Machine Learning

R20

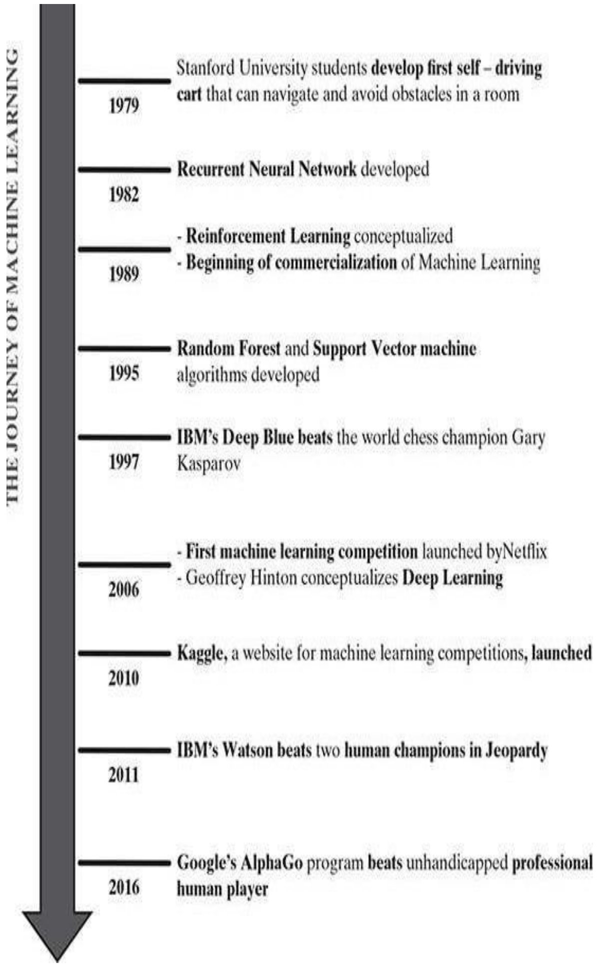
(20A05602T)

Unit-I

Chapter 1 Introduction to Machine Learning

INTRODUCTION

The Rapid development in the area of machine learning has triggered a question in everyone's mind – can machines learn better than human? To find its answer, the first step would be to understand what learning is from a human perspective. Then, more light can be shed on what machine learning is. In the end, we need to know whether machine learning has already surpassed or has the potential to surpass human learning in every facet of life.



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

2006 - **First machine learning competition** launched byNetflix
- Geoffrey Hinton conceptualizes **Deep Learning**

2010 **Kaggle,** a website for machine learning competitions**, launched**

2011 **IBM's Watson beats** two **human champions in Jeopardy**

2016 **Google's AlphaGo** program **beats** unhandicapped **professional human player**

FIG. 1.1 Evolution of machine learning

WHAT IS HUMAN LEARNING?

In cognitive science, learning is typically referred to as the process of gaining information through observation. And why do we need to learn? In our daily life, we need to carry out multiple activities. It may be a task as simple as walking down the street or doing the homework. Or it may be some complex task like deciding the angle in which a rocket should be launched so that it can have a particular trajectory. To do a task in a proper way, we need to have prior information on one or more things related to the task. Also, as we keep learning more or in other words acquiring more information, the efficiency in doing the tasks keep improving. For example, with more knowledge, the ability to do homework with less number of mistakes increases. In the same way, information from past rocket launches helps in taking the right precautions and makes more successful rocket launch. Thus, with more learning, tasks can be performed more efficiently.

TYPES OF HUMAN LEARNING

Thinking intuitively, human learning happens in one of the three ways – (1) either somebody who is an expert in the subject directly teaches us, (2) we build our own notion indirectly based on what we have learnt from the expert in the past, or (3) we do it ourselves, may be after multiple attempts, some being unsuccessful. The first type of learning, we may call, falls under the category of learning directly under expert guidance, the second type falls under learning guided by knowledge gained from experts and the third type is learning by self or self- learning. Let’s look at each of these types deeply using real-life examples and try to understand what they mean.

Learning under expert guidance

An infant may inculcate certain traits and characteristics, learning straight from its guardians. He calls his hand, a ‘hand’, because that is the information he gets from his parents. The sky is ‘blue’ to him because that is what his parents have taught him. We say that the baby ‘learns’ things from his parents.

The next phase of life is when the baby starts going to school. In school, he starts with basic familiarization of alphabets and digits. Then the baby learns how to form words from the alphabets and numbers from the digits. Slowly more complex learning happens in the form of sentences, paragraphs, complex mathematics, science, etc. The baby is able to learn all these things from his teacher who already has knowledge on these areas.

Then starts higher studies where the person learns about more complex, application-oriented skills. Engineering students get skilled in one of the disciplines like civil, computer science, electrical, mechanical, etc. medical students learn about anatomy, physiology, pharmacology, etc. There are some experts, in general the teachers, in the respective field who have in-depth subject matter knowledge, who help the students in learning these skills.

Then the person starts working as a professional in some field. Though he might have gone through enough theoretical learning in the respective field, he still needs to learn more about the hands-on application of the knowledge that he has acquired. The professional mentors, by virtue of the knowledge that they have gained through years of hands-on experience, help all new comers in the field to learn on-job.

In all phases of life of a human being, there is an element of guided learning. This learning is imparted by someone, purely because of the fact that he/she has already gathered the knowledge by virtue of his/her experience in that field. So guided learning is the process of gaining information from a person having sufficient knowledge due to the past experience.

Learning guided by knowledge gained from experts

An essential part of learning also happens with the knowledge which has been impacted by teacher or mentor at some point of time in some other form/context. For example, a baby can group together all objects of same color even if his parents have not specifically taught him to do so. He is able to do so because at some point of time or other his parents have told him which color is blue, which is red, which is green, etc. A grown-up kid can select one odd word from a set of words because it is a verb and other words being all nouns. He could do this because of his ability to label the words as verbs or nouns, taught by his English teacher long back. In a professional role, a person is able to make out to which customers he should market a campaign from the knowledge about preference that was given by his boss long back.

In all these situations, there is no direct learning. It is some past information shared on some different context, which is used as a learning to make decisions.

Learning by self

In many situations, humans are left to learn on their own. A classic example is a baby learning to walk through obstacles. He bumps on to obstacles and falls down multiple times till he learns that whenever there is an obstacle, he needs to cross over it. He faces the same challenge while learning to ride a cycle as a kid or drive a car as an adult. Not all things are taught by others. A lot of things need to be learnt only from mistakes made in the past. We tend to form a check list on things that we should do, and things that we should not do, based on our experiences.

WHAT IS MACHINE LEARNING?

Before answering the question ‘What is machine learning?’ more fundamental questions that peep into one’s mind are Do machines really learn? If so, how do they learn? Which problem can we consider as a well-posed learning problem? What are the important features that are required to well-define a learning problem?

At the onset, it is important to formalize the definition of machine learning. This will itself address the first question, i.e. if machines really learn. There are multiple ways to define machine learning. But the one which is perhaps most relevant, concise and accepted universally is the one stated by Tom M. Mitchell, Professor of Machine Learning Department, School of Computer Science, Carnegie Mellon University. Tom M. Mitchell has defined machine learning as

‘A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E.’

What this essentially means is that a machine can be considered to learn if it is able to gather experience by doing a certain task and improve its performance in doing the similar tasks in the future. When we talk about past experience, it means past data related to the task. This data is an input to the machine from some source.

In the context of the learning to play checkers, E represents the experience of playing the game, T represents the task of playing checkers and P is the performance measure indicated by the percentage of games won by the player. The same mapping can be applied for any other machine learning problem, for example, image classification problem. In context of image classification, E represents the past data with images having labels or assigned classes (for example whether the image is of a class cat or a class dog or a class elephant etc.), T is the task of assigning class to new, unlabelled images and P is the performance measure indicated by the percentage of images correctly classified.

The first step in any project is defining your problem. Even if the most powerful algorithm is used, the results will be meaningless if the wrong problem is solved.

How do machines learn?

The basic machine learning process can be divided into three parts.

1. Data Input: Past data or information is utilized as a basis for future decision-making
2. Abstraction: The input data is represented in a broader way through the underlying algorithm
3. Generalization: The abstracted representation is generalized to form a framework for making decisions.

Figure 1.2 is a schematic representation of the machine learning process.

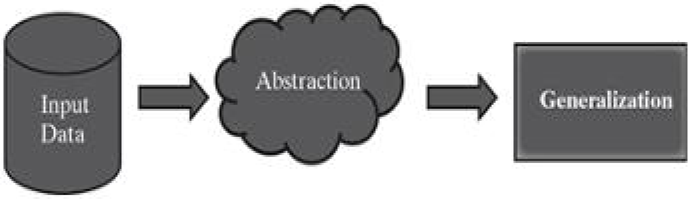


FIG. 1.2 Process of machine learning

Let’s put the things in perspective of the human learning process and try to understand the machine learning process more clearly. Reason is, in some sense, machine learning process tries to emulate the process in which humans learn to a large extent.

Let’s consider the situation of typical process of learning from classroom and books and preparing for the examination. It is a tendency of many students to try and memorize (we often call it ‘learn by heart’) as many things as possible. This may work well when the scope of learning is not so vast. Also, the kinds of questions which are asked in the examination are pretty much simple and straightforward. The questions can be answered by simply writing the same things which have been memorized. However, as the scope gets broader and the questions asked in the examination gets more complex, the strategy of memorizing doesn’t work well. The number of topics may get too vast for a student to memorize. Also, the capability of memorizing varies from student to student. Together with that, since the questions get more complex, a direct reproduction of the things memorized may not help. The situation continues to get worse as the student graduates to higher classes.

So, what we see in the case of human learning is that just by great memorizing and perfect recall, i.e. just based on knowledge input, students can do well in the examinations only till a certain stage. Beyond that, a better learning strategy needs to be adopted:

1. To be able to deal with the vastness of the subject matter and the related issues in memorizing it

2. To be able to answer questions where a direct answer has not been learnt

A good option is to figure out the key points or ideas amongst a vast pool of knowledge. This helps in creating an outline of topics and a conceptual mapping of those outlined topics with the entire knowledge pool. For example, a broad pool of knowledge may consist of all living animals and their characteristics such as whether they live in land or water, whether they lay eggs, whether they have scales or fur or none, etc. It is a difficult task for any student to memorize the characteristics of all living animals – no matter how much photographic memory he/she may possess. It is better to draw a notion about the basic groups that all living animals belong to and the characteristics which define each of the basic groups. The basic groups of animals are invertebrates and vertebrates. Vertebrates are further grouped as mammals, reptiles, amphibians, fishes, and birds. Here, we have mapped animal groups and their salient characteristics.

1. Invertebrate: Do not have backbones and skeletons

2. Vertebrate

- Fishes: Always live in water and lay eggs
- Amphibians: Semi-aquatic i.e. may live in water or land; smooth skin; lay eggs
- Reptiles: Semi-aquatic like amphibians; scaly skin; lay eggs; cold-blooded
- Birds: Can fly; lay eggs; warm-blooded
- Mammals: Have hair or fur; have milk to feed their young; warm-blooded

This makes it easier to memorize as the scope now reduces to know the animal groups that the animals belong to. Rest of the answers about the characteristics of the animals may be derived from the concept of mapping animal groups and their characteristics.

Moving to the machine learning paradigm, the vast pool of knowledge is available from the data input.

However, rather than using it in entirety, a concept map, much in line with the animal group to characteristic mapping explained above, is drawn from the input data. This is nothing but knowledge abstraction as performed by the machine. In the end, the abstracted mapping from the input data can be applied to make critical conclusions. For example, if the group of an animal is given, understanding of the characteristics can be automatically made. Reversely, if the characteristic of an unknown animal is given, a definite conclusion can be made about the animal group it belongs to. This is generalization in context of machine learning.

Abstraction

During the machine learning process, knowledge is fed in the form of input data. However, the data cannot be used in the original shape and form. As we saw in the example above, abstraction helps in deriving a conceptual map based on the input data. This map, or a model as it is known in the machine learning paradigm, is summarized knowledge representation of the raw data. The model may be in any one of the following forms Computational blocks like if/else rules Mathematical equations Specific data structures like trees or graphs Logical groupings of similar observations. The choice of the model used to solve a specific learning problem is a human task. The decision related to the choice of model is taken based on multiple aspects, some of which are listed below:

The type of problem to be solved: Whether the problem is related to forecast or prediction, analysis of trend, understanding the different segments or groups of objects, etc.

Nature of the input data: How exhaustive the input data is, whether the data has no values for many fields, the data types, etc.

Domain of the problem: If the problem is in a business critical domain with a high rate of data input and need for immediate inference, e.g. fraud detection problem in banking domain.

Once the model is chosen, the next task is to fit the model based on the input data. Let’s understand this with an example. In a case where the model is represented by a mathematical equation, say ‘y = c1 + c2x’ (the model is known as simple linear regression which we will study in a later chapter), based on the input data, we have to find out the values of c1 and c2. Otherwise, the equation (or the model) is of no use. So, fitting the model, in this case, means finding the values of the unknown coefficients or constants of the equation or the model. This process of fitting the model based on the input data is known as training. Also, the input data based on which the model is being finalized is known as training data.

Generalization:

The first part of machine learning process is abstraction i.e. abstract the knowledge which comes as input data in the form of a model. However, this abstraction process, or more popularly training the model, is just one part of machine learning. The other key part is to tune up the abstracted knowledge to a form which can be used to take future decisions. This is achieved as a part of generalization. This part is quite difficult to achieve. This is because the model is trained based on a finite set of data, which may possess a limited set of characteristics. But when we want to apply the model to take decision on a set of unknown data, usually termed as test data, we may encounter two problems:

1. The trained model is aligned with the training data too much, hence may not portray the actual trend.

2. The test data possess certain characteristics apparently unknown to the training data.

Hence, a precise approach of decision-making will not work. An approximate or heuristic approach, much like gut-feeling-based decision-making in human beings, has to be adopted. This approach has the risk of not making a correct decision – quite obviously because certain assumptions that are made may not be true in reality.

But just like machines, same mistakes can be made by humans too when a decision is made based on intuition or gut-feeling – in a situation where exact reason-based decision-making is not possible.

Well-posed learning problem:

For defining a new problem, which can be solved using machine learning, a simple framework, highlighted below, can be used. This framework also helps in deciding whether the problem is a right candidate to be solved using machine learning. The framework involves answering three questions:

1. What is the problem?

2. Why does the problem need to be solved?

3. How to solve the problem?

Step 1: What is the Problem?

A number of information should be collected to know what is the problem.

Informal description of the problem, e.g. I need a program that will prompt the next word as and when I type a word.

Formalism: Use Tom Mitchell’s machine learning formalism stated above to define the T, P, and E for the problem.

For example:

Task (T): Prompt the next word when I type a word. Experience (E): A corpus of commonly used English words and phrases.

Performance (P): The number of correct words prompted considered as a percentage (which in machine learning paradigm is known as learning accuracy).

Assumptions - Create a list of assumptions about the problem.

Similar problems:

What other problems have you seen or can you think of that are similar to the problem that you are trying to solve?

Step 2: Why does the problem need to be solved?

Motivation:

What is the motivation for solving the problem? What requirement will it fulfil?

For example, does this problem solve any long- standing business issue like finding out potentially fraudulent transactions? Or the purpose is more trivial like trying to suggest some movies for upcoming weekend.

Solution benefits:

Consider the benefits of solving the problem. What capabilities does it enable?

It is important to clearly understand the benefits of solving the problem. These benefits can be articulated to sell the project.

Solution use

How will the solution to the problem be used and the life time of the solution is expected to have?

Step 3: How would I solve the problem?

Try to explore how to solve the problem manually.

Detail out step-by-step data collection, data preparation, and program design to solve the problem. Collect all these details and update the previous sections of the problem definition, especially the assumptions.

Summary

Step 1: What is the problem? Describe the problem informally and formally and list assumptions and similar problems.

Step 2: Why does the problem need to be solved? List the motivation for solving the problem, the benefits that the solution will provide and how the solution will be used.

Step 3: How would I solve the problem? Describe how the problem would be solved manually to flush domain knowledge.

TYPES OF MACHINE LEARNING

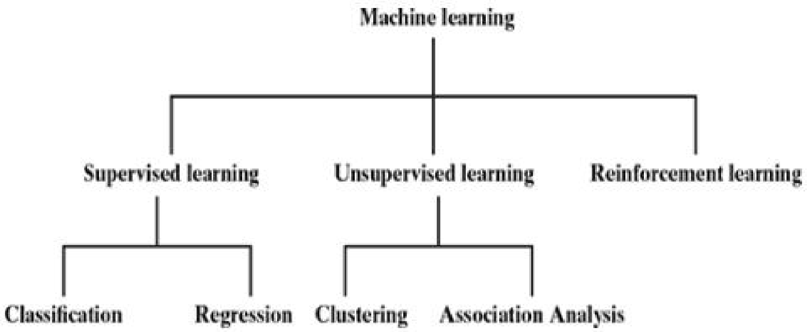


FIG. 1.3 Types of machine learning

As highlighted in Figure 1.3, 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.

Supervised learning:

The major motivation of supervised learning is to learn from past information. So what kind of past information does the machine need for supervised learning? It is the information about the task which the machine has to execute. In context of the definition of machine learning, this past information is the experience. Let’s try to understand it with an example.

Say a machine is getting images of different objects as input and the task is to segregate the images by either shape or color of the object. If it is by shape, the images which are of round-shaped objects need to be separated from images of triangular-shaped objects, etc. If the segregation needs to happen based on color, images of blue objects need to be separated from images of green objects. But how can the machine know what is round shape, or triangular shape? Same way, how can the machine distinguish image of an object based on whether it is blue or green in color? A machine is very much like a little child whose parents or adults need to guide him with the basic information on shape and color before he can start doing the task. A machine needs the basic information to be provided to it. This basic input, or the experience in the paradigm of machine learning, is given in the form of training data. Training data is the past information on a specific task. In context of the image segregation problem, training data will have past data on different aspects or features on a number of images, along with a tag on whether the image is round or triangular, or blue or green in color. The tag is called 'label' and we say that the training data is labeled in case of supervised learning.

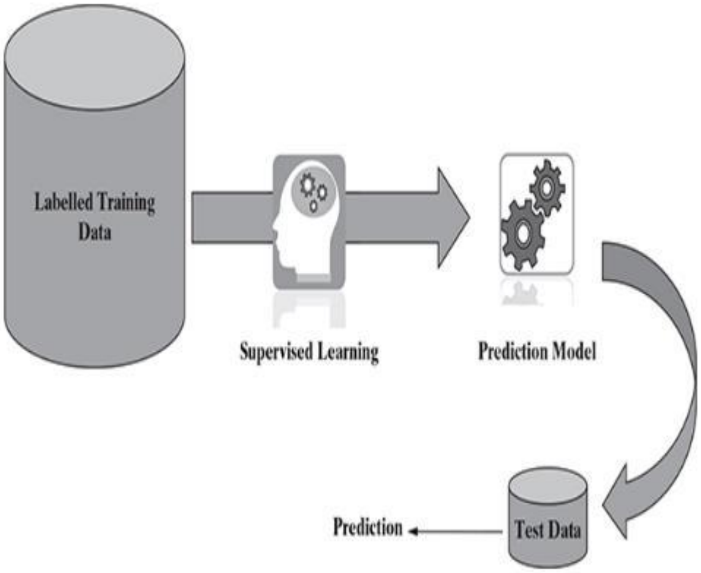


FIG. 1.4 Supervised learning

Figure 1.4 is a simple depiction of the supervised learning process. Labeled training data containing past information comes as an input. Based on the training data, the machine builds a predictive model that can be used on test data to assign a label for each record in the test data.

Some examples of supervised learning are

- Predicting the results of a game
- Predicting whether a tumor is malignant or benign
- Predicting the price of domains like real estate, stocks, etc.
- Classifying texts such as classifying a set of emails as spam or non- spam

Now, let's consider two of the above examples, say 'predicting whether a tumour is malignant or benign' and 'predicting price of domains such as real estate'. Are these two problems same in nature? The answer is 'no'.

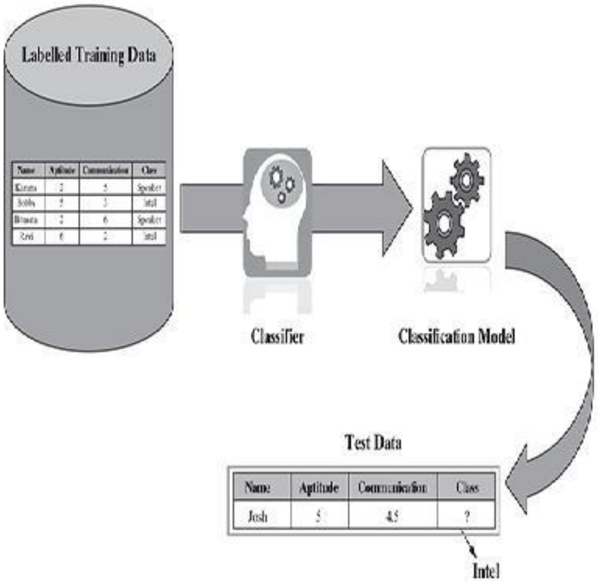
Though both of them are prediction problems, in one case we are trying to predict which category or class an unknown data belongs to whereas in the other case we are trying to predict an absolute value and not a class. When we are trying to predict a categorical or nominal variable, the problem is known as a classification problem. Whereas when we are trying to predict a real- valued variable, the problem falls under the category of regression.

Let’s try to understand these two areas of supervised learning, i.e. classification and regression in more details.

a) Classification

Let’s discuss how to segregate the images of objects based on the shape . If the image is of a round object, it is put under one category, while if the image is of a triangular object, it is put under another category. In which category the machine should put an image of unknown category, also called a test data in machine learning parlance, depends on the information it gets from the past data, which we have called as training data. Since the training data has a label or category defined for each and every image, the machine has to map a new image or test data to a set of images to which it is similar to and assign the same label or category to the test data.

So we observe that the whole problem revolves around assigning a label or category or class to a test data based on the label or category or class information that is imparted by the training data. Since the target objective is to assign a class label, this type of problem as classification problem. Figure 1.5 depicts the typical process of classification.



Name	Aptitude	Communication	Class
Karuna	2	5	Speaker
Bobby	5	3	Intel
Bhuvna	2	6	Speaker
Ravi	6	2	Intel

Name	Aptitude	Communication	Class
Josh	5	4.5	?

FIG. 1.5 Classification

There are number of popular machine learning algorithms which help in solving classification problems. To name a few, Naïve Bayes, Decision tree, and k- Nearest Neighbour algorithms are adopted by many machine learning practitioners.

A critical classification problem in context of banking domain is identifying potential fraudulent transactions. Since there are millions of transactions which have to be scrutinized and assured whether it might be a fraud transaction, it is not possible for any human being to carry out this task. Machine learning is effectively leveraged to do this task and this is a classic case of classification. Based on the past transaction data, specifically the ones labeled as fraudulent, all new incoming transactions are marked or labeled as normal or suspicious. The suspicious transactions are subsequently segregated for a closer review.

In summary, classification is a type of supervised learning where a target feature, which is of type categorical, is predicted for test data based on the information imparted by training data. The target categorical feature is known as class.

Some typical classification problems include:

- Image classification Prediction of disease Win–loss prediction of games
- Prediction of natural calamity like earthquake, flood, etc. Recognition of handwriting

b) Regression

In linear regression, the objective is to predict numerical features like real estate or stock price, temperature, marks in an examination, sales revenue, etc. The underlying predictor variable and the target variable are continuous in nature. In case of linear regression, a straight line relationship is 'fitted' between the predictor variables and the target variables, using the statistical concept of least squares method. As in the case of least squares method, the sum of square of error between actual and predicted values of the target variable is tried to be minimized. In case of simple linear regression, there is only one predictor variable whereas in case of multiple linear regression, multiple predictor variables can be included in the model.

Let's take the example of yearly budgeting exercise of the sales managers. They have to give sales prediction for the next year based on sales figure of previous years vis- à-vis investment being put in. Obviously, the data related to past as well as the data to be predicted are continuous in nature. In a basic approach, a simple linear regression model can be applied with investment as predictor variable and sales revenue as the target variable.

Figure 1.6 shows a typical simple regression model, where regression line is fitted based on values of target variable with respect to different values of predictor variable. A typical linear regression model can be represented in the form –where 'x' is the predictor variable and 'y' is the target variable.

The input data come from a famous multivariate data set named Iris introduced by the British statistician and biologist Ronald Fisher. The data set consists of 50 samples from each of three species of Iris – Iris setosa, Iris virginica, and Iris versicolor. Four features were measured for each sample – sepal length, sepal width, petal length, and petal width. These features can uniquely discriminate the different species of the flower.

The Iris data set is typically used as a training data for solving the classification problem of predicting the flower species based on feature values. However, we can also demonstrate regression using this data set, by predicting the value of one feature using another feature as predictor. In Figure 1.6, 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.

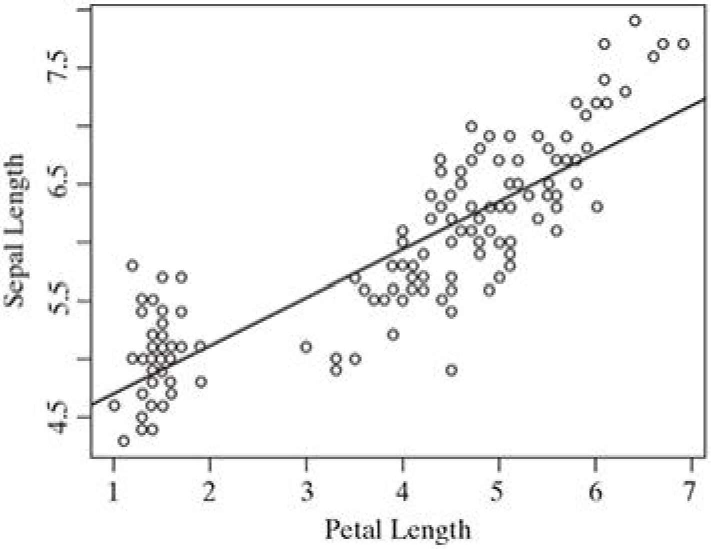


FIG. 1.6 Regression

Typical applications of regression can be seen in

- Demand forecasting in retails
- Sales prediction for managers
- Price prediction in real estate
- Weather forecast
- Skill demand forecast in job market

Unsupervised learning:

Unlike supervised learning, in unsupervised learning, there is no labelled training data to learn from and no prediction to be made. In unsupervised learning, the objective is to take a dataset as input and try to find natural groupings or patterns within the data elements or records. Therefore, unsupervised learning is often termed as descriptive model and the process of unsupervised learning is referred as pattern discovery or knowledge discovery. One critical application of unsupervised learning is customer segmentation.

Clustering is the main type of unsupervised learning. It intends to group or organize similar objects together. For that reason, objects belonging to the same cluster are quite similar to each other while objects belonging to different clusters are quite dissimilar. Hence, the objective of clustering to discover the intrinsic grouping of unlabelled data and form clusters, as depicted in Figure 1.7. Different measures of similarity can be applied for clustering. One of the most commonly adopted similarity measure is distance. Two data items are considered as a part of the same cluster if the distance between them is less. In the same way, if the distance between the data items is high, the items do not generally belong to the same cluster. This is also known as distance-based clustering. Figure 1.8 depicts the process of clustering at a high level.

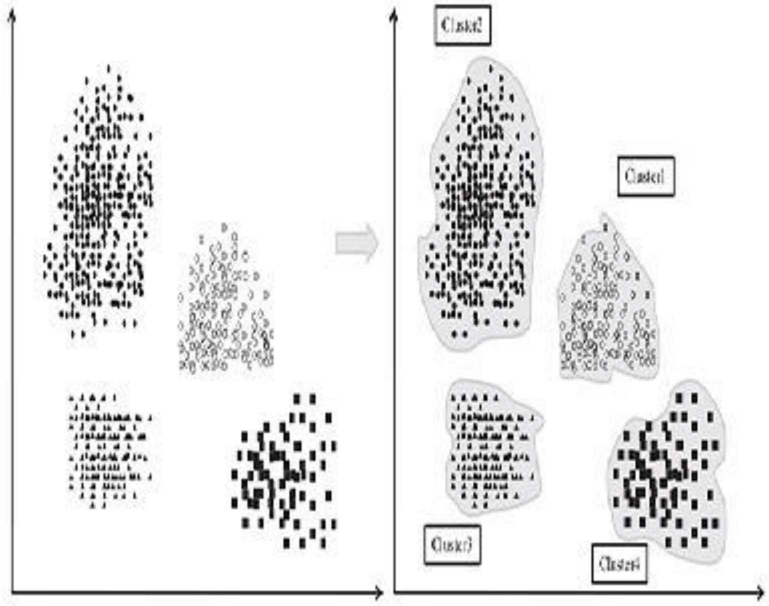


FIG. 1.7 Distance-based clustering

Other than clustering of data and getting a summarized view from it, one more variant of unsupervised learning is association analysis. As a part of association analysis, the association between data elements is identified. Let's try to understand the approach of association analysis in context of one of the most common examples, i.e. market basket analysis as shown in Figure 1.9. From past transaction data in a grocery store, it may be observed that most of the customers who have bought item A, have also bought item B and item C or at least one of them. This means that there is a strong association of the event 'purchase of item A' with the event 'purchase of item B', or 'purchase of item C'. Identifying these sorts of associations is the goal of association analysis. This helps in boosting up sales pipeline, hence a critical input for the sales group.

Critical applications of association analysis include market basket analysis and recommender systems.

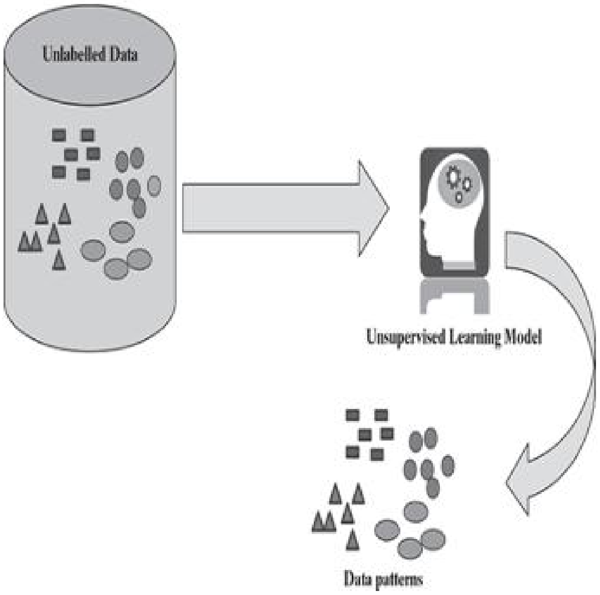
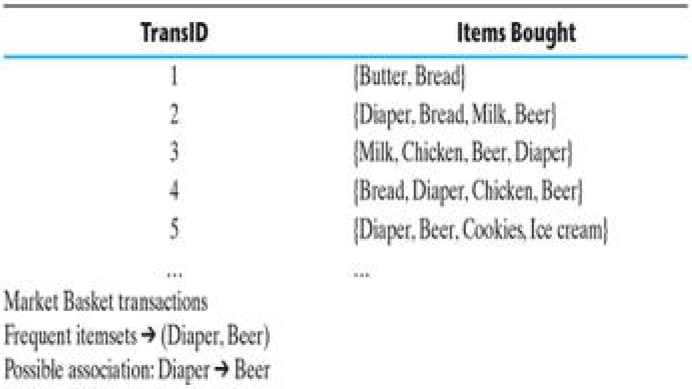


FIG. 1.8 Unsupervised learning



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:

We have seen babies learn to walk without any prior knowledge of how to do it. Often we wonder how they really do it. They do it in a relatively simple way.

First they notice somebody else walking around, for example parents or anyone living around. They understand that legs have to be used, one at a time, to take a step. While walking, sometimes they fall down hitting an obstacle, whereas other times they are able to walk smoothly avoiding bumpy obstacles. When they are able to walk overcoming the obstacle, their parents are elated and appreciate the baby with loud claps / or may be a chocolates. When they fall down while circumventing an obstacle, obviously their parents do not give claps or chocolates. Slowly a time comes when the babies learn from mistakes and are able to walk with much ease.

In the same way, machines often learn to do tasks autonomously. Let's try to understand in context of the example of the child learning to walk. The action tried to be achieved is walking, the child is the agent and the place with hurdles on which the child is trying to walk resembles the environment. It tries to improve its performance of doing the task. When a sub-task is accomplished successfully, a reward is given. When a sub-task is not executed correctly, obviously no reward is given. This continues till the machine is able to complete execution of the whole task. This process of learning is known as reinforcement learning. Figure 1.10 captures the high-level process of reinforcement learning.

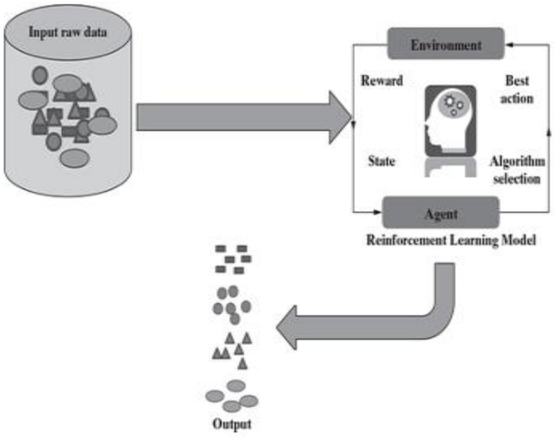
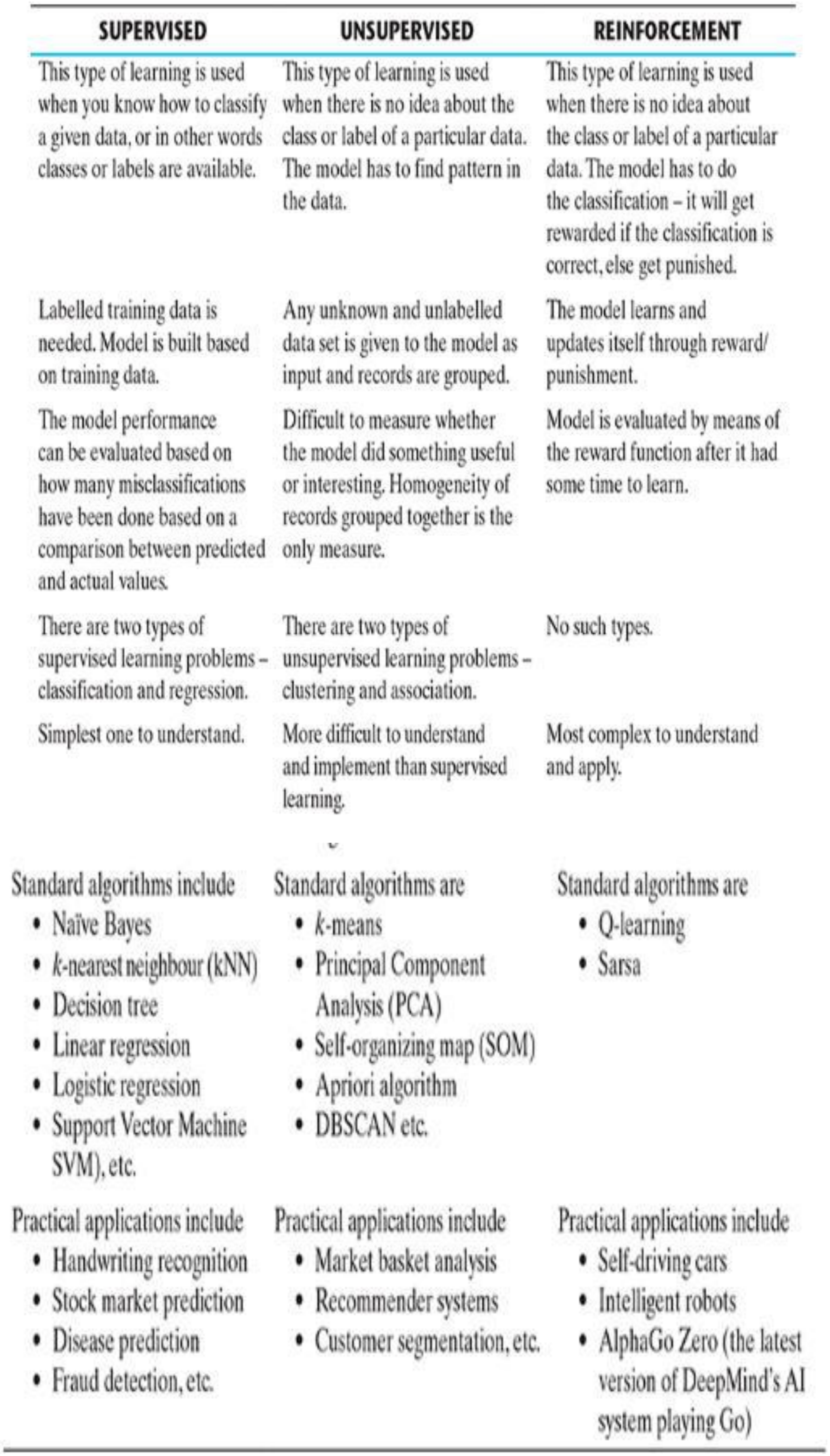


FIG. 1.10 Reinforcement learning

One contemporary example of reinforcement learning is self-driving cars. The critical information which it needs to take care of our speed and speed limit in different road segments, traffic conditions, road conditions, weather conditions, etc. The tasks that have to be taken care of are start/stop, accelerate/decelerate, turn to left / right, etc.



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

PROBLES NOT TO BE SOLVED USING MACHINE LEARNING

Machine learning should not be applied to tasks in which humans are very effective or frequent human intervention is needed. For example, air traffic control is a very complex task needing intense human involvement. At the same time, for very simple tasks which can be implemented using traditional programming paradigms, there is no sense of using machine learning. For example, simple rule-driven or formula-based applications like price calculator engine, dispute tracking application, etc. do not need machine learning techniques.

Machine learning should be used only when the business process has some lapses. If the task is already optimized, incorporating machine learning will not serve to justify the return on investment.

For situations where training data is not sufficient, machine learning cannot be used effectively. This is because, with small training data sets, the impact of bad data is exponentially worse. For the quality of prediction or recommendation to be good, the training data should be sizeable.

APPLICATIONS OF MACHINE LEARNING

Wherever there is a substantial amount of past data, machine learning can be used to generate actionable insight from the data. Though machine learning is adopted in multiple forms in every business domain, we have covered below three major domains just to give some idea about what type of actions can be done using machine learning.

Banking and finance

In the banking industry, fraudulent transactions, especially the ones related to credit cards, are extremely prevalent. Since the volumes as well as velocity of the transactions are extremely high, high performance machine learning solutions are implemented by almost all leading banks across the globe. The models work on a real-time basis, i.e. the fraudulent transactions are spotted and prevented right at the time of occurrence.

This helps in avoiding a lot of operational hassles in settling the disputes that customers will otherwise raise against those fraudulent transactions.

Customers of a bank are often offered lucrative proposals by other competitor banks. Proposals like higher bank interest, lower processing charge of loans, zero balance savings accounts, no overdraft penalty, etc. are offered to customers, with the intent that the customer switches over to the competitor bank. Also, sometimes customers get demotivated by the poor quality of services of the banks and shift to competitor banks. Machine learning helps in preventing or at least reducing the customer churn. Both descriptive and predictive learning can be applied for reducing customer churn. Using descriptive learning, the specific pockets of problem, i.e. a specific bank or a specific zone or a specific type of offering like car loan, may be spotted where maximum churn is happening. Quite obviously, these are troubled areas where further investigation needs to be done to find and fix the root cause. Using predictive learning, the set of vulnerable customers who may leave the bank very soon, can be identified. Proper action can be taken to make sure that the customers stay back.

Insurance

Insurance industry is extremely data intensive. For that reason, machine learning is extensively used in the insurance industry. Two major areas in the insurance industry where machine learning is used are risk prediction during new customer on boarding and claims management. During customer on boarding, based on the past information the risk profile of a new customer needs to be predicted. Based on the quantum of risk predicted, the quote is generated for the prospective customer.

When a customer claim comes for settlement, past information related to historic claims along with the adjustor notes are considered to predict whether there is any possibility of the claim to be fraudulent. Other than the past information related to the specific customer, information related to similar customers, i.e. customer belonging to the same geographical location, age group, ethnic group, etc., are also considered to formulate the model.

Healthcare

Wearable device data form a rich source for applying machine learning and predict the health conditions of the person real time. In case there is some health issue which is predicted by the learning model, immediately the person is alerted to take preventive action. In case of some extreme problem, doctors or healthcare providers in the vicinity of the person can be alerted. Suppose an elderly person goes for a morning walk in a park close to his house. Suddenly, while walking, his blood pressure shoots up beyond a certain limit, which is tracked by the wearable. The wearable data is sent to a remote server and a machine learning algorithm is constantly analyzing the streaming data. It also has the history of the elderly person and persons of similar age group. The model predicts some fatality unless immediate action is taken. Alert can be sent to the person to immediately stop walking and take rest. Also, doctors and healthcare providers can be alerted to be on standby. Machine learning along with computer vision also plays a crucial role in disease diagnosis from medical imaging.

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

The algorithms related to different machine learning tasks are known to all and can be implemented using any language/platform. It can be implemented using a Java platform or C / C++ language or in .NET. However, there are certain languages and tools which have been developed with a focus for implementing machine learning. Few of them, which are most widely used, are covered below.

Python

Python is one of the most popular, open source programming languages widely adopted by machine learning community. It was designed by Guido van Rossum and was first released in 1991. The reference implementation of Python, i.e. CPython, is managed by Python Software Foundation, which is a non-profit organization.Python has very strong libraries for advanced mathematical functionalities (NumPy), algorithms and mathematical tools (SciPy) and numerical plotting (matplotlib). Built on these libraries, there is a machine learning library named scikit-learn, which has various classification, regression, and clustering algorithms embedded in it.

R

R is a language for statistical computing and data analysis. It is an open source language, extremely popular in the academic community – especially among statisticians and data miners. R is considered as a variant of S, a GNU project which was developed at Bell Laboratories. Currently, it is supported by the R Foundation for statistical computing.

R is a very simple programming language with a huge set of libraries available for different stages of machine learning. Some of the libraries standing out in terms of popularity are plyr/dplyr (for data transformation), caret (‘Classification and Regression Training’ for classification), RJava (to facilitate integration with Java), tm (for text mining), ggplot2 (for data visualization).Other than the libraries, certain packages like Shiny and R Markdown have been developed around R to develop interactive web applications, documents and dashboards on R without much effort.

Matlab

MATLAB (matrix laboratory) is licensed commercial software with a robust support for a wide range of numerical computing. MATLAB has a huge user base across industry and academia. MATLAB is developed by Math Works, a company founded in 1984. Being proprietary software, MATLAB is developed much more professionally, tested rigorously, and has comprehensive documentation.

MATLAB also provides extensive support of statistical functions and has a huge number of machine learning algorithms in-built. It also has the ability to scale up for large datasets by parallel processing on clusters and cloud.

SAS

SAS (earlier known as ‘Statistical Analysis System’) is another licenced commercial software which provides strong support for machine learning functionalities. Developed in C by SAS Institute, SAS had its first release in the year 1976.

SAS is a software suite comprising different components. The basic data management functionalities are embedded in the Base SAS component whereas the other components like SAS/INSIGHT, Enterprise Miner, SAS/STAT, etc. help in specialized functions related to data mining and statistical analysis.

Other languages/tools

There are a host of other languages and tools that also support machine learning functionalities. Owned by IBM, SPSS (originally named as Statistical Package for the Social Sciences) is a popular package supporting specialized data mining and statistical analysis.

Originally popular for statistical analysis in social science (as the name reflects), SPSS is now popular in other fields as well. Released in 2012, Julia is an open source, liberal licence programming language for numerical analysis and computational science. It has baked in all good things of MATLAB, Python, R, and other programming languages used for machine learning for which it is gaining steady attention from machine learning development community. Another big point in favor of Julia is its ability to implement high-performance machine learning algorithms.

ISSUES IN MACHINE LEARNING

Machine learning is a field which is relatively new and still evolving. Also, the level of research and kind of use of machine learning tools and technologies varies drastically from country to country. The laws and regulations, cultural background, emotional maturity of people differ drastically in different countries. All these factors make the use of machine learning and the issues originating out of machine learning usage are quite different.

The biggest fear and issue arising out of machine learning is related to privacy and the breach of it. The primary focus of learning is on analyzing data, both past and current, and coming up with insight from the data. This insight may be related to people and the facts revealed might be private enough to be kept confidential. Also, different people have a different preference when it comes to sharing of information. While some people may be open to sharing some level of information publicly, some other people may not want to share it even to all friends and keep it restricted just to family members.

Classic examples are a birth date (not the day, but the date as a whole), photographs of a dinner date with family, educational background, etc. Some people share them with all in the social platforms like Face book while others do not, or if they do, they may restrict it to friends only. When machine learning algorithms are implemented using those information, inadvertently people may get upset. For example, if there is a learning algorithm to do preference-based customer segmentation and the output of the analysis is used for sending targeted marketing campaigns, it will hurt the emotion of people and actually do more harm than good. In certain countries, such events may result in legal actions to be taken by the people affected.

Even if there is no breach of privacy, there may be situations where actions were taken based on machine learning may create an adverse reaction. Let’s take the example of knowledge discovery exercise done before starting an election campaign. If a specific area reveals an ethnic majority or skewness of a certain demographic factor, and the campaign pitch carries a message keeping that in mind, it might actually upset the voters and cause an adverse result.

So a very critical consideration before applying machine learning is that proper human judgment should be exercised before using any outcome from machine learning. Only then the decision taken will be beneficial and also not result in any adverse impact.

CHAPTER 2

PREPARING TO MODEL

INTRODUCTION

In the last chapter, we got introduced to machine learning. In the beginning, we got a glimpse of the journey of machine learning as an evolving technology. It all started as a proposition from the renowned computer scientist Alan Turing – machines can 'learn' and become artificially intelligent. Gradually, through the next few decades path-breaking innovations came in from Arthur Samuel, Frank Rosenblatt, John Hopfield, Christopher Watkins, Geoffrey Hinton and many other computer scientists. They shaped up concepts of Neural Networks, Recurrent Neural Network, Reinforcement Learning, Deep Learning, etc. which took machine learning to new heights. In parallel, interesting applications of machine learning kept on happening, with organizations like IBM and Google taking a lead. What started with IBM's Deep Blue beating the world chess champion Gary Kasparov, continued with IBM's Watson beating two human champions in a Jeopardy competition.Google also started with a series of innovations applying machine learning. The Google Brain, Sibyl, Waymo, AlphaGo programs – are all extremely advanced applications of machine learning which have taken the technology a few notches up. Now we can see an all-pervasive presence of machine learning technology in all walks of life.

We have also seen the types of human learning and how that, in some ways, can be related to the types of machine learning – supervised, unsupervised, and reinforcement. Supervised learning, as we saw, implies learning from past data, also called training data, which has got known values or classes. Machines can 'learn' or get 'trained' from the past data and assign classes or values to unknown data, termed as test data. This helps in solving problems related to prediction. This is much like human learning through expert guidance as happens for infants from parents or students through teachers.

So, supervised learning in case of machines can be perceived as guided learning from human inputs. Unsupervised machine learning doesn't have labeled data to learn from. It tries to find patterns in unlabelled data. This is much like human beings trying to group together objects of similar shape. This learning is not guided by labeled inputs but uses the knowledge gained from the labels themselves. Last but not the least is reinforcement learning in which machine tries to learn by itself through penalty/ reward mechanism – again pretty much in the same way as human self-learning happens.

Lastly, we saw some of the applications of machine learning in different domains such as banking and finance, insurance, and healthcare. Fraud detection is a critical business case which is implemented in almost all banks across the world and uses machine learning predominantly. Risk prediction for new customers is a similar critical case in the insurance industry which finds the application of machine learning. In the healthcare sector, disease prediction makes wide use of machine learning, especially in the developed countries.

While development in machine learning technology has been extensive and its implementation has become widespread, to start as a practitioner, we need to gain some basic understanding. We need to understand how to apply the array of tools and technologies available in the machine learning to solve a problem. In fact, that is going to be very specific to the kind of problem that we are trying to solve. If it is a prediction problem, the kind of activities that will be involved is going to be completely different vis-à-vis if it is a problem where we are trying to unfold a pattern in a data without any past knowledge about the data. So how a machine learning project looks like or what are the salient activities that form the core of a machine learning project will depend on whether it is in the area of supervised or unsupervised or reinforcement learning area. However, irrespective of the variation, some foundational knowledge needs to be built before we start with the core machine learning concepts and key algorithms. In this section, we will have a quick look at a few typical machine learning activities and focus on some of the foundational concepts that all practitioners need to gain as pre-requisites before starting their journey in the area of machine learning.

MACHINE LEARNING ACTIVITIES

The first step in machine learning activity starts with data. In case of supervised learning, it is the labelled training data set followed by test data which is not labelled. In case of unsupervised learning, there is no question of labelled data but the task is to find patterns in the input data. A thorough review and exploration of the data is needed to understand the type of the data, the quality of the data and relationship between the different data elements. Based on that, multiple pre-processing activities may need to be done on the input data before we can go ahead with core machine learning activities.

Following are the typical preparation activities done once the input data comes into the machine learning system:

Understand the type of data in the given input data set. Explore the data to understand the nature and quality.

Explore the relationships amongst the data elements, e.g. inter- feature relationship.

Find potential issues in data.

Do the necessary remediation, e.g. impute missing data values, etc., if needed.

Apply pre-processing steps, as necessary.

Once the data is prepared for modelling, then the learning tasks start off. As a part of it, do the following activities:

The input data is first divided into parts – the training data and the test data (called holdout). This step is applicable for supervised learning only.

Consider different models or learning algorithms for selection.

Train the model based on the training data for supervised learning problem and apply to unknown data. Directly apply the chosen unsupervised model on the input data for unsupervised learning problem.

After the model is selected, trained (for supervised learning), and applied on input data, the performance of the model is evaluated. Based on options available, specific actions can be taken to improve the performance of the model, if possible.

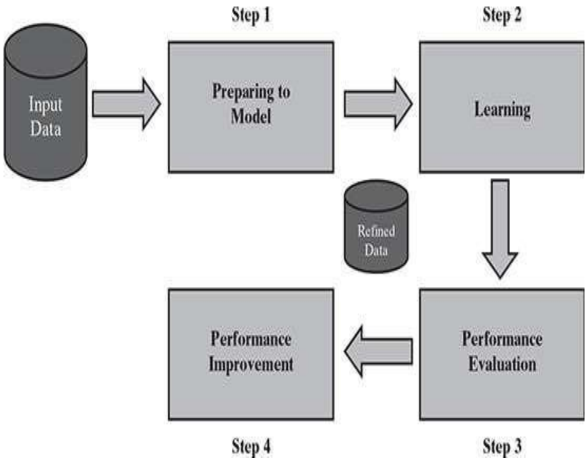
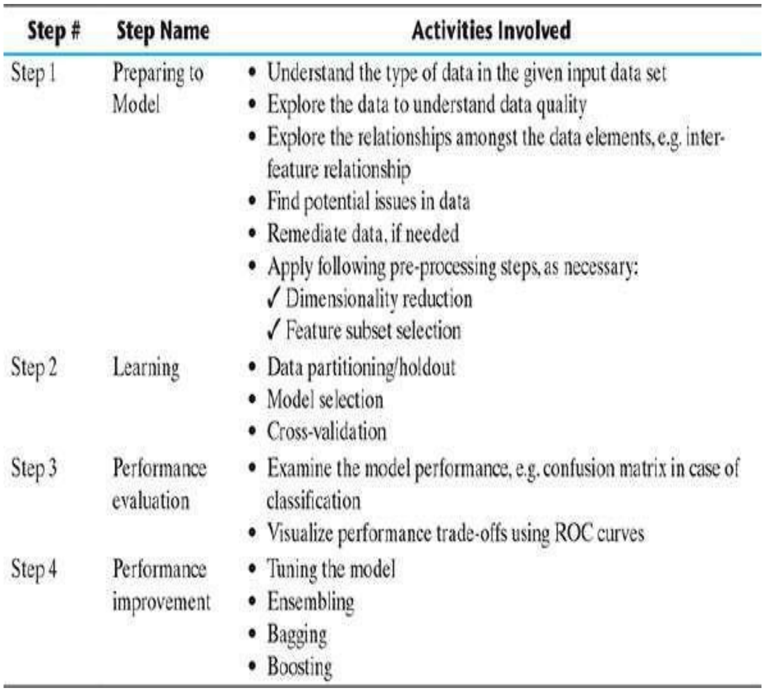


Table 2.1 contains a summary of steps and activities involved:



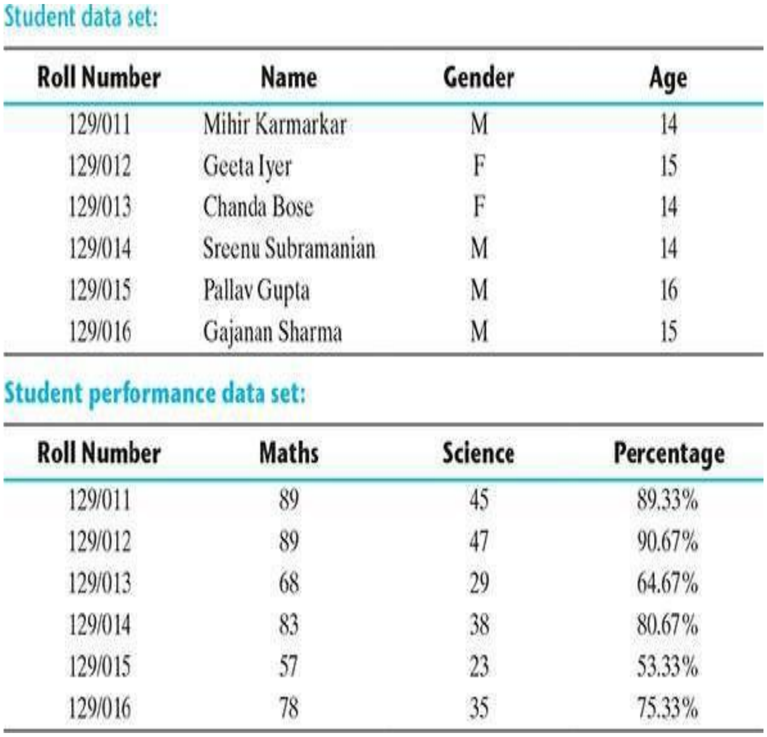
Step #	Step Name	Activities Involved
Step 1	Preparing to Model	• Understand the type of data in the given input data set • Explore the data to understand data quality • Explore the relationships amongst the data elements, e.g. inter-feature relationship • Find potential issues in data • Remediate data, if needed • Apply following pre-processing steps, as necessary: ✓ Dimensionality reduction ✓ Feature subset selection
Step 2	Learning	• Data partitioning/holdout • Model selection • Cross-validation
Step 3	Performance evaluation	• Examine the model performance, e.g. confusion matrix in case of classification • Visualize performance trade-offs using ROC curves
Step 4	Performance improvement	• Tuning the model • Ensembling • Bagging • Boosting

In this chapter, we will cover the first part, i.e. preparing to model.

BASIC TYPES OF DATA IN MACHINE LEARNING

Before starting with types of data, let’s first understand what a data set is and what are the elements of a data set. A data set is a collection of related information or records. The information may be on some entity or some subject area. For example (Fig. 2.2), we may have a data set on students in which each record consists of information about a specific student. Again, we can have a data set on student performance which has records providing performance, i.e. marks on the individual subjects.

Each row of a data set is called a record. Each data set also has multiple attributes, each of which gives information on a specific characteristic. For example, in the data set on students, there are four attributes namely Roll Number, Name, Gender, and Age, each of which understandably is a specific characteristic about the student entity. Attributes can also be termed as feature, variable, dimension or field. Both the data sets, Student and Student Performance, are having four features or dimensions; hence they are told to have four- dimensional data space. A row or record represents a point in the four-dimensional data space as each row has specific values for each of the four attributes or features. Value of an attribute, quite understandably, may vary from record to record. For example, if we refer to the first two records in the Student data set, the value of attributes Name, Gender, and Age are different (Fig.2.3).



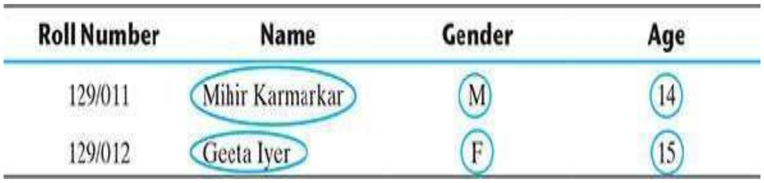
Student data set:

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

Now that a context of data sets is given, let’s try to understand the different types of data that we generally come across in machine learning problems. Data can broadly be divided into following two types:

1. Qualitative data
2. Quantitative data

Qualitative data provides information about the quality of an object or information which cannot be measured. For example, if we consider the quality of performance of students in terms of ‘Good’, ‘Average’, and ‘Poor’, it falls under the category of qualitative data. Also, name or roll number of students are information that cannot be measured using some scale of measurement. So they would fall under qualitative data. Qualitative data is also called categorical data.

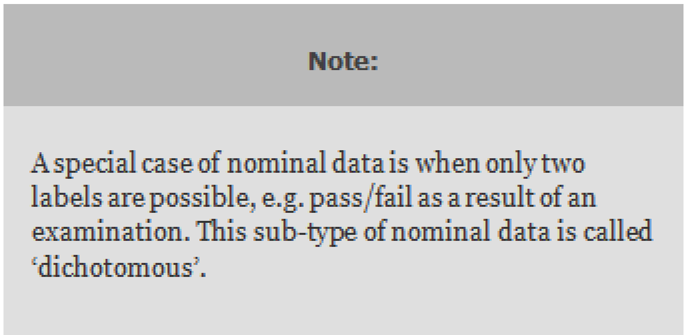
Qualitative data can be further subdivided into two types as follows:

1. Nominal data
2. Ordinal data

Nominal data is one which has no numeric value, but a named value. It is used for assigning named values to attributes. Nominal values cannot be quantified.

Examples of nominal data are

1. Blood group: A, B, O, AB, etc.
2. Nationality: Indian, American, British, etc.
3. Gender: Male, Female, Other



Note:

A special case of nominal data is when only two labels are possible, e.g. pass/fail as a result of an examination. This sub-type of nominal data is called ‘dichotomous’.

It is obvious, mathematical operations such as addition, subtraction, multiplication, etc. cannot be performed on nominal data. For that reason, statistical functions such as mean, variance, etc. can also not be applied on nominal data. However, a basic count is possible. So mode, i.e. most frequently occurring value, can be identified for nominal data.

Ordinal data, in addition to possessing the properties of nominal data, can also be naturally ordered. This means ordinal data also assigns named values to attributes but unlike nominal data, they can be arranged in a sequence of increasing or decreasing value so that we can say whether a value is better than or greater than another value. Examples of ordinal data are

1. Customer satisfaction: ‘Very Happy’, ‘Happy’, ‘Unhappy’, etc.

2. Grades: A, B, C, etc.

3. Hardness of Metal: ‘Very Hard’, ‘Hard’, ‘Soft’, etc.

Like nominal data, basic counting is possible for ordinal data. Hence, the mode can be identified. Since ordering is possible in case of ordinal data, median, and quartiles can be identified in addition. Mean can still not be calculated.

Quantitative data relates to information about the quantity of an object – hence it can be measured. For example, if we consider the attribute ‘marks’, it can be measured using a scale of measurement. Quantitative data is also termed as numeric data. There are two types of quantitative data:

1. Interval data

2. Ratio data

Interval data is numeric data for which not only the order is known, but the exact difference between values is also known. An ideal example of interval data is Celsius temperature. The difference between each value remains the same in Celsius temperature. For example, the difference between 12°C and 18°C degrees is measurable and is 6°C as in the case of difference between 15.5°C and 21.5°C. Other examples include date, time, etc.

For interval data, mathematical operations such as addition and subtraction are possible. For that reason, for interval data, the central tendency can be measured by mean, median, or mode. Standard deviation can also be calculated.

However, interval data do not have something called a ‘true zero’ value. For example, there is nothing called ‘0 temperature’ or ‘no temperature’. Hence, only addition and subtraction applies for interval data. The ratio cannot be applied. This means, we can say a temperature of 40°C is equal to the temperature of 20°C + temperature of 20°C. However, we cannot say the temperature of 40°C means it is twice as hot as in temperature of 20°C.

Ratio data represents numeric data for which exact value can be measured. Absolute zero is available for ratio data. Also, these variables can be added, subtracted, multiplied, or divided. The central tendency can be measured by mean, median, or mode and methods of dispersion such as standard deviation. Examples of ratio data include height, weight, age, salary, etc.

Figure 2.4 gives a summarized view of different types of data that we may find in a typical machine learning problem.

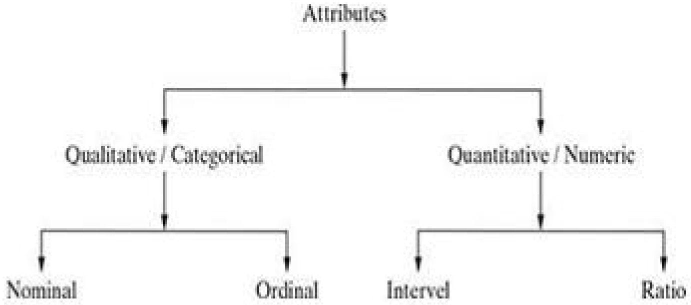
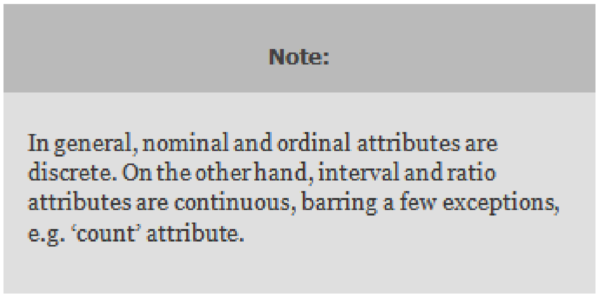


FIG. 2.4 Types of data

Apart from the approach detailed above, attributes can also be categorized into types based on a number of values that can be assigned. The attributes can be either discrete or continuous based on this factor.Discrete attributes can assume a finite or countably infinite number of values. Nominal attributes such as roll number, street number, pin code, etc. can have a finite number of values whereas numeric attributes such as count, rank of students, etc. can have countably infinite values. A special type of discrete attribute which can assume two values only is called binary attribute.

Examples of binary attribute include male/ female, positive/negative, yes/no, etc.

Continuous attributes can assume any possible value which is a real number. Examples of continuous attribute include length, height, weight, price, etc.



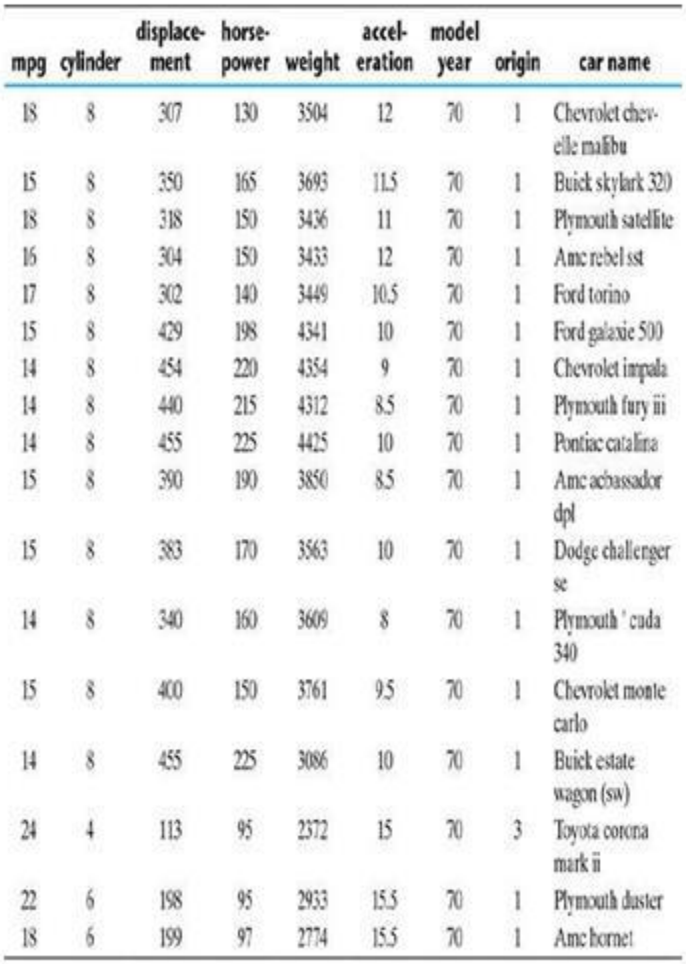
Note:

In general, nominal and ordinal attributes are discrete. On the other hand, interval and ratio attributes are continuous, barring a few exceptions, e.g. 'count' attribute.

EXPLORING STRUCTURE OF DATA

By now, we understand that in machine learning, we come across two basic data types – numeric and categorical. With this context in mind, we can delve deeper into understanding a data set. We need to understand that in a data set, which of the attributes are numeric and which are categorical in nature. This is because the approach of exploring numeric data is different than the approach of exploring categorical data. In case of a standard data set, we may have the data dictionary available for reference. Data dictionary is a metadata repository, i.e. the repository of all information related to the structure of each data element contained in the data set. The data dictionary gives detailed information on each of the attributes – the description as well as the data type and other relevant details. In case the data dictionary is not available, we need to use standard library function of the machine learning tool that we are using and get the details. For the time being, let us move ahead with a standard data set from UCI machine learning repository.

The data set that we take as a reference is the Auto MPG data set available in the UCI repository. Figure 2.5 is a snapshot of the first few rows of the data set.



mpg	cylinder	displace-ment	horse-power	weight	accel-eration	model year	origin	car name
18	8	307	130	3504	12	70	1	Chevrolet chev-elle 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

As is quite evident from the data, the attributes such as ‘mpg’, ‘cylinders’, ‘displacement’, ‘horsepower’, ‘weight’, ‘acceleration’, ‘model year’, and ‘origin’ are all numeric. Out of these attributes, ‘cylinders’, ‘model year’, and ‘origin’ are discrete in nature as the only finite number of values can be assumed by these attributes. The remaining of the numeric attributes, i.e. ‘mpg’, ‘displacement’, ‘horsepower’, ‘weight’, and ‘acceleration’ can assume any real value.

Exploring numerical data

There are two most effective mathematical plots to explore numerical data – box plot and histogram. We will explore all these plots one by one, starting with the most critical one, which is the box plot.

Understanding central tendency

To understand the nature of numeric variables, we can apply the measures of central tendency of data, i.e. mean and median. In statistics, measures of central tendency help us understand the central point of a set of data.

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.

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

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.

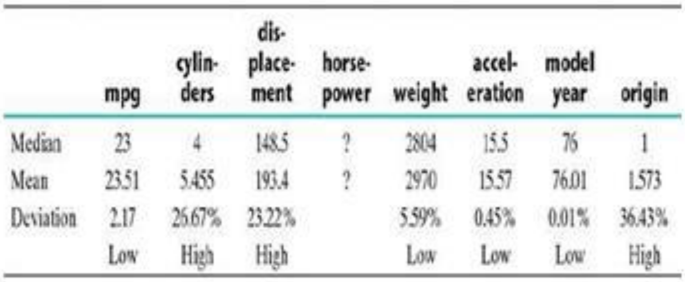
Median, on contrary, is the value of the element appearing in the middle of an ordered list of data elements. If we consider the above 5 data elements, the ordered list would be – 21, 34, 67, 89, and 96. Since there are 5 data elements, the 3rd element in the ordered list is considered as the median. Hence, the median value of this set of data is 67.

There might be a natural curiosity to understand why two measures of central tendency are reviewed. The reason is mean and median are impacted differently by data values appearing at the beginning or at the end of the range. Mean being calculated from the cumulative sum of data values, is impacted if too many data elements are having values closer to the far end of the range, i.e. close to the maximum or minimum values. It is especially sensitive to outliers, i.e. the values which are unusually high or low, compared to the other values.

Mean is likely to get shifted drastically even due to the presence of a small number of outliers. If we observe that for certain attributes the deviation between values of mean and median are quite high, we should investigate those attributes further and try to find out the root cause along with the need for remediation.

So, in the context of the Auto MPG data set, let's try to find out for each of the numeric attributes the values of mean and median. We can also find out if the deviation between these values is large. In Figure 2.6, the comparison between mean and median for all the attributes has been shown. We can see that for the attributes such as 'mpg', 'weight', 'acceleration', and 'model.year' the deviation between mean and median is not significant which means the chance of these attributes having too many outlier values is less.

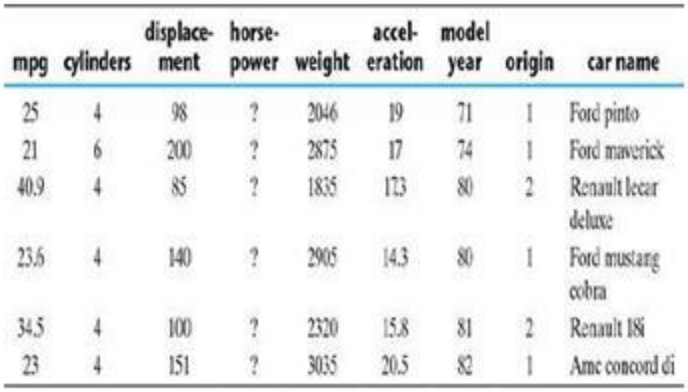
However, the deviation is significant for the attributes 'cylinders', 'displacement' and 'origin'. So, we need to further drill down and look at some more statistics for these attributes. Also, there is some problem in the values of the attribute 'horsepower' because of which the mean and median calculation is not possible.



	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

With a bit of investigation, we can find out that the problem is occurring because of the 6 data elements, as shown in Figure 2.7, do not have value for the attribute ‘horsepower’.



mpg	cylinders	displace-ment	horse-power	weight	accel-eration	model year	origin	car name
25	4	98	?	2046	19	71	1	Ford pinto
21	6	200	?	2875	17	74	1	Ford maverick
40.9	4	85	?	1835	173	80	2	Renault lecar deluxe
23.6	4	140	?	2905	14.3	80	1	Ford mustang cobra
34.5	4	100	?	2320	15.8	81	2	Renault 18i
23	4	151	?	3035	20.5	82	1	Amc concord di

FIG. 2.7 Missing values of attribute ‘horsepower’ in Auto MPG

For that reason, the attribute ‘horsepower’ is not treated as a numeric. That’s why the operations applicable on numeric variables, like mean or median, are failing. So we have to first remediate the missing values of the attribute ‘horsepower’ before being able to do any kind of exploration. However, we will cover the approach of remediation of missing values a little later.

Understanding data spread

Now that we have explored the central tendency of the different numeric attributes, we have a clear idea of which attributes have a large deviation between mean and median. Let’s look closely at those attributes. To drill down more, we need to look at the entire range of values of the attributes, though not at the level of data elements as that may be too vast to review manually. So we will take a granular view of the data spread in the form of

1. Dispersion of data
2. Position of the different data values
3. Measuring data dispersion

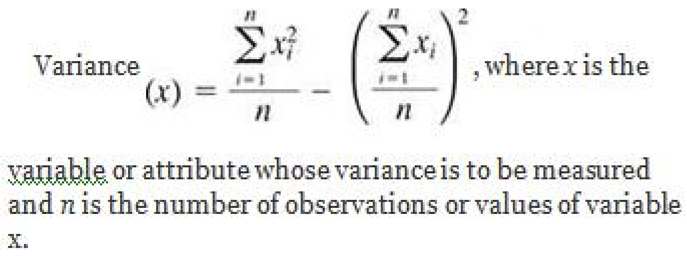
Consider the data values of two attributes

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

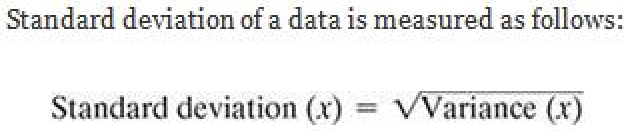
2. Attribute 2 values : 34, 46, 59, 39, and 52

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:



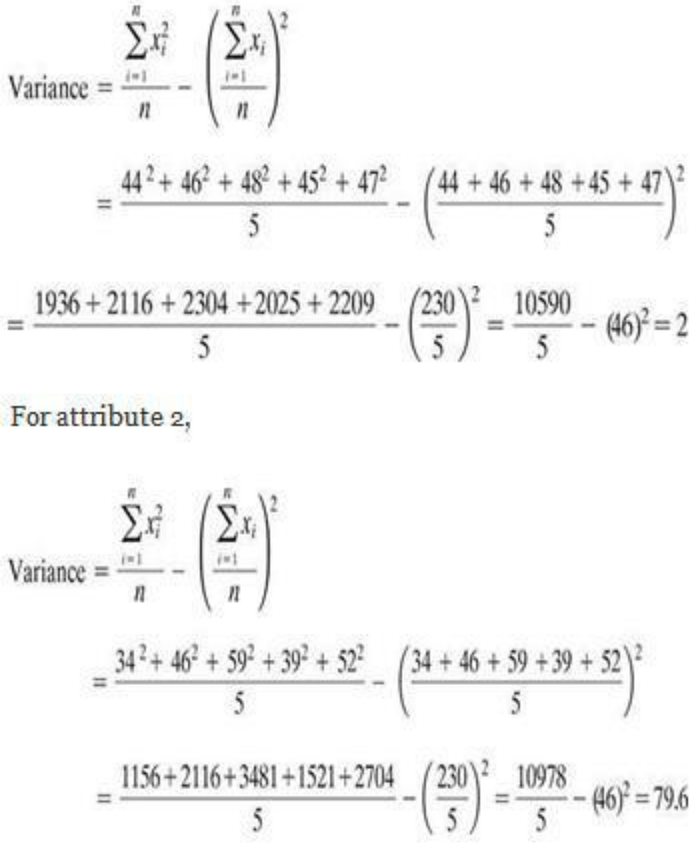
Variance $(x) = \frac{\sum_{i=1}^{n} x_i^2}{n} - \left(\frac{\sum_{i=1}^{n} x_i}{n}\right)^2$, where x is the variable or attribute whose variance is to be measured and n is the number of observations or values of variable x.



Standard deviation of a data is measured as follows:

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

Larger value of variance or standard deviation indicates more dispersion in the data and vice versa. In the above example, let’s calculate the variance of attribute 1 and that of attribute 2. For attribute 1, So it is quite clear from the measure that attribute 1 values are quite concentrated around the mean while attribute 2 values are extremely spread out. Since this data was small, a visual inspection and understanding were possible and that matches with the measured value.



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

$$= \frac{44^2 + 46^2 + 48^2 + 45^2 + 47^2}{5} - \left(\frac{44 + 46 + 48 + 45 + 47}{5}\right)^2$$

$$= \frac{1936 + 2116 + 2304 + 2025 + 2209}{5} - \left(\frac{230}{5}\right)^2 = \frac{10590}{5} - (46)^2 = 2$$

For attribute 2,

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

$$= \frac{34^2 + 46^2 + 59^2 + 39^2 + 52^2}{5} - \left(\frac{34 + 46 + 59 + 39 + 52}{5}\right)^2$$

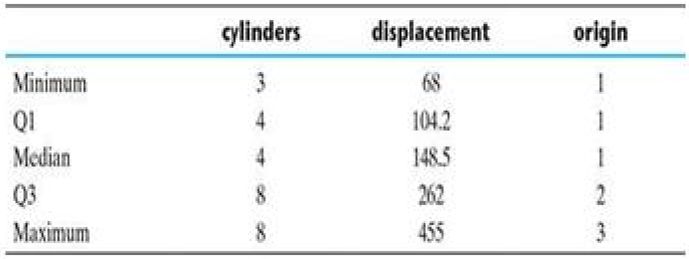
$$= \frac{1156 + 2116 + 3481 + 1521 + 2704}{5} - \left(\frac{230}{5}\right)^2 = \frac{10978}{5} - (46)^2 = 79.6$$

Measuring data value position

When the data values of an attribute are arranged in an increasing order, we have seen earlier that median gives the central data value, which divides the entire data set into two halves. Similarly, if the first half of the data is divided into two halves so that each half consists of one- quarter of the data set, then that median of the first half is known as first quartile or Q1. In the same way, if the second half of the data is divided into two halves, then that median of the second half is known as third quartile or Q3. The overall median is also known as second quartile or Q2. So, any data set has five values -minimum, first quartile (Q1), median (Q2), third quartile (Q3), and maximum.

Let's review these values for the attributes 'cylinders', 'displacement', and 'origin'. Figure 2.8 captures a summary of the range of statistics for the attributes. If we take the example of the attribute 'displacement', we can see that the difference between minimum value and Q1 is 36.2 and the difference between Q1 and median is

On the contrary, the difference between median and Q3 is 113.5 and Q3 and the maximum value is 193. In other words, the larger values are more spread out than the smaller ones. This helps in understanding why the value of mean is much higher than that of the median for the attribute 'displacement'. Similarly, in case of attribute 'cylinders', we can observe that the difference between minimum value and median is 1 whereas the difference between median and the maximum value is 4. For the attribute 'origin', the difference between minimum value and median is 0 whereas the difference between median and the maximum value is 2.



	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

However, we still cannot ascertain whether there is any outlier present in the data. For that, we can better adopt some means to visualize the data. Box plot is an excellent visualization medium for numeric data.

DATA PRE-PROCESSING

Dimensionality reduction

Till the end of the 1990s, very few domains were explored which included data sets with a high number of attributes or features. In general, the data sets used in machine learning used to be in few 10s. However, in the last two decades, there has been a rapid advent of computational biology like genome projects. These projects have produced extremely high-dimensional data sets with 20,000 or more features being very common.

Also, there has been a wide-spread adoption of social networking leading to a need for text classification for customer behavior analysis.

High-dimensional data sets need a high amount of computational space and time. At the same time, not all features are useful – they degrade the performance of machine learning algorithms. Most of the machine learning algorithms performs better if the dimensionality of data set, i.e. the number of features in the data set, is reduced. Dimensionality reduction helps in reducing irrelevance and redundancy in features. Also, it is easier to understand a model if the number of features involved in the learning activity is less.

Dimensionality reduction refers to the techniques of reducing the dimensionality of a data set by creating new attributes by combining the original attributes. The most common approach for dimensionality reduction is known as Principal Component Analysis (PCA). PCA is a statistical technique to convert a set of correlated variables into a set of transformed, uncorrelated variables called principal components. The principal components are a linear combination of the original variables. They are orthogonal to each other. Since principal components are uncorrelated, they capture the maximum amount of variability in the data. However, the only challenge is that the original attributes are lost due to the transformation. Another commonly used technique which is used for dimensionality reduction is Singular Value Decomposition (SVD).

Feature subset selection

Feature subset selection or simply called feature selection, both for supervised as well as unsupervised learning; try to find out the optimal subset of the entire feature set which significantly reduces computational cost without any major impact on the learning accuracy. It may seem that a feature subset may lead to loss of useful information as certain features are going to be excluded from the final set of features used for learning. However, for elimination only features which are not relevant or redundant are selected.

A feature is considered as irrelevant if it plays an insignificant role (or contributes almost no information) in classifying or grouping together a set of data instances. All irrelevant features are eliminated while selecting the final feature subset. A feature is potentially redundant when the information contributed by the feature is more or less same as one or more other features. Among a group of potentially redundant features, a small number of features can be selected as a part of the final feature subset without causing any negative impact to learn model accuracy.

UNIT-2

Modelling and Evaluation

OBJECTIVE OF THE CHAPTER :

The previous chapter gives a comprehensive understanding of the basic data types in the contextof machine learning. It also enables a beginner in the field of machine learning to acquire an understanding about the nature and quality of the data by effective exploration of the data set. In this chapter, the objective is to introduce the basic concepts of learning. In this regard, the informationshared concerns the aspects of model selection and application. It also imparts knowledge regarding how to judge the effectiveness of the model in doinga specific learning task, supervised or unsupervised, and how to boost the model performance using different tuning parameters.

INTRODUCTION

The learning process of machines may seem quite magical to somebody who is new to machine learning. The thought that a machine is able to think and take intelligent action may be mesmerizing – much like a science fiction or a fantasy story. However, delving a bit deeper helps them realize that it is not as magical as it may seem to be. In fact, it tries to emulate human learning by applying mathematical and statistical formulations. In that sense, both human and machine learning strives to build formulations or mapping basedon a limited number of observations. As introduced in Chapter 1, the basic learning process, irrespective of the fact that the learner is a human or a machine, can bedivided into three parts:

1. Data Input
2. Abstraction
3. Generalization

Though in Chapter 1 we have understood these aspectsin details, let’s quickly refresh our memory with an example. It’s a fictitious situation. The detective department of New City Police has got a tip that in a campaign gathering for the upcoming election, a criminal is going to launch an attack on the main candidate. However, it is not known who the person is and quite obviously the person might use some disguise. The only thing that is for sure is the person is a history- sheeter or a criminal having a long record of serious crime. From the criminal database, a list of such criminals along with their photographs has been collected. Also, the photos taken by security cameras positioned at different places near the gathering are available with the detective department. They have to match the photos from the criminal database with the faces in the gathering to spot the potential attacker. So the main problem here is to spot the face of the criminal based on the match with the photos in the criminal database.

This can be done using human learning where a person from the detective department can scan through each shortlisted photo and try to match that photo with the faces in the gathering. A person having a strong memory can take a glance at the photos of all criminals in one shot and then try to find a face in the gathering which closely resembles one of the criminal photos that she has viewed. Easy, isn’t it? But that is not possible in reality. The number of criminals in the database and hence the count of photos runs in hundreds, if not thousands. So taking a look at all the photos and memorizing them is not possible. Also, an exact match isout of the question as the criminal, in most probability, will come in disguise. The strategy to be taken here is to match the photos in smaller counts and also based on certain salient physical features like the shape of the jaw,the slope of the forehead, the size of the eyes, the structure of the ear, etc. So, the photos from the criminal database form the input data. Based on it, key features can be abstracted. Since human matching for each and every photo may soon lead to a visual as well as mental fatigue, a generalization of abstracted feature-based datais a good way to detect potential criminal faces in the gathering. For example, from the abstracted feature- based data, say it is observed that most of the criminals have a shorter distance between the inner corners of the eyes, a smaller angle between the nose and the corners ofthe mouth, a higher curvature to the upper lip, etc.

Hence, a face in the gathering may be classified as ‘potentially criminal’ based on whether they match withthese generalized observations. Thus, using the input data, feature-based abstraction could be built and by applying generalization of the abstracted data, human learning could classify the faces as potentially criminal ultimately leading to spotting of the criminal.

The same thing can be done using machine learning too. Unlike human detection, a machine has no subjective baggage, no emotion, no bias due to past experience, and above all no mental fatigue. The machine can also use the same input data, i.e. criminal database photos, apply computational techniques to abstract feature-based concept map from the input data and generalize the same in the form of a classification algorithm to decide whether a face in the gathering is potentially criminal or not.

When we talk about the learning process, abstraction is a significant step as it represents raw input data in a summarized and structured format, such that a meaningful insight is obtained from the data. This structured representation of raw input data to the meaningful pattern is called a **model**. The model mighthave different forms. It might be a mathematical equation, it might be a graph or tree structure, it might be a computational block, etc. The decision regarding which model is to be selected for a specific data set is taken by the learning task, based on the problem to be solved and the type of data. For example, when the problem is related to prediction and the target field is numeric and continuous, the regression model is assigned. The process of assigning a model, and fitting aspecific model to a data set is called model **training.**

Once the model is trained, the raw input data issummarized into an abstracted form.

However, with abstraction, the learner is able to only summarize the knowledge. This knowledge might be stillvery broad-based – consisting of a huge number of feature-based data and inter-relations. To generate actionable insight from such broad-based knowledge is very difficult. This is where generalization comes into play. Generalization searches through the huge set of abstracted knowledge to come up with a small and manageable set of key findings. It is not possible to do anexhaustive search by reviewing each of the abstracted findings one-by-one. A heuristic search is employed, an approach which is also used for human learning (often termed as ‘gut-feel’). It is quite obvious that the heuristics sometimes result in erroneous result. If the outcome is systematically incorrect, the learning is said to have a **bias**.

Points to Ponder:

- A machine learning algorithm creates its cognitive capability by building a mathematical formulation or function, known as target function, based on the features in the input data set.
- Just like a child learning things for the first time needs her parents guidance to decide whether she is right or wrong, in machine learning someone has to provide some non-learnable parameters, also called hyper-parameters. Without these human inputs, machine learning algorithms cannot be successful.

SELECTING A MODEL

Now that you are familiar with the basic learning process and have understood model abstraction and generalization in that context, let's try to formalize it in context of a motivating example. Continuing the thread of the potential attack during the election campaign, New City Police department has succeeded in foiling the bid to attack the electoral candidate. However, this was a wake-up call for them and they want to take a proactive action to eliminate all criminal activities in the region.

They want to find the pattern of criminal activities in the recent past, i.e. they want to see whether the number of criminal incidents per month has any relation with an average income of the local population, weapon sales, the inflow of immigrants, and other such factors. Therefore, an association between potential causes of disturbance and criminal incidents has to be determined. In other words, the goal or target is to develop a model to infer how the criminal incidents change based on the potential influencing factors mentioned above.

In machine learning paradigm, the potential causes of disturbance, e.g. average income of the local population, weapon sales, the inflow of immigrants, etc. are input variables. They are also called predictors, attributes, features, independent variables, or simply variables. The number of criminal incidents is an output variable (also called response or dependent variable). Input variables can be denoted by X, while individual input variables are represented as $X_1, X_2, X_3, \ldots, X_n$ and output variable by symbol Y. The relationship between X and Y is represented in the general form: $Y = f(X) + e$, where 'f' is the **target function** and 'e' is a random error term.

Note:

Just like a target function with respect to a machinelearning model, some other functions which are frequently tracked are

- A **cost function** (also called **error function**) helps to measure the extent to which the model is going wrong in estimating the relationship between *X* and *Y*. In that sense, cost function can tell how bad the model is performing. For example, R-squared (to be discussed later in this chapter) is acost function of regression model.
  Loss function is almost synonymous to cost function – only difference being loss function is usually a function defined on adata point, while cost function is for the entire training data set.

Machine learning is an optimization problem. We try to define a model and tune the parameters to find the most suitable solution to a problem. However, we need to have

- a way to evaluate the quality or optimality of a solution. This is done using **objective function**. Objective means goal.
- Objective function takes in data and model (along with parameters) as input and returns a value. Target is to find values of model parameter to maximize or minimize the returnvalue. When the objective is to minimize the value, it becomes synonymous to cost function. Examples: maximize the rewardfunction in reinforcement learning, maximize the posterior probability in Naive Bayes, minimize squared error in regression

But the problem that we just talked about is one specific type of problem in machine learning. We have seen in Chapter 1 that there are three broad categories of machine learning approaches used for resolving differenttypes of problems. Quickly recapitulating, they are

1. Supervised
   1. Classification
   2. Regression
2. Unsupervised
   1. Clustering
   2. Association analysis
3. Reinforcement

For each of the cases, the model that has to be created/trained is different. Multiple factors play a role when we try to select the model for solving a machine learning problem. The most important factors are (i) the kind of problem we want to solve using machine learning and (ii) the nature of the underlying data. The problem may be related to the prediction of a class value like whether a tumour is malignant or benign, whether the next day will be snowy or rainy, etc. It may be related to prediction – but of some numerical value like what the price of a house should be in the next quarter, what is theexpected growth of a certain IT stock in the next 7 days, etc. Certain problems are related to grouping of data like finding customer segments that are using a certain product, movie genres which have got more box office success in the last one year, etc. So, it is very difficult to give a generic guidance related to which machine learning has to be selected. In other words, there is no one model that works best for every machine learning problem. This is what '**No Free Lunch'** theorem also states.

Any learning model tries to simulate some real-worldaspect. However, it is simplified to a large extent removing all intricate details. These simplifications are based on certain assumptions – which are quite dependent on situations. Based on the exact situation,i.e. the problem in hand and the data characteristics, assumptions may or may not hold. So the same model may yield remarkable results in a certain situation while it may completely fail in a different situation. That's why,while doing the data exploration, which we covered inthe previous chapter, we need to understand the data characteristics, combine this understanding with the problem we are trying to solve and then decide which model to be selected for solving the problem.

Let's try to understand the philosophy of model selection in a structured way. Machine learning algorithms are broadly of two types: models for supervised learning, which primarily focus on solvingpredictive problems and models for unsupervised learning, which solve descriptive problems.

Predictive models

Models for supervised learning or predictive models, as is understandable from the name itself, try to predict certain value using the values in an input data set. The learning model attempts to establish a relation between the target feature, i.e. the feature being predicted, andthe predictor features. The predictive models have a clearfocus on what they want to learn and how they want to learn.

Predictive models, in turn, may need to predict thevalue of a category or class to which a data instance belongs to. Below are some examples:

1. Predicting win/loss in a cricket match
2. Predicting whether a transaction is fraud
3. Predicting whether a customer may move to another product

The models which are used for prediction of target features of categorical value are known as classification models. The target feature is known as a class and the categories to which classes are divided into are called levels. Some of the popular classification models include *k*-Nearest Neighbor (kNN), Naïve Bayes, and Decision Tree.

Predictive models may also be used to predict numerical values of the target feature based on thepredictor features. Below are some examples:

1. Prediction of revenue growth in the succeeding year
2. Prediction of rainfall amount in the coming monsoon
3. Prediction of potential flu patients and demand for flu shots nextwinter

The models which are used for prediction of the numerical value of the target feature of a data instance are known as regression models. Linear Regression andLogistic Regression models are popular regression models.

Points to Ponder:

- Categorical values can be converted to numerical values and vice versa. For example, for stock price growth prediction, any growth percentage lying between certain ranges may be represented by a categorical value, e.g. 0%–5% as ‘low’, 5%– 10% as ‘moderate’, 10%– 20% as ‘high’ and > 20% as ‘booming’. In a similar way, a categorical value can be converted to numerical value, e.g. in the tumor malignancy detection problem, replace ‘benign’ as 0 and ‘malignant’ as 1. This way, the models can be used interchangeably, though it may not work always.
- There are multiple factors to be considered while selecting a model. For example, while selecting the model for prediction, the training data size is an important factor to be considered. If the training data set is small, low variance models like Naïve Bayes are supposed to perform better because model overfitting needs to be avoided in this situation. Similarly, when the training data is large, low bias models like logistic regression should be preferred because they can represent complex relationships in a more effective way.

Few models like Support Vector Machines and Neural Network can be used for both classificationsas well as for regression.

Descriptive models

Models for unsupervised learning or descriptive models are used to describe a data set or gain insight from a dataset. There is no target feature or single feature of interestin case of unsupervised learning. Based on the value of all features, interesting patterns or insights are derived about the data set.

Descriptive models which group together similar data instances, i.e. data instances having a similar value of the different features are called clustering models. Examplesof clustering include

1.Customer grouping or segmentation based on social, demographic, ethnic, etc. factors

2.Grouping of music based on different aspects like genre, language, time-period, etc.

3.Grouping of commodities in an inventory

The most popular model for clustering is *k*-Means.

Descriptive models related to pattern discovery is used for market basket analysis of transactional data. In market basket analysis, based on the purchase pattern available in the transactional data, the possibility of purchasing one product based on the purchase of another product is determined. For example, transactional data may reveal a pattern that generally a customer who purchases milk also purchases biscuit at the same time. This can be useful for targeted promotions or in-store set up. Promotions related to biscuits can be sent to customers of milk products or viceversa. Also, in the store products related to milk can be placed close to biscuits.

TRAINING A MODEL (FOR SUPERVISED LEARNING)

Holdout method

In case of supervised learning, a model is trained using the labelled input data. However, how can we understand the performance of the model? The test data may not be available immediately. Also, the label value ofthe test data is not known. That is the reason why a part of the input data is held back (that is how the name holdout originates) for evaluation of the model. This subset of the input data is used as the test data for evaluating the performance of a trained model. In general 70%–80% of the input data (which is obviously labelled) is used for model training. The remaining 20%–30% is used as test data for validation of the performance of the model. However, a different proportion of dividing the input data into training and test data is also acceptable. To make sure that the data inboth the buckets are similar in nature, the division is done randomly. Random numbers are used to assign data items to the partitions. This method of partitioning the input data into two parts – training and test data (depicted in Figure 3.1), which is by holding back a partof the input data for validating the trained model is known as holdout method

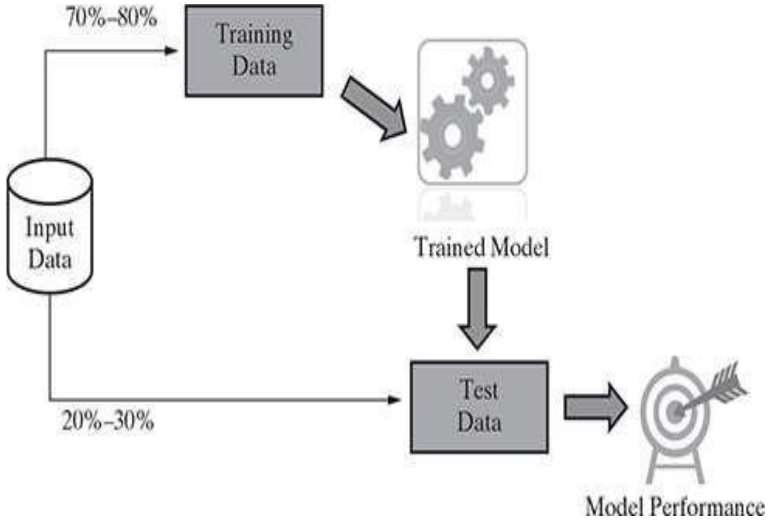


FIG. 3.1 Holdout method

Once the model is trained using the training data, the labels of the test data are predicted using the model's target function. Then the predicted value is compared with the actual value of the label. This is possible becausethe test data is a part of the input data with known labels.The performance of the model is in general measured by the accuracy of prediction of the label value.

In certain cases, the input data is partitioned into threeportions – a training and a test data, and a third validation data. The validation data is used in place of test data, for measuring the model performance. It is used in iterations and to refine the model in each iteration. The test data is used only for once, after the model is refined and finalized, to measure and report the final performance of the model as a reference for future learning efforts.

An obvious problem in this method is that the divisionof data of different classes into the training and test datamay not be proportionate. This situation is worse if the overall percentage of data related to certain classes is much less compared to other classes. This may happen despite the fact that random sampling is employed for test data selection. This problem can be addressed to some extent by applying stratified random sampling in place of sampling. In case of stratified random sampling,the whole data is broken into several homogenous groups or strata and a random sample is selected from each such stratum. This ensures that the generated random partitions have equal proportions of each class.

K-fold Cross-validation method

Holdout method employing stratified random sampling approach still heads into issues in certain specific situations. Especially, the smaller data sets may have thechallenge to divide the data of some of the classes proportionally amongst training and test data sets. A special variant of holdout method, called repeated holdout, is sometimes employed to ensure the randomness of the composed data sets. In repeated holdout, several random holdouts are used to measure the model performance. In the end, the average of all performances is taken. As multiple holdouts have been drawn, the training and test data (and also validation data, in case it is drawn) are more likely to contain representative data from all classes and resemble the original input data closely. This process of repeated holdout is the basis of *k*-fold cross-validation technique. In *k*-fold cross-validation, the data set is divided into *k*-completely distinct or non-overlapping random partitions called folds. Figure 3.2 depicts an overall approach for *k*-fold cross-validation.

The value of '*k*' in *k*-fold cross-validation can be set toany number. However, there are two approaches which are extremely popular:

1. 10-fold cross-validation (10-fold CV)
2. Leave-one-out cross-validation (LOOCV)

10-fold cross-validation is by far the most popular approach. In this approach, for each of the 10-folds, eachcomprising of approximately 10% of the data, one of the folds is used as the test data for validating model performance trained based on the remaining 9 folds (or 90% of the data). This is repeated 10 times, once for eachof the 10 folds being used as the test data and the remaining folds as the training data.The average performance across all folds is being reported. Figure 3.3depicts the detailed approach of selecting the '*k*' folds in *k*-fold cross-validation. As can be observed in the figure, each of the circles resembles a record in the input data set whereas the different colors indicate the different classes that the records belong to.The entire data set is broken into '*k*' folds – out of which one fold is selected ineach iteration as the test data set. The fold selected as test data set in each of the '*k*' iterations is different. Also, note that though in figure 3.3 the circles resemble the records in the input data set, the contiguous circles represented as folds do not mean that they are subsequent records in the data set. This is more a virtual representation and not a physical representation. As already mentioned, the records in a fold are drawn by using random sampling technique.

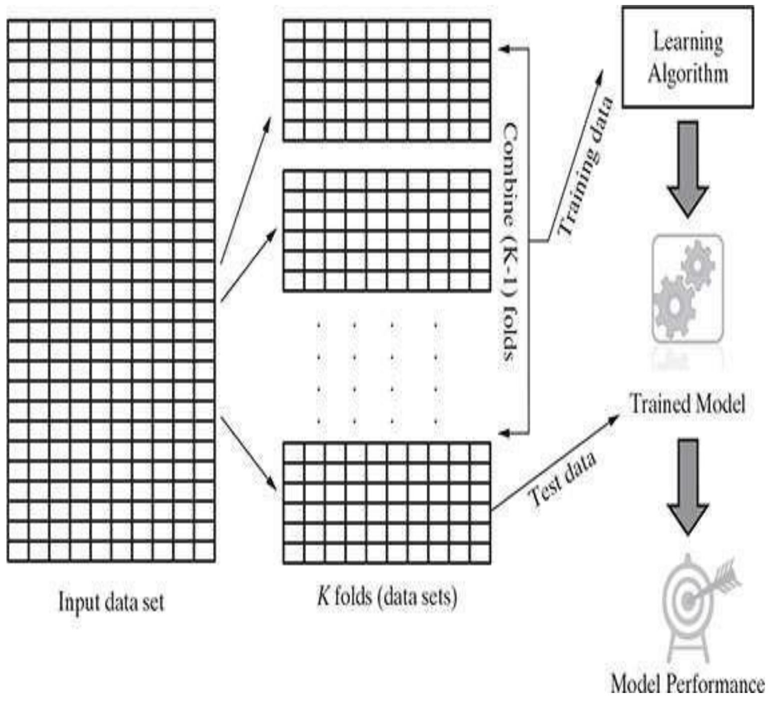


FIG. 3.2 Overall approach for *K*-fold cross-validation

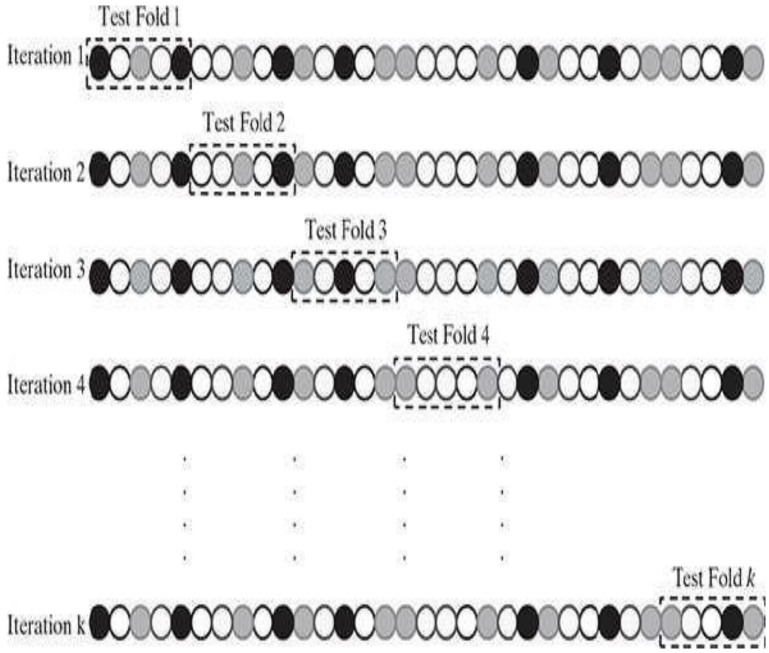


FIG. 3.3 Detailed approach for fold selection

Leave-one-out cross-validation (LOOCV) is an extremecase of *k*-fold cross-validation using one record or data instance at a time as a test data. This is done to maximizethe count of data used to train the model. It is obvious that the number of iterations for which it has to be run is equal to the total number of data in the input data set.

Hence, obviously, it is computationally very expensiveand not used much in practice.

Bootstrap sampling

Bootstrap sampling or simply bootstrapping is a popular way to identify training and test data sets from the input data set. It uses the technique of Simple Random Sampling with Replacement (SRSWR), which is a well- known technique in sampling theory for drawing randomsamples. We have seen earlier that *k*-fold cross- validation divides the data into separate partitions – say 10 partitions in case of 10-fold cross-validation. Then it uses data instances from partition as test data and the remaining partitions as training data. Unlike this approach adopted in case of *k*-fold cross- validation, bootstrapping randomly picks data instances from the input data set, with the possibility of the same data instance to be picked multiple times. This essentially means that from the input data set having '*n*' data instances, bootstrapping can create one or more training data sets having '*n*' data instances, some of the data instances being repeated multiple times. Figure 3.4 briefly presents the approach followed in bootstrap sampling.

This technique is particularly useful in case of input data sets of small size, i.e. having very less number of data instances.

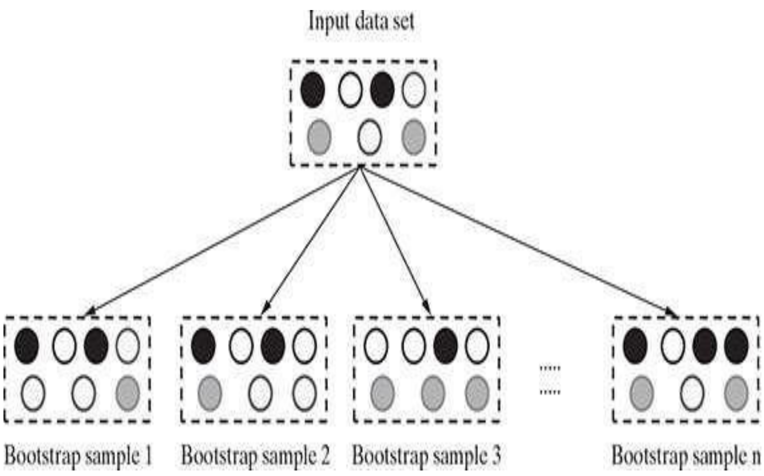
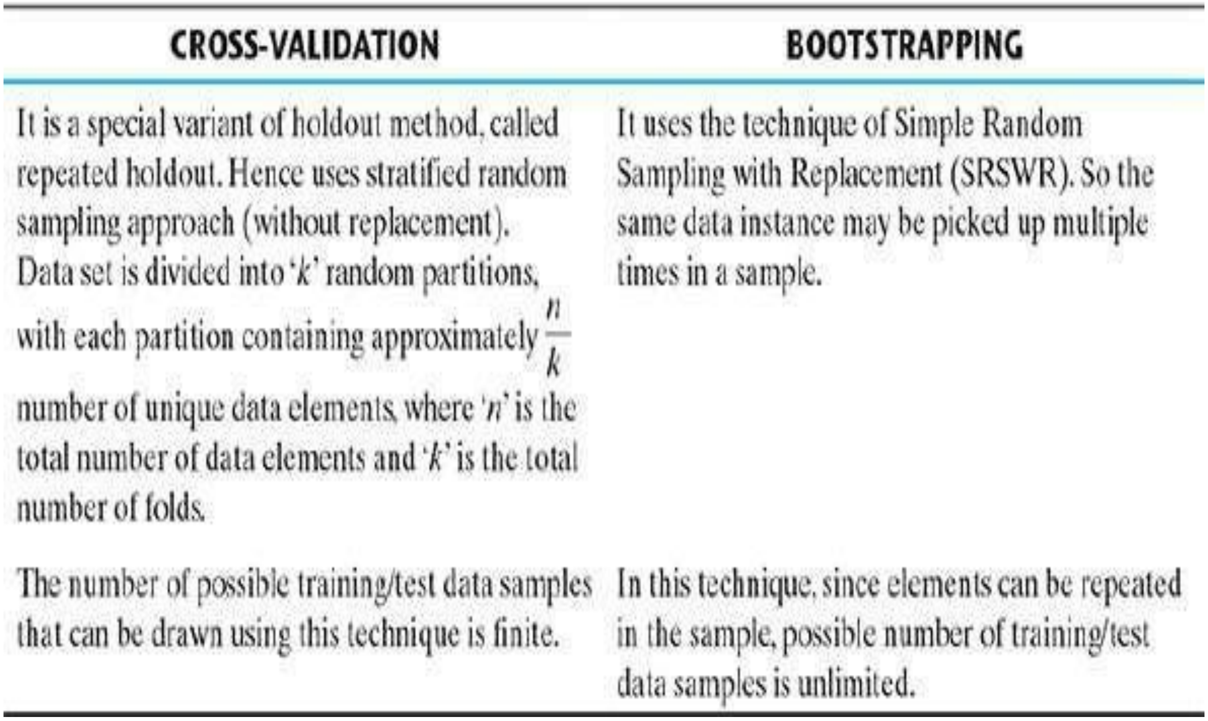


FIG. 3.4 Bootstrap sampling



CROSS-VALIDATION	BOOTSTRAPPING
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.	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.
The number of possible training/test data samples that can be drawn using this technique is finite.	In this technique, since elements can be repeated in the sample, possible number of training/test data samples is unlimited.

Lazy vs. Eager learner

Eager learning follows the general principles of machine learning – it tries to construct a generalized, input- independent target function during the model training phase. It follows the typical steps of machine learning, i.e. abstraction and generalization and comes up with a trained model at the end of the learning phase. Hence, when the test data comes in for classification, the eager learner is ready with the model and doesn't need to referback to the training data. Eager learners take more time in the learning phase than the lazy learners. Some of thealgorithms which adopt eager learning approach include

Some of thealgorithms which adopt eager learning approach include Decision Tree, Support Vector Machine, Neural Network,etc.

Lazy learning, on the other hand, completely skips the abstraction and generalization processes, as explained incontext of a typical machine learning process. In that respect, strictly speaking, lazy learner doesn't 'learn' anything. It uses the training data in exact, and uses the knowledge to classify the unlabelled test data. Since lazy learning uses training data as-is, it is also known as rote learning (i.e. memorization technique based on repetition). Due to its heavy dependency on the given training data instance, it is also known as instance learning. They are also called non-parametric learning. Lazy learners take very little time in training because notmuch of training actually happens. However, it takes quite some time in classification as for each tuple of test data, a comparison-based assignment of label happens. One of the most popular algorithm for lazy learning is *k*- nearest neighbor.

Note:

Parametric learning models have finite number of parameters. In case of non-parametric models, quitecontradicting to its name, the number of parametersis potentially infinite.

Models such as Linear Regression and Support Vector Machine, since the coefficients form the learning parameters, they are fixed in size. Hence, these models are clubbed as parametric.On the otherhand, in case of models such as *k*-Nearest Neighbor (*k*NN) and decision tree, number of parameters grows with the size of the training data. Hence, they are considered as non-parametric learning models.

MODEL REPRESENTATION AND INTERPRETABILITY

We have already seen that the goal of supervised machine learning is to learn or derive a target function which can best determine the target variable from the set of input variables. A key consideration in learning the target function from the training data is the extent of generalization. This is because the input data is just a limited, specific view and the new, unknown data in the test data set may be differing quite a bit from the trainingdata.

Fitness of a target function approximated by a learning algorithm determines how correctly it is able to classify a set of data it has never seen.

Underfitting

If the target function is kept too simple, it may not be able to capture the essential nuances and represent theunderlying data well. A typical case of underfitting mayoccur when trying to represent a non-linear data with alinear model as demonstrated by both cases of underfitting shown in figure 3.5. Many times underfitting happens due to unavailability of sufficient training data. Underfitting results in both poor performance with training data as well as poor generalization to test data. Underfitting can be avoidedby

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

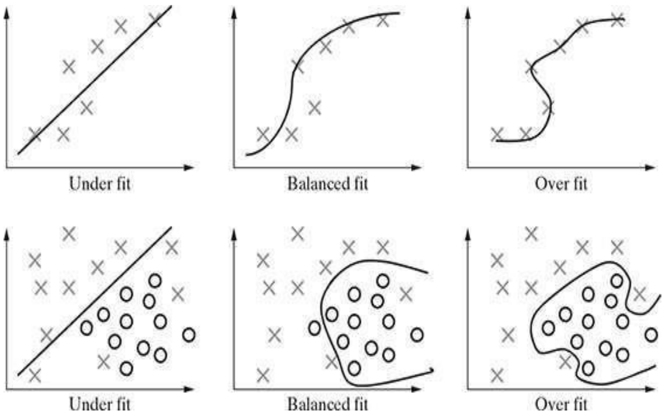


FIG. 3.5 Underfitting and Overfitting of models

Overfitting

Overfitting refers to a situation where the model has been designed in such a way that it emulates the trainingdata too closely. In such a case, any specific deviation in the training data, like noise or outliers, gets embedded inthe model. It adversely impacts the performance of the model on the test data. Overfitting, in many cases, occur as a result of trying to fit an excessively complex model to closely match the training data. This is represented with a sample data set in figure 3.5 . The target function, in these cases, tries to make sure all training data points are correctly partitioned by the decision boundary.

However, more often than not, this exact nature is not replicated in the unknown test data set. Hence, the targetfunction results in wrong classification in the test data set. Overfitting results in good performance with trainingdata set, but poor generalization and hence poor performance with test data set. Overfitting can be avoided by

1. using re-sampling techniques like *k*-fold cross validation
2. hold back of a validation data set
3. remove the nodes which have little or no predictive power forthe given machine learning problem.

Both underfitting and overfitting result in poorclassification quality which is reflected by low classification accuracy.

Bias – variance trade-off

In supervised learning, the class value assigned by the learning model built based on the training data may differ from the actual class value. This error in learning can be of two types – errors due to 'bias' and error due to'variance'. Let's try to understand each of them in details.

Errors due to 'Bias'

Errors due to bias arise from simplifying assumptions made by the model to make the target function less complex or easier to learn. In short, it is due to underfitting of the model. Parametric models generally have high bias making them easier to understand/interpret and faster to learn. These algorithms have a poor performance on data sets, which are complex in nature and do not align with the simplifying assumptions made by the algorithm. Underfitting results in high bias.

Errors due to 'Variance'

Errors due to variance occur from difference in training data sets used to train the model. Different training data sets (randomly sampled from the input data set) are usedto train the model. Ideally the difference in the data sets should not be significant and the model trained using different training data sets should not be too different.

However, in case of overfitting, since the model closelymatches the training data, even a small difference in training data gets magnified in the model.

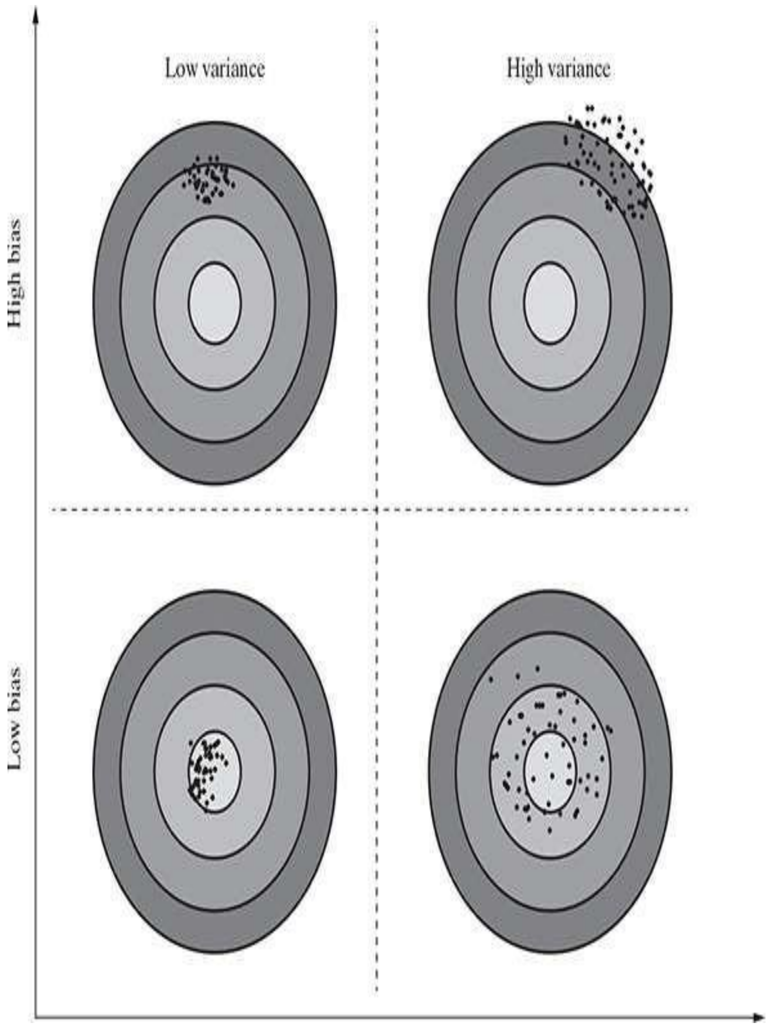


FIG. 3.6 Bias-variance trade-off

So, the problems in training a model can either happenbecause either (a) the model is too simple and hence failsto interpret the data grossly or (b) the model is extremelycomplex and magnifies even small differences in the training data.

As is quite understandable:

- Increasing the bias will decrease the
- variance, and Increasing the variance will decrease the bias

On one hand, parametric algorithms are generally seento demonstrate high bias but low variance. On the other hand, non-parametric algorithms demonstrate low bias and high variance.

As can be observed in Figure 3.6 , the best solution is to have a model with low bias as well as low variance. However, that may not be possible in reality. Hence, the goal of supervised machine learning is to achieve a balance between bias and variance. The learning algorithm chosen and the user parameters which can be configured helps in striking a trade-off between bias and variance. For example, in a popular supervised algorithm k-Nearest Neighbors or kNN, the user configurable parameter 'k' can be used to do a trade-off between bias and variance. In one hand, when the value of 'k' is decreased, the model becomes simpler to fit and bias increases. On the other hand, when the value of ‘k’ is increased, the variance increases.

EVALUATING PERFORMANCE OF A MODEL

Supervised learning – classification

In supervised learning, one major task is classification. The responsibility of the classification model is to assign class label to the target feature based on the value of the predictor features. For example, in the problem of predicting the win/loss in a cricket match, the classifier will assign a class value win/loss to target feature based on the values of other features like whether the team wonthe toss, number of spinners in the team, number of wins the team had in the tournament, etc. To evaluate the performance of the model, the number of correct classifications or predictions made by the model has to be recorded. A classification is said to be correct if, say for example in the given problem, it has been predictedby the model that the team will win and it has actually won.

Based on the number of correct and incorrect classifications or predictions made by a model, the accuracy of the model is calculated. If 99 out of 100 timesthe model has classified correctly, e.g. if in 99 out of 100 games what the model has predicted is same as what the outcome has been, then the model accuracy is said to be 99%. However, it is quite relative to say whether a model has performed well just by looking at the accuracy value. For example, 99% accuracy in case of a sports win predictor model may be reasonably good but the same number may not be acceptable as a good threshold when the learning problem deals with predicting a critical illness. In this case, even the 1% incorrect prediction may lead to loss of many lives. So the model performance needs to be evaluated in light of the learning problem in question. Also, in certain cases, erring on the side of caution may be preferred at the cost of overall accuracy. For that reason, we need to look more closely at the model accuracy and also at the same time look at other measures of performance of a model like sensitivity, specificity, precision, etc. So, let’s start with looking at model accuracy more closely. And let’s try to understand it with an example.

There are four possibilities with regards to the cricketmatch win/loss prediction:

1. the model predicted win and the team won
2. the model predicted win and the team lost
3. the model predicted loss and the team won
4. the model predicted loss and the team lost

In this problem, the obvious class of interest is ‘win’.

The first case, i.e. the model predicted win and the team won is a case where the model has correctly classified data instances as the class of interest. Thesecases are referred as True Positive (TP) cases.

The second case, i.e. the model predicted win and the team lost is a case where the model incorrectly classified data instances as the class of interest. These cases arereferred as False Positive (FP) cases.

The third case, i.e. the model predicted loss and theteam won is a case where the model has incorrectly classified as not the class of interest. These cases are referred as False Negative (FN) cases.

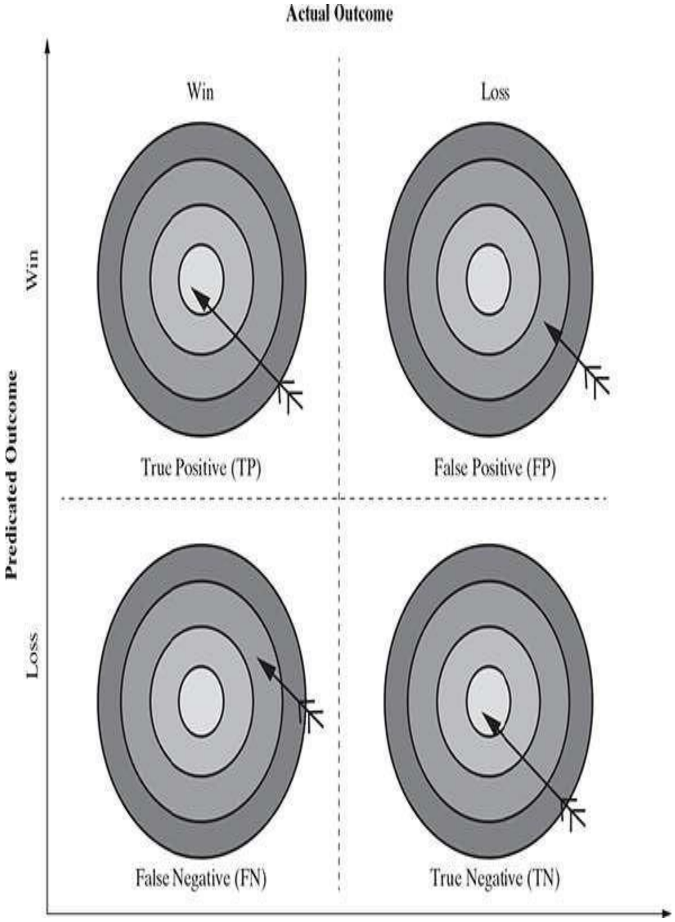


FIG. 3.7 Details of model classification

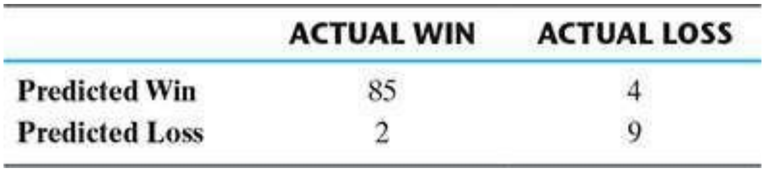
The fourth case, i.e. the model predicted loss and the team lost is a case where the model has correctly classified as not the class of interest. These cases are referred as True Negative (TN) cases. All these four casesare depicted in Figure 3.7 .

For any classification model, **model accuracy** is given by total number of correct classifications (either asthe class of interest, i.e. True Positive or as not the class of interest, i.e. True Negative) divided by total number of classifications done.

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

A matrix containing correct and incorrect predictions in the form of TPs, FPs, FNs and TNs is known as **confusion matrix**. The win/loss prediction of cricket match has two classes of interest – win and loss. For thatreason it will generate a 2 × 2 confusion matrix. For a classification problem involving three classes, the confusion matrix would be 3 × 3, etc.

Let’s assume the confusion matrix of the win/lossprediction 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.

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

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

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

In context of the above confusion matrix,

$$\text{Error rate} = \frac{\text{FP} + \text{FN}}{\text{TP} + \text{FP} + \text{FN} + \text{TN}} = \frac{4 + 2}{85 + 4 + 2 + 9} = \frac{6}{100} = 6\%$$

$$= 1\text{–Model accuracy}$$

Sometimes, correct prediction, both TPs as well as TNs, may happen by mere coincidence. Since these occurrences boost model accuracy, ideally it should not happen. **Kappa** value of a model indicates the adjustedthe 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

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

$\text{P}(\text{p}_r)$ = Proportion of expected agreement between actual and predicted data both in case of class of interest as well as the other classes

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

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.

$$\therefore \text{P(a)} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{FN} + \text{TN}} = \frac{85 + 9}{85 + 4 + 2 + 9} = \frac{94}{100} = 0.94$$

$$\text{P}(\text{p}_r) = \frac{85 + 4}{85 + 4 + 2 + 9} \times \frac{85 + 2}{85 + 4 + 2 + 9} + \frac{2 + 9}{85 + 4 + 2 + 9} \times \frac{4 + 9}{85 + 4 + 2 + 9}$$

$$= \frac{89}{100} \times \frac{87}{100} + \frac{11}{100} \times \frac{13}{100} = 0.89 \times 0.87 + 0.11 \times 0.13 = 0.7886$$

$$\therefore \text{k} = \frac{0.94 - 0.7886}{1 - 0.7886} = 0.7162$$

Note:

Kappa value can be 1 at the maximum, which represents perfect agreement between model’sprediction and actual values.

As discussed earlier, in certain learning problems it is critical to have extremely low number of FN cases, if needed, at the cost of a conservative classification model.Though it is a clear case of misclassification and will impact model accuracy adversely, it is still required as missing each class of interest may have serious consequence. This happens more in problems from medical domains like disease prediction problem. For example, if a tumor is malignant but wrongly classified as benign by the classifier, then the repercussion of such misclassification is fatal. It does not matter if higher number of tumours which are benign are wrongly classified as malignant. In these problems there are somemeasures of model performance which are more important than accuracy. Two such critical measurements are sensitivity and specificity of the model.

The **sensitivity** of a model measures the proportionof TP examples or positive cases which were correctly classified. It is measured as

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

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

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

So, again taking the example of the malignancy prediction of tumours, class of interest is ‘malignant’. Sensitivity measure gives the proportion of tumours which are actually malignant and have been predicted asmalignant. It is quite obvious that for such problems themost critical measure of the performance of a good model is sensitivity. A high value of sensitivity is more desirable than a high value of accuracy.

Specificity is also another good measure to indicate agood balance of a model being excessively conservative or excessively aggressive. Specificity of a model measuresthe proportion of negative examples which have been correctly classified. In the context, of malignancy prediction of tumours, specificity gives the proportion of benign tumours which have been correctly classified. In the context of the above confusion matrix for the cricket match win prediction problem,

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

A higher value of specificity will indicate a better model performance. However, it is quite understandablethat a conservative approach to reduce False Negatives might actually push up the number of FPs. Reason for this is that the model, in order to reduce FNs, is going to classify more tumours as malignant. So the chance that benign tumours will be classified as malignant or FPs will increase.

There are two other performance measures of a supervised learning model which are similar to sensitivity and specificity. These are **precision** and **recall**. While precision gives the proportion of positivepredictions which are truly positive, recall gives the proportion of TP cases over all actually positive cases.

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

Precision indicates the reliability of a model in predicting a class of interest. When the model is relatedto win / loss prediction of cricket, precision indicates how often it predicts the win correctly. In context of theabove confusion matrix for the cricket match win prediction problem,

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} = \frac{85}{85 + 4} = \frac{85}{89} = 95.5\%$$

It is quite understandable that a model with higherprecision is perceived to be more reliable.

Recall indicates the proportion of correct prediction ofpositives to the total number of positives. In case of win/loss prediction of cricket, recall resembles what proportion of the total wins were predicted correctly.

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

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

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

F-measure

F-measure is another measure of model performance which combines the precision and recall. It takes the harmonic mean of precision and recall as calculated as

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

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

$$F\text{-measure} = \frac{2 \times 0.955 \times 0.977}{0.955 + 0.977} = \frac{1.866}{1.932} = 96.6\%$$

As a combination of multiple measures into one, *F*- score gives the right measure using which performance of different models can be compared. However, one assumption the calculation is based on is that precision and recall have equal weight, which may not always be true in reality. In certain problems, the disease prediction problems, e.g., precision may be given far more weightage. In that case, different weightages may be assigned to precision and recall. However, there may be a serious dilemma regarding what value to be adopted for each and what is the basis for the specific value adopted.

Receiver operating characteristic (ROC) curves

As we have seen till now, though accuracy is the most popular measure, there are quite a number of other measures to evaluate the performance of a supervised learning model. However, visualization is an easier and more effective way to understand the model performance. It also helps in comparing the efficiency oftwo models.

Receiver Operating Characteristic (ROC) curve helps in visualizing the performance of a classification model. It shows the efficiency of a model in the detection of true positives while avoiding the occurrence of false positives. To refresh our memory, true positives are those cases where the model has correctly classified data instances asthe class of interest. For example, the model has correctly classified the tumours as malignant, in case of a tumour malignancy prediction problem. On the other hand, FPs are those cases where the model incorrectly classified data instances as the class of interest. Using thesame example, in this case, the model has incorrectly classified the tumours as malignant, i.e. tumours whichare actually benign have been classified as malignant.

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

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

In the ROC curve, the FP rate is plotted (in the horizontal axis) against true positive rate (in the vertical axis) at different classification thresholds. If we assume a lower value of classification threshold, the model classifies more items as positive. Hence, the values of both False Positives and True Positives increase. The area under curve (AUC) value, as shown in figure 3.8a_, isthe area of the two-dimensional space under the curve extending from (0, 0) to (1, 1), where each point on the curve gives a set of true and false positive values at a specific classification threshold. This curve gives an indication of the predictive quality of a model. AUC valueranges from 0 to 1, with an AUC of less than 0.5 indicating that the classifier has no predictive ability.

Figure 3.8b shows the curves of two classifiers – classifier 1 and classifier 2. Quite obviously, the AUC ofclassifier 1 is more than the AUC of classifier 2. So, we can draw the inference that classifier 1 is better than classifier 2.

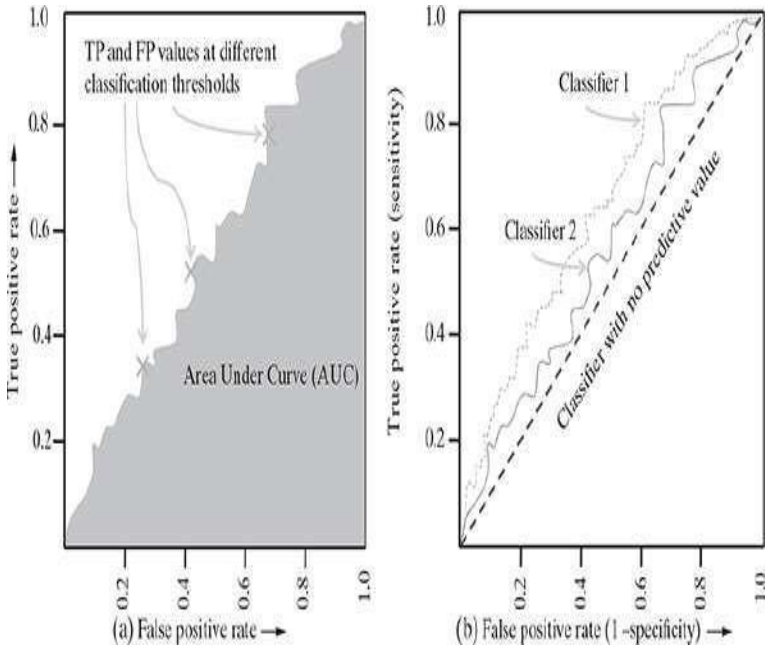


FIG. 3.8 ROC curve

A quick indicative interpretation of the predictivevalues from 0.5 to 1.0 is given below:

- – 0.6 ➔ Almost no predictive ability
- – 0.7 ➔ Weak predictive ability
- – 0.8 ➔ Fair predictive ability
- – 0.9 ➔ Good predictive ability
- – 1.0 ➔ Excellent predictive ability

Supervised learning – regression

A well-fitted regression model churns out predicted values close to actual values. Hence, a regression model which ensures that the difference between predicted andactual values is low can be considered as a good model. Figure 3.9 represents a very simple problem of real estate value prediction solved using linear regression model. If ‘area’ is the predictor variable (say *x*) and ‘value’ is the target variable (say *y*), the linear regression model can be represented in the form:

$$y = \alpha + \beta x$$

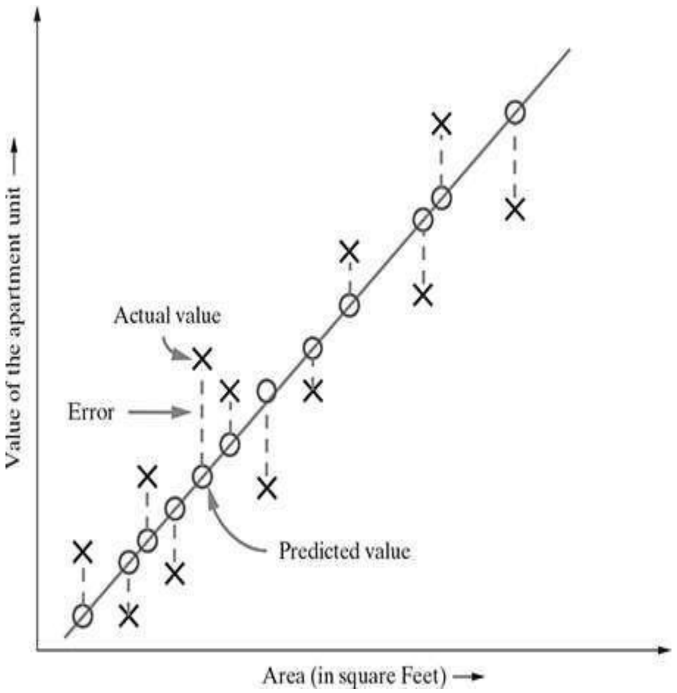


FIG. 3.9 Error – Predicted vs. actual value

For a certain value of *x*, say $\hat{x}$, the value of y is predicted as $\hat{y}$ whereas the actual value of *y* is *Y* (say). The distance between the actual value and the fitted orpredicted value, i.e. $\hat{y}$ is known as **residual**. The regression model can be considered to be fitted well if the difference between actual and predicted value, i.e. the residual value is less.

R-squared is a good measure to evaluate the model fitness. It is also known as the coefficient of determination, or for multiple regression, the coefficientof multiple determination. The R-squared value lies between 0 to 1 (0%–100%) with a larger value representing a better fit. It is calculated as:

$$R^2 = \frac{SST - SSE}{SST}$$

Sum of Squares Total (SST) = squared differences ofeach observation from the overall mean=

$$\sum_{i=1}^{n} (y_i - \bar{y})^2$$

where $\bar{y}$ is the mean.

Sum of Squared Errors (SSE) (of prediction) = sum of the squared residuals=

$$\sum_{i=1}^{n} (Y_i - \hat{y})^2$$

Where ^yi is thepredicted value of *yi* and *Yi* is the actual value of *yi*.

Unsupervised learning – clustering

Clustering algorithms try to reveal natural groupings amongst the data sets. However, it is quite tricky to evaluate the performance of a clustering algorithm. Clustering, by nature, is very subjective and whether thecluster is good or bad is open for interpretations. It was noted, ‘clustering is in the eye of the beholder’. This stems from the two inherent challenges which lie in the process of clustering:

1.It is generally not known how many clusters can be formulatedfrom a particular data set. It is completely open-ended in mostcases and provided as a user input to a clustering algorithm.

2.Even if the number of clusters is given, the same number of clusters can be formed with different groups of data instances.

In a more objective way, it can be said that a clusteringalgorithm is successful if the clusters identified using thealgorithm is able to achieve the right results in the overall problem domain. For example, if clustering is applied for identifying customer segments for a marketing campaign of a new product launch, the clustering can be considered successful only if the marketing campaign ends with a success,i.e. it is able to create the right brand recognition resulting in steady revenue from new product sales. However, there are couple of popular approaches which are adopted for cluster quality evaluation.

1.Internal evaluation

In this approach, the cluster is assessed based on the underlying datathat was clustered. The internal evaluation methods generally measure cluster quality based on homogeneity of data belonging to the same cluster and heterogeneity of data belonging to different clusters. The homogeneity/heterogeneity is decided by some similarity measure. For example, **silhouette coefficient**, which is one of the most popular internal evaluation methods, uses distance (Euclidean or Manhattan distances most commonly used) between data elements as a similarity measure. The value of silhouette width ranges between –1 and +1, with a high value indicating high intra- cluster homogeneity and inter-cluster heterogeneity.

For a data set clustered into ‘k’ clusters, silhouette width is calculated as:

$$\text{Silhouette width} = \frac{b(i) - a(i)}{\max\{a(i),\ b(i)\}}$$

$a(i)$ is the average distance between the i th data instance and all other data instances belonging to the same cluster and $b(i)$ is the lowest average distance between the i-the data instance and data instances of all other clusters.

Let’s try to understand this in context of the example depicted in figure 3.10. There are four clusters namely cluster 1, 2, 3, and 4. Let’sconsider an arbitrary data element ‘i’ in cluster 1, resembled by the asterisk. $a(i)$ is the average of the distances $ai1$, $ai2$, …, $ain1$ of the different data elements from the i th data element in cluster 1, assuming there are n_1 data elements in cluster 1. Mathematically,

$$a(i) = \frac{a_{i1} + a_{i2} + \ldots + a_{in_1}}{n_1}$$

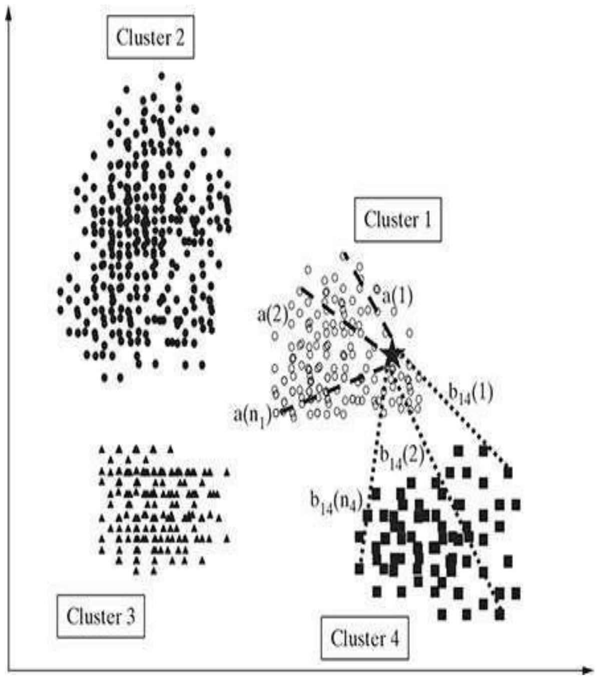


FIG. 3.10 Silhouette width calculation

In the same way, let's calculate the distance of an arbitrary data element '*i*' in cluster 1 with the different data elements from another cluster, say cluster 4 and take an average of all those distances. Hence,

$$b_{14}\,(\text{average}) = \frac{b_{14}(1) + b_{14}(2) + \ \ldots\ + b_{14}(n_4)}{(n_4)}$$

where n_4 is the total number of elements in cluster 4. In the same way, we can calculate the values of *b*12 (average) and *b*13 (average). b(i) is the minimum of all these values. Hence, we can say that,

b(i) = minimum [*b*12(average), *b*13(average), *b*14(average)]

2.External evaluation

In this approach, class label is known for the data set subjected to clustering. However, quite obviously, the known class labels are not a part of the data used in clustering. The cluster algorithm is assessed based on how close the results are compared to those known class labels. For example, **purity** is one of the most popular measures of cluster algorithms – evaluates the extent to which clusters contain a single class.

For a data set having 'n' data instances and 'c' known class labels which generates 'k' clusters, purity is measured as:

$$\text{Purity} = \frac{1}{n}\sum_{k}\max(k \cap c)$$

IMPROVING PERFORMANCE OF A MODEL

Now we have almost reached the end of the journey of building learning models. We have got some idea about what modelling is, how to approach about it to solve a learning problem and how to measure the success of our model. Now comes a million dollar question. Can we improve the performance of our model? If so, then what are the levers for improving the performance? In fact, even before that comes the question of model selection – which model should be selected for which machine learning task? We have already discussed earlier that the model selection is done one several aspects:

1.Type of learning the task in hand, i.e. supervised orunsupervised

2.Type of the data, i.e. categorical or numeric

3.Sometimes on the problem domain

4.Above all, experience in working with different models to solveproblems of diverse domains

So, assuming that the model selection is done, what are the different avenues to improve the performance ofmodels?

One effective way to improve model performance is bytuning model parameter. **Model parameter tuning** is the process of adjusting the model fitting options. For example, in the popular classification model *k*-Nearest Neighbour (*k*NN), using different values of '*k*' or the number of nearest neighbours to be considered, the model can be tuned. In the same way, a number of hidden layers can be adjusted to tune the performance inneural networks model. Most machine learning models have at least one parameter which can be tuned.

As an alternate approach of increasing the performance of one model, several models may be combined together. The models in such combination are complimentary to each other, i.e. one model may learn one type data sets well while struggle with another type of data set. Another model may perform well with the data set which the first one struggled with. This approachof combining different models with diverse strengths is known as **ensemble** (depicted in Figure 3.11).

Ensemble helps in averaging out biases of the differentunderlying models and also reducing the variance.

Ensemble methods combine weaker learners to create stronger ones. A performance boost can be expected evenif models are built as usual and then ensembled.

Following are the typical steps in ensemble process:

- Build a number of models based on the training data
- For diversifying the models generated, the training data subset canbe varied using the allocation function. Sampling techniques like bootstrapping may be used to generate unique training data sets.
- Alternatively, the same training data may be used but the models combined are quite varying, e.g, SVM, neural network, *k*NN, etc. The outputs from the different models are combined using a combination function.
- A very simple strategy of combining, say in case of a prediction task using ensemble, can be majority voting ofthe different models combined. For example, 3 out of 5 classes predict 'win' and 2 predict 'loss' – then the final outcome of the ensemble using majority vote would be a 'win'.

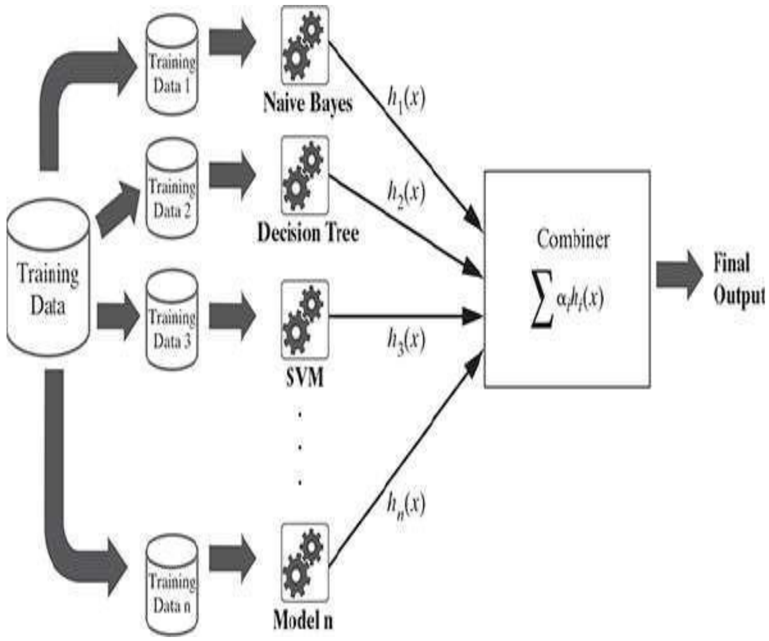


FIG. 3.11 Ensemble

One of the earliest and most popular ensemble modelsis **bootstrap aggregating** or **bagging**. Bagging uses bootstrap sampling method (refer section 3.3.3) to generate multiple training data sets. These training data sets are used to generate (or train) a set of models using the same learning algorithm. Then the outcomes of the models are combined by majority voting (classification) or by average (regression). Bagging is a very simple ensemble technique which can perform really well for unstable learners like a decision tree, in which a slight change in data can impact the outcome of a model significantly.

Just like bagging, **boosting** is another key ensemble- based technique. In this type of ensemble, weaker learning models are trained on resampled data and the outcomes are combined using a weighted voting approach based on the performance of different models. **Adaptive boosting** or **AdaBoost** is a special variant ofboosting algorithm. It is based on the idea of generating weak learners and slowly learning.

Random forest is another ensemble-based technique. It is an ensemble of decision trees – hence the name random forest to indicate a forest of decision trees.It has been discussed in more details in chapter 7.

In this chapter, you have been introduced to the crux of machine learning, i.e. modelling. Thorough understanding of the technical aspects elaborated in this chapter is extremely crucial for the success of any machine learning project. For example, the first dilemma comes about which model to select. Again, in case of supervised learning, how can we deal with the unavailability of sufficient training data. In the same way, once the model is trained in case of supervised learning or the grouping is done in case of clustering, how we can understand whether the model training (for supervised) or grouping done (for unsupervised) is good or bad. All these and more have been addressed as a part of this chapter.