frac{12}{1.41 \times 8.6486} \approx 0.984$$

Thank you for your attention

Acknowledgments

- *Hichem Felouat hichemfel@gmail.com*
- *and S. Sridhar and M. Vijayalakshmi*