



**SCHOOL OF COMPUTER SCIENCE,ENGINEERING
AND APPLICATIONS ,**

BHARATHIDASAN UNIVERSITY,

KHAJAMALAI CAMPUS,

TRICHY-620 023

MACHINE LEARNING

(SUBJECT CODE: MCS24024)

STUDY MATERIAL

FACULTY NAME : Mrs. R. RAMYA

DESIGNATION : GUEST LECTURER

SEMESTER : II

CLASS : M.S.C (CS)

INTRODUCTION TO MACHINE LEARNING

UNIT-1

THE JOURNEY OF ML

THE JOURNEY OF MACHINE LEARNING

- | | |
|------|---|
| 1950 | Alan Turing proposes "learning machine" |
| 1952 | Arthur Samuel developed first machine learning program that could play Checkers |
| 1957 | Frank Rosenblatt designed the first neural network program simulating human brain |
| 1967 | Nearest neighbour algorithm created – start of basic pattern recognition |
| 1979 | Stanford University students develop first self – driving cart that can navigate and avoid obstacles in a room |
| 1982 | Recurrent Neural Network developed |
| 1989 | - Reinforcement Learning conceptualized
- Beginning of commercialization of Machine Learning |
| 1995 | Random Forest and Support Vector machine algorithms developed |
| 1997 | IBM's Deep Blue beats the world chess champion Gary Kasparov |

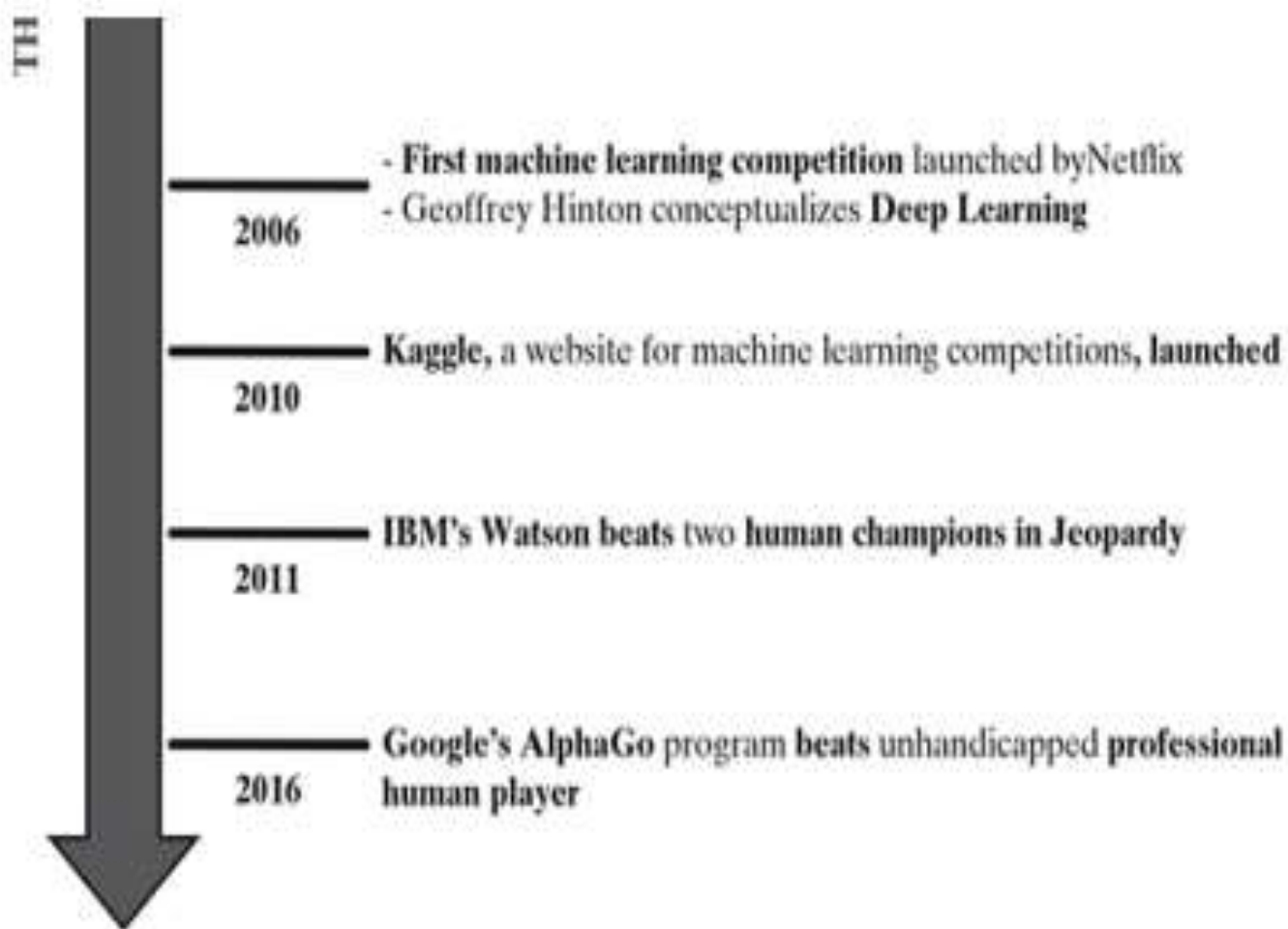


FIG. 1.1 Evolution of machine learning

WHAT IS HUMAN LEARNING?

- Human learning, in cognitive science, is the process of acquiring information through observation.
- The necessity to learn arises from the need to perform various daily tasks efficiently, ranging from simple activities like walking down the street to complex tasks such as determining the angle for launching a rocket.
- As individuals acquire more information, their efficiency in task execution improves. There are three types of human learning:

TYPES OF HUMAN LEARNING

1. Learning under expert guidance:

- In this type, individuals acquire knowledge directly from an expert in a particular field.
- Examples include an infant learning basic traits from guardians, a student learning alphabets and complex subjects from teachers, and professionals gaining hands-on experience under the guidance of mentors.
- Guided learning is a key element in all phases of human life, involving the transfer of knowledge from someone with expertise based on past experiences.

2. Learning guided by knowledge gained from experts:

- Individuals learn indirectly based on information imparted by teachers or mentors at some point in time and in different contexts.
- For instance, a child grouping objects by color based on previous information from parents, a student categorizing words as verbs or nouns based on prior teaching, and a professional making decisions informed by past preferences communicated by a boss.
- This type involves the application of past information in diverse contexts for decision-making.

3. Learning by self:

- Individuals are left to learn on their own through personal experiences and mistakes.
- A classic example is a baby learning to walk by navigating obstacles and learning from falls.
- This type emphasizes self-directed learning, where individuals develop a checklist of do's and don'ts based on their own experiences.
- In summary, human learning involves a combination of direct guidance from experts, indirect learning based on past knowledge, and self-directed learning through personal experiences. Each type contributes to the overall process of gaining information and adapting to different situations throughout life.

WHAT IS MACHINE LEARNING?

Machine Learning Definition:

- Machine learning is the process of computers learning from experience E in performing tasks T , where the performance measure P improves with experience.

How Machines Learn ?

- Machines learn by gathering experience (E) from past data related to a specific task (T).
- Example: In playing checkers, experience is gained by playing the game, the task is playing checkers, and the performance measure is the percentage of games won.

Components of Machine Learning Process :

Three main components: Data Input, Abstraction, Generalization.

- Data Input: Utilizes past data for future decision-making.
- Abstraction: Represents input data more broadly through an underlying algorithm.
- Generalization: Abstracted representation is generalized to form a framework for decision-making.

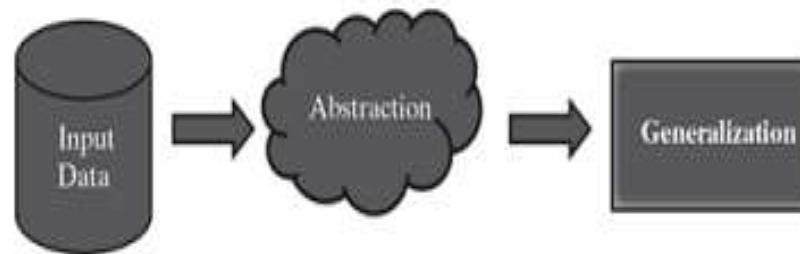


FIG. 1.2 Process of machine learning

Perspective on Machine Learning Process

- Machine learning process mirrors human learning, emphasizing abstraction and generalization.
- Analogous to human learning strategies, where memorization is insufficient for complex tasks.

Abstraction in Machine Learning :

- Knowledge from input data is abstracted into a model.
- Model forms a conceptual map or representation of raw data.
- Model types include computational blocks, mathematical equations, data structures, or logical groupings.

Choosing a Model

- Decision on model type based on problem type, nature of input data, and domain of the problem.

Fitting the Model

- Fitting the model involves determining values of coefficients or constants in the chosen model.
- Process known as training, with input data referred to as training data.

Generalization in Machine Learning

- Generalization tunes the abstracted knowledge to make future decisions.
- Challenges include aligning the trained model with actual trends and handling unknown characteristics in test data.

Heuristic Approach in Generalization

- Generalization involves an approximate or heuristic approach, similar to gut-feeling-based decision-making in humans.
- Acknowledges the risk of incorrect decisions due to assumptions not holding true in reality.

Well-posed learning problem

To define a machine learning problem, follow this simple framework:

- Step 1: Describe the problem informally and formally, list assumptions, and identify similar problems.
- Step 2: List the motivation, benefits, and solution use.
- Step 3: Describe manual problem-solving steps to gather domain knowledge.

- **Step 1: What is the Problem?** Gather information about the problem, including an informal description and formalism using Tom Mitchell's machine learning framework:
- Task (T): Prompt the next word when typing a word.
- Experience (E): Corpus of commonly used English words and phrases.
- Performance (P): Learning accuracy, measured as the percentage of correct prompted words.
- Assumptions: List assumptions about the problem and identify similar problems.

- **Step 2: Why does the Problem need to be Solved?**

Understand the motivation, benefits, and solution use:

- **Motivation:** Identify the purpose, such as solving business issues or suggesting movies.
- **Solution Benefits:** Consider the advantages of solving the problem.
- **Solution Use:** Explore how the solution will be used and its expected lifetime.

- **Step 3: How would I Solve the Problem?** Explore manual problem-solving steps, including data collection, preparation, and program design. Update previous sections with these details.

TYPES OF MACHINE LEARNING

- Machine learning can be classified into three broad categories:
 1. Supervised learning – Also called predictive learning. A machine predicts the class of unknown objects based on prior class-related information of similar objects.
 2. Unsupervised learning – Also called descriptive learning. A machine finds patterns in unknown objects by grouping similar objects together.
 3. Reinforcement learning – A machine learns to act on its own to achieve the given goals

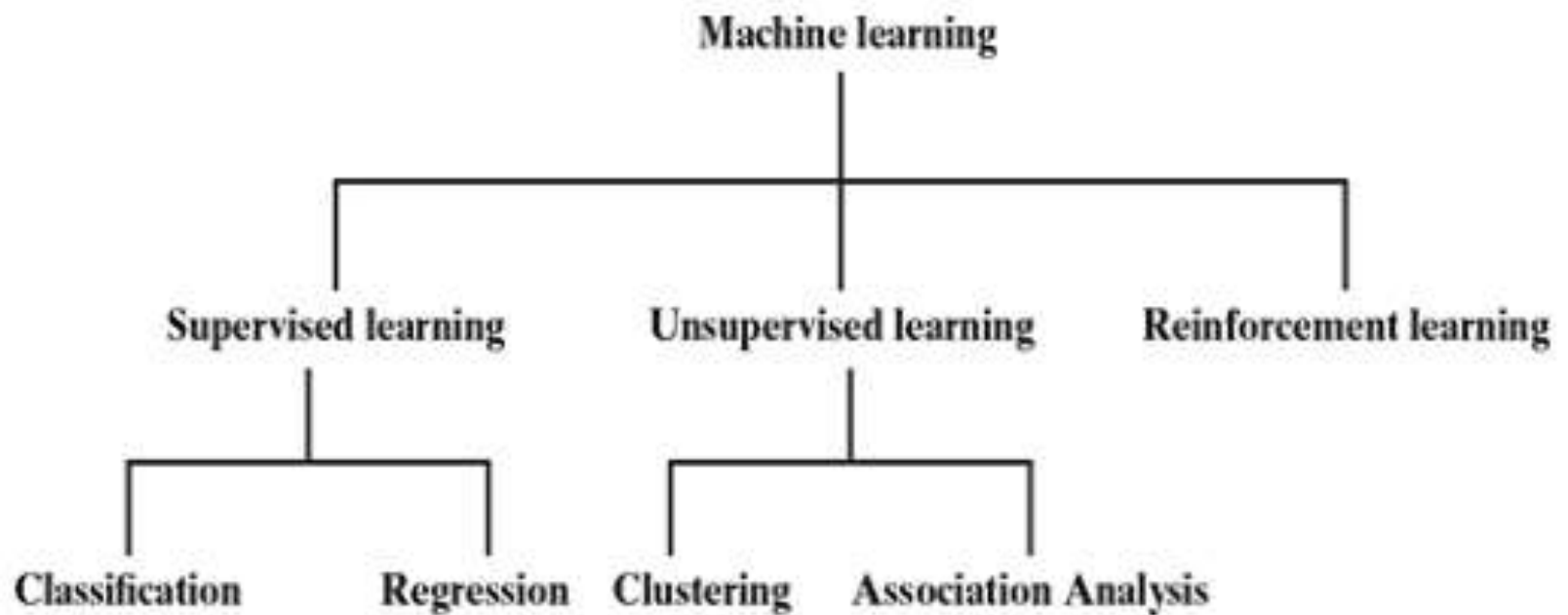


FIG. 1.3 Types of machine learning

Supervised Learning:

Motivation of Supervised Learning:

- Goal: Learn from past information.
- Past information: Experience about the task the machine needs to execute.

Example: Image Segregation Task:

- Input: Images of different objects.
- Task: Segregate images by shape or color (e.g., round or triangular, blue or green).
- Challenge: Machine needs basic information on shape and color, similar to guiding a child.
- Experience (Past Information): Training data containing labeled images and their characteristics.
- Label: Tag indicating round/triangular or blue/green.
- Training Data: Essential for machine learning.

Supervised Learning Process :

- Input: Labeled training data.
- Output: Predictive model.
- Usage: Apply model on test data to assign labels.

Examples of Supervised Learning:

- Predicting game results.
- Predicting tumor malignancy.
- Predicting prices (e.g., real estate, stocks).
- Text classification (e.g., spam detection).

Classification vs. Regression:

- Classification: Predicting categorical variables (e.g., types, classes).
- Regression: Predicting real-valued variables (e.g., prices).

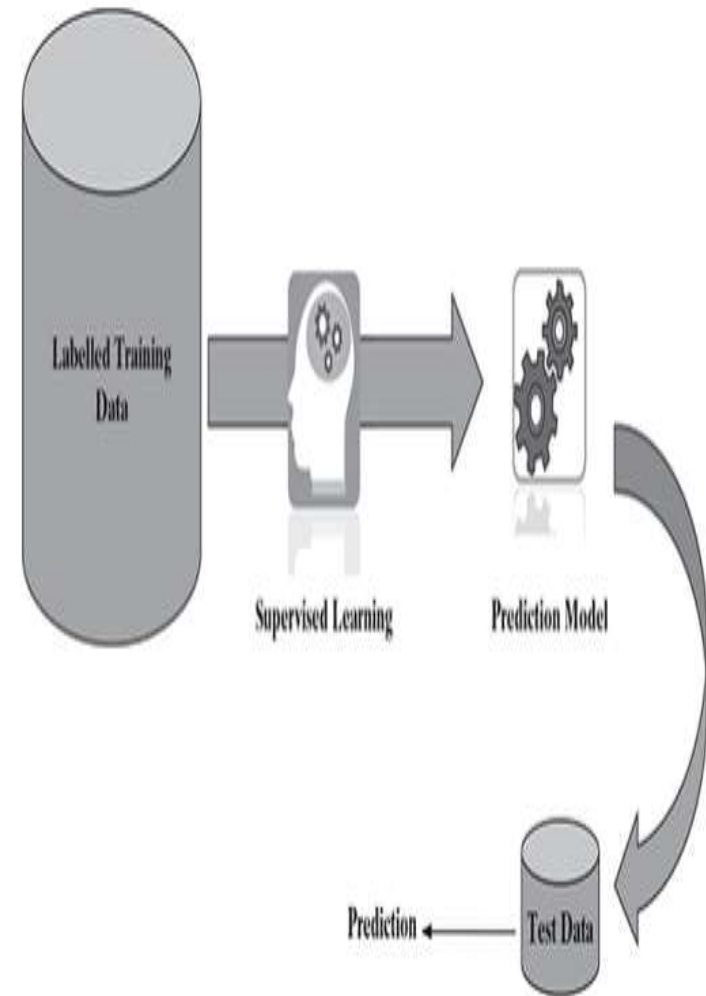


FIG. 1.4 Supervised learning

Classification:

- Problem: Assigning labels or categories to test data based on training data.
- Process: Map new image/test data to similar labeled images.
- Examples: Image classification, disease prediction, win-loss prediction.

Popular Classification Algorithms:

- Naïve Bayes, Decision tree, k-Nearest Neighbour.

Critical Classification Example:

- Banking: Identifying potential fraudulent transactions.
- Method: Use labeled past transaction data to flag new transactions as normal or suspicious.

Summary - Classification:

- Supervised learning predicting a categorical target feature.
- Target feature: Class or category.

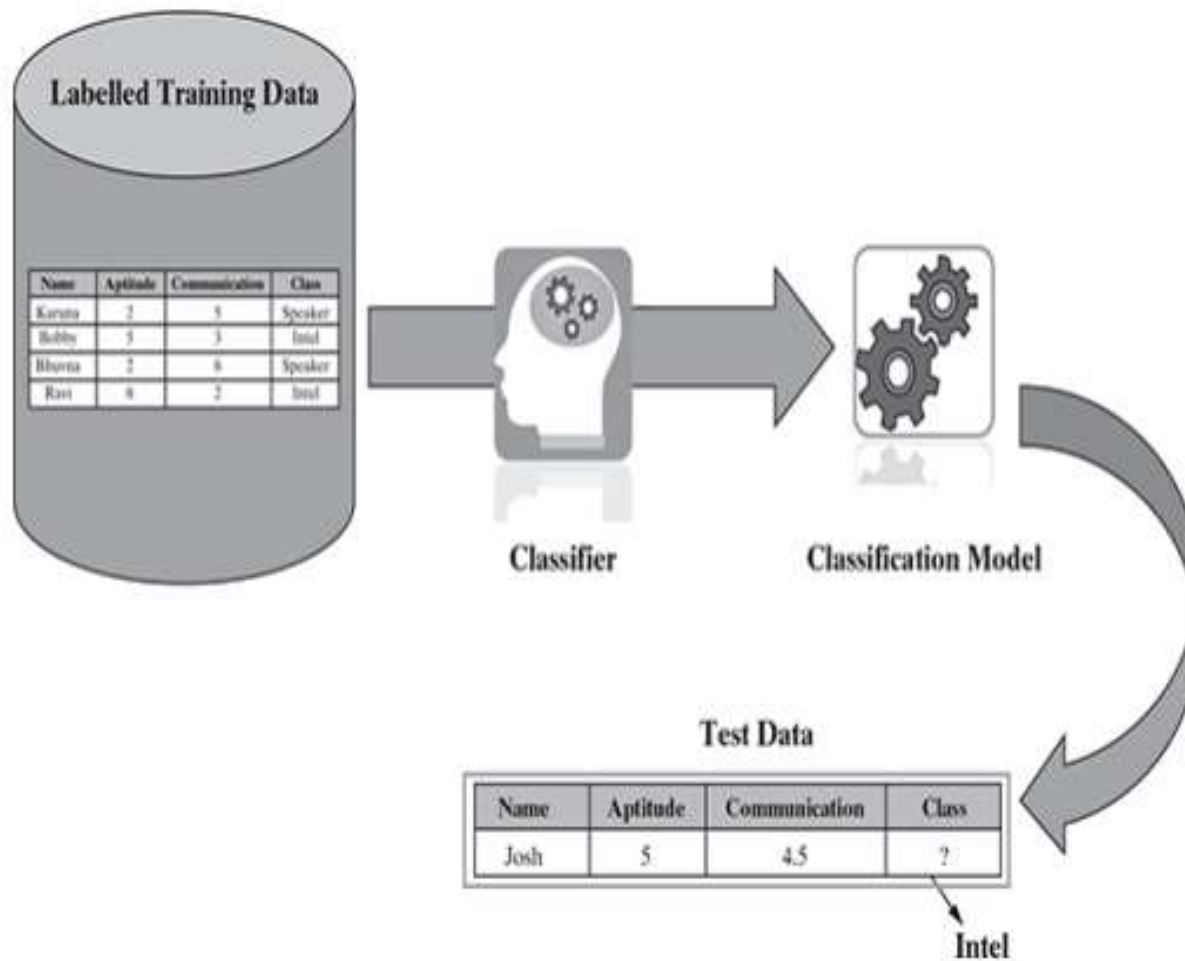


FIG. 1.5 Classification

Regression:

- Objective: Predict numerical features (e.g., real estate or stock price).
- Variables: Continuous predictor and target variables.
- Model: Linear regression uses least squares method.

Example: Yearly Sales Prediction - Regression:

- Predictor: Investment.
- Target: Sales revenue.
- Model: Simple linear regression.

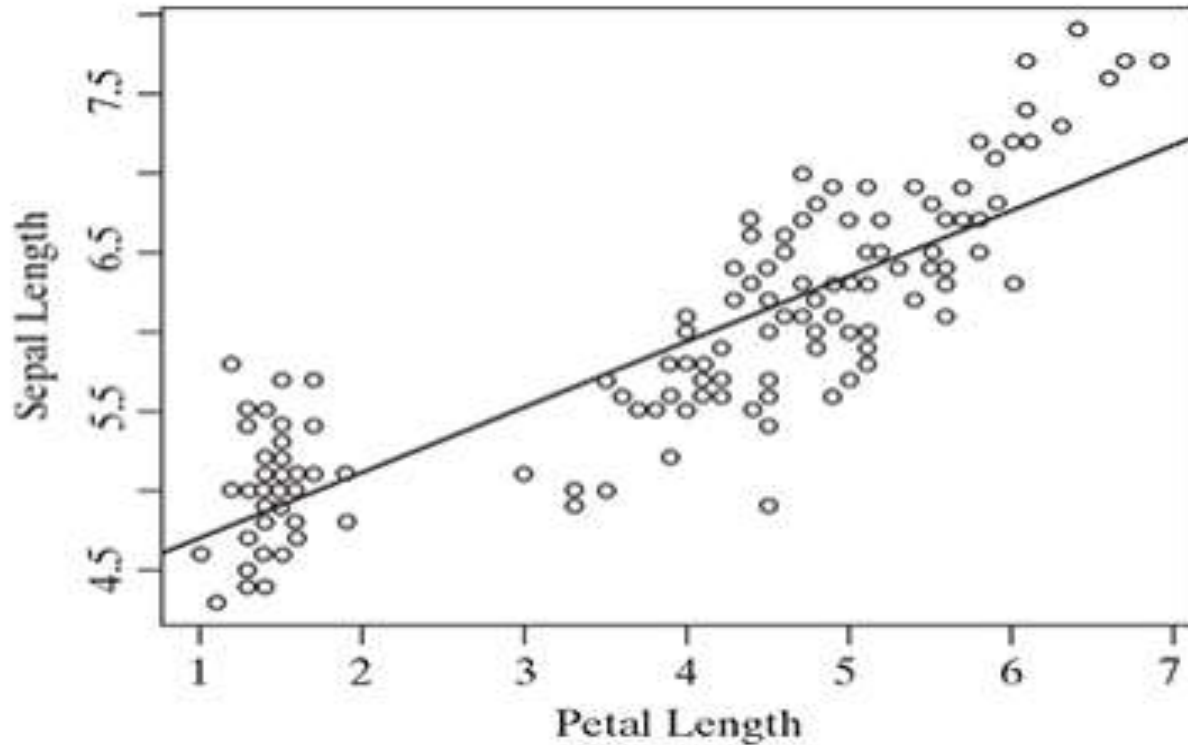
Linear Regression Model (Figure 1.6):

- Equation: $y=mx+b$.
- Example: Sales prediction based on investment.

Applications of Regression:

- Demand forecasting, sales prediction, price prediction, weather forecast, skill demand forecast.

petal length is a predictor variable which, when fitted in the simple linear regression model, helps in predicting the value of the target variable sepal length.



Unsupervised learning

- Unsupervised learning doesn't use labeled training data or make predictions. Its goal is to find natural patterns in a dataset. Often called a descriptive model, unsupervised learning involves discovering patterns or knowledge.
- An important application is customer segmentation. Clustering, the primary unsupervised learning method, groups similar objects. Objects in the same cluster are similar, and those in different clusters are dissimilar.

Clustering

- Clustering aims to reveal intrinsic groupings in unlabelled data, as seen in Figure. Various similarity measures, like distance, can be used, with items close in distance belonging to the same cluster.

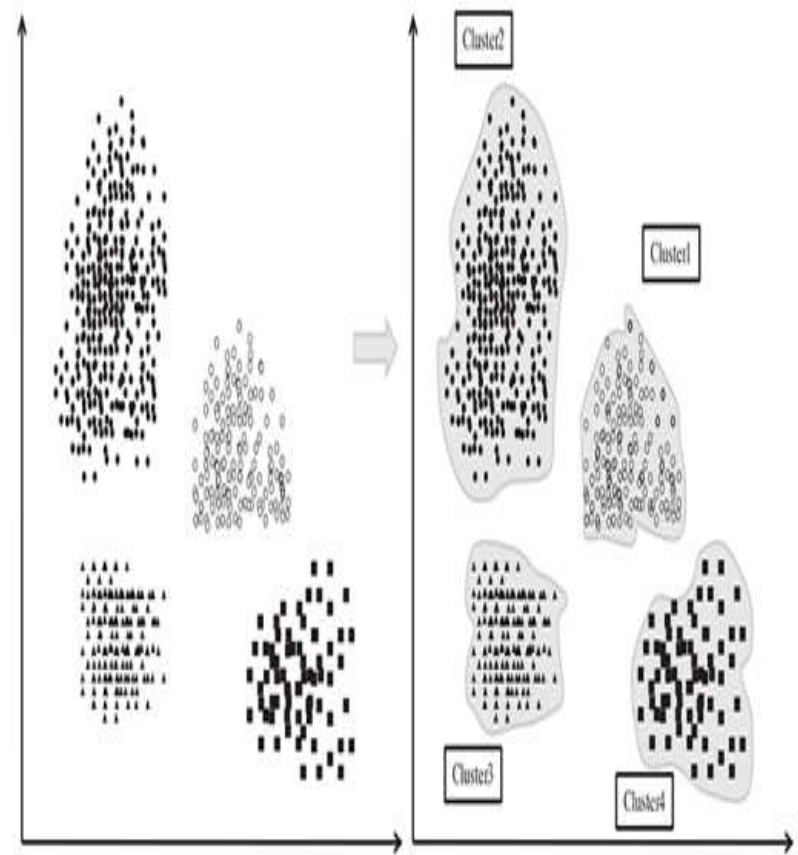


FIG. 1.7 Distance-based clustering

Figure shows the high-level process of clustering.

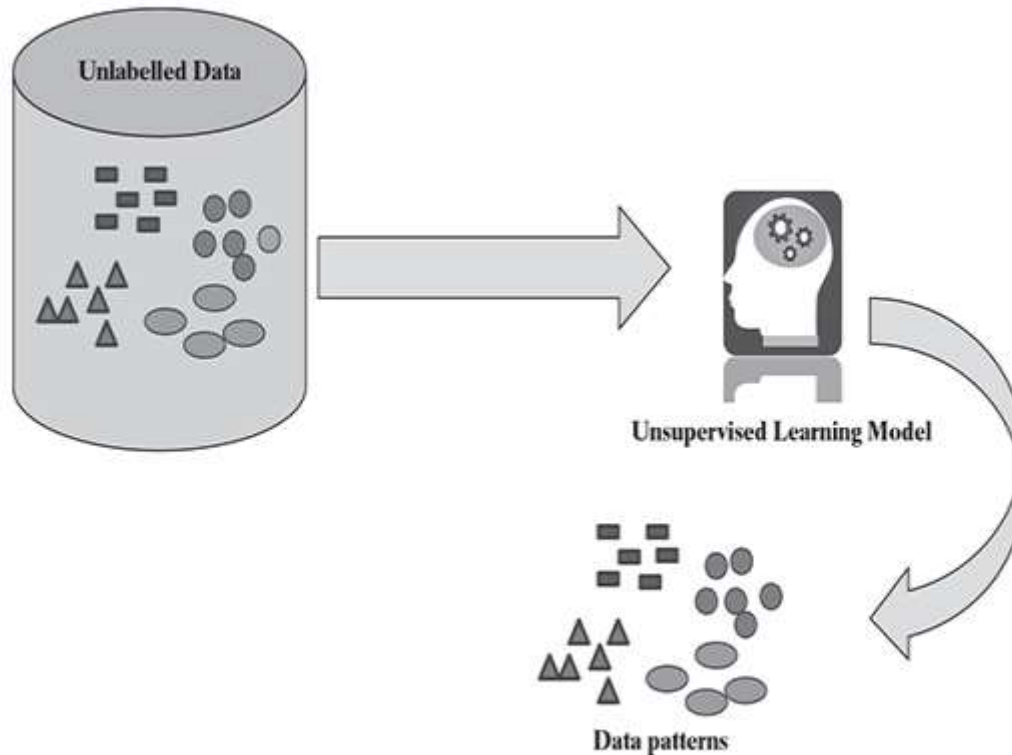


FIG. 1.8 Unsupervised learning

Association analysis

- In addition to clustering, another form of unsupervised learning is association analysis. This method identifies relationships between data elements.
- For example, in market basket analysis (Figure 1.9), past transaction data may reveal a strong association between customers purchasing item A also buying item B or item C. Association analysis, like market basket analysis, is crucial for sales strategies, aiding in identifying patterns to enhance the sales pipeline. Other key applications include recommender systems.

TransID	Items Bought
1	{Butter, Bread}
2	{Diaper, Bread, Milk, Beer}
3	{Milk, Chicken, Beer, Diaper}
4	{Bread, Diaper, Chicken, Beer}
5	{Diaper, Beer, Cookies, Ice cream}
...	...

Market Basket transactions
Frequent itemsets → {Diaper, Beer}
Possible association: Diaper → Beer

FIG. 1.9 Market basket analysis

Reinforcement learning

- Babies learn to walk by observing others and gradually practicing. They notice that legs are used one at a time to take a step. While walking, they may encounter obstacles, sometimes falling and other times navigating smoothly. When they overcome obstacles, they receive positive reinforcement like claps or chocolates from their parents. Learning from mistakes, babies eventually walk with ease.
- Similarly, machines learn autonomously, illustrated by the analogy of a child learning to walk. The task is walking, the child is the agent, and the environment has obstacles. The machine seeks to improve task performance, receiving rewards for successfully completing sub-tasks and none for errors. This process, known as reinforcement learning, continues until the machine can execute the entire task. Figure depicts the high-level process of reinforcement learning.

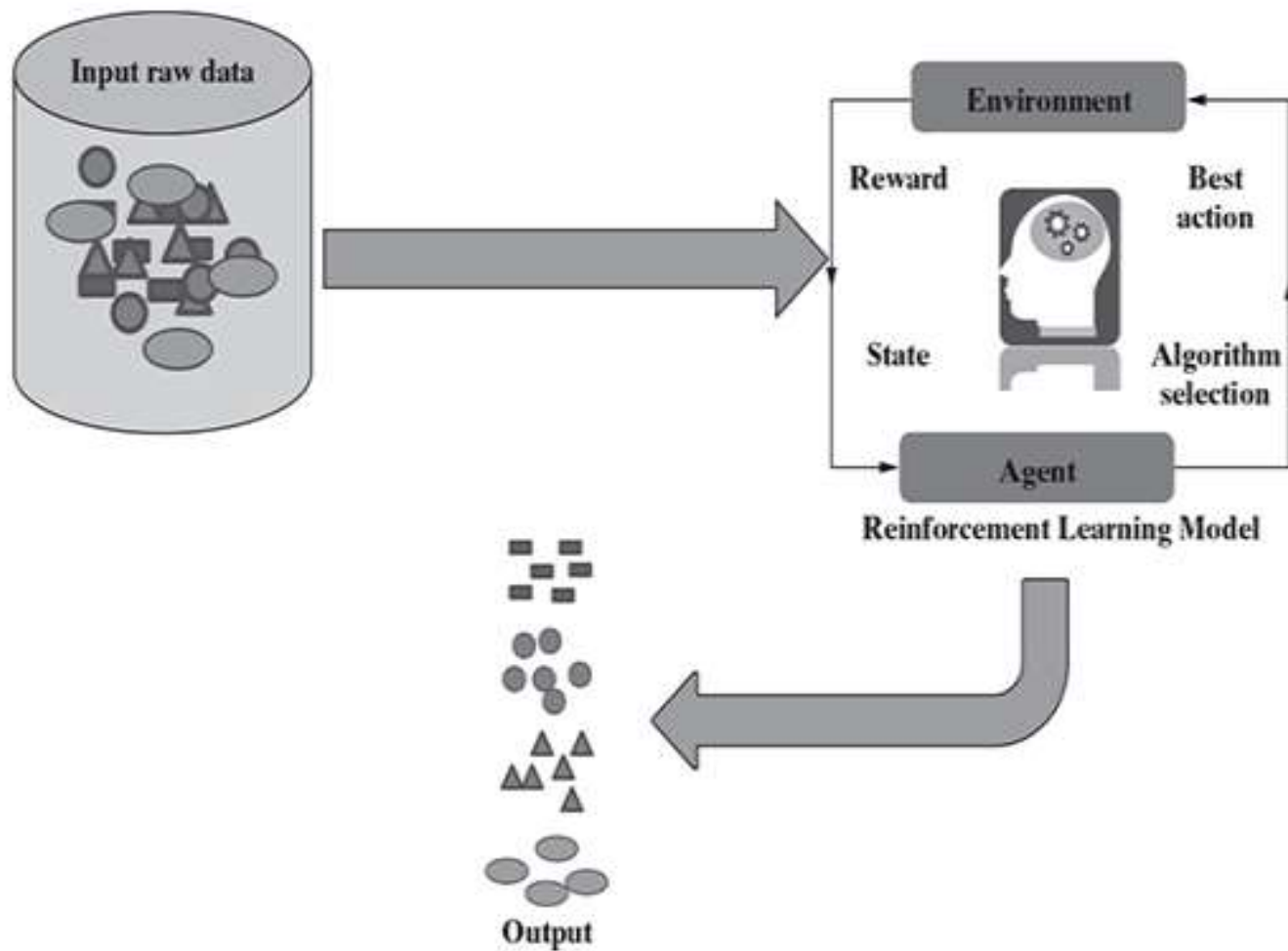


FIG. 1.10 Reinforcement learning

SUPERVISED	UNSUPERVISED	REINFORCEMENT
<p>This type of learning is used when you know how to classify a given data, or in other words classes or labels are available.</p>	<p>This type of learning is used when there is no idea about the class or label of a particular data. The model has to find pattern in the data.</p>	<p>This type of learning is used when there is no idea about the class or label of a particular data. The model has to do the classification – it will get rewarded if the classification is correct, else get punished.</p>
<p>Labelled training data is needed. Model is built based on training data.</p>	<p>Any unknown and unlabelled data set is given to the model as input and records are grouped.</p>	<p>The model learns and updates itself through reward/punishment.</p>
<p>The model performance can be evaluated based on how many misclassifications have been done based on a comparison between predicted and actual values.</p>	<p>Difficult to measure whether the model did something useful or interesting. Homogeneity of records grouped together is the only measure.</p>	<p>Model is evaluated by means of the reward function after it had some time to learn.</p>
<p>There are two types of supervised learning problems – classification and regression.</p>	<p>There are two types of unsupervised learning problems – clustering and association.</p>	<p>No such types.</p>
<p>Simplest one to understand.</p>	<p>More difficult to understand and implement than supervised learning.</p>	<p>Most complex to understand and apply.</p>

Standard algorithms include

- Naïve Bayes
- k -nearest neighbour (kNN)
- Decision tree
- Linear regression
- Logistic regression
- Support Vector Machine (SVM), etc.

Standard algorithms are

- k -means
- Principal Component Analysis (PCA)
- Self-organizing map (SOM)
- Apriori algorithm
- DBSCAN etc.

Standard algorithms are

- Q-learning
- Sarsa

Practical applications include

- Handwriting recognition
- Stock market prediction
- Disease prediction
- Fraud detection, etc.

Practical applications include

- Market basket analysis
- Recommender systems
- Customer segmentation, etc.

Practical applications include

- Self-driving cars
- Intelligent robots
- AlphaGo Zero (the latest version of DeepMind's AI system playing Go)

PROBLEMS UNSUITABLE FOR MACHINE LEARNING

- **Tasks Where Humans Excel:**

- Avoid applying machine learning to tasks where humans are highly effective or frequent human intervention is essential. For instance, complex tasks like air traffic control demand intense human involvement.

- **Simple Rule-Based Tasks:**

- Tasks that can be easily handled with traditional programming paradigms, such as simple rule-driven or formula-based applications (e.g., price calculators, dispute tracking), do not necessitate the use of machine learning techniques.

- **Optimized Business Processes:**

- Machine learning is justified only when there are inefficiencies in the business process. If a task is already optimized, introducing machine learning may not yield a significant return on investment.

- **Insufficient Training Data:**

- Machine learning is ineffective when training data is insufficient. Small training datasets amplify the negative impact of bad data, leading to poor predictions or recommendations. A sizeable training dataset is crucial for quality outcomes.

APPLICATIONS OF MACHINE LEARNING

➤ **Banking and Finance:**

- Machine learning is crucial in banking to detect and prevent real-time fraudulent transactions.
- It contributes to customer retention by addressing issues, using both descriptive and predictive learning to identify problem areas and vulnerable customers, thereby reducing customer churn.

➤ **Insurance:**

- In the insurance sector, machine learning is extensively utilized for risk prediction during new customer onboarding and claims management.
- It predicts the risk profile of new customers based on past information, influencing the generated quote.
- During claims settlement, machine learning analyzes historical claims data and adjustor notes to predict potential fraudulent claims, considering similar customer information in the modeling process.

➤ **Healthcare:**

- Wearable device data is harnessed for real-time health condition prediction through machine learning.
- Individuals receive alerts for preventive action, and in critical situations, nearby doctors or healthcare providers can be notified.
- For instance, machine learning algorithms analyze streaming data from wearables to predict health risks for an elderly person during a morning walk, triggering immediate alerts.
- Furthermore, machine learning and computer vision play a vital role in disease diagnosis from medical imaging.

STATE-OF-THE-ART LANGUAGES/TOOLS IN MACHINE LEARNING

- **Python:**
 - Open-source, widely adopted by the machine learning community.
 - Strong libraries like NumPy, SciPy, and matplotlib for advanced mathematical functionalities, algorithms, and numerical plotting.
 - Machine learning library: scikit-learn, with various classification, regression, and clustering algorithms.
- **R:**
 - Open-source language for statistical computing and data analysis.
 - Popular in academia, especially among statisticians and data miners.
 - Comprehensive set of libraries, including plyr/dplyr, caret, RJava, tm, and ggplot2 for different machine learning stages.
 - Development of interactive web applications and dashboards facilitated by packages like Shiny and R Markdown.

- **Matlab:**
 - Licensed commercial software with robust support for numerical computing.
 - Extensive user base in industry and academia.
 - Comprehensive documentation, in-built statistical functions, and numerous machine learning algorithms.
 - Ability to scale up for large datasets through parallel processing on clusters and cloud.
- **SAS:**
 - Licensed commercial software with a focus on machine learning functionalities.
 - Developed in C, with a suite of components like Base SAS, SAS/INSIGHT, Enterprise Miner, SAS/STAT, etc.
 - First released in 1976, offering strong support for data mining and statistical analysis.

- **Other Languages/Tools:**
 - **SPSS (Statistical Package for the Social Sciences):** Owned by IBM, supports specialized data mining and statistical analysis.
 - **Julia:** Open-source, released in 2012, gaining attention for numerical analysis and computational science. Combines features from MATLAB, Python, R, and other languages, known for high-performance machine learning algorithms.

ISSUES IN MACHINE LEARNING

- **Diversity in Usage and Regulation:**
 - Machine learning is an evolving field with varying levels of research and utilization across countries.
 - Legal and cultural differences, emotional maturity, and regulations differ significantly, impacting the use and issues related to machine learning.
- **Privacy Concerns:**
 - The major concern revolves around privacy and potential breaches.
 - Machine learning analyzes both past and current data, often revealing private information.
 - Individual preferences regarding information sharing vary, and inadvertent use of such data, especially for targeted marketing, can upset individuals.
 - Privacy breaches may lead to legal actions in certain countries.

- **Unintended Consequences:**

- Actions based on machine learning outcomes may lead to adverse reactions.
- Example: Knowledge discovery before an election campaign revealing demographic factors may result in a campaign message that upsets voters, leading to adverse outcomes.

- **Need for Human Judgment:**

- Before implementing machine learning outcomes, exercising proper human judgment is crucial.
- Decisions should consider potential benefits and avoid adverse impacts.
- Human oversight is essential to ensure responsible and ethical use of machine learning results.

UNIT-2

PREPARING TO MODEL

1. Introduction to Machine Learning:

- The chapter provides an overview of the evolution of machine learning, starting with Alan Turing's proposition of machines capable of learning.
- Key Innovations and Figures: It highlights contributions from computer scientists like Arthur Samuel, Frank Rosenblatt, and Geoffrey Hinton, who shaped concepts such as Neural Networks and Deep Learning.
- Notable Applications: IBM's Deep Blue defeating chess champion Gary Kasparov and Google's innovations like Google Brain and AlphaGo showcase significant milestones in applying machine learning.
- Types of Machine Learning: The paragraph discusses supervised, unsupervised, and reinforcement learning, drawing parallels to human learning processes.
- Applications in Various Sectors: It mentions real-world applications in banking, insurance, and healthcare, emphasizing the role of machine learning in areas like fraud detection and disease prediction.

2. ML Activities:

1. Data Preparation:

- Understand the type and quality of the input data.
- Explore relationships among data elements to identify patterns.
- Identify and address any potential issues in the data, such as missing values.
- Perform necessary data remediation, like imputing missing values.
- Apply pre-processing steps to prepare the data for modeling.

2. Model Training:

- Divide the prepared data into training and test sets (applicable for supervised learning).
- Explore different models or learning algorithms for selection.
- Train the model using the training data for supervised learning or apply the chosen unsupervised model directly to the input data.

3. Model Evaluation:

- After training or applying the model, evaluate its performance.
- Take specific actions to improve the model's performance if necessary, based on available options.

4. Machine Learning Process Overview:

- Figure 2.1 illustrates the four-step process of machine learning, depicting data preparation, model training, and evaluation stages.

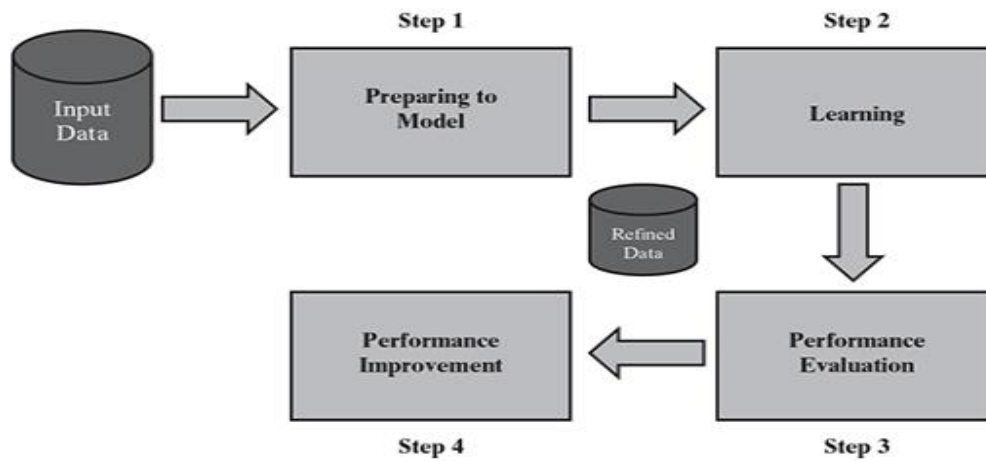


FIG. 2.1 Detailed process of machine learning

contains a summary of steps and activities involved:

Table 2.1 *Activities in Machine Learning*

Step #	Step Name	Activities Involved
Step 1	Preparing to Model	<ul style="list-style-type: none"> • Understand the type of data in the given input data set • Explore the data to understand data quality • Explore the relationships amongst the data elements, e.g. inter-feature relationship • Find potential issues in data • Remediate data, if needed • Apply following pre-processing steps, as necessary: <ul style="list-style-type: none"> ✓ Dimensionality reduction ✓ Feature subset selection
Step 2	Learning	<ul style="list-style-type: none"> • Data partitioning/holdout • Model selection • Cross-validation
Step 3	Performance evaluation	<ul style="list-style-type: none"> • Examine the model performance, e.g. confusion matrix in case of classification • Visualize performance trade-offs using ROC curves
Step 4	Performance improvement	<ul style="list-style-type: none"> • Tuning the model • Ensembling • Bagging • Boosting

3. BASIC TYPES OF DATA IN MACHINE LEARNING

In machine learning, data sets consist of related information or records. Each record contains specific information about a subject, like a student in a student data set. These records are organized into rows, and each row represents a point in a multi-dimensional data space. Each data set has attributes, which provide information on specific characteristics. Attributes are also known as features, variables, dimensions, or fields.

Student data set:

Roll Number	Name	Gender	Age
129/011	Mihir Karmarkar	M	14
129/012	Geeta Iyer	F	15
129/013	Chanda Bose	F	14
129/014	Sreenu Subramanian	M	14
129/015	Pallav Gupta	M	16
129/016	Gajanan Sharma	M	15

Student performance data set:

Roll Number	Maths	Science	Percentage
129/011	89	45	89.33%
129/012	89	47	90.67%
129/013	68	29	64.67%
129/014	83	38	80.67%
129/015	57	23	53.33%
129/016	78	35	75.33%

FIG. 2.2 Examples of data set

Roll Number	Name	Gender	Age
129/011	Mihir Karmarkar	M	14
129/012	Geeta Iyer	F	15

FIG. 2.3 Data set records and attributes

There are two main types of data in machine learning:

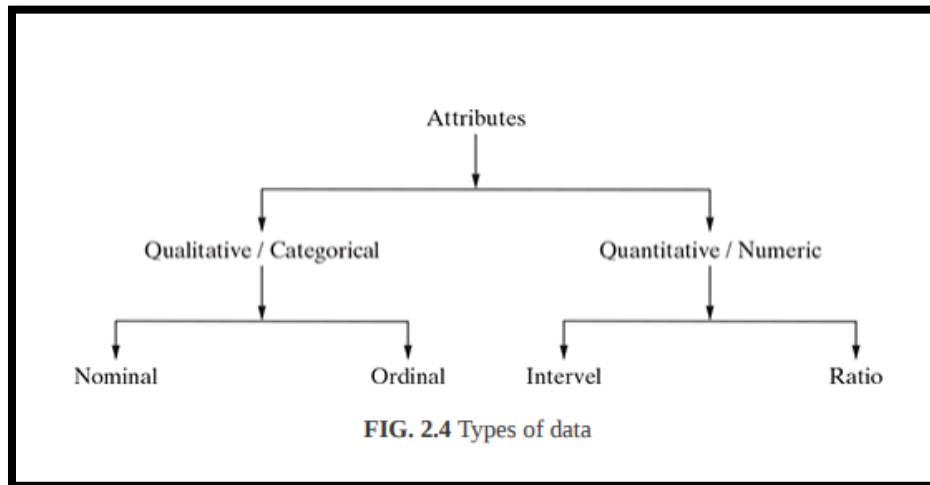


FIG. 2.4 Types of data

Qualitative data: This type of data describes qualities or characteristics and cannot be measured. It includes:

- Nominal data: Named values without numeric significance. Examples include blood group, nationality, and gender.
- Ordinal data: Named values with a natural order. Examples include customer satisfaction ratings, grades, and hardness of metal.

Quantitative data: This type of data represents quantities that can be measured. It includes:

- Interval data: Numeric data where the order and the exact difference between values are known, but there is no true zero. Examples include temperature in Celsius and time.
- Ratio data: Numeric data with a true zero and where mathematical operations like addition, subtraction, multiplication, and division are meaningful. Examples include height, weight, age, and salary.

Attributes can also be categorized as discrete or continuous:

Discrete attributes: Can assume a finite or countably infinite number of values. Examples include roll number, street number, and count.

Binary attributes are a special case of discrete attributes with only two possible values, such as male/female or yes/no.

Continuous attributes: Can take any real number value. Examples include length, height, weight, and price.

This summary outlines the basic types of data encountered in machine learning problems.

4. EXPLORING STRUCTURE OF DATA

In machine learning, data sets typically contain two basic types of data: numeric and categorical. Understanding the nature of these attributes is crucial for exploring and analyzing the data effectively.

mpg	cylinder	displacement	horsepower	weight	acceleration	model year	origin	car name
18	8	307	130	3504	12	70	1	Chevrolet chevelle malibu
15	8	350	165	3693	11.5	70	1	Buick skylark 320
18	8	318	150	3436	11	70	1	Plymouth satellite
16	8	304	150	3433	12	70	1	Amc rebel sst
17	8	302	140	3449	10.5	70	1	Ford torino
15	8	429	198	4341	10	70	1	Ford galaxie 500
14	8	454	220	4354	9	70	1	Chevrolet impala
14	8	440	215	4312	8.5	70	1	Plymouth fury iii
14	8	455	225	4425	10	70	1	Pontiac catalina
15	8	390	190	3850	8.5	70	1	Amc acbassador dpl
15	8	383	170	3563	10	70	1	Dodge challenger se
14	8	340	160	3609	8	70	1	Plymouth 'cuda 340
15	8	400	150	3761	9.5	70	1	Chevrolet monte carlo
14	8	455	225	3086	10	70	1	Buick estate wagon (sw)
24	4	113	95	2372	15	70	3	Toyota corona mark ii
22	6	198	95	2933	15.5	70	1	Plymouth duster
18	6	199	97	2774	15.5	70	1	Amc hornet

FIG. 2.5 Auto MPG data set

Numeric attributes: Attributes such as 'mpg', 'cylinders', 'displacement', 'horsepower', 'weight', 'acceleration', and 'model year' in the Auto MPG data set are all numeric. Among these, 'cylinders', 'model year', and 'origin' are discrete, meaning they can only assume a finite number of values. The rest of the numeric attributes can take any real value.

Categorical attributes: The attribute 'car name' in the Auto MPG data set is categorical, specifically nominal. This means it represents named values without numerical significance.

//It's worth noting that even though attributes like 'cylinders' or 'origin' have a limited number of possible values, they are treated as numeric for the purpose of exploring the data in this section. However, one may choose to treat them as categorical attributes based on their preference or specific analysis requirements.//

5. EXPLORING NUMERICAL DATA

Understanding Central Tendency:

- i. Mean: It's the sum of all data values divided by the count of data elements.
Mean, by definition, is a sum of all data values divided by the count of data elements. For example, mean of a set of observations – 21, 89, 34, 67, and 96 is calculated as below.

$$\text{Mean} = \frac{21 + 89 + 34 + 67 + 96}{5} = 61.4$$

If the above set of numbers represents marks of 5 students in a class, the mean marks, or the falling in the middle of the range is 61.4.

- ii. Median: It's the middle value in an ordered list of data elements.
- iii. The reason for reviewing both mean and median is their different sensitivities to outliers.
- iv. Mean is influenced by outliers, while median is not as much affected.
- v. Significant deviation between mean and median suggests potential outliers.

	mpg	cylinders	displacement	horsepower	weight	acceleration	model year	origin
Median	23	4	148.5	?	2804	15.5	76	1
Mean	23.51	5.455	193.4	?	2970	15.57	76.01	1.573
Deviation	2.17	26.67%	23.22%		5.59%	0.45%	0.01%	36.43%
	Low	High	High		Low	Low	Low	High

FIG. 2.6 Mean vs. Median for Auto MPG

There is some problem in the values of the attribute 'horsepower' because of which the mean and median calculation is not possible.

Exploring Data Spread:

- i. Measuring Data Dispersion:
 - Variance and standard deviation quantify how spread out the data values are.
 - Larger variance or standard deviation indicates more dispersion.

Consider the data values of two attributes

1. Attribute 1 values : 44, 46, 48, 45, and 47

2. Attribute 2 values : 34, 46, 59, 39, and 52

Both the set of values have a mean and median of 46. However, the first set of values that is of attribute 1 is more concentrated or clustered around the mean/median value whereas the second set of values of attribute 2 is quite spread out or dispersed. To measure the extent of dispersion of a data, or to find out how much the different values of a data are spread out, the variance of the data is measured. The variance of a data is measured using the formula given below:

$$\text{Variance}_{(x)} = \frac{\sum_{i=1}^n x_i^2}{n} - \left(\frac{\sum_{i=1}^n x_i}{n} \right)^2$$

where x is the variable or attribute whose variance is to be measured and n is the number of observations or values of variable x. Standard deviation of a data is measured as follows:

$$\text{Standard deviation } (x) = \sqrt{\text{Variance } (x)}$$

$$\begin{aligned} \text{Variance} &= \frac{\sum_{i=1}^n x_i^2}{n} - \left(\frac{\sum_{i=1}^n x_i}{n} \right)^2 \\ &= \frac{44^2 + 46^2 + 48^2 + 45^2 + 47^2}{5} - \left(\frac{44 + 46 + 48 + 45 + 47}{5} \right)^2 \\ &= \frac{1936 + 2116 + 2304 + 2025 + 2209}{5} - \left(\frac{230}{5} \right)^2 = \frac{10590}{5} - (46)^2 = 2 \end{aligned}$$

For attribute 2,

$$\begin{aligned} \text{Variance} &= \frac{\sum_{i=1}^n x_i^2}{n} - \left(\frac{\sum_{i=1}^n x_i}{n} \right)^2 \\ &= \frac{34^2 + 46^2 + 59^2 + 39^2 + 52^2}{5} - \left(\frac{34 + 46 + 59 + 39 + 52}{5} \right)^2 \\ &= \frac{1156 + 2116 + 3481 + 1521 + 2704}{5} - \left(\frac{230}{5} \right)^2 = \frac{10978}{5} - (46)^2 = 79.6 \end{aligned}$$

ii. Measuring Data Value Position:

When data values are sorted from smallest to largest:

- Median (Q2) divides the data set into two halves.
- First quartile (Q1) marks the midpoint of the first half.
- Third quartile (Q3) marks the midpoint of the second half.
- The smallest and largest values complete the set.
- This sequence gives five key values: minimum, Q1, Q2 (median), Q3, and maximum.

	cylinders	displacement	origin
Minimum	3	68	1
Q1	4	104.2	1
Median	4	148.5	1
Q3	8	262	2
Maximum	8	455	3

FIG. 2.8 Attribute value drill-down for Auto MPG

iii. Visualizing Data:

- Box plot is an effective tool to visualize numerical data, especially for identifying outliers.
- This exploration helps understand the distribution and characteristics of numerical attributes in the data set.

6. PLOTTING AND EXPLORING NUMERICAL DATA

Box Plots:

- Box plots offer a quick view of a dataset's key statistics.
- They represent the minimum, first quartile (Q1), median (Q2), third quartile (Q3), and maximum.
- The box itself spans from Q1 to Q3, indicating the inter-quartile range (IQR).
- The median is shown inside the box.
- The lower whisker extends to 1.5 times the IQR below Q1, but stops at the lowest data value within that range.

- Similarly, the upper whisker extends to 1.5 times the IQR above Q3, stopping at the highest data value within that range.
- Values beyond the whiskers are considered outliers.
- There are different types of box plots, with the one explained being the Tukey box plot.

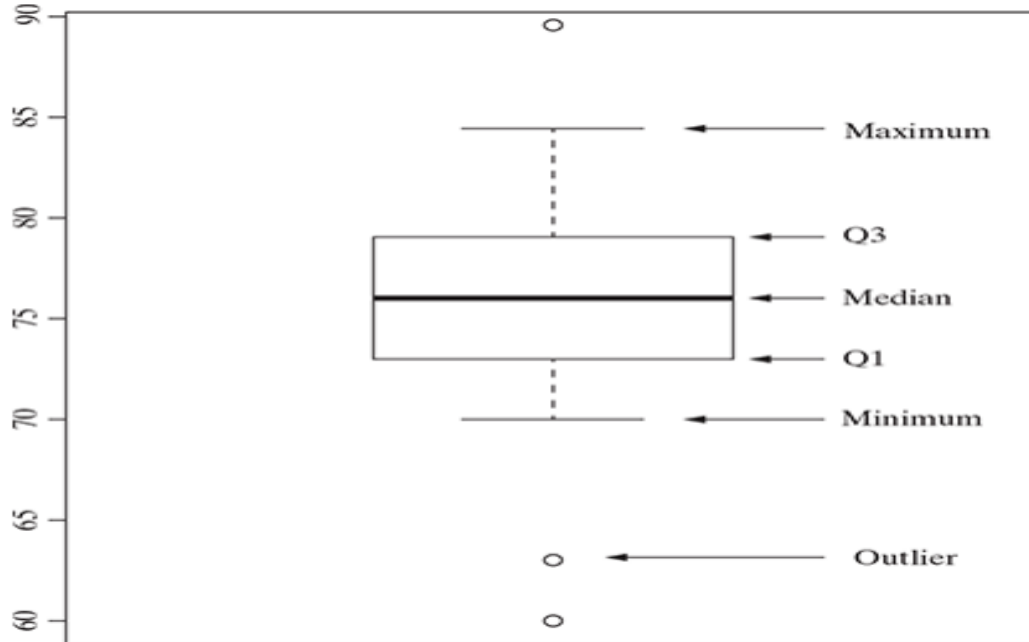


FIG. 2.9 Box plot

Analyzing Box Plots for Specific Attributes:

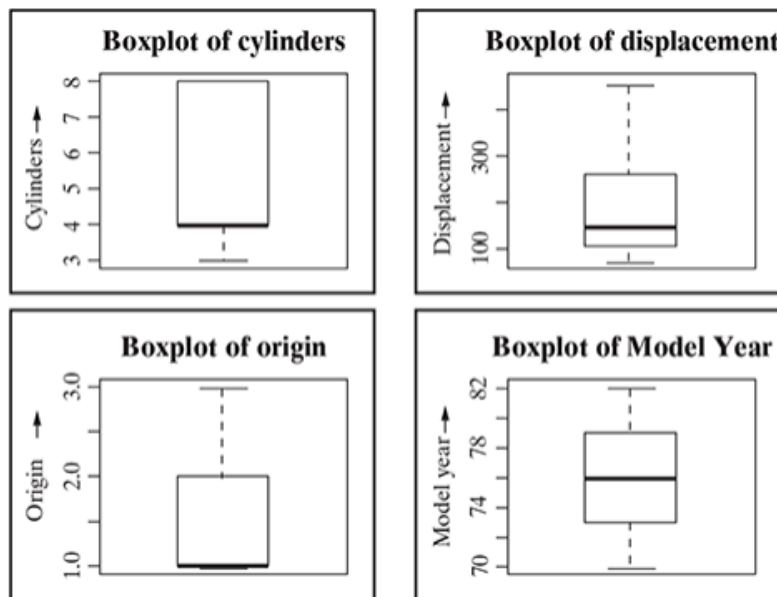


FIG. 2.10 Box plot of Auto MPG attributes

- For attributes like 'cylinders' and 'origin,' the shape of the box plot might look odd due to discrete data values.
- Understanding the frequency distribution of these attributes helps interpret the box plot.
- For 'cylinders,' where values range from 3 to 8, high frequency values cluster around certain numbers, affecting the appearance of the plot.
- Similarly, for 'origin,' where values range from 1 to 3, the distribution of frequencies affects the plot's appearance.

- For attributes like 'displacement' and 'model year,' abnormalities in the box plot might indicate specific characteristics of the data.
- Analyzing quartiles and data distribution provides insights into these abnormalities.

Histograms:

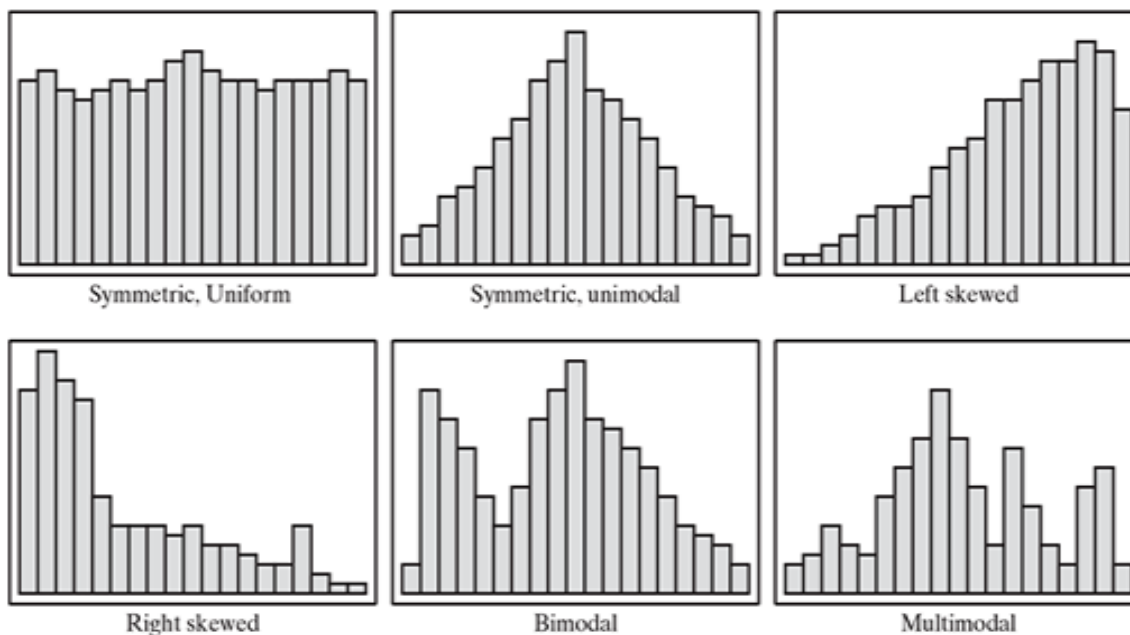


FIG. 2.11 General Histogram shapes

- Histograms visualize data distribution by grouping values into intervals or bins.
- Unlike box plots, histograms show the frequency of data elements within each interval.
- Histograms can have different shapes, indicating the distribution's characteristics, such as skewness.
- Examining histograms provides insights into the distribution of data values across intervals.
- Each bar in a histogram represents the count of data elements within a specific interval.
- The height of each bar reflects the frequency of data elements within that interval.
- A uniform histogram, like for 'model year,' suggests that all values are equally likely to occur.

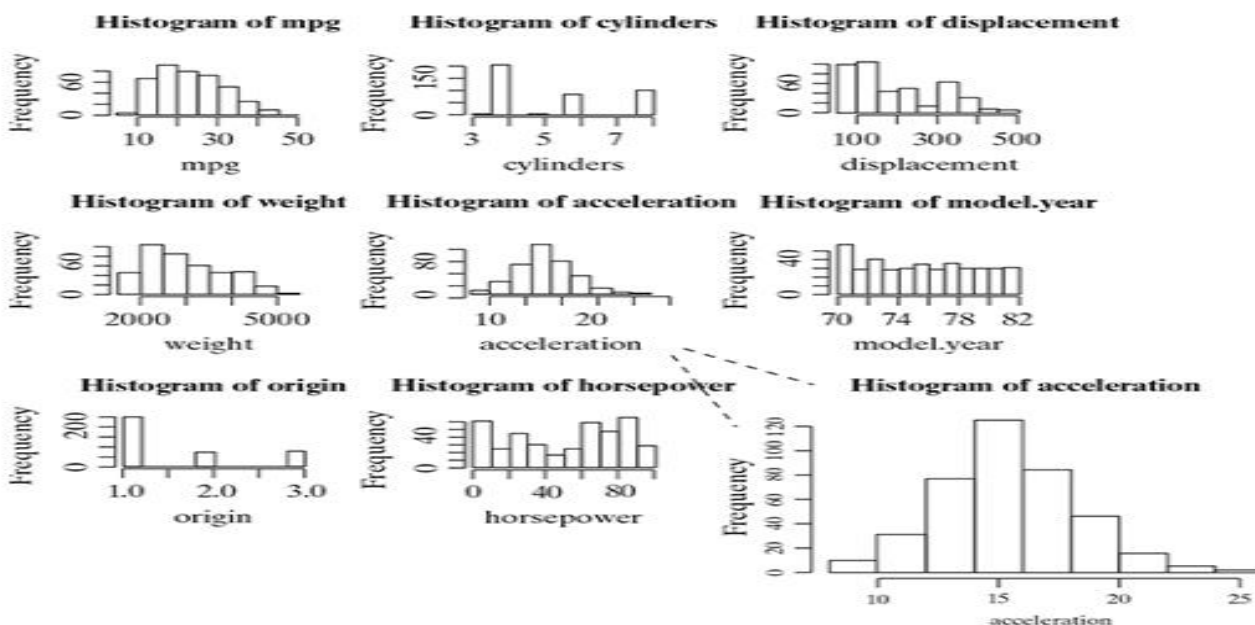


FIG. 2.12 Histogram Auto MPG attributes

7. EXPLORING CATEGORICAL DATA

- Unlike numeric data, exploring categorical data offers fewer options.
- In the Auto MPG dataset, the 'car name' attribute is categorical, and 'cylinders' can also be considered categorical.
- Initial exploration involves noting the number of unique values for each attribute.
- For 'car name', listing unique names gives insight into the variety of car models.
- Similarly, for 'cylinders', noting unique values provides information on the different cylinder counts.
- Further exploration involves creating tables to summarize the categories and their respective counts.
- Tables are created for both 'car name' and 'cylinders' attributes to display the count of data elements in each category.
- Understanding the proportion or percentage of data elements in each category provides additional insight.
- Proportion tables for both attributes show the percentage of data elements in each category.
- Mode, a statistical measure applicable to categorical attributes, represents the category with the highest frequency.
- Finding the mode for attributes like 'car name' and 'cylinders' helps identify the most common category.
- A unimodal attribute has a single mode, bimodal has two modes, and multimodal has multiple modes.

8. Exploring Relationship Between Variables:

Scatter Plot :

- A scatter plot visualizes bivariate relationships between two variables.
- It plots points based on attribute values, with one attribute on the x-axis and the other on the y-axis.
- For example, in the Auto MPG dataset, we analyze the relationship between 'displacement' and 'mpg'.
- By observing the scatter plot, we notice a trend where 'mpg' decreases as 'displacement' increases.
- Correlation between variables can further quantify this relationship.

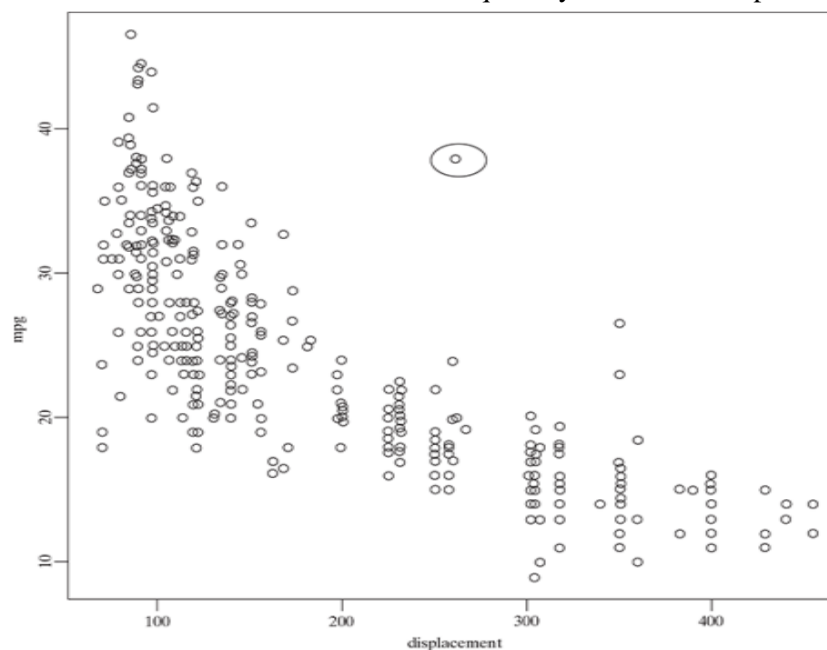


FIG. 2.13 Scatter plot of 'displacement' and 'mpg'

Pairwise Scatter Plot :

- Pairwise scatter plots capture relationships among multiple attributes.
- In the Auto MPG dataset, scatter plots among 'mpg', 'displacement', 'horsepower', 'weight', and 'acceleration' reveal significant relationships.
- Some attribute pairs exhibit strong relationships, while others like 'weight' and 'acceleration' show weaker connections.

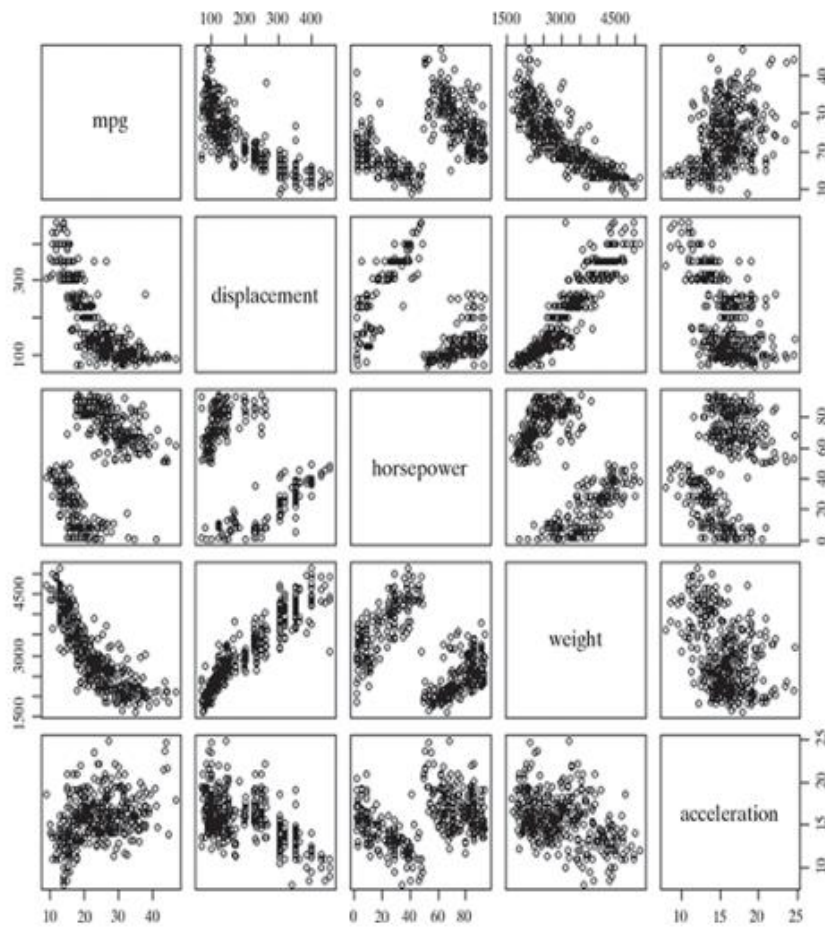


FIG. 2.14 Pair wise scatter plot between different attributes of Auto MPG

Two-Way Cross-Tabulations :

- Cross-tabs provide a summarized view of the relationship between two categorical attributes.
- They help understand how data values of one attribute change concerning another.
- For instance, considering attributes like 'cylinders', 'model.year', and 'origin' in the Auto MPG dataset:
- Cross-tabs reveal the distribution of vehicles per year across different regions.
- They show the count of vehicles per region with varying cylinder counts.
- Further analyses can involve summarizing data values, such as grouping cars by cylinder count (e.g., 4 or fewer vs. more than 4) in each region or year.

'Model year' vs. 'origin'

Table 2.8 Cross-tab for 'Model year' vs. 'Origin'

Origin \ Model Year	70	71	72	73	74	75	76	77	78	79	80	81	82
1	22	20	18	29	15	20	22	18	22	23	7	13	20
2	5	4	5	7	6	6	8	4	6	4	9	4	2
3	2	4	5	4	6	4	4	6	8	2	13	12	9

'Cylinders' vs. 'Origin'

Table 2.9 Cross-tab for 'Cylinders' vs. 'Origin'

Cylinders \ Origin	1	2	3
3	0	0	4
4	72	63	69
5	0	3	0
6	74	4	6
8	103	0	0

Table 2.10 Cross-tab for 'Cylinders' vs. 'Model year'

Cylinders \ Model Year	70	71	72	73	74	75	76	77	78	79	80	81	82
3	0	0	1	1	0	0	0	1	0	0	1	0	0
4	7	13	14	11	15	12	15	14	17	12	25	21	28
5	0	0	0	0	0	0	0	0	1	1	1	0	0
6	4	8	0	8	7	12	10	5	12	6	2	7	3
8	18	7	13	20	5	6	9	8	6	10	0	1	0

9. DATA QUALITY AND REMEDIATION

Data Quality

- The success of machine learning heavily relies on data quality. Flawless data enhances prediction accuracy, especially in supervised learning. However, data imperfections like missing values and outliers are common. These issues stem from factors such as:
 1. Incorrect Sample Set Selection: Choosing improper sample sets can distort the data's representation. For instance, using sales transaction data from a holiday period to predict future sales can yield inaccurate results. Similarly, predicting poll results requires a diverse sample set representing various demographics.
 2. Errors in Data Collection: Manual data collection often leads to errors like outliers and missing values due to inaccuracies in recording or non-response in surveys.

Data Remediation

To address data quality issues:

- Handling Outliers: Outliers, exceptionally high or low data points, can skew models. Solutions include removing outliers, imputing values using statistical measures, or capping extreme values.
- Handling Missing Values: Missing data can be dealt with by eliminating records with missing values if the proportion is tolerable. Otherwise, imputation techniques such as mean, median, or mode substitution can be employed.

10. Data Pre-processing

Dimensionality Reduction

- Until the late 1990s, data sets used in machine learning typically had few attributes or features, usually in the tens.
- However, with the emergence of computational biology projects like genome sequencing and the rise of social networking, high-dimensional data sets with 20,000 or more features became common.
- High-dimensional data sets require significant computational resources and time. Not all features are useful; some can degrade the performance of machine learning algorithms.
- Dimensionality reduction techniques aim to reduce the number of features, thereby reducing irrelevance and redundancy. This makes models easier to understand and improves algorithm performance.
- Principal Component Analysis (PCA) is a widely used technique for dimensionality reduction. It converts correlated variables into uncorrelated principal components while capturing maximum data variability.
- Singular Value Decomposition (SVD) is another commonly used technique for dimensionality reduction.
- More details on these concepts are covered in Chapter 4.

Feature Subset Selection

- Feature subset selection, also known as feature selection, aims to identify the optimal subset of features from the entire feature set. This reduces computational costs without significantly impacting learning accuracy.
- It eliminates irrelevant or redundant features from the final feature set.
- An irrelevant feature contributes little to classifying or grouping data instances and is therefore removed.
- Redundant features contribute similar information to other features and can be eliminated without affecting model accuracy.
- Feature subset selection methods will be discussed in detail in Chapter 4.

UNIT- 3

MODELLING AND EVALUATION

1. Introduction to Machine Learning:

- Machine learning involves applying mathematical and statistical formulations to emulate human learning processes.
- Both human and machine learning aim to build formulations or mappings based on limited observations.
- The basic learning process involves Data Input, Abstraction, and Generalization.

Example of Criminal Detection:

- The scenario involves using photos from a criminal database to spot potential attackers in a campaign gathering.
- Human learning involves scanning photos and matching them with faces, while machine learning employs computational techniques for abstraction and generalization.

Abstraction and Generalization:

- Abstraction involves summarizing raw input data into a structured format called a model.
- Different forms of models exist, such as mathematical equations, graphs, or computational blocks.
- Model selection depends on the problem and data type, with model training summarizing raw input data into an abstracted form.
- Generalization sifts through abstracted knowledge to derive actionable insights from a broad-based set of findings.

Machine Learning Process:

- Machine learning algorithms create cognitive capabilities by building a mathematical formulation or function based on input features.
- Human input, in the form of non-learnable parameters or hyper-parameters, is crucial for the success of machine learning algorithms.
- These points outline the learning process of machines, the role of abstraction and generalization, and the importance of human input in machine learning algorithms.

3.2 Selecting a Model

Introduction to Model Selection:

- The New City Police department aims to analyze factors affecting criminal activities to take proactive measures.
- They need a model to infer how criminal incidents change based on factors like average income, weapon sales, and immigration.

Machine Learning Model Components:

- Input variables (predictors) like income, weapon sales, etc., influence the output variable (number of criminal incidents).
- The relationship is represented as: $Y = f(X) + e$, where 'f' is the target function and 'e' is the error term.

Cost and Loss Functions:

- Cost function measures model performance, like R-squared for regression.

- Loss function is defined on data points, while the cost function considers the entire dataset.

Model Selection Process:

- Three main categories of machine learning: supervised, unsupervised, and reinforcement learning.
- Model selection depends on the problem type and data nature.
- No single model works best for all problems (No Free Lunch theorem).

Understanding Model Selection:

- Models simplify real-world aspects based on assumptions, which may vary based on the situation and data characteristics.
- Data exploration helps in understanding data characteristics and selecting appropriate models.

3.2.1 Predictive Models

Supervised Learning:

- Predictive models focus on predicting values using input data.
- Classification models predict category/class labels (e.g., win/loss in cricket).
- Regression models predict numerical values (e.g., revenue growth).

3.2.2 Descriptive Models

Unsupervised Learning:

- Descriptive models describe or gain insights from data.
- Clustering models group similar data instances.
- Market basket analysis discovers patterns in transactional data.

3.3 Training a Model (for Supervised Learning)

Holdout Method:

- Divides data into training and test sets.
- Uses a subset for model evaluation, commonly 70%-80% for training and 20%-30% for testing.
- May also use a validation set for refining the model.

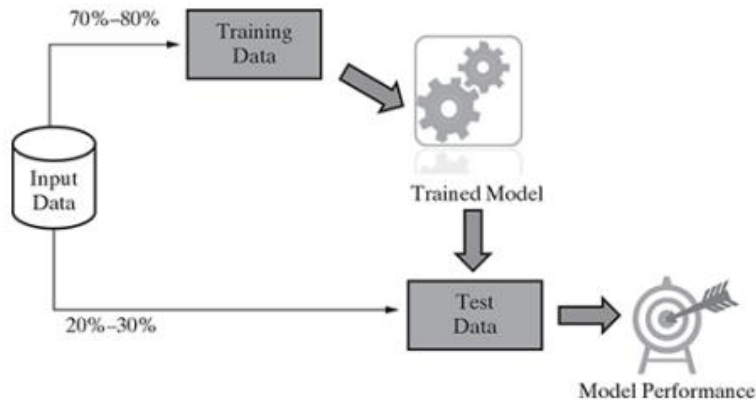


FIG. 3.1 Holdout method

K-fold Cross-validation Method:

- Divides data into 'k' non-overlapping folds for validation.
- Popular approach: 10-fold cross-validation.
- Leave-one-out cross-validation is computationally expensive but uses all data for validation.
- The holdout method, even with stratified random sampling, encounters issues, especially with smaller datasets where proportional division among classes is challenging. To address this, a variant called repeated holdout is used. In repeated holdout, multiple random holdouts are used

to assess model performance, and the average performance is calculated. This process forms the basis of k-fold cross-validation.

- In k-fold cross-validation, the dataset is divided into k distinct partitions or folds. The value of 'k' can vary, but 10-fold cross-validation is the most popular. Here, each fold serves as the test data once, with the rest used for training. This process is repeated 10 times, and the average performance across all folds is reported.
- Leave-one-out cross-validation (LOOCV) is an extreme form of k-fold cross-validation where each data instance is used as a test data point. However, LOOCV is computationally expensive and not commonly used in practice.

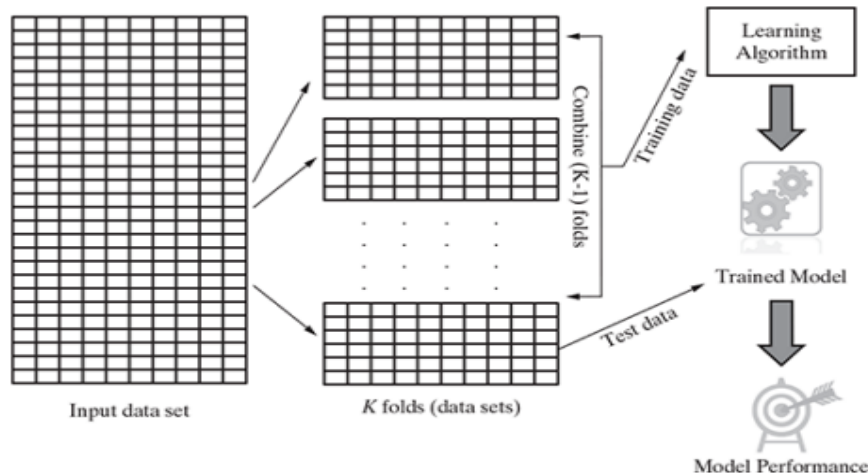


FIG. 3.2 Overall approach for K-fold cross-validation

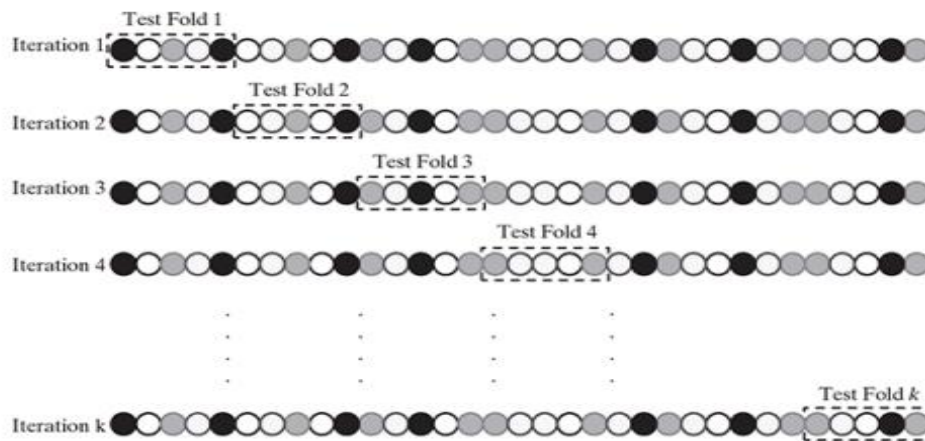


FIG. 3.3 Detailed approach for fold selection

Bootstrap Sampling:

- Bootstrap sampling, also known as bootstrapping, is a widely used method for creating training and test datasets from the original input data. It employs Simple Random Sampling with Replacement (SRSWR), a well-known sampling technique, to draw random samples from the dataset.
- Unlike k-fold cross-validation, where the data is divided into distinct partitions, bootstrap sampling randomly selects data instances from the input dataset. The key feature of bootstrapping is that it allows the same data instance to be picked multiple times, creating training datasets with repeated instances.

- For instance, from an input dataset containing 'n' data instances, bootstrapping can generate one or more training datasets, each consisting of 'n' data instances, some of which may be repeated. This flexibility makes bootstrapping particularly useful for small datasets with limited data instances.

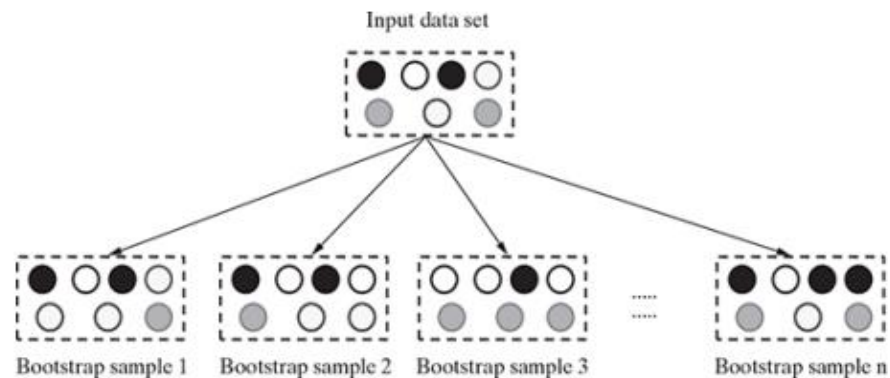


FIG. 3.4 Bootstrap sampling

- Figure 3.4 illustrates the basic approach of bootstrap sampling. This technique is invaluable for making the most of small datasets by generating diverse training samples for robust model training.

CROSS-VALIDATION	BOOTSTRAPPING
<p>It is a special variant of holdout method, called repeated holdout. Hence uses stratified random sampling approach (without replacement). Data set is divided into 'k' random partitions, with each partition containing approximately $\frac{n}{k}$ number of unique data elements, where 'n' is the total number of data elements and 'k' is the total number of folds.</p> <p>The number of possible training/test data samples that can be drawn using this technique is finite.</p>	<p>It uses the technique of Simple Random Sampling with Replacement (SRSWR). So the same data instance may be picked up multiple times in a sample.</p> <p>In this technique, since elements can be repeated in the sample, possible number of training/test data samples is unlimited.</p>

Lazy vs. Eager Learner:

- Eager learners construct a generalized target function during training.
- Lazy learners use training data directly for classification and are computationally faster but slower in classification.
- Examples include Decision Trees and k-Nearest Neighbor.

MODEL REPRESENTATION AND INTERPRETABILITY

In supervised machine learning, the primary objective is to derive a target function that accurately predicts the target variable based on the input variables. However, the challenge lies in ensuring that the model can generalize well to unseen data beyond the training set.

Underfitting:

- Occurs when the target function is too simplistic to capture the complexities of the underlying data.
- Often happens with insufficient training data or overly simple models.
- Results in poor performance both on training and test datasets.
- Remedied by increasing training data, refining feature selection, or using more complex models.

1. using more training data
2. reducing features by effective feature selection

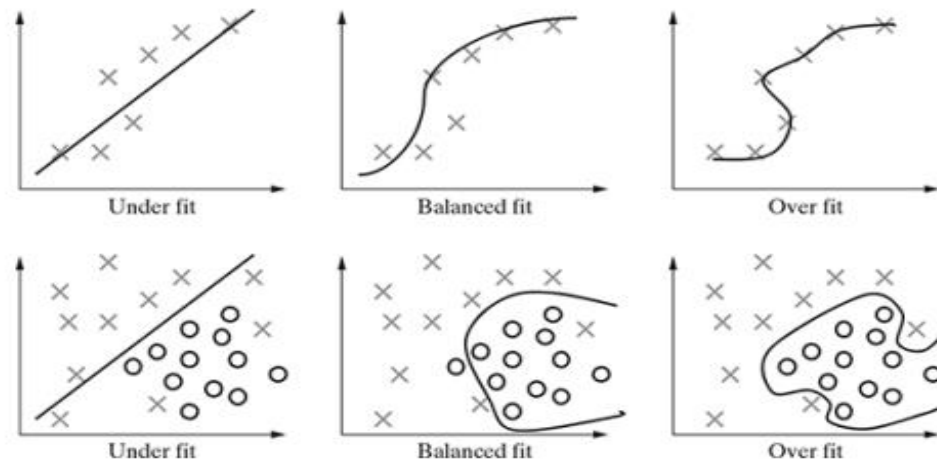


FIG. 3.5 Underfitting and Overfitting of models

Overfitting:

- Arises when the model excessively fits the training data, capturing noise or outliers.
- Leads to excellent performance on training data but poor generalization to unseen test data.
- Mitigated through resampling techniques like k-fold cross-validation, holding back a validation dataset, or removing nodes with little predictive power.

Bias-Variance Trade-off:

- Bias errors stem from oversimplified models, resulting in underfitting.
- Variance errors emerge from differences in training datasets, especially pronounced in overfitting scenarios.
- Ideal models strike a balance between low bias and low variance, though achieving this balance is challenging.
- Algorithms and user parameters, like 'k' in k-Nearest Neighbors, help manage the bias-variance trade-off.

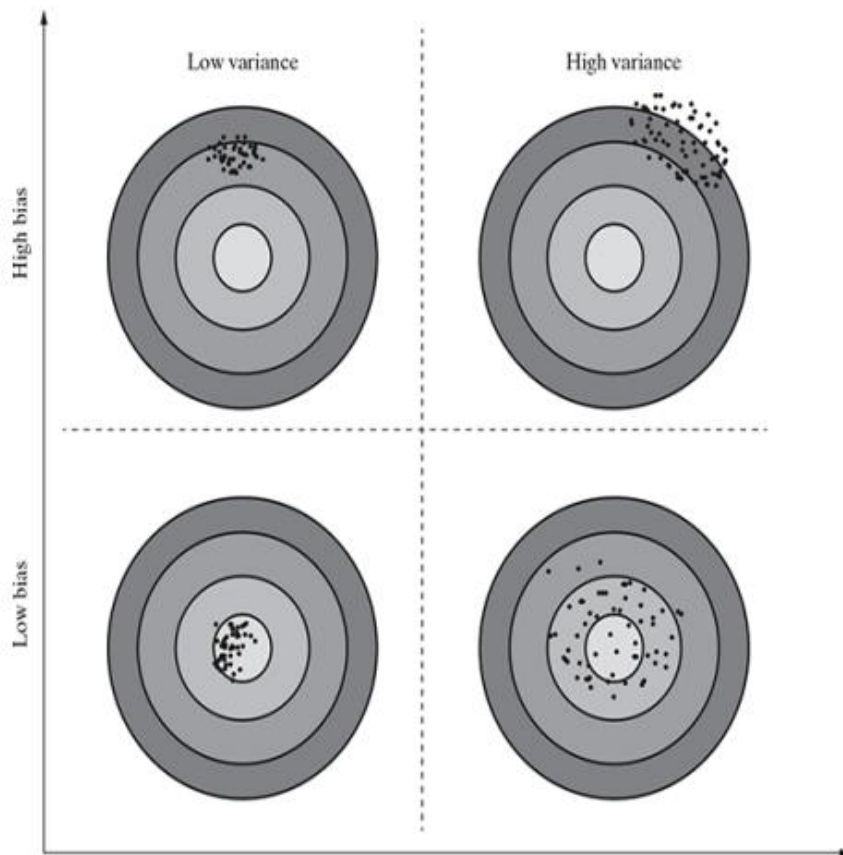


FIG. 3.6 Bias-variance trade-off

In summary, effective supervised learning involves navigating the trade-off between overly simplistic and overly complex models to achieve optimal performance on both training and test datasets.

supervised learning, particularly in classification tasks, evaluating the performance of a model is crucial. Here's a breakdown of key evaluation metrics and methods:

3.5 EVALUATING PERFORMANCE OF A MODEL

Model Accuracy:

- Determines the correctness of predictions made by the model.
- Measured by the ratio of correct predictions to total predictions.
- It's essential to interpret accuracy within the context of the problem domain and its consequences.

$$\text{Model accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{FN} + \text{TN}}$$

Confusion Matrix:

- Summarizes correct and incorrect predictions in a matrix format.
- Contains True Positives (TP), False Positives (FP), False Negatives (FN), and True Negatives (TN).
- Helps calculate various performance metrics.

Let's assume the confusion matrix of the win/loss prediction of cricket match problem to be as below:

	ACTUAL WIN	ACTUAL LOSS
Predicted Win	85	4
Predicted Loss	2	9

In context of the above confusion matrix, total count of TPs = 85, count of FPs = 4, count of FNs = 2 and count of TNs = 9.

$$\therefore \text{Model accuracy} = \frac{TP + TN}{TP + FP + FN + TN} = \frac{85 + 9}{85 + 4 + 2 + 9} = \frac{94}{100} = 94\%$$

The percentage of misclassifications is indicated using **error rate** which is measured as

$$\text{Error rate} = \frac{FP + FN}{TP + FP + FN + TN}$$

In context of the above confusion matrix,

Sensitivity and Specificity:

- Sensitivity measures the proportion of true positive cases correctly classified.
- Specificity measures the proportion of true negative cases correctly classified.
- Sensitivity is crucial in scenarios where identifying positives is vital, such as medical diagnosis.

$$\text{Sensitivity} = \frac{TP}{TP + FN}$$

In the context of the above confusion matrix for the cricket match win prediction problem,

$$\text{Sensitivity} = \frac{TP}{TP + FN} = \frac{85}{85 + 2} = \frac{85}{87} = 97.7\%$$

$$\text{Specificity} = \frac{TN}{TN + FP} = \frac{9}{9 + 4} = \frac{9}{13} = 69.2\%$$

Precision and Recall:

- Precision indicates the reliability of positive predictions.
- Recall measures the proportion of true positives among all actual positives.
- Both metrics provide insights into the model's performance.

$$\text{Precision} = \frac{TP}{TP + FP}$$

$$\text{Recall} = \frac{TP}{TP + FN}$$

F-measure:

- Harmonic mean of precision and recall.
- Offers a combined measure of precision and recall, aiding model comparison.
- Weightage between precision and recall may vary based on the problem domain.

$$F\text{-measure} = \frac{2 \times \text{precision} \times \text{recall}}{\text{precision} + \text{recall}}$$

Receiver Operating Characteristic (ROC) Curves:

- Visualizes classification model performance.
- Plots true positive rate against false positive rate at various classification thresholds.
- Area under the ROC curve (AUC) indicates predictive quality:
- AUC < 0.5: No predictive ability
- 0.5 < AUC < 1.0: Varying degrees of predictive ability
- Evaluating model performance involves a nuanced understanding of these metrics, considering the problem context and potential consequences of misclassification. ROC curves provide a visual assessment of classifier performance, aiding in model selection and comparison based on predictive ability.

$$\text{True Positive Rate TPR} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

$$\text{False Positive Rate FPR} = \frac{\text{FP}}{\text{FP} + \text{TN}}$$

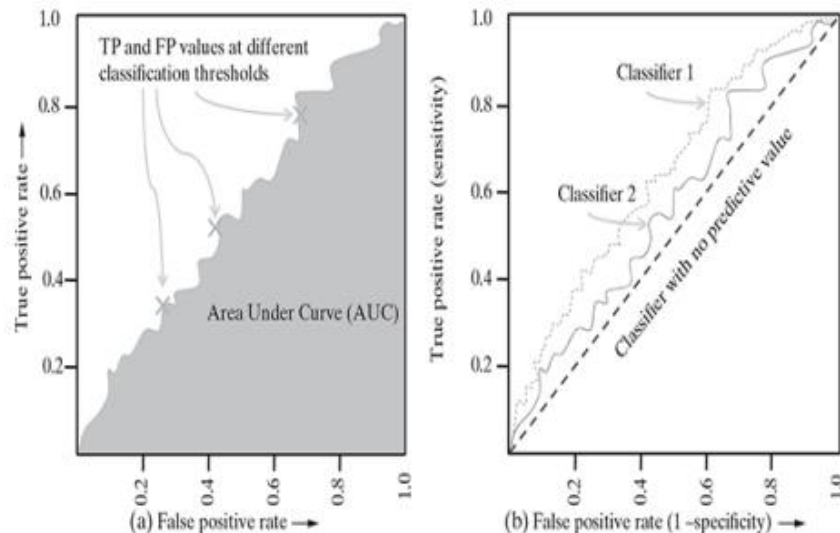


FIG. 3.8 ROC curve

Kappa value

Kappa value can be 1 at the maximum, which represents perfect agreement between model's prediction and actual values.

Kappa value of a model indicates the adjusted the model accuracy. It is calculated using the formula below:

$$\text{Kappa value (k)} = \frac{P(a) - P(p_r)}{1 - P(p_r)}$$

$P(a)$ = Proportion of observed agreement between actual and predicted in overall data set

$$= \frac{TP + TN}{TP + FP + FN + TN}$$

$P(p_r)$ = Proportion of expected agreement between actual and predicted data both in case of class of interest as well as the other classes

$$= \frac{TP + FP}{TP + FP + FN + TN} \times \frac{TP + FN}{TP + FP + FN + TN} + \frac{FN + TN}{TP + FP + FN + TN} \\ \times \frac{FP + TN}{TP + FP + FN + TN}$$

BASICS OF FEATURE ENGINEERING

4.1 Introduction

In the preceding chapters, we delved into the machine learning process, covering various foundational aspects. We commenced with an exploration of human learning and how machine learning mimics it. Following this, we examined the diverse problem types solvable through machine learning techniques. Preceding the application of machine learning to problem-solving, we outlined preparatory steps in detail. Subsequently, we navigated through the systematic process of modeling a problem using machine learning. However, mere modeling does not suffice to gauge the efficacy of machine learning as a problem-solving tool. Hence, we delved into assessing the effectiveness of machine learning models in problem-solving scenarios. Additionally, we explored methods to enhance model performance when necessary. Now, poised to commence problem-solving endeavors employing machine learning, we must address another pivotal aspect crucial to any machine learning problem—feature engineering. While feature engineering is a component of the preparatory activities covered earlier in Chapter 2, its criticality and vastness warrant dedicated attention. This realm pertains to dataset features, which serve as vital inputs for any machine learning problem, whether supervised or unsupervised. Feature engineering is a pivotal preparatory process in machine learning, responsible for converting raw input data into well-aligned features ready for utilization by machine learning models.

Before delving into feature engineering, let's elucidate the concept of a feature.

4.1.1 Understanding Features

A feature constitutes an attribute of a dataset utilized in a machine learning process. Some machine learning practitioners assert that only attributes pertinent to a machine learning problem qualify as features, though this viewpoint requires cautious consideration. Indeed, selecting a subset of features relevant to machine learning constitutes a sub-area of feature engineering, garnering significant research interest. Features within a dataset are also referred to as dimensions. Consequently, a dataset encompassing 'n' features is denoted as an n-dimensional dataset.

Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
6.7	3.3	5.7	2.5	Virginica
4.9	3	1.4	0.2	Setosa
5.5	2.6	4.4	1.2	Versicolor
6.8	3.2	5.9	2.3	Virginica
5.5	2.5	4	1.3	Versicolor
5.1	3.5	1.4	0.2	Setosa
6.1	3	4.6	1.4	versicolor

FIG. 4.1 Data set features

Consider the example of the renowned Iris dataset, introduced by British statistician and biologist Ronald Fisher, as illustrated in Figure 4.1. This dataset comprises five attributes or

features, namely Sepal.Length, Sepal.Width, Petal.Length, Petal.Width, and Species. Among these, the feature 'Species' represents the class variable, while the remaining features serve as predictor variables. Hence, it constitutes a five-dimensional dataset.

4.1.2 Feature Engineering: Concept and Elements

Feature engineering encompasses the process of translating a dataset into features such that these features effectively represent the dataset and enhance learning performance.

As previously established, feature engineering is a crucial preprocessing step for machine learning, comprising two primary elements:

Feature Transformation: This involves transforming data—whether structured or unstructured—into a new set of features capable of effectively representing the underlying problem targeted by machine learning. Feature transformation encompasses two variants:

Feature Construction: This process uncovers missing information regarding relationships between features, augmenting the feature space by creating additional features. Consequently, if a dataset comprises 'n' features or dimensions, 'm' additional features or dimensions may be added through feature construction, resulting in a 'n + m' dimensional dataset.

Feature Extraction: This process entails extracting or creating a new set of features from the original set of features using functional mapping.

Feature Subset Selection: In contrast to feature transformation, feature subset selection (or feature selection) does not generate new features. Instead, its objective is to derive a subset of features from the complete feature set that holds the most significance in the context of a specific machine learning problem. Thus, the aim of feature selection is to derive a subset F (F_1, F_2, \dots, F_m) from F (F_1, F_2, \dots, F_n), where $m < n$, ensuring that F yields optimal results for a given machine learning problem. Further elaboration on these concepts will follow in subsequent sections.

4.2 Feature Transformation

Developing an effective feature space is a pivotal prerequisite for the success of any machine learning model. However, determining which features hold utmost importance often remains ambiguous. Consequently, all available attributes within the dataset are typically employed as features, leaving the task of identifying crucial features to the learning model. This approach proves impractical, especially in domains such as medical image classification and text categorization. For instance, when training a model to classify documents as spam or non-spam, representing a document as a bag of words results in a feature space containing all unique words across documents, potentially numbering in the hundreds of thousands or even millions with the inclusion of bigrams or trigrams. To address this challenge, feature transformation serves as an effective tool for dimensionality reduction, thereby enhancing learning model performance. Feature transformation broadly encompasses two distinct objectives:

Achieving the best reconstruction of the original features in the dataset.

Achieving the highest efficiency in the learning task.

4.2.1 Feature Construction

Feature construction entails transforming a given set of input features to generate a new set of more potent features. To elucidate further, consider a real estate dataset containing details of all apartments sold in a specific region.



The diagram illustrates feature construction. On the left, a table with three columns: 'apartment_length', 'apartment_breadth', and 'apartment_price'. An arrow points to the right, where a new table is shown with four columns: 'apartment_length', 'apartment_breadth', 'apartment_area', and 'apartment_price'. The 'apartment_area' column contains values calculated from length and breadth (e.g., 80 * 59 = 4720).

apartment_length	apartment_breadth	apartment_price
80	59	23,60,000
54	45	12,15,000
78	56	21,84,000
63	63	19,84,000
83	74	30,71,000
92	86	39,56,000

apartment_length	apartment_breadth	apartment_area	apartment_price
80	59	4,720	23,60,000
54	45	2,430	12,15,000
78	56	4,368	21,84,000
63	63	3,969	19,84,500
83	74	6,142	30,71,000
92	86	7,912	39,56,000

FIG. 4.2 Feature construction (example 1)

Example 1: Real Estate Dataset

Original Features: Apartment length, apartment breadth, and apartment price.

Transformation: Instead of utilizing apartment length and breadth as predictors, it proves more convenient and meaningful to incorporate the area of the apartment, an attribute not originally present in the dataset. Thus, the new feature, apartment area, is added to the dataset, effectively transforming the three-dimensional dataset into a four-dimensional one.

Certain scenarios necessitate feature construction as a prerequisite before commencing with machine learning tasks, including:

When features possess categorical values necessitating numeric inputs for machine learning.

When features with numeric (continuous) values require conversion into ordinal values.

When text-specific feature construction is warranted.

4.2.1.1 Encoding Categorical (Nominal) Variables

Consider a dataset on athletes, featuring attributes such as age, city of origin, parents' athletic status, and chance of winning. Since machine learning algorithms mandate numerical inputs, categorical features such as city of origin, parents' athletic status, and chance of winning require transformation into usable features. This transformation involves creating dummy features, assigning values of 0 or 1 based on the categorical value of the original feature in each row.

Age (Years)	City of origin	Parents athlete	Chance of win
18	City A	Yes	Y
20	City B	No	Y
23	City B	Yes	Y
19	City A	No	N
18	City C	Yes	N
22	City B	Yes	Y

(a)

Age (Years)	origin_city_A	origin_city_B	origin_city_C	parents_athlete_Y	parents_athlete_N	win_chance_Y	win_chance_N
18	1	0	0	1	0	1	0
20	0	1	0	0	1	1	0
23	0	1	0	1	0	1	0
19	1	0	0	0	1	0	1
18	0	0	1	1	0	0	1
22	0	1	0	1	0	1	0

(b)

Age (Years)	origin_city_A	origin_city_B	origin_city_C	parents_athlete_Y	win_chance_Y
18	1	0	0	1	1
20	0	1	0	0	1
23	0	1	0	1	1
19	1	0	0	0	0
18	0	0	1	1	0
22	0	1	0	1	1

(c)

FIG. 4.3 Feature construction (encoding nominal variables)

4.2.1.2 Encoding Categorical (Ordinal) Variables

In a student dataset, if the grade is an ordinal variable with values A, B, C, and D, transformation involves mapping each ordinal value to a corresponding numeric value. For instance, grades A, B, C, and D can be mapped to values 1, 2, 3, and 4, respectively.

marks_science	marks_maths	Grade
78	75	B
56	62	C
87	90	A
91	95	A
45	42	D
62	57	B

(a)

(b)

FIG. 4.4 Feature construction (encoding ordinal variables)

4.2.1.3 Transforming Numeric (Continuous) Features to Categorical Features

Sometimes, continuous numerical variables may need transformation into categorical variables. For example, in a real estate price prediction problem, numerical data can be binned into

multiple categories based on data range, transforming the original numerical feature into a categorical variable.

apartment_area	apartment_price
4,720	23,60,000
2,430	12,15,000
4,368	21,84,000
3,969	19,84,500
6,142	30,71,000
7,912	39,56,000

(a)

apartment_area	apartment_grade
4,720	Medium
2,430	Low
4,368	Medium
3,969	Low
6,142	High
7,912	High

(b)

apartment_area	apartment_grade
4,720	2
2,430	1
4,368	2
3,969	1
6,142	3
7,912	3

(c)

FIG. 4.5 Feature construction (numeric to categorical)

4.2.1.4 Text-Specific Feature Construction

Text data, being prevalent in various communication mediums, requires transformation into a numerical representation through vectorization. This process involves tokenization, counting occurrences of each token, and normalization, culminating in the formation of a document-term matrix, which serves as input to machine learning models.

This	House	Build	Feeling	Well	Theatre	Movie	Good	Lonely	...
2	1	1	0	0	1	1	1	0	
0	0	0	1	1	0	0	0	0	
1	0	0	2	1	1	0	0	1	
0	0	0	0	1	0	1	1	0	
.	
.	
.	

FIG. 4.6 Feature construction (text-specific)

4.2.2 Feature Extraction

In feature extraction, new features are generated from combinations of original features, employing various operators tailored to different types of features:

- **Boolean Features:** Operators such as conjunctions, disjunctions, and negations are commonly used.
- **Nominal Features:** Operations like Cartesian product and M of N are applied.
- **Numerical Features:** Operators include min, max, addition, subtraction, multiplication, division, average, equivalence, inequality, etc.

Consider a dataset with a feature set $F(F_1, F_2, \dots, F_n)$. After feature extraction using a mapping function $f(F_1, F_2, \dots, F_n)$, new features are generated, typically resulting in a set with $m < n$ features.

Principal Component Analysis (PCA)

PCA aims to transform a dataset's feature space to a lower-dimensional space, emphasizing dissimilarity among features. This transformation reduces the number of related attributes, enhancing machine learning performance.

Concept: PCA extracts a set of orthogonal features termed principal components, preserving the original data's variability.

Process:

1. Compute the covariance matrix of the dataset.
2. Calculate the eigenvalues of the covariance matrix.
3. Identify the eigenvectors with the highest eigenvalues, representing the principal components.

Singular Value Decomposition (SVD)

SVD, a matrix factorization technique, decomposes a matrix into three constituent matrices: U, Σ , and V, facilitating dimensionality reduction.

Usage: Often applied in PCA after removing attribute means.

Properties:

1. Right-singular vectors (columns of V) capture attribute patterns.
2. Left-singular vectors (columns of U) capture instance patterns.
3. Larger singular values correspond to more significant parts of the matrix.

Linear Discriminant Analysis (LDA)

Unlike PCA, LDA focuses on class separability rather than overall data variability, aiming to transform the dataset into a lower-dimensional space while optimizing class separation to prevent model overfitting.

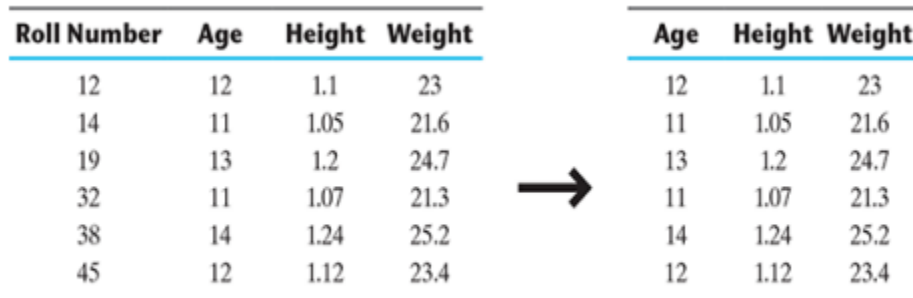
Steps:

1. Calculate mean vectors for individual classes.
2. Compute intra-class and inter-class scatter matrices.
3. Determine eigenvalues and eigenvectors for intra-class and inter-class scatter matrices.
4. Select top 'k' eigenvectors with highest eigenvalues for dimensionality reduction.

Feature extraction techniques like PCA, SVD, and LDA play vital roles in preprocessing datasets for machine learning tasks, facilitating dimensionality reduction while preserving essential information and improving model performance.

4.3 Feature Subset Selection

Feature selection is a crucial preprocessing step in machine learning projects, aiming to identify a subset of system attributes that contribute most meaningfully to the learning task. To illustrate, consider predicting student weights based on attributes like Roll Number, Age, Height, and Weight. It's evident that Roll Number holds no relevance in this prediction task, hence it's eliminated to form a more effective feature subset.



The diagram illustrates feature selection. On the left, a table with four columns: Roll Number, Age, Height, and Weight. An arrow points to the right, where a table with three columns: Age, Height, and Weight. The data rows are identical in both tables, but the 'Roll Number' column is removed in the second table.

Roll Number	Age	Height	Weight
12	12	1.1	23
14	11	1.05	21.6
19	13	1.2	24.7
32	11	1.07	21.3
38	14	1.24	25.2
45	12	1.12	23.4

Age	Height	Weight
12	1.1	23
11	1.05	21.6
13	1.2	24.7
11	1.07	21.3
14	1.24	25.2
12	1.12	23.4

FIG. 4.8 Feature selection

4.3.1 Issues in High-Dimensional Data

With the proliferation of data, particularly in high-dimensional spaces like DNA analysis and text categorization, challenges arise due to the large number of variables. Such datasets often contain hundreds or thousands of dimensions, posing computational and interpretational challenges for machine learning algorithms.

The objective of feature selection is three-fold:

- Having faster and more cost-effective (i.e. less need for computational resources) learning model
- Improving the efficiency of the learning model
- Having a better understanding of the underlying model that generated the data

4.3.2 Key Drivers of Feature Selection – Feature Relevance and Redundancy

Feature Relevance: In supervised learning, features are evaluated based on their contribution to predicting the class label. Irrelevant or weakly relevant features are identified and excluded.

Feature Redundancy: Features may offer similar information, making some redundant. Identifying and eliminating redundant features helps streamline the feature set.

4.3.3 Measures of Feature Relevance and Redundancy

Feature Relevance: Mutual information is used in supervised learning to gauge a feature's information contribution. For unsupervised learning, entropy is calculated to measure feature relevance.

Feature Redundancy: Various measures like correlation, distance-based metrics (e.g., Euclidean distance), and coefficient-based measures (e.g., Jaccard index) assess the similarity between features.

4.3.4 Overall Feature Selection Process

Feature selection involves generating candidate subsets, evaluating them against criteria like classification accuracy, and validating the chosen subset. Different strategies like sequential forward/backward selection or bi-directional selection are employed, with stopping criteria guiding the process.

1. generation of possible subsets
2. subset evaluation
3. stop searching based on some stopping criterion
4. validation of the result

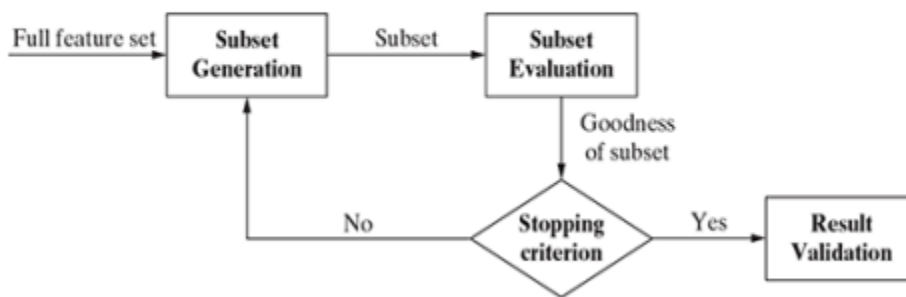


FIG. 4.12 Feature selection process

4.3.5 Feature Selection Approaches

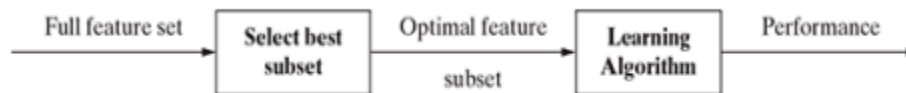


FIG. 4.13 Filter approach

Filter Approach: Features are selected based on statistical measures without employing a learning algorithm.

Wrapper Approach: Features are selected by evaluating candidate subsets using a learning algorithm, making it computationally intensive but often yielding superior performance.

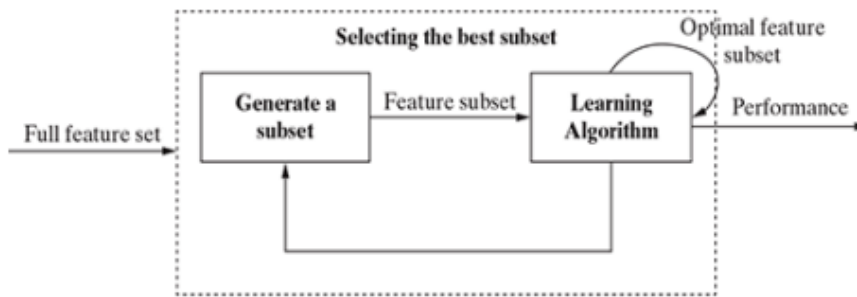


FIG. 4.14 Wrapper approach

Hybrid Approach: Combines statistical tests from the filter approach with learning algorithms to identify the best feature subset.

Embedded Approach: Similar to the wrapper approach, but feature selection and classification are performed simultaneously.

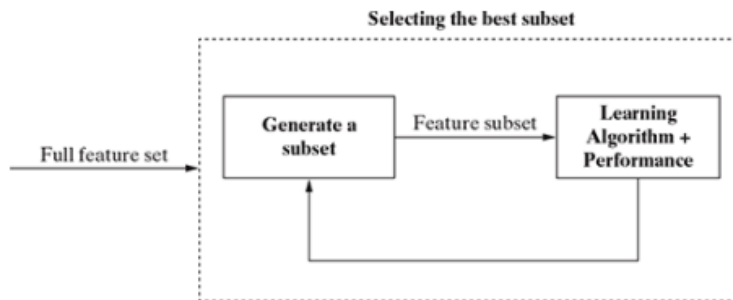


FIG. 4.15 Embedded approach

These approaches offer different trade-offs between computational complexity and performance, allowing practitioners to choose the most suitable method based on the specific requirements of their task.