Unit I

1 Introduction to Machine Learning

Syllabus

Introduction : *What is Machine Learning, Definitions and Real-life applications, Comparison of Machine learning with traditional programming, ML vs AI vs Data Science.*

Learning Paradigms : *Learning Tasks - Descriptive and Predictive Tasks, Supervised, Unsupervised, Semi-supervised and Reinforcement Learnings.*

Models of Machine learning : *Geometric model, Probabilistic Models, Logical Models, Grouping and grading models, Parametric and non-parametric models.*

Feature Transformation : *Dimensionality reduction techniques - PCA and LDA.*

Contents

1.1 What is Machine Learning ?
1.2 Real-life Applications
1.3 Comparison of Machine Learning with Traditional Programming
1.4 Learning Paradigms
1.5 Supervised Learning
1.6 Unsupervised Learning
1.7 Semi-supervised Learning
1.8 Reinforcement Learnings
1.9 Models of Machine Learning
1.10 Grouping and Grading Models
1.11 Parametric Models
1.12 Non-parametric Models
1.13 Feature
1.14 PCA
1.15 LDA
1.16 Application of Machine Learning

1.1 What is Machine Learning ?

- Machine Learning (ML) is a sub-field of Artificial Intelligence (AI) which concerns with developing computational theories of learning and building learning machines.
- **Learning** is a phenomenon and process which has manifestations of various aspects. Learning process includes gaining of new symbolic knowledge and development of cognitive skills through instruction and practice. It is also discovery of new facts and theories through observation and experiment.
- **Machine Learning Definition :** 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.
- Machine learning is programming computers to optimize a performance criterion using example data or past experience. Application of machine learning methods to large databases is called **data mining.**
- It is very hard to write programs that solve problems like recognizing a human face. We do not know what program to write because we don't know how our brain does it. Instead of writing a program by hand, it is possible to collect lots of examples that specify the correct output for a given input.
- A machine learning algorithm then takes these examples and produces a program that does the job. The program produced by the learning algorithm may look very different from a typical hand-written program. It may contain millions of numbers. If we do it right, the program works for new cases as well as the ones we trained it on.
- Main goal of machine learning is to devise learning algorithms that do the learning automatically without human intervention or assistance. The machine learning paradigm can be viewed as "programming by example." Another goal is to develop computational models of human learning process and perform computer simulations.
- The goal of machine learning is to build computer systems that can adapt and learn from their experience.
- Algorithm is used to solve a problem on computer. An algorithm is a sequence of instruction. It should carry out to transform the input to output. For example, for addition of four numbers is carried out by giving four number as input to the algorithm and output is sum of all four numbers. For the same task, there may be various algorithms. It is interested to find the most efficient one, requiring the least number of instructions or memory or both.
- For some tasks, however, we do not have an algorithm.

Why is Machine Learning Important ?

- Machine learning algorithms can figure out how to perform important tasks by generalizing from examples.
- Machine Learning provides business insight and intelligence. Decision makers are provided with greater insights into their organizations. This adaptive technology is being used by global enterprises to gain a competitive edge.
- Machine learning algorithms discover the relationships between the variables of a system (input, output and hidden) from direct samples of the system.
- **Following are some of the reasons :**
  1. Some tasks cannot be defined well, except by examples. For example: recognizing people.
  2. Relationships and correlations can be hidden within large amounts of data. To solve these problems, machine learning and data mining may be able to find these relationships.
  3. Human designers often produce machines that do not work as well as desired in the environments in which they are used.
  4. The amount of knowledge available about certain tasks might be too large for explicit encoding by humans.
  5. Environments change time to time.
  6. New knowledge about tasks is constantly being discovered by humans.
- Machine learning also helps us find solutions of many problems in computer vision, speech recognition and robotics. Machine learning uses the theory of statistics in building mathematical models, because the core task is making inference from a sample.

How Machines Learn ?

- Machine learning typically follows three phases :
  1. **Training :** A training set of examples of correct behavior is analyzed and some representation of the newly learnt knowledge is stored. This is some form of rules.
  2. **Validation :** The rules are checked and, if necessary, additional training is given. Sometimes additional test data are used, but instead, a human expert may validate the rules, or some other automatic knowledge - based component may be used. The role of the tester is often called the opponent.
  3. **Application :** The rules are used in responding to some new situation.
- Fig. 1.1.1 shows phases of ML.

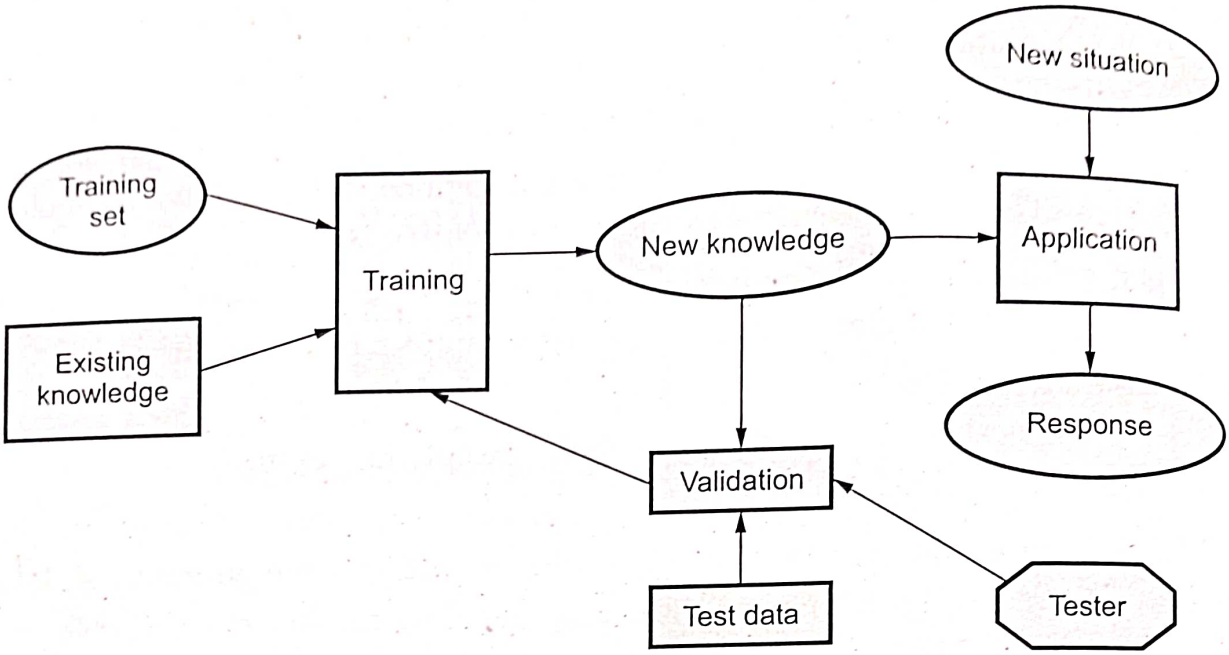


Fig. 1.1.1 Phases of ML

1.1.1 Why Machine Learning is Important ?

- Machine learning algorithms can figure out how to perform important tasks by generalizing from examples.
- Machine learning provides business insight and intelligence. Decision makers are provided with greater insights into their organizations. This adaptive technology is being used by global enterprises to gain a competitive edge.
- Machine learning algorithms discover the relationships between the variables of a system (input, output and hidden) from direct samples of the system.
- **Following are some of the reasons :**
  1. Some tasks cannot be defined well, except by examples. For example : Recognizing people.
  2. Relationships and correlations can be hidden within large amounts of data. To solve these problems, machine learning and data mining may be able to find these relationships.
  3. Human designers often produce machines that do not work as well as desired in the environments in which they are used.
  4. The amount of knowledge available about certain tasks might be too large for explicit encoding by humans.
  5. Environments change time to time.
  6. New knowledge about tasks is constantly being discovered by humans.

- Machine learning also helps us find solutions of many problems in computer vision, speech recognition and robotics. Machine learning uses the theory of statistics in building mathematical models, because the core task is making inference from a sample.
- Learning is used when :
  1. Human expertise does not exist (navigating on Mars),
  2. Humans are unable to explain their expertise (speech recognition)
  3. Solution changes in time (routing on a computer network)
  4. Solution needs to be adapted to particular cases (user biometrics)

1.1.2 Ingredients of Machine Learning

The ingredients of machine learning are as follows :

1. **Tasks :** The problems that can be solved with machine learning. A task is an abstract representation of a problem. The standard methodology in machine learning is to learn one task at a time. Large problems are broken into small, reasonably independent sub-problems that are learned separately and then recombined.
   - Predictive tasks perform inference on the current data in order to make predictions. Descriptive tasks characterize the general properties of the data in the database.
2. **Models :** The output of machine learning. Different models are geometric models, probabilistic models, logical models, grouping and grading.
   - The model-based approach seeks to create a modified solution tailored to each new application. Instead of having to transform your problem to fit some standard algorithm, in model-based machine learning you design the algorithm precisely to fit your problem.
   - Model is just made up of set of assumptions, expressed in a precise mathematical form. These assumptions include the number and types of variables in the problem domain, which variables affect each other and what the effect of changing one variable is on another variable.
   - Machine learning models are classified as : Geometric model, Probabilistic model and Logical model.
3. **Features :** The workhorses of machine learning. A good feature representation is central to achieving high performance in any machine learning task.
   - Feature extraction starts from an initial set of measured data and builds derived values intended to be informative, non redundant, facilitating the subsequent learning and generalization steps.

- Feature selection is a process that chooses a subset of features from the original features so that the feature space is optimally reduced according to a certain criterion.

Review Questions

1. *Justify the following*
   i) Predict the height of a person. Is it a regression task.
   ii) Find the gender of a person by analyzing his writing style. Is it a classification task ?
   iii) Filter out spam emails. Is it a example of unsupervised learning.
2. *What is machine learning ? Explain types of machine learning.*

1.2 Real-life Applications

- Examples of successful applications of machine learning :

Here are several examples :

1 **Optical character recognition :** Categorize images of handwritten characters by the letters represented.

2 **Face detection :** Find faces in images (or indicate if a face is present).

3 **Spam filtering :** Identify email messages as spam or non-spam topic spotting: categorize news articles (say) as to whether they are about politics, sports, entertainment, etc.

4 **Spoken language understanding :** Within the context of a limited domain, determine the meaning of something uttered by a speaker to the extent that it can be classified into one of a fixed set of categories.

1.2.1 Learning Associations

- Learning association is the process of developing insights into various associations between products. A good example is how seemingly unrelated products may reveal an association to one another when analyzed in relation to the buying behaviors of customers.
- This application of machine learning involves studying the association between the products people buy and is also known as basket analysis.
- If a buyer buys X, would they buy Y because of a relationship that can be identified between them ? Knowing these relationships could help in suggesting an associated product to the customer.
- For a higher likelihood of the customer buying it, it can also help in bundling products for a better package.

- This learning of associations between products by a machine is called learning associations. Once we find an association by examining a large amount of sales data, big data analysts can develop a rule to derive a probability test in learning a conditional probability.
- In finding an association rule, learning a conditional probability of the form P(Y | X) where Y is the product we would like to condition on X, which is the product or the set of products which we know that the customer has already purchased.
- So an example of an association rule can be {eggs, bacon} → {cheese} expressing, that customers who buy eggs and bacon also often buy cheese (to make ham-and-eggs).
- The two basic characteristics of an association rule are support and confidence.

1.2.2 Classification

- Classification is the process of placing each individual from the population under study in many classes. This is identified as independent variables.
- Classification helps analysts use measurements of an object to identify the category to which that object belongs. To establish an efficient rule, analysts use data. Data consists of many examples of objects with their correct classification.
- For example, before a bank decides to disburse a loan, it assesses customers on their ability to repay the loan. By considering factors such as customer's earning, age, savings, and financial history, we can do it. This information is taken from the past data of the loan. Hence, the seeker uses this data to create a relationship between customer attributes and related risks.

Face Recognition

- Face recognition task is effortlessly and every day we recognize our friends, relative and family members. We also recognition by looking at the photographs. In photographs, they are in different pose, hair styles, background light, makeup and without makeup.
- We do it subconsciously and cannot explain how we do it. Because we can't explain how we do it, we can't write an algorithm.
- Face has some structure. It is not a random collection of pixel. It is symmetric structure. It contains predefined components like nose, mouth, eye, ears. Every person face is a pattern composed of a particular combination of the features. By analyzing sample face images of a person, a learning program captures the pattern specific to that person and uses it to recognize if a new real face or new image belongs to this specific person or not.

- Machine learning algorithm creates an optimized model of the concept being learned based on data or past experience.
- In the case of face recognition, the input is an image, the classes are people to be recognized, and the learning program should learn to associate the face images to identities. This problem is more difficult than optical character recognition because there are more classes, input image is larger and a face is 3D and differences in pose and lighting cause significant changes in the image.

Medical diagnosis

- In medical diagnosis, the inputs are the relevant information about the patient and the classes are the illnesses. The inputs contain the age of patient's, gender, past medical history and current symptoms.
- Some tests may not have been applied to the patient, and thus these inputs would be missing. Tests take time, may be costly and may inconvience the patient so we do not want to apply them unless we believe that they will give us valuable information.
- In the case of a medical diagnosis, a wrong decision may lead to a wrong or no treatment, and in cases of doubt it is preferable that the classifier reject and defer decision to a human expert.

1.2.3 Regression

- Regression : Trying to predict a real value. For instance, predict the value of a stock tomorrow given its past performance. Or predict Alice's score on the machine learning final exam based on her homework scores.
- If the desired output consists of one or more continuous variables, then the task is called regression. An example of a regression problem would be the prediction of the yield in a chemical manufacturing process in which the inputs consist of the concentrations of reactants, the temperature, and the pressure.
- The goal of regression is to predict the value of one or more continuous target variables t given the value of a D-dimensional vector x of input variables.
- Navigation of a mobile robot is one of the examples of regression. An autonomous car, where the output is the angle by which the steering wheel should be turned at each time, to advance without hitting obstacles and deviating from the route.
- Inputs in such a case are provided by sensors on the car, for example, a video camera, GPS etc. Training data can be collected by monitoring and recording the actions of a human driver.

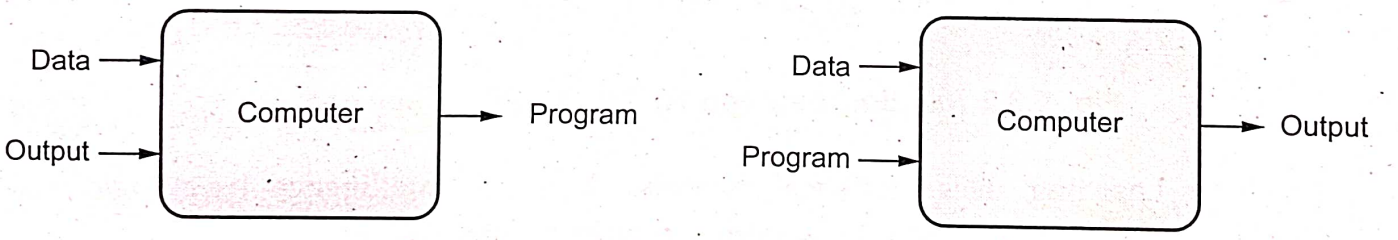
- In regression, we can use the principle of machine learning to optimize parameters and to cut the approximation error and calculate the closest possible outcome.

Review Questions

1. *Write mathematical form of the following :*
   i) Classification ii) Class probability estimation iii) Regression
   Which one out of these three is more precise ? Which one leads to overfitting ?

1.3 Comparison of Machine Learning with Traditional Programming

- Machine learning seeks to construct a model or logic for the problem by analyzing its input data and answers. In contrast, traditional programming is that programming aims to answer a problem using a predefined set of rules or logic.
- Machine learning is the ability of machines to automate a learning process. The input of this learning process is data and the output is a model. Through machine learning, a system can perform a learning function with the data it ingests and thus it becomes progressively better at said function.
- Traditional programming is a manual process. It requires a programmer to create the rules or logic of the program. We have to manually come up with the rules and feed it to the computer alongside input data. The machine then processes the given data according to the coded rules and comes up with answers as output.
- Fig 1.3.1 shows machine learning and traditional programming.



(a) Machine learning process (b) Traditional programming

Fig 1.3.1

- For projects that involve predicting output or identifying objects in images, **machine learning** has proven to be much more efficient. In traditional programming, the rule-based approach is preferred in situations where the problem is of an algorithmic manner and there are not so many parameters to consider when writing the logical rules.

- Machine Learning is a proven technique for helping to solve complex problems such as facial and voice recognition, recommendation systems, self-driving cars and email spam detection.

1.3.1 ML vs AI vs Data Science

- Artificial Intelligence (AI) is the broad concept of developing machines that can simulate human thinking, reasoning and behavior.
- Machine Learning (ML) is a subset of AI wherein computer systems learn from the environment and in turn, use these learnings to improve experiences and processes. All machine learning is AI, but not all AI is machine learning.
- Data Science is the processing, analysis and extraction of relevant assumptions from data. It's about finding hidden patterns in the data. A Data Scientist makes use of machine learning in order to predict future events.
- Fig. 1.3.2 shows relation between AI, ML and Data science.

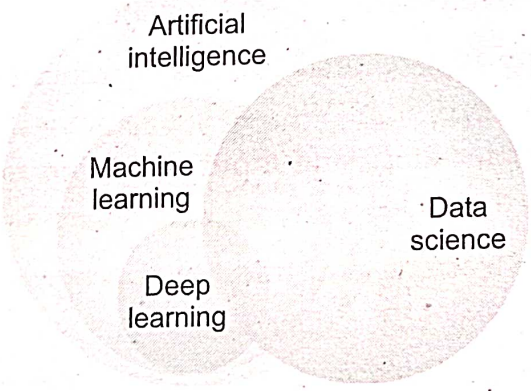


Fig. 1.3.2 Relation between AI, ML and Data science

- Machine Learning uses statistical models. Artificial Intelligence uses logic and decision trees. Data Science deals with structured data.
- Machine Learning : A form of analytics in which software programs learn about data and find patterns.
- AI : Development of computerized applications that simulate human intelligence and interaction.
- Data Science : The process of using advanced analytics to extract relevant information from data.

In Table Format :

Machine Learning	Artificial Intelligence	Data Science
Focuses on providing a means for algorithms and systems to learn from experience with data and use that experience to improve over time.	Focuses on giving machines cognitive and intellectual capabilities similar to those of humans.	Focuses on extracting information needles from data haystacks to aid in decision-making and planning.
Machine Learning uses statistical models.	Artificial Intelligence uses logic and decision trees.	Data Science deals with structured data.
A form of analytics in which software programs learn about data and find patterns.	Development of computerized applications that simulate human intelligence and interaction.	The process of using advanced analytics to extract relevant information from data.
Objective is to maximize accuracy.	Objective is to maximize the chance of success.	Objective is to extract actionable insights from the data.
ML can be done through supervised, unsupervised or reinforcement learning approaches.	AI encompasses a collection of intelligence concepts, including elements of perception, planning and prediction.	Uses statistics, mathematics, data wrangling, big data analytics, machine learning and various other methods to answer analytics questions.
ML is concerned with knowledge accumulation.	AI is concerned with knowledge dissemination and conscious machine actions.	Data science is all about data engineering.

1.4 Learning Paradigms

1.4.1 Descriptive Tasks

- Descriptive Analytics is the conventional form of business intelligence and data analysis, seeks to provide a depiction or "summary view" of facts and figures in an understandable format, to either inform or prepare data for further analysis.
- Two primary techniques are used for reporting past events : Data aggregation and data mining.
- It presents past data in an easily digestible format for the benefit of a wide business audience.
- A set of techniques for reviewing and examining the data set to understand the data and analyze business performance.

- "Descriptive analytics helps organisations to understand what happened in the past. It helps to understand the relationship between product and customers.
- The objective of this analysis is to understanding, what approach to take in the future. If we learn from past behaviour , it helps us to influence future outcomes.
- Company reports is an example of descriptive analytics which simply provides a historic reviw of company operations, stakeholders, customers and financials.
- It also helps to describe and present data in such format, which can be easily understood by a wide variety of business readers.

1.4.2 Predictive Tasks

- To make prediction, predictive mining tasks performs inference on the current data. Predictive analysis provides answers of the future queries that move across using historical data as the chief principle for decisions.
- It involves the supervised learning functions used for the prediction of the target value. The methods fall under this mining category are the classification, time-series analysis and regression.
- Data modeling is the necessity of the predictive analysis, which works by utilizing some variables to anticipate the unknown future data values for other variables.
- It provides organizations with actionable insights based on data. It provides an estimation regarding the likelihood of a future outcome.
- To do this, a variety of techniques are used, such as machine learning, data mining, modeling and game theory.
- Predictive modeling can, for example, help to identify any risks or opportunities in the future.
- Predictive analytics can be used in all departments, from predicting customer behaviour in sales and marketing, to forecasting demand for operations or determining risk profiles for finance.
- A very well-known application of predictive analytics is credit scoring used by financial services to determine the likelihood of customers making future credit payments on time. Determining such a risk profile requires a vast amount of data, including public and social data.
- Historical and transactional data are used to identify patterns, and statistical models and algorithms are used to capture relationships in various datasets.
- Predictive analytics has taken off in the big data era and there are many tools available for organisations to predict future outcomes.

1.4.3 Difference between Descriptive and Predictive Tasks

Sr. No.	Descriptive model	Predictive model
1.	It use data aggregation and data mining to provide insight into the past and answer.	Use statistical models and forecasts techniques to understand the future and answer.
2.	What has happened ?	What could happened ?
3.	Descriptive analytics is the analysis of past or historical data to understand trends and evaluate metrics over time.	Predictive analytics predicts future trends.
4.	Examples of tools used : Data aggregation and data mining.	Examples of tools used : Machine learning, statistical models and simulation.
5.	Used when user want to summarize results for all or part of your business.	Used when user want to make an educated guess at likely results.
6.	Limitation : Snapshot of the past, often with limited ability to help guide decisions.	Limitation : Guess at the future, helps inform low complexity decisions.

Review Question

1. *Explain predictive and descriptive task.*

1.5 Supervised Learning

- Supervised learning is the machine learning task of inferring a function from supervised training data. The training data consist of a set of training examples. The task of the supervised learner is to predict the output behavior of a system for any set of input values, after an initial training phase.
- **Supervised learning** in which the network is trained by providing it with input and matching output patterns. These input-output pairs are usually provided by an external teacher.
- Human learning is based on the past experiences. A computer does not have experiences.

- A computer system learns from data, which represent some "past experiences" of an application domain.
- To learn a target function that can be used to predict the values of a discrete class attribute, e.g., approve or not-approved and high-risk or low risk. The task is commonly called : Supervised learning, Classification or inductive learning.
- Training data includes both the input and the desired results. For some examples the correct results (targets) are known and are given in input to the model during the learning process. The construction of a proper training, validation and test set is crucial. These methods are usually fast and accurate.
- Have to be able to generalize : Give the correct results when new data are given in input without knowing a priori the target.
- Supervised learning is the machine learning task of inferring a function from supervised training data. The training data consist of a set of training examples. In supervised learning, each example is a pair consisting of an input object and a desired output value.
- A supervised learning algorithm analyzes the training data and produces an inferred function, which is called a classifier or a regression function. Fig. 1.5.1 shows supervised learning process.

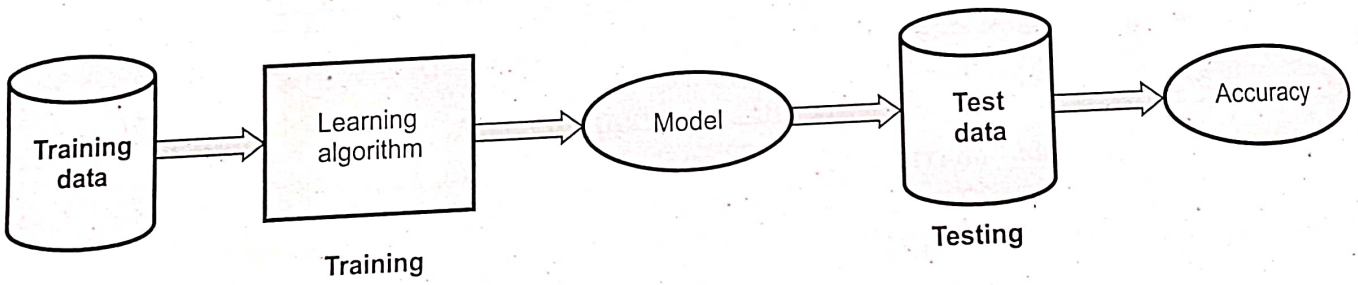


Fig. 1.5.1 Supervised learning process

- The learned model helps the system to perform task better as compared to no learning.
- Each input vector requires a corresponding target vector.

  Training Pair = (Input Vector, Target Vector)
- Fig. 1.5.2 shows input vector. (See Fig. 1.5.2 on next page)
- Supervised learning denotes a method in which some input vectors are collected and presented to the network. The output computed by the net-work is observed and the deviation from the expected answer is measured. The weights are corrected according to the magnitude of the error in the way defined by the learning algorithm.

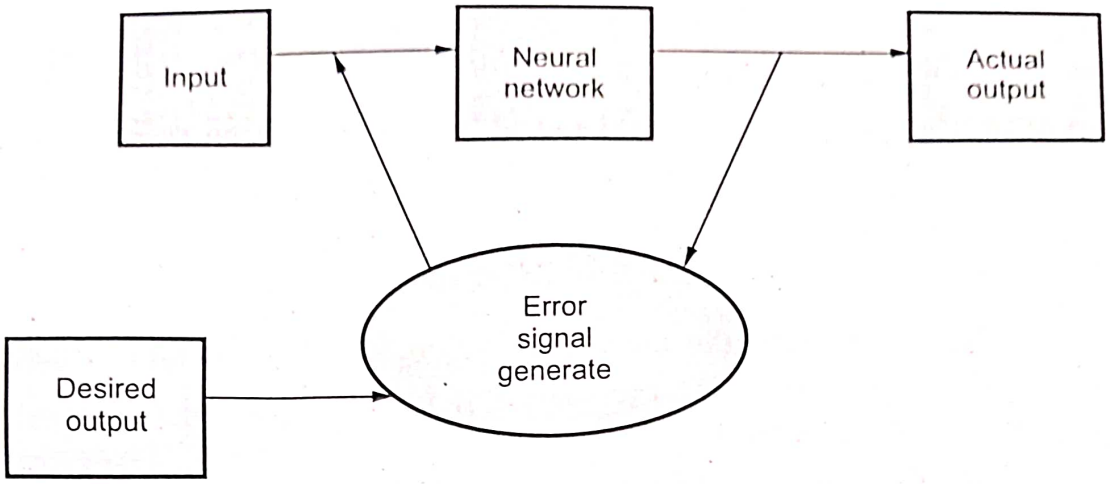


Fig. 1.5.2 Input vector

- Supervised learning is further divided into methods which use reinforcement or error correction. The perceptron learning algorithm is an example of supervised learning with reinforcement.

Data formats in supervised learning :

- Supervised learning always uses a dataset to define finite set of real vectors with m features each :

$$X = \{\bar{x}_1, \bar{x}_2, \ldots, \bar{x}_n\} \text{ where } \bar{x}_i \in \mathbb{R}^m$$

- Considering that user approach is always probabilistic, we need to consider each X as drawn from a statistical multivariate distribution D. It is also useful to add an important condition upon the whole dataset X. Here we consider that all samples to be independent and identically distributed. This means all variables belong to the same distribution D and considering an arbitrary subset of m values, it happens that :

$$P(\bar{x}_1, \bar{x}_2, \ldots, \bar{x}_m) = \prod_{i=1}^{m} P(\bar{x}_i)$$

- The corresponding output values can be both numerical - continuous and categorical. In the first case, the process is called regression, while in the second, it is called classification.
- Example : Dataset contains city populations by year for the past 100 years and user want to know what the population of a specific city will be four years from now. The outcome uses labels that already exist in the data set : **Population, city and year.**
- In order to solve a given problem of supervised learning, following steps are performed :
  1. Find out the type of training examples.

2. Collect a training set.
3. Determine the input feature representation of the learned function.
4. Determine the structure of the learned function and corresponding learning algorithm.
5. Complete the design and then run the learning algorithm on the collected training set.
6. Evaluate the accuracy of the learned function. After parameter adjustment and learning, the performance of the resulting function should be measured on a test set that is separate from the training set.

- Supervised learning is divided into two types : Classification and Regression.

1. Classification :

- Classification predicts categorical labels (classes), prediction models continuous - valued functions. Classification is considered to be supervised learning.
- Classifies data based on the training set and the values in a classifying attribute and uses it in classifying new data. **Prediction** means models continuous-valued functions, i.e., predicts unknown or missing values.
- Preprocessing of the data in preparation for classification and prediction can involve data cleaning to reduce noise or handle missing values, relevance analysis to remove irrelevant or redundant attributes and data transformation, such as generalizing the data to higher level concepts or normalizing data.
- Numeric prediction is the task of predicting continuos values for given input. For example, we may wish to predict the salary of college employee with 15 years of work experience or the potential sales of a new product given its price.
- Some of the classification methods like back - propagation, support vector machines and k-nearest-neighbor classifiers can be used for prediction.

2. Regression :

- For an input x, if the output is continuous, this is called a regression problem. For example, based on historical information of demand for tooth paste in supermarket, user are asked to predict the demand for the next month.
- Regression is concerned with the prediction of continuous quantities. Linear regression is the oldest and most widely used predictive model in the field of machine learning. The goal is to minimize the sum of the squared errors to fit a straight line to a set of data points.
- Regression algorithm used in supervised learning is linear regression, Bayesian linear regression, polynomial regression, regression tree etc.

1.5.1 Advantages and Disadvantages of Supervised Learning

1. Advantages of supervised learning

- It performs classification and regression tasks.
- It allows estimating or mapping the result to a new sample.
- We have complete control over choosing the number of classes we want in the training data.

2. Disadvantages of supervised learning

- Supervised learning cannot handle all complex tasks in Machine Learning.
- Computation time is vast for supervised learning.
- It requires a labelled data set.
- It requires a training process.

Review Questions

1. *Explain supervised learning with example.*
2. *Explain data formats for supervised learning problem with example.*

1.6 Unsupervised Learning

- Unsupervised learning is a type of machine learning in which models are trained using unlabeled dataset and are allowed to act on that data without any supervision.
- In unsupervised learning, a dataset is provided without labels and a model learns useful properties of the structure of the dataset. The main goal of unsupervised learning is to discover hidden and interesting patterns in unlabeled data.
- They are called unsupervised because they do not need a teacher or super-visor to label a set of training examples. Only the original data is required to start the analysis.
- Unsupervised learning tasks typically involve grouping similar examples together, dimensionality reduction and density estimation.
- Common algorithms used in unsupervised learning include clustering, anomaly detection, neural networks.

- Fig. 1.6.1 shows unsupervised learning.

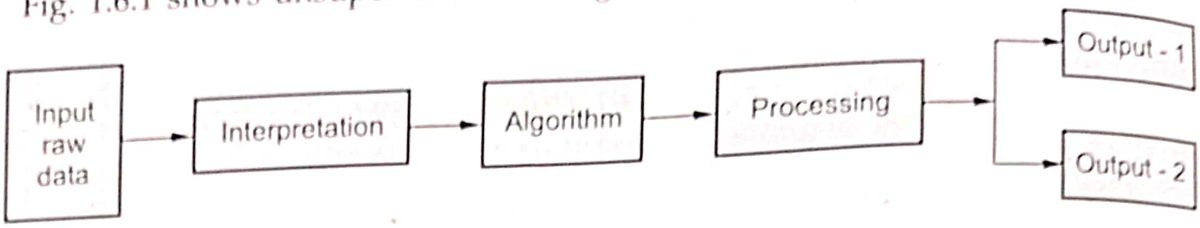


Fig. 1.6.1 Unsupervised learning

- The most common unsupervised learning method is cluster analysis, which applies clustering methods to explore data and find hidden patterns or groupings in data. Unsupervised learning is typically applied before supervised learning, to identify features in exploratory data analysis and establish classes based on groupings.
- Unsupervised machine learning is mainly used to :
  1. Cluster datasets on similarities between features or segment data.
  2. Understand relationship between different data point such as automated music recommendations.
  3. Perform initial data analysis.
- Unsupervised learning algorithms have the capability of analyzing large amounts of data and identifying unusual points among the dataset. Once those anomalies have been detected, they can be brought to the awareness of the user, who can then decide to act or not on this warning.
- Anomaly detection can be very useful in the financial and banking sectors. Indeed, financial fraud has become a daily problem, due to the ease with which credit card details can be stolen. Using unsupervised learning models, unauthorized or fraudulent transactions on a bank account can be identified as it will most often constitute a change in the user's normal pattern of spending.
- Example : Using customer data and user want to create segments of customers who like similar products. The data that user are providing is not labeled and the labels in the outcome are generated based on the similarities that were discovered between data points.
- Types of unsupervised learning is clustering and association analysis.
- There is a wide range of algorithms that can be deployed under unsupervised learning. A few of them includes : K-means clustering, Principal component analysis, Hierarchical clustering and Dendrogram.

1.6.1 Advantages and Disadvantages of Unsupervised Learning

1. Advantages of unsupervised learning

- It does not require a training data to be labelled.

- Dimensionality reduction can be easily accomplished using unsupervised learning.
- Capable of finding previously unknown patterns in data.

2. Disadvantages of unsupervised learning

- Difficult to measure accuracy or effectiveness due to lack of predefined answers during training.
- The results often have lesser accuracy.
- The user needs to spend time interpreting and label the classes which follow that classification.

1.6.2 Difference between Supervised and Unsupervised Learning

Sr. No.	Supervised learning	Unsupervised learning
1.	Desired output is given.	Desired output is not given.
2.	It is not possible to learn larger and more complex models than with supervised learning.	It is possible to learn larger and more complex models with unsupervised learning.
3.	Use training data to infer model.	No training data is used.
4.	Every input pattern that is used to train the network is associated with an output pattern.	The target output is not presented to the network.
5.	Trying to predict a function from labeled data.	Try to detect interesting relations in data.
6.	Supervised learning requires that the target variable is well defined and that a sufficient number of its values are given.	For unsupervised learning typically either the target variable is unknown or has only been recorded for too small a number of cases.
7.	Example : Optical character recognition.	Example : Find a face in an image.
8.	We can test our model.	We can not test our model.
9.	Supervised learning is also called classification.	Unsupervised learning is also called clustering.

1.7 Semi-supervised Learning

- Semi-supervised learning uses both labeled and unlabeled data to improve supervised learning. The goal is to learn a predictor that predicts future test data better than the predictor learned from the labeled training data alone.
- Semi-supervised learning is motivated by its practical value in learning faster, better and cheaper.

- In many real world applications, it is relatively easy to acquire a large amount of unlabeled data x.
- For example, documents can be crawled from the Web, images can be obtained from surveillance cameras, and speech can be collected from broadcast. However, their corresponding labels y for the prediction task, such as sentiment orientation, intrusion detection and phonetic transcript, often requires slow human annotation and expensive laboratory experiments.
- In many practical learning domains, there is a large supply of unlabeled data but limited labeled data, which can be expensive to generate. For example : Text processing, video-indexing, bioinformatics etc.
- Semi-supervised Learning makes use of both labeled and unlabeled data for training, typically a small amount of labeled data with a large amount of unlabeled data. When unlabeled data is used in conjunction with a small amount of labeled data, it can produce considerable improvement in learning accuracy.
- Semi-supervised learning sometimes enables predictive model testing at reduced cost.
- **Semi-supervised classification :** Training on labeled data exploits additional unlabeled data, frequently resulting in a more accurate classifier.
- **Semi-supervised clustering :** Uses small amount of labeled data to aid and bias the clustering of unlabeled data.

1.7.1 Comparision between Supervised, Unsupervised, Semi-supervised Learning

Sr. No.	Supervised learning	Unsupervised learning	Semi-supervised learning
1.	Input data is labeled.	Input data is unlabeled.	A large amount of input data is unlabeled while a small amount is labeled.
2.	Trying to predict a specific quantity.	Trying to understand the data.	Using unsupervised methods to improve supervised algorithm.
3.	Used in Fraud detection.	Used in Identity management.	Used in spam detection.
4.	Subtype : Classification and regression.	Subtype : Clustering and association.	Subtype : Classification, regression, clustering and association.
5.	Higher accuracy.	Lesser accuracy.	Lesser accuracy.

1.8 Reinforcement Learnings

- Reinforcement Learning (RL) is the science of decision making. It is about learning the optimal behavior in an environment to obtain maximum reward. In RL, the data is accumulated from machine learning systems that use a trial-and-error method. Data is not part of the input that we would find in supervised or unsupervised machine learning.
- Reinforcement learning uses algorithms that learn from outcomes and decide which action to take next. After each action, the algorithm receives feedback that helps it determine whether the choice it made was correct, neutral or incorrect. It is a good technique to use for automated systems that have to make a lot of small decisions without human guidance.
- Reinforcement learning is an autonomous, self - teaching system that essentially learns by trial and error. It performs actions with the aim of maximizing rewards, or in other words, it is learning by doing in order to achieve the best outcomes.
- A good example of using reinforcement learning is a robot learning how to walk. The robot first tries a large step forward and falls. The outcome of a fall with that big step is a data point the reinforcement learning system responds to. Since the feedback was negative, a fall, the system adjusts the action to try a smaller step. The robot is able to move forward. This is an example of reinforcement learning in action.
- Reinforcement learning is learning what to do and how to map situations to actions. The learner is not told which actions to take. Fig. 1.8.1 shows concept of reinforced learning.

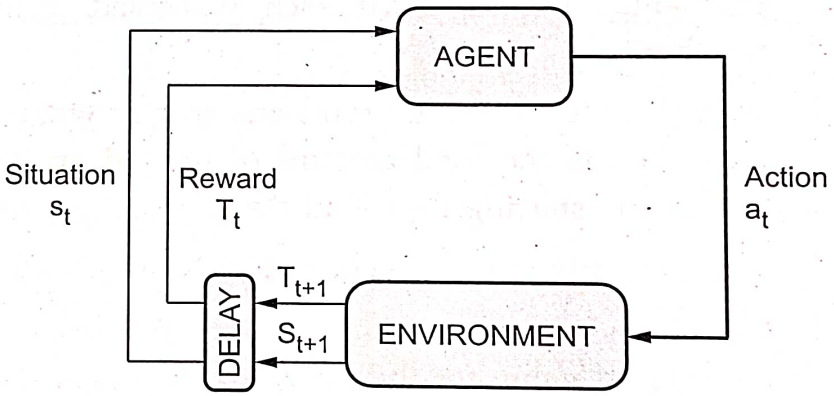


Fig. 1.8.1 Reinforced learning

- Reinforced learning is deals with agents that must sense and act upon their environment. It combines classical Artificial Intelligence and machine learning techniques.
- It allows machines and software agents to automatically determine the ideal behavior within a specific context, in order to maximize its performance. Simple reward feedback is required for the agent to learn its behavior; this is known as the reinforcement signal.
- Two most important distinguishing features of reinforcement learning is trial-and-error and delayed reward.

- With reinforcement learning algorithms an agent can improve its performance by using the feedback it gets from the environment. This environmental feedback is called the reward signal.
- Based on accumulated experience, the agent needs to learn which action to take in a given situation in order to obtain a desired long term goal. Essentially actions that lead to long term rewards need to reinforced. Reinforcement learning has connections with control theory, Markov decision processes and game theory.

1.8.1 Elements of Reinforcement Learning

- Reinforcement learning elements are as follows :
  1. Policy 2. Reward function
  3. Value function 4. Model of the environment
- Fig. 1.8.2 shows elements of RL.

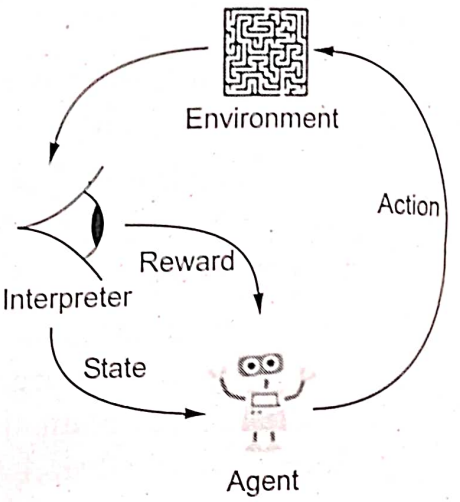


Fig. 1.8.2 Elements of reinforcement learning

- **Policy :** Policy defines the learning agent behavior for given time period. It is a mapping from perceived states of the environment to actions to be taken when in those states.
- **Reward function :** Reward function is used to define a goal in a reinforcement learning problem. It also maps each perceived state of the environment to a single number.
- **Value function :** Value functions specify what is good in the long run. The value of a state is the total amount of reward an agent can expect to accumulate over the future, starting from that state.
- **Model of the environment :** Models are used for planning.
- **Credit assignment problem :** Reinforcement learning algorithms learn to generate an internal value for the intermediate states as to how good they are in leading to the goal.
- The learning decision maker is called the agent. The agent interacts with the environment that includes everything outside the agent.
- The agent has sensors to decide on its state in the environment and takes an action that modifies its state.
- The reinforcement learning problem model is an agent continuously interacting with an environment. The agent and the environment interact in a sequence of time steps. At each time step t, the agent receives the state of the environment and a scalar numerical reward for the previous action, and then the agent then selects an action.

- Reinforcement learning is a technique for solving Markov decision problems.
- Reinforcement learning uses a formal framework defining the interaction between a learning agent and its environment in terms of states, actions and rewards. This framework is intended to be a simple way of representing essential features of the artificial intelligence problem.

1.8.2 Application of Reinforcement Learning

1. Robotics : Robots with pre-programmed behavior are useful in structured environments, such as the assembly line of an automobile manufacturing plant, where the task is repetitive in nature.
2. A master chess player makes a move. The choice is informed both by planning, anticipating possible replies and counter replies.
3. An adaptive controller adjusts parameters of a petroleum refinery's operation in real time.

1.8.3 Advantages and Disadvantages of Reinforcement Learning

Advantages of Reinforcement learning

1. Reinforcement learning can be used to solve very complex problems that cannot be solved by conventional techniques.
2. The model can correct the errors that occurred during the training process.
3. In RL, training data is obtained via the direct interaction of the agent with the environment.

Disadvantages of Reinforcement learning

1. Reinforcement learning is not preferable to use for solving simple problems.
2. Reinforcement learning needs a lot of data and a lot of computation.

Review Question

1. *Discuss the reinforcement learning and write the brief applications.*

1.9 Models of Machine Learning

- A machine learning model is a program that can find patterns or make decisions from a previously unseen dataset. For example, in natural language processing, machine learning models can parse and correctly recognize the intent behind previously unheard sentences or combinations of words.

- A machine learning model can perform such tasks by having it 'trained' with a large dataset. During training, the machine learning algorithm is optimized to find certain patterns or outputs from the dataset, depending on the task. The output of this process - often a computer program with specific rules and data structures - is called a machine learn.
- For classification and regression problem, there are different choices of Machine Learning Models each of which can be viewed as a black box that solve the same problem. However, each model come from a different algorithm approaches and will perform differently under different data set. The best way is to use cross - validation to determine which model performs best on test data.
- The model - based approach seeks to create a modified solution tailored to each new application. Instead of having to transform user problem to fit some standard algorithm, in model-based machine learning user design the algorithm precisely to fit problem.
- The core idea at the heart of model - based machine learning is that all the assumptions about the problem domain are made explicit in the form of a model.
- Model is just made up of set of assumptions, expressed in a precise mathematical form. These assumptions include the number and types of variables in the problem domain, which variables affect each othe and what the effect of changing one variable is on another variable.
- Machine learning models are classified as :
  1. Geometric model (Using the Geometry of the instance space).
  2. Probabilistic model (Using Probability to classify the instance space).
  3. Logical model (Using a Logical expression).

1.9.1 Geometric Model

- Here, we consider models that define similarity by considering the geometry of the instance space. In Geometric models, features could be described as points in two dimensions (x - and y - axis) or a three-dimensional space (x, y and z).
- Geometric models are constructed directly in instance space, using geometric concepts like lines, planes and distances.
- Even when features are not intrinsically geometric, they could be modelled in a geometric manner (for example, temperature as a function of time can be modelled in two axes).
- This models use intuitions from geometry such as separating hyper planes, linear transformations and distance metric. The main goal of this method is to find a set of representative features of geometric form to represent an object by collecting

geometric features from images and learning them using efficient machine learning methods.

- In geometric models, there are two ways we could impose similarity. We could use geometric concepts like lines or planes to segment (classify) the instance space. These are called **linear models.**
- Linear models are parametric, which means that they have a fixed form with a small number of numeric parameters that need to be learned from data. Linear models have low variance and high bias. This implies that linear models are less likely to overfit the training data than some other models.
- In other method, we can use the geometric notion of distance to represent similarity. In this case, if two points are close together, they have similar values for features and thus can be classed as similar. We call such models as **Distance - based models.**
- Examples of distance - based models include the nearest - neighbour models, which use the training data.
- Geometric learning methods can not only solve recognition problems but also predict subsequent actions by analyzing a set of sequential input sensory images, usually some extracting features of images.
- Example of Geometric models : K - nearest neighbors, linear regression, support vector machine, logistic regression.
- Geometric features :
  1. Corners : Corners is a very simple but significant feature of objects. Especially, Complex objects usually have different corner features with each other. Corners of an object can be extracted by using the technique, calling corner detection.
  2. Edges : Edges are one-dimensional structure features of an image. They represent the boundary of different image regions. The outline of an object can be easily detected by finding the edge using the technique of edge detection.
  3. Blobs : Blobs represent regions of images, which can be detected using blob detection method.
  4. Ridges : From a practical viewpoint, a ridge can be thought of as a one - dimensional curve that represents an axis of symmetry, i.e. Ridges detection method.

1.9.2 Probabilistic Models

- Probabilistic models view learning as a process of reducing uncertainty, modeled by means probability distributions. A model describes data that one could observe

from a system. If we use the mathematics of probability theory to express all forms of uncertainty and noise associated with our model.

- Probabilistic models see features and target variables as random variables. The process of modelling represents and manipulates the level of uncertainty with respect to these variables.
- Fig. 1.9.1 shows Probabilistic logic learning.

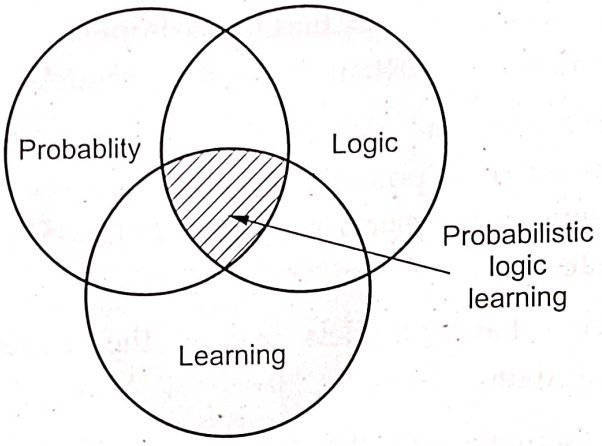


Fig. 1.9.1 Probabilistic logic learning as the intersection of probability logic and learning

- There are two types of probabilistic models : Predictive and Generative.
- Predictive probability models use the idea of a conditional probability distribution P (Y |X) from which Y can be predicted from X. Generative models estimate the joint distribution P (Y, X).
- Once we know the joint distribution for the generative models, we can derive any conditional or marginal distribution involving the same variables. Thus, the generative model is capable of creating new data points and their labels, knowing the joint probability distribution.
- The joint distribution looks for a relationship between two variables. Once this relationship is inferred, it is possible to infer new data points. Naïve Bayes is an example of a probabilistic classifier.
- Example of Probabilistic models : Naïve Bayes, Gaussian process regression, conditional random field.
- Probabilistic modeling is a statistical technique used to take into account the impact of random events or actions in predicting the potential occurrence of future outcomes.
- In machine learning, we train the system by using a limited data set called 'training data' and based on the confidence level of the training data we expect the machine learning algorithm to depict the behaviour of the larger set of actual data.

- Probability theory provides a mathematical foundation for quantifying uncertainty of the knowledge.
- ML is focused on making predictions as accurate as possible, while traditional statistical models are aimed at inferring relationships between variables.
- We make observations using the sensors in the world. Based on the observations, we intend to make decisions. Given the same observations, the decision should be the same. However, the world changes, observations change, our sensors change, the output should not change.
- We build models for predictions; can we trust them? Are they certain? Many applications of machine learning depend on good estimation of the uncertainty :
  a) Forecasting
  b) Decision making
  c) Learning from limited, noisy, and missing data
  d) Learning complex personalised models
  e) Data compression
  f) Automating scientific modelling, discovery, and experiment design.
- A signal is called random if its occurrence can not be predicted. Such signal can not be by any mathematical equation.
- The random signals are represented collectively by a random variable takes its value will be taken at particular time is not known.
- The random variables are analyzed statistically with the help of probability, probability density functions and statistical averages such as mean, variance etc.

Relative frequency : For event 'A' relative frequency is defined as,

$$\text{Relative frequency} = \frac{\text{Number of times event occurs } (N_A)}{\text{Total number of trials}} = \frac{N_A}{N}$$

As number of trials approach infinity, relative frequency is called probability.

Probability of event 'A' is defined as the ratio of number of possible favourable outcomes to total number of outcomes i.e.,

$$\text{Probability, } P(A) = \lim_{N \to \infty} \frac{N_A}{N}$$

$$= \frac{\text{Number of possible favourable outcomes}}{\text{Total number of outcomes}}$$

Permutations and Combinations

Combination of 'n' taken 'r' at a time, $^{n}C_{r} = \frac{n!}{(n-r)!\, r!}$

Permutation of 'n' taken 'r' at a time, $^{n}P_{r} = \frac{n!}{(n-r)!}$

1.9.2.1 Naive Bayes Classifier

- Naive Bayes classifiers are a family of simple probabilistic classifiers based on applying Bayes' theorem with strong independence assumptions between the features. It is highly scalable, requiring a number of parameters linear in the number of variables (features/predictors) in a learning problem.
- A Naive Bayes Classifier is a program which predicts a class value given a set of attributes.
- For each known class value,
  1. Calculate probabilities for each attribute, conditional on the class value.
  2. Use the product rule to obtain a joint conditional probability for the attributes.
  3. Use Bayes rule to derive conditional probabilities for the class variable.
- Once this has been done for all class values, output the class with the highest probability.
- Naive bayes simplifies the calculation of probabilities by assuming that the probability of each attribute belonging to a given class value is independent of all other attributes. This is a strong assumption but results in a fast and effective method.
- The probability of a class value given a value of an attribute is called the conditional probability. By multiplying the conditional probabilities together for each attribute for a given class value, we have a probability of a data instance belonging to that class.

Conditional Probability

- Let A and B be two events such that $P(A) > 0$. We denote $P(B \mid A)$ the probability of B given that A has occurred. Since A is known to have occurred, it becomes the new sample space replacing the original S. From this, the definition is,

$$P(B/A) \equiv \frac{P(A \cap B)}{P(A)}$$

OR

$$P(A \cap B) = P(A)\, P(B/A)$$

- The notation P(B | A) is read **"the probability of event B given event A"**. It is the probability of an event B given the occurrence of the event A.
- We say that, the probability that both A and B occur is equal to the probability that A occurs times the probability that B occurs given that A has occurred. We call **P(B|A) the conditional probability of B given A**, i.e., the probability that B will occur given that A has occurred.
- Similarly, the conditional probability of an event A, given B by,

$$P(A/B) \equiv \frac{P(A \cap B)}{P(B)}$$

- The probability P(A|B) simply reflects the fact that the probability of an event A may depend on a second event B. If A and B are mutually exclusive $A \cap B = \phi$ and $P(A|B) = 0$.
- Another way to look at the conditional probability formula is :

$$P(\text{Second/First}) = \frac{P(\text{First choice and second choice})}{P(\text{First choice})}$$

- Conditional probability is a defined quantity and cannot be proven.
- The key to solving conditional probability problems is to :
  1. Define the events.
  2. Express the given information and question in probability notation.
  3. Apply the formula.

Joint Probability

- A joint probability is a probability that measures the likelihood that two or more events will happen concurrently.
- If there are two independent events A and B, the probability that A and B will occur is found by multiplying the two probabilities. Thus for two events A and B, the special rule of multiplication shown symbolically is :

$$P(A \text{ and } B) = P(A)\, P(B).$$

- The general rule of multiplication is used to find the joint probability that two events will occur. Symbolically, the general rule of multiplication is,

$$P(A \text{ and } B) = P(A)\, P(B|A).$$

- The probability $P(A \cap B)$ is called the joint probability for two events A and B which intersect in the sample space. Venn diagram will readily shows that

$$P(A \cap B) = P(A) + P(B) - P(A \cup B)$$

Equivalently :

$$P(A \cap B) = P(A) + P(B) - P(A \cap B) \leq P(A) + P(B)$$

- The probability of the union of two events never exceeds the sum of the event probabilities.
- A tree diagram is very useful for portraying conditional and joint probabilities. A tree diagram portrays outcomes that are mutually exclusive.

Bayes Theorem

- Bayes' theorem is a method to revise the probability of an event given additional information. Bayes's theorem calculates a conditional probability called a posterior or revised probability.
- Bayes' theorem is a result in probability theory that relates conditional probabilities. If A and B denote two events, $P(A|B)$ denotes the conditional probability of A occurring, given that B occurs. The two conditional probabilities $P(A|B)$ and $P(B|A)$ are in general different.
- Bayes theorem gives a relation between $P(A|B)$ and $P(B|A)$. An important application of Bayes' theorem is that it gives a rule how to update or revise the strengths of evidence-based beliefs in light of new evidence a posteriori.
- A **prior probability** is an initial probability value originally obtained before any additional information is obtained.
- A **posterior probability** is a probability value that has been revised by using additional information that is later obtained.
- Suppose that $B_1, B_2, B_3 \ldots B_n$ partition the outcomes of an experiment and that A is another event. For any number, k, with $1 \leq k \leq n$, we have the formula :

$$P(B_k/A) = \frac{P(A/B_k) \cdot P(B_k)}{\sum_{i=1}^{n} P(A/B_i) \cdot P(B_i)}$$

1.9.3 Logical Models

- Logical models are defined in terms of easily interpretable logical expansions. Logical models use a logical expression to divide the instance space into segments and hence construct grouping models.
- A logical expression is an expression that returns a Boolean value, i.e., a True or False outcome. Models involving logical statements easily translated into human - understandable rules.

- Once the data is grouped using a logical expression, the data is divided into homogeneous groupings for the problem we are trying to solve. For example, for a classification problem, all the instances in the group belong to one class.
- There are mainly two kinds of logical models : Tree models and Rule models.
- Rule models consist of a collection of implications or IF-THEN rules. For tree - based models, the 'if - part' defines a segment and the 'then - part' defines the behavior of the model for this segment. Rule models follow the same reasoning.
- Example of Logical models : Decision tree, random forest.

1.10 Grouping and Grading Models

- Grading vs grouping is an orthogonal categorization to geometric - probabilistic - logical-compositional model. Difference between grouping and grading models is the way they handle the instance space.

Grouping Model :

- Grouping models break the instance space up into groups or segments and in each segment apply a very simple method. Example : Decision tree, KNN.
- Grouping models have fixed resolution. They cannot distinguish instances beyond a resolution. At the finest resolution, grouping models assign the majority class to all instances that fall into the segment. Find the right segments and label all the objects in that segment.

Grading Model :

- Grading models form one global model over the instance space. They don't use the notion of segment.
- Grading models are usually able to distinguish between arbitrary instances, no matter how similar they are.

1.11 Parametric Models

- Model can be represented using a pre - determined number of parameters. These methods in Machine Learning typically take a model - based approach. We make an assumption there with respect to the form of the function to be guessed. Then we choose an appropriate model based on this assumption correct to estimate the set of parameters.
- The advantage of the parametric approach is that the model is defined up to a small number of parameters, for example mean and variance, the sufficient statistics of the distribution. Once those parameters are estimated from the sample, the whole distribution is known.

- We estimate the parameters of the distribution from the given sample, plug in these estimates to the assumed model and get an estimated distribution, which we then use to make a decision. The method we use to estimate the parameters of a distribution is maximum likelihood estimation.
- Examples of parametric machine learning algorithms are Logistic Regression, Linear Discriminant Analysis, Perceptron, Naive Bayes and Simple Neural Networks.
- **Advantages :**
  1. These methods are simpler and easier to understand.
  2. These models are very rapid to learn from data.
  3. They do not need as much training data.
  4. The methods are well - matched to simpler problems.

1.11.1 Maximum Likelihood Estimation

- Maximum - Likelihood Estimation (MLE) is a method of estimating the parameters of a statistical model. When applied to a data set and given a statistical model, maximum - likelihood estimation provides estimates for the model's parameters. $X_1, X_2, X_3, \cdots, X_n$ have joint density denoted $f_\theta(x_1, x_2, \cdots, x_n) = f(x_1, x_2, \cdots, x_n|\theta)$. Given observed values $X_1 = x_1, x_2 = x_2, \cdots, X_n = x_n$,

$$\text{lik}(\theta) = f(x_1, x_2, \cdots, x_n|\theta)$$

Considered as a function of θ.

- If the distribution is discrete, f will be the frequency distribution function.
- The maximum likelihood estimate of θ is that value of that maximises lik(θ) : It is the value that makes the observed data the most probable.

Examples of maximizing likelihood :

- A random variable with this distribution is a formalization of a coin toss. The value of the random variable is 1 with probability θ and 0 with probability $1 - \theta$. Let X be a Bernoulli random variable and let x be an outcome of X, then we have

$$P(X = x) = \begin{bmatrix} \theta & \text{if } x = 1 \\ 1-\theta & \text{if } x = 0 \end{bmatrix}$$

- Usually, we use the notation P(.) for a probability mass and the notation P(.) for a probability density. For mathematical convenience write P(X) as

$$P(X = x) = \theta^x (1-\theta)^{1-x}$$

1.12 Non-parametric Models

- Size of the model depends on data, cannot be represented using a pre-determined number of parameters.
- Instead, non-parametric methods state to a set of algorithms. That does not make any primary assumptions with respect to the form of the function to be assessed. These methods are accomplished by approximating the unidentified function f that could be of any form.
- In machine learning, nonparametric methods are also called instance - based or memory - based learning algorithms.
- Density estimation is the problem of reconstructing the probability density function using a set of given data points. Namely, we observe $X_1;\ldots;X_n$ and we want to recover the underlying probability density function generating our dataset.
- Here we discuss, histogram methods for density estimation. A histogram is a chart that plots the distribution of a numeric variable's values as a series of bars. Each bar typically covers a range of numeric values called a bin or class; a bar's height indicates the frequency of data points with a value within the corresponding bin.
- For simplicity, we assume that $X_i \in [0; 1]$, so $p(x)$ is non - zero only within $[0; 1]$. We also assume that $p(x)$ is smooth and $|p'(X)| \leq L$ for all x. The histogram is to partition the set $[0; 1]$ into several bins and using the count of the bin as a density estimate.
- When we have M bins, this yields a partition

$$B_1 = \left[0, \frac{1}{M}\right), B_2 = \left[\frac{1}{M}, \frac{2}{M}\right), \ldots, B_{M-1} = \left[\frac{M-2}{M}, \frac{M-1}{M}\right), B_M = \left[\frac{M-1}{M}, 1\right]$$

- In such case, then for a given point $x \in B_\ell$, the density estimator from the histogram will be

$$\hat{p}_n(x) = \frac{\text{Number of observations within } B_\ell}{n} \times \frac{1}{\text{Length of the bin}}$$

$$= \frac{M}{n}\sum_{i=1}^{n} I(X_i \in B_\ell)$$

- The intuition of this density estimator is that the histogram assign equal density value to every points within the bin. So for B_ℓ, that contains x, the ratio of observations within this bin is $\frac{1}{n}\sum_{i=1}^{n} I(X_i \in B_\ell)$ which should be equal to the density estimate times the length of the bin.
- Non - parametric methods lean towards additional precision because they try to find the best fit for the data points. Though, this approaches at the cost of needing

a very huge amount of observations. That is desired so as to approximate the unidentified function (f) exactly.

- Non - parametric methods can occasionally present overfitting. They can on occasion learn the errors and noise in a way that they cannot simplify well to new, unseen data points as these methods tend to be more flexible.
- Examples of non - parametric methods are k-Nearest Neighbors, Decision Trees like CART and C4.5, Support Vector Machines.
- **Advantages of nonparametric methods :**

1. Accomplished in fitting a huge number of functional forms.
2. There are no assumptions about the original function.
3. They may outcome in higher performance models for prediction.

Limitations of nonparametric methods

1. They require a lot more training data to estimate the mapping function.
2. Overfitting : Extra risk to overfit the training data.

1.12.1 Difference between Non-parametric Methods and Parametric Methods

Sr. No.	Non-parametric method	Parametric methods
1.	Algorithms that do not make particular assumptions about the kind of mapping function are known as non-parametric algorithms.	Parametric model is a learner that summarizes data through a collection of parameters.
2.	Non-parametric analysis to test group medians.	Parametric analysis to test group means.
3.	It can be used on small samples.	Tend to need larger samples.
4.	No information about the population is available.	Information about population is completely known.
5.	It can be used on ordinal and nominal scale data.	Used mainly on interval and ratio scale data.
6.	Not necessarily the samples are independent.	Samples are independent.
7.	K-nearest neighbors is an example of a non - parametric algorithm.	Examples of parametric models include logistic regression and linear SVM.

1.13 Feature

- In machine learning, features are individual independent variables that act like input in your system. Feature is an attribute of a data set and used in a machine learning process. Selection of the subset of features which are meaningful for machine learning is a sub-area of feature engineering.
- The features in a data set are also called its dimensions. So a data set having 'n' features is called an n-dimensional data set.
- A good feature representation is central to achieving high performance in any machine learning task.
- Consider an example of text categorization. Assume that we need to train a model for classifying a given document as spam and not spam. If we represent a document as a bag of words, the feature space consists of a vocabulary of all unique words present in all the documents in the training set.
- For a collection of 100,000 to 1,000,000 documents, we can easily expect hundreds of thousands of features. If we further extend this document model to include all possible bigrams and trigrams, we could easily get over a million features.
- A feature tree is a tree such that each internal node is labelled with a feature, and each edge emanating from an internal node is labelled with a literal. The set of literals at a node is called a split. Each leaf of the tree represents a logical expression, which is the conjunction of literals encountered on the path from the root of the tree to the leaf. The extension of that conjunction is called the instance space segment associated with the leaf.
- Two features are redundant if they are highly correlated, regardless of whether they are correlated with the task or not.
- Feature engineering is the process of creating features (also called "attributes") that don't already exist in the dataset. This means that if your dataset already contains enough "useful" features, you don't necessarily need to engineer additional features.
- Feature engineering refers to the process of translating a data set into features such that these features are able to represent the data set more effectively and result in a better learning performance.
- If feature engineering is performed properly, it helps to improve the power of prediction of machine learning algorithms by creating the features using the raw data that facilitate the machine learning process.
- Elements of feature engineering is **feature transformation and feature subset selection.**

1.13.1 Feature Transformation

- **Feature transformation** transforms the data, structured or unstructured, into a new set of features which can represent the underlying problem which machine learning is trying to solve.
- There are two distinct goals of feature transformation :
  1. Achieving best reconstruction of the original features in the data set.
  2. Achieving highest efficiency in the learning task.
- There are two variants of feature transformation :
  1. Feature construction.
  2. Feature extraction.

1.13.2 Feature Construction

- Feature construction involves transforming a given set of input features to generate a new set of more powerful features which can then used for prediction.
- Feature construction methods may be applied to pursue two distinct goals : Reducing data dimensionality and improving prediction performance.
- Steps :
  1. Start with an initial feature space F_0 .
  2. Transform F_0 to construct a new feature space F_N .
  3. Select a subset of features F_i from F_N .
  4. If some terminating criteria is achieved : Go back to step 3 otherwise set $F_T = F_i$.
  5. F_T is the newly constructed feature space.
- Feature construction process discovers missing information about the relationships between features and augments the feature space by creating additional features.
- Hence, if there are 'n' features or dimensions in a data set, after feature construction 'm' more features or dimensions may get added. So at the end, the data set will become 'n + m' dimensional.
- The task of constructing appropriate features is often highly application specific and labour intensive. Thus building auto-mated feature construction methods that require minimal user effort is challenging. In particular we want methods that :
  1. Generate a set of features that help improve prediction accuracy.
  2. Are computationally efficient.
  3. Are generalizable to different classifiers.
  4. Allow for easy addition of domain knowledge.

- Genetic programming is an evolutionary algorithm-based technique that starts with a population of individuals, evaluates them based on some fitness function and constructs a new population by applying a set of mutation and crossover operators on high scoring individuals and eliminating the low scoring ones.
- In the feature construction paradigm, genetic programming is used to derive a new feature set from the original one. Individuals are often tree like representations of features, the fitness function is usually based on the prediction performance of the classifier trained on these features while the operators can be applications specific.
- The method essentially performs a search in the new feature space and helps generate a high performing subset of features. The newly generated features may often be more comprehensible and intuitive than the original feature set, which makes GP-related methods well-suited for such tasks.
- In decision trees, the model explicitly selects features that are highly correlated with the label. In particular, by limiting the depth of the decision tree, one can at least hope that the model will be able to throw away irrelevant features.

1.13.3 Feature Extraction

- Feature extraction is a process that extracts a set of new features from the original features through some functional mapping. Feature extraction method creates a new feature set.
- Feature extraction increases the accuracy of learned models by extracting features from the input data. This phase of the general framework reduces the dimensionality of data by removing the redundant data.
- A characteristic of these large data sets is a large number of variables that require a lot of computing resources to process.
- Feature extraction is the name for methods that select and combine variables into features, effectively reducing the amount of data that must be processed, while still accurately and completely describing the original data set.
- The process of feature extraction is useful when you need to reduce the number of resources needed for processing without losing important or relevant information.
- Feature extraction can also reduce the amount of redundant data for a given analysis. Also, the reduction of the data and the machine's efforts in building variable combinations (features) facilitate the speed of learning and generalization steps in the machine learning process.

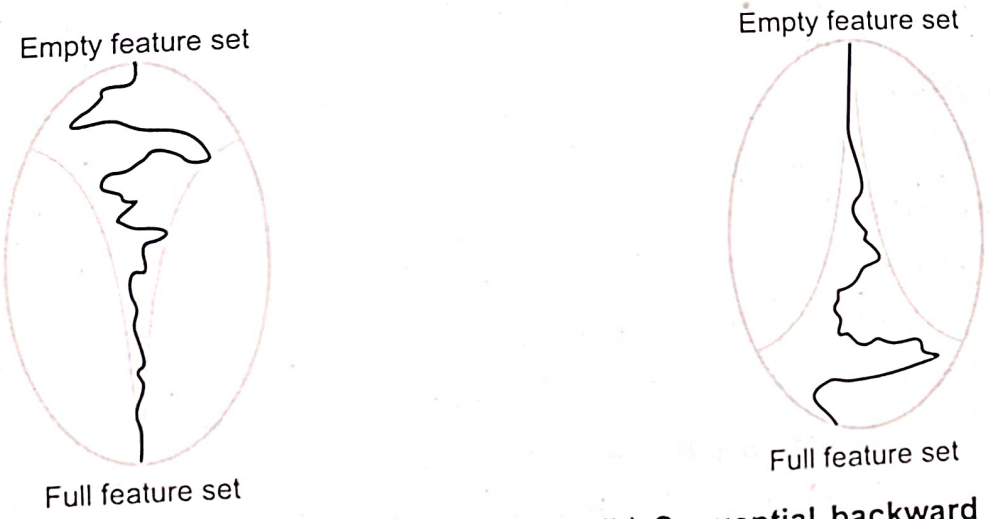
1.13.4 Feature Selection

- Feature selection is a process that chooses a subset of features from the original features so that the feature space is optimally reduced according to a certain criterion.
- Feature selection is a critical step in the feature construction process. In text categorization problems, some words simply do not appear very often. Perhaps the word "groovy" appears in exactly one training document, which is positive. Is it really worth keeping this word around as a feature ? It's a dangerous endeavor because it's hard to tell with just one training example if it is really correlated with the positive class or is it just noise. You could hope that your learning algorithm is smart enough to figure it out. Or you could just remove it.
- There are three general classes of feature selection algorithms : Filter methods, wrapper methods and embedded methods.
- The role of feature selection in machine learning is,
  1. To reduce the dimensionality of feature space.
  2. To speed up a learning algorithm.
  3. To improve the predictive accuracy of a classification algorithm.
  4. To improve the comprehensibility of the learning results.
- Features Selection Algorithms are as follows :
  1. **Instancebased approaches :** There is no explicit procedure for feature subset generation. Many small data samples are sampled from the data. Features are weighted according to their roles in differentiating instances of different classes for a data sample. Features with higher weights can be selected.
  2. **Nondeterministic approaches :** Genetic algorithms and simulated annealing are also used in feature selection.
  3. **Exhaustive complete approaches :** Branch and Bound evaluates estimated accuracy and ABB checks an inconsistency measure that is monotonic. Both start with a full feature set until the preset bound cannot be maintained.

1.13.5 Subsect Selection

- Finding the best subset of the set of features is main aim of subset selection. The best subset contains the least number of dimensions that most contribute to accuracy.

- Using a suitable error function, this can be used in both regression and classification problems. There are 2d possible subsets of d variables, but we cannot test for all of them unless d is small and we employ heuristics to get a reasonable (but not optimal) solution in reasonable (polynomial) time.
- Subset selection are of two types : Forward and backward selection.
  1. **Forward selection :** It start without variables and add them one by one, at each step adding the one that decreases the error the most, until any further addition does not decrease the error.
  2. **Backward selection :** It start with all variables and remove them one by one, at each step removing the one that decreases the error the most, until any further removal increases the error significantly.
- **Sequential Forward Selection (SFS) :** SFS is the simplest greedy search algorithm. It start from the empty set, sequentially add the feature x^+. SFS performs best when the optimal subset is small.
- The main disadvantage of SFS is that it is unable to remove features that become obsolete after the addition of other features.
- **Sequential Backward Selection (SBS) :** It works in the opposite direction of SFS. Starting from the full set, sequentially remove the feature x^- that least reduces the value of the objective function.
- SBS works best when the optimal feature subset is large, since SBS spends most of its time visiting large subsets. The main limitation of SBS is its inability to reevaluate the usefulness of a feature after it has been discarded.
- SFS is performed from the empty set. SBS is performed from the full set.
- There are two floating methods :
  1. Sequential Floating Forward Selection (SFFS) starts from the empty set. After each forward step, SFFS performs backward steps as long as the objective function increases.
  2. Sequential Floating Backward Selection (SFBS) starts from the full set. After each backward step, SFBS performs forward steps as long as the objective function increases.
- Subset selection is supervised in that outputs are used by the regressor or classifier to calculate the error, but it can be used with any regression or classification method.



(a) Sequential forward selection (b) Sequential backward selection

Fig. 1.13.1

- In an application like face recognition, feature selection is not a good method for dimensionality reduction because individual pixels by themselves do not carry much discriminative information; it is the combination of values of several pixels together that carry information about the face identity. This is done by feature extraction methods.

1.13.6 Curse of Dimensionality

- In machine learning, "dimensionality" simply refers to the number of features (i.e. input variables) in your dataset.
- When the number of features is very large relative to the number of observations in your dataset, certain algorithms struggle to train effective models. This is called the "Curse of Dimensionality," and it's especially relevant for clustering algorithms that rely on distance calculations.
- Dimensionality reduction is the process of reducing the number of random variables under consideration, by obtaining a set of principal variables. It can be divided into feature selection and feature extraction.
- Classification problem example : We have an input data $\{X_1, X_2, X_3, ..., X_N\}$ such that $x_i = (x_i^1, x_i^2, ..., x_i^d)$ and a set of corresponding output labels. Assume the dimension d of the data point x is very large and we want to classify x.
- Problem with high dimensional input vectors are large number of parameters to learn, if a dataset is small, this can result in overfit and large variance of estimates.
- Solution to this problem is as follows :
  1. Selection of a smaller subset of inputs from a large set of inputs; train classifier on the reduced input set.

2. Combination of high dimensional inputs to a smaller set of features $\phi_k(x)$; train classifier on new features.

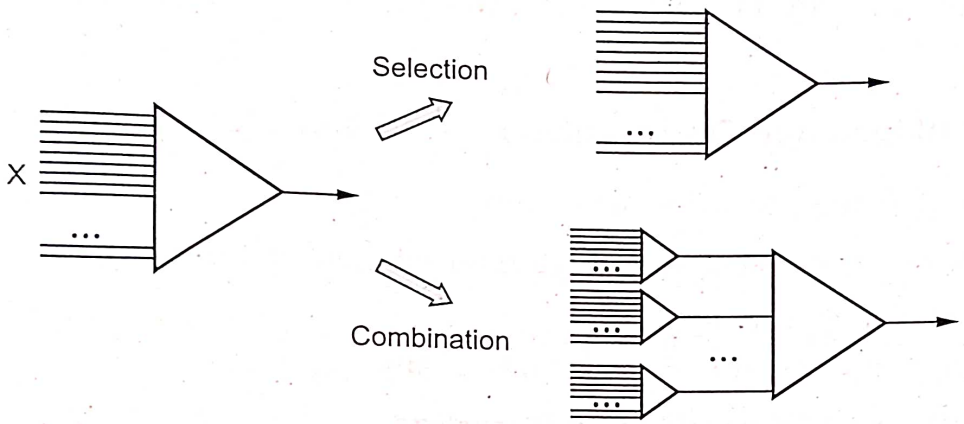


Fig. 1.13.2 Dimensionality reduction

There are two components of dimensionality reduction :

1. **Feature selection :** User try to find a subset of the original set of variables, or features, to get a smaller subset which can be used to model the problem. It usually involves three ways : Filter, wrapper and embedded.
2. **Feature extraction :** This reduces the data in a high dimensional space to a lower dimension space, i.e. a space with lesser number of dimensions.

There are many methods to perform dimension reduction.

1. **Missing Values :** While exploring data, if we encounter missing values, what we do ? Our first step should be to identify the reason then impute missing values / drop variables using appropriate methods. But, what if we have too many missing values ? Should we impute missing values or drop the variables ?
2. **Low Variance :** Let's think of a scenario where we have a constant variable in our data set.
3. **Decision Trees :** It can be used as a ultimate solution to tackle multiple challenges like missing values, outliers and identifying significant variables.
4. **Random Forest :** Similar to decision tree is random forest.
5. **High Correlation :** Dimensions exhibiting higher correlation can lower down the performance of model. Moreover, it is not good to have multiple variables of similar information or variation also known as "multicollinearity".

6. **Backward Feature Elimination :** In this method, we start with all n dimensions. Compute the sum of square of error (SSR) after eliminating each variable (n times). Then, identifying variables whose removal has produced the smallest increase in the SSR and removing it finally, leaving us with n-1 input features.

1.13.7 Advantages and Disadvantages of Dimensionality Reduction

Advantages of Dimensionality Reduction

It helps in data compression and hence reduced storage space.

It reduces computation time.

- It also helps remove redundant features, if any.

- **Disadvantages of Dimensionality Reduction**
- It may lead to some amount of data loss.
- PCA tends to find linear correlations between variables, which is sometimes undesirable.
- PCA fails in cases where mean and covariance are not enough to define datasets.
- We may not know how many principal components to keep - in practice, some thumb rules are applied.

1.14 PCA

- This method was introduced by Karl Pearson. It works on a condition that while the data in a higher dimensional space is mapped to data in a lower dimension space, the variance of the data in the lower dimensional space should be maximum.
- Principal Component Analysis (PCA) is to reduce the dimensionality of a data set by finding a new set of variables, smaller than the original set of variables, retains most of the sample's information and useful for the compression and classification of data.

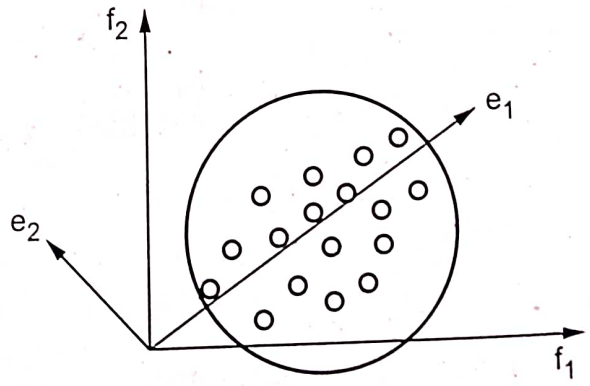


Fig. 1.14.1 PCA

- In PCA, it is assumed that the information is carried in the variance of the features, that is, the higher the variation in a feature, the more information that feature carries.

- Hence, PCA employs a linear transformation that is based on preserving the most variance in the data using the least number of dimensions.
- It involves the following steps :
  1. Construct the covariance matrix of the data.
  2. Compute the eigenvectors of this matrix.
  3. Eigenvectors corresponding to the largest eigen values are used to reconstruct a large fraction of variance of the original data.
- The data instances are projected onto a lower dimensional space where the new features best represent the entire data in the least squares sense.
- It can be shown that the optimal approximation, in the least square error sense, of a d-dimensional random vector $x_2 < d$ by a linear combination of independent vectors is obtained by projecting the vector x onto the eigenvectors e_i corresponding to the largest eigen values l_i of the covariance matrix (or the scatter matrix) of the data from which x is drawn.
- The eigenvectors of the covariance matrix of the data are referred to as principal axes of the data, and the projection of the data instances on to these principal axes are called the principal components. Dimensionality reduction is then obtained by only retaining those axes (dimensions) that account for most of the variance, and discarding all others.
- In the Fig. 1.14.1, Principal axes are along the eigenvectors of the covariance matrix of the data. There are two principal axes shown in the figure, first one is closed to origin, the other is far from origin.
- If $X = X_1, X_2, \ldots\ldots, X_N$ is the set of n patterns of dimension d, the sample mean of the data set is

$$m = \frac{1}{n}\sum_{i=1}^{n} X_i$$

- The sample covariance matrix is

$$C = (X-m)(X-m)^T$$

- C is a symmetric matrix. The orthogonal basis can be calculated by finding the eigenvalues and eigenvectors.
- The eigenvectors g_i and the corresponding eigenvalues λ_i are solutions of the equation

$$C * g_i = \lambda_i * g_i \quad i = 1, \ldots\ldots, d$$

- The eigenvector corresponding to the largest eigenvalue gives the direction of the largest variance of the data. By ordering the eigenvectors according to the eigenvalues, the directions along which there is maximum variance can be found.

- If E is the matrix consisting of eigenvectors as row vectors, we can transform the data X to get Y.

$$Y = E(X - m)$$

- The original data X can be got from Y as follows :

$$X = E^t Y + m$$

- Instead of using all d eigenvectors, the data can be represented by using the first k eigenvectors where $k < d$.
- If only the first k eigenvectors are used represented by E_K, then

$$Y = E_K (X - m) \quad \text{and} \quad X' = E_k^t Y + m$$

1.14.1 Non Negative Matrix Factorization (NMF)

- Nonnegative Matrix Factorization is a matrix factorization method where we constrain the matrices to be nonnegative. In order to understand NMF, we should clarify the underlying intuition between matrix factorization.
- Suppose we factorize a matrix X into two matrices W and H so that X = W H.
- There is no guarantee that we can recover the original matrix, so we will approximate it as best as we can.
- Now, suppose that X is composed of m rows, $x_1, x_2, \ldots x_m$, W is composed of k rows $w_1, w_2, \ldots w_k$, H is composed of m rows $h_1, h_2, \ldots h_m$.
- Each row in X can be considered a data point. For instance, in the case of decomposing images, each row in X is a single image, and each column represents some feature,

$$X = \begin{bmatrix} x_1 \\ x_2 \\ \cdots \\ x_k \end{bmatrix}, \quad W = \begin{bmatrix} w_1 \\ w_2 \\ \cdots \\ w_k \end{bmatrix}, \quad H = \begin{bmatrix} h_1 \\ h_2 \\ \cdots \\ h_k \end{bmatrix}$$

- Take the i^{th} row in X, x_i. If you think about the equation, you will find that x_i can be written as

$$x_i = \sum_{j=1}^{k} w_{ij} \times h_i$$

- Basically, we can interpret x_i to be a weighted sum of some components, where each row in H is a component, and each row in W contains the weights of each component.

How Does it Work ?

- NMF decomposes multivariate data by creating a user-defined number of features. Each feature is a linear combination of the original attribute set; the coefficients of these linear combinations are non-negative.
- NMF decomposes a data matrix V into the product of two lower rank matrices W and H so that V is approximately equal to W times H.
- NMF uses an iterative procedure to modify the initial values of W and H so that the product approaches V. The procedure terminates when the approximation error converges or the specified number of iterations is reached.
- During model apply, an NMF model maps the original data into the new set of attributes (features) discovered by the model.

1.14.2 Difference between PCA and NMF

Sr. No.	PCA	NMF
1.	It uses unsupervised dimensionality reduction.	It also uses unsupervised dimensionality reduction.
2.	Orthogonal vectors with positive and negative coefficients.	Non-negative coefficients.
3.	Difficult to interpret.	Easier to interpret.
4.	PCA is non-iterative.	NMF is iterative.
5.	Designed for producing optimal basis images.	Designed for producing coefficients with a specific property.

1.14.3 Sparse PCA

- In sparse PCA one wants to get a small number of features which still capture most of the variance. Thus one needs to enforce sparsity of the PCA component, which yields a trade-off between explained variance and sparsity.
- To address the non-sparsity issue of traditional PCA, sparse PCA imposes additional constraint on the number of non-zero element in the vector v.
- This is achieved through the l_0 norm, which gives the number of non-zero element in the vector v. A sparse PCA with at most k non-zero loadings can then be formulated as the following optimization problem.
- Optimization problems with l_0 norm constraint is in general NP-hard. Therefore, most methods for sparse PCA relaxes the l_0 norm constraint with l_1 norm appended to the objective function.

1.14.4 Kernel PCA

- Kernel PCA is the nonlinear form of PCA, which better exploits the complicated spatial structure of high-dimensional features.
- It can extract up to n (number of samples) nonlinear principal components without expensive computations.
- The standard steps of kernel PCA dimensionality reduction can be summarized as :
  1. Construct the kernel matrix K from the training data set.
  2. Compute the gram matrix.
  3. Solve N-dimensional column vector.
  4. Compute the kernel principal components.
- Kernel PCA supports both transform and inverse_transform.
- Fig 1.14.2 (a), (b) shows PCA and KPCA.

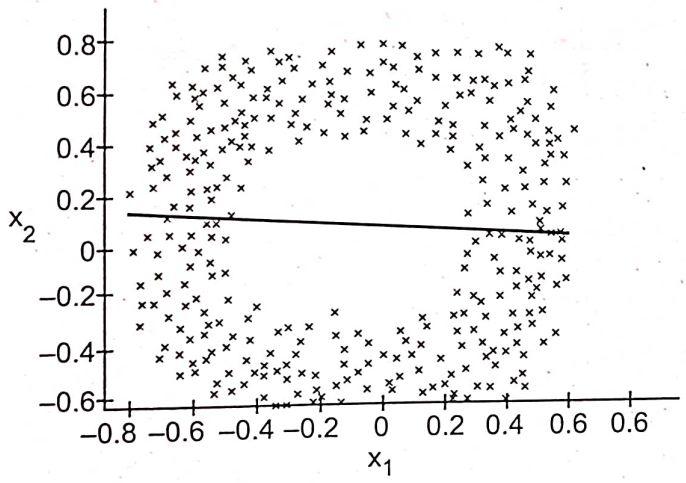


Fig. 1.14.2 (a) PCA

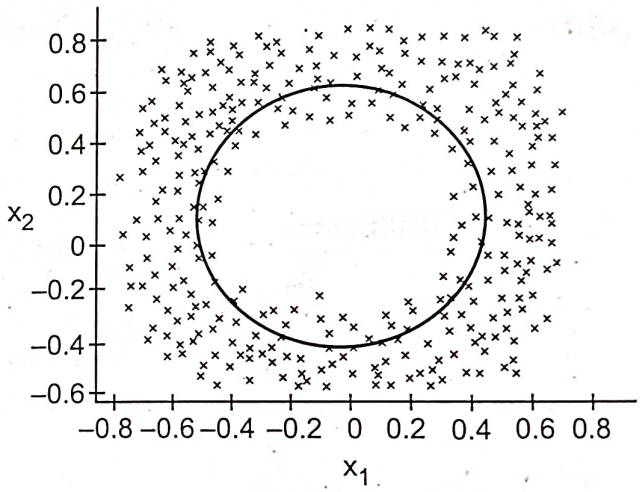


Fig. 1.14.2 (b) KPCA

Preliminaries :

```
# Load libraries
from sklearn.decomposition import PCA, KernelPCA
from sklearn.datasets import make_circles
```

Create Linearly Inseparable Data :

```
# Create linearly inseparable data
X, _ = make_circles(n_samples=1000, random_state=1, noise=0.1, factor=0.1)
```

Conduct Kernel PCA :

```
# Apply kernal PCA with radius basis function (RBF) kernel
kpca = KernelPCA(kernel="rbf", gamma=15, n_components=1)
X_kpca = kpca.fit_transform(X)
```

Review Question

1. *What is Principal Component Analysis (PCA), when it is used.*

1.15 LDA

- Fisher Linear Discriminant Analysis is also called Linear Discriminant Analysis (LDA). LDA is closely related to PCA, for both of them are based on linear, i.e. matrix multiplication, transformations.
- In LDA, the transformation is based on maximizing a ratio of "between-class variance" to "within-class variance" with the goal of reducing data variation in the same class and increasing the separation between classes.
- First applied in 1935 by M. Barnard at the suggestion of R. A. Fisher (1936), Fisher linear discriminant analysis :

1. finds linear combinations of the gene expression profiles $X = X_1, \ldots, X_p$ with large ratios of between-groups to within-groups sums of squares - discriminant variables;
2. predicts the class of an observation X by the class whose mean vector is closest to X in terms of the discriminant variables

- Suppose we have two classes and d-dimensional samples $x_1, \ldots, x_n$ where

a. n_1 samples come from the first class.

b. n_2 samples come from the second class.

- Consider projection on a line. Let the line direction be given by unit vector v.

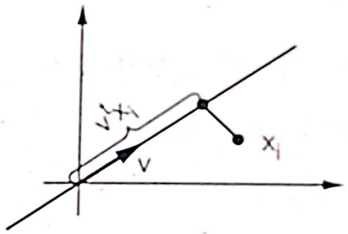


Fig. 1.15.1

- Scalar $v^t x_i$ is the distance of projection of x_i from the origin. Thus it $v^t x_i$ is the projection of x_i into a one dimensional subspace.
- Thus the projection of sample x_i onto a line in direction v is given by $v^t x_i$.
- How to measure separation between projection of different classes ?
- Let p_1 and p_2 be the means of projection of classes 1 and 2.
- Let μ_1 and μ_2 be the means of classes 1 and 2.

- $|p_1 - p_2|$ seems like a good measure

$$p_1 = \frac{1}{n_1} \sum_{x_i \in C_1}^{n_1} v^t x_i = v^t \left(\frac{1}{n_1} \sum_{x_i \in C_1}^{n_1} x_i \right) = v^t \mu_1$$

Similarly, $p_2 = v^t \mu_2$

however $|\hat{\mu}_1 - \hat{\mu}| > |\tilde{\mu}_1 - \tilde{\mu}_2|$

- The problem with $|\tilde{\mu}_1 - \tilde{\mu}_2|$ is that it does not consider the variance of the classes.

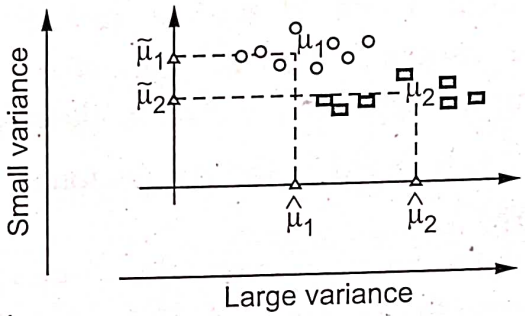


Fig. 1.15.2

- We need to normalize $|\tilde{\mu}_1 - \tilde{\mu}_2|$ by a factor which is proportional to variance.
- Have samples $z_1, \ldots, z_n$. Samples mean is $\mu_z = \frac{1}{2} \sum_{i=1}^{n} z_i$.
- Define their scatter as

$$s = \sum_{i=1}^{n} (z_i - \mu_z)^2$$

- Fisher solution : Normalize $|\tilde{\mu}_1 - \tilde{\mu}_2|$ by scatter
- Let $y_i = v^t x_i$, i.e. y_i's are the projected samples
- Scatter for projected samples of class 1 is

$$\tilde{s}_1^2 = \sum_{y_i \in \text{Class } 1} (y_i - \tilde{\mu}_1)^2$$

- Scatter for projected samples of class 2 is

$$\tilde{s}_2^2 = \sum_{y_i \in \text{Class } 2} (y_i - \tilde{\mu}_2)^2$$

- We need to normalize by both scatter of class 1 and scatter of class 2. Thus Fisher linear discriminant is to project on line in the direction v which maximizes

$$J(v) = \frac{(\tilde{\mu}_1 - \tilde{\mu}_2)^2}{\tilde{S}_1^2 + \tilde{S}_2^2}$$

- Define the separate class scatter matrices S_1 and S_2 for classes 1 and 2.

Discriminant function :

- Discriminant function is a function of the pattern x that leads to a classification rule. The form of the discriminant function is specified and is not imposed by the underlying distribution.
- When g(x) is linear, the decision surface is a hyperplane

$$g(x) = w^T x + w_0 = \sum_{i=1}^{d} w_i x_i + w_o$$

- For a 2-class case, we seek a weight vector w and threshold w_o such that

$$w^T x + w_0 \begin{cases} > 0 \\ < 0 \end{cases} \Rightarrow x \in \begin{cases} w_1 \\ w_2 \end{cases}$$

- If x_1 and x_2 are both on the decision surface, then

$$w^T x_1 + w_0 = W^T x_2 + w_o$$

$$w^T (x_1 - x_2) = 0$$

i.e. Weight vector is normal to vectors in the hyperplane

We can write $x = x_p + r\frac{w}{||w||}$

$$g(x) = w^T x + w_o$$

$$= w^T \left(x_p + r\frac{w}{||w||} \right) + w_o$$

$$= r\frac{||w||^2}{||w||}$$

or, $$r = \frac{g(x)}{||w||}$$

- The value of the discriminant function for a pattern x is a measure of its distance from the hyperplane. A pattern classifier using linear discriminant functions is called a *linear machine.*
- The decision boundaries are assumed to be linear. The discriminant function divides the feature space by a hyperplane whose orientation is determined by the weight vector w and the distance from the origin by the threshold w_o.
- Different optimization schemes lead to different method such as the perceptron, Fisher's Linear discriminant function and support vector machines. Linear combinations of nonlinear functions serves as a stepping stone to nonlinear models.

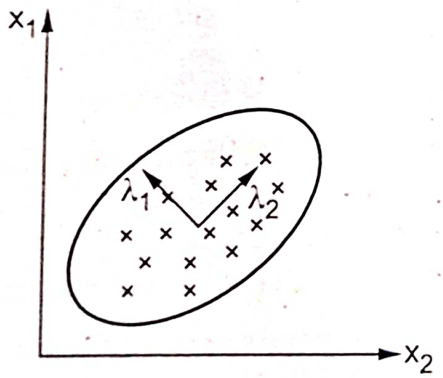
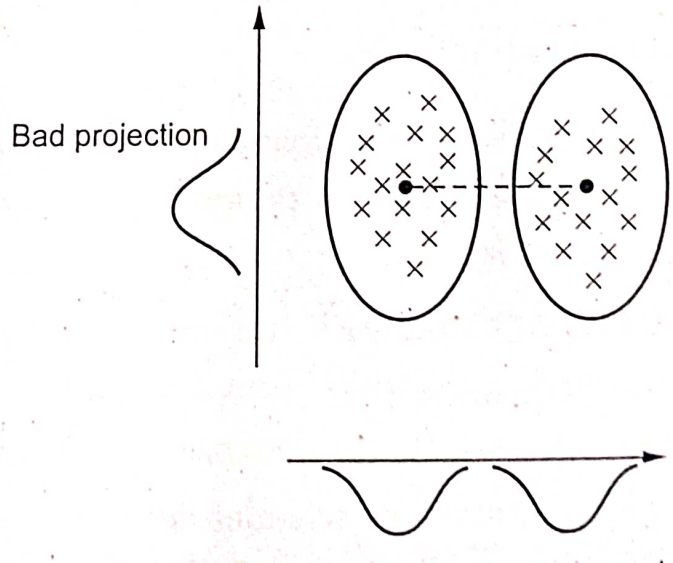
- Limitations of LDA :

a) LDA is a parametric method since it assumes unimodal Gaussian likelihoods.

b) LDA will fail when the discriminatory information is not in the mean but rather in the variance of the data.

c) LDA produces at most C-1 feature projections

1.15.1 Difference between PCA and LDA

Sr. No.	PCA	LDA
1.	Perform dimensionality reduction while preserving as much of the variance in the high dimensional space as possible.	Perform dimensionality reduction while preserving as much of the class discriminatory information as possible
2.	The transformation is based on minimizing mean square error between original data vectors and data vectors that can be estimated from the reduced dimensionality data vectors.	The transformation is based on maximizing a ratio of "between-class variance" to "within-class variance" with the goal of reducing data variation in the same class and increasing the separation between classes.
3.	Higher variance	Smaller variance
4.	Bad for discriminability	Good discriminability
5.	PCA as a technique that finds the directions of maximal variance	LDA attempts to find a feature subspace that maximizes class separability
6.	PCA is unsupervised algorithm	LDA is supervised algorithm.
7.		

1.16 Application of Machine Learning

- Examples of successful applications of machine learning :

Here are several examples :

1 **Optical character recognition :** Categorize images of handwritten characters by the letters represented.

2 **Face detection :** Find faces in images (or indicate if a face is present).

3 **Spam filtering :** Identify email messages as spam or non-spam topic spotting: categorize news articles (say) as to whether they are about politics, sports, entertainment, etc.

4 **Spoken language understanding :** Within the context of a limited domain, determine the meaning of something uttered by a speaker to the extent that it can be classified into one of a fixed set of categories.

1.16.1 Face Recognition and Medical Diagnosis

Face recognition

- Face recognition task is effortlessly and every day we recognize our friends, relative and family members. We also recognition by looking at the photographs. In photographs, they are in different pose, hair styles, background light, makeup and without makeup.
- We do it subconsciously and cannot explain how we do it. Because we can't explain how we do it, we can't write an algorithm.
- Face has some structure. It is not a random collection of pixel. It is symmetric structure. It contains predefined components like nose, mouth, eye, ears. Every person face is a pattern composed of a particular combination of the features. By analyzing sample face images of a person, a learning program captures the pattern specific to that person and users it to recognize if a new real face or new image belongs to this specific person or not.
- Machine learning algorithm creates an optimized model of the concept being learned based on data or past experience.
- In the case of face recognition, the input is an image, the classes are people to be recognized and the learning program should learn to associate the face images to identities. This problem is more difficult than optical character recognition because there are more classes, input image is larger and a face is 3D and differences in pose and lighting cause significant changes in the image.

Medical diagnosis

- In medical diagnosis, the input are the relevant information about the patient and the classes are the illness. The inputs contain the age of patient's, gender, past medical history and current symptoms.
- Some tests may not have been applied to the patient and thus these inputs would be missing. Tests take time, may be costly and may inconvience the patient so we do not want to apply them unless we believe that they will give us valuable information.
- In the case of a medical diagnosis, a wrong decision may lead to a wrong or no treatment and in cases of doubt it is preferable that the classifier reject and defer decision to a human expert.

1.16.2 Google Home and Amazon Alexa

Amazon Alexa / Siri

- Every time Alexa or Siri make a mistake when responding to our request, it uses the data it receives based on how it responded to the original query to improve the next time. If an error was made, it takes that data and learns from it. If the response was favourable, the system notes that as well.
- Data and machine learning are responsible for the explosive growth of digital voice assistants. They continue o get better with the more experiences they have and the data they accumulate.
- When user make a request of Alexa, the microphone on the device records command. This recording is sent to over the internet to the cloud. If user are talking to Alexa, the recording is sent to Alexa Voice Services (AVS). This cloud-based service will review the recording and interpret user request. Then, the system will send a relevant response back to the device.
- Amazon breaks down user "orders" into individual sounds. It then consults a database containing various words' pronunciations to find which words most closely correspond to the combination of individual sounds.
- It then identifies important words to make sense of the tasks and carry out corresponding functions. For instance, if Alexa notices words like "sport" or "basketball", it would open the sports app.
- Amazon's servers send the information back to our device and Alexa may speak. If Alexa needs to say anything back, it would go through the same process described above, but in reverse order.

Google Home :

- Google services such as its image search and translation tools use sophisticated machine learning which allow computers to see, listen and speak in much the same way as human do.
- To perform its functions, Google Assistant relies on Artificial Intelligence (AI) technologies such as natural language processing and machine learning to understand what the user is saying and to make suggestions or act on that language input.
- The Google Home can play music, but it's primarily designed as a vehicle for Google Assistant -- Google's voice - activated virtual helper that's connected to the internet.
- The Google Home is always listening to its environment, but it won't record what we are saying or respond to our commands until we speak one of its pre-programmed wake words -- either "OK, Google" or "Hey, Google."
- TF-IDF, is a numerical statistic that is intended to reflect how important a word is to a document from collection corpus. It is often used as a weighting factor for searches of information retrieval, text mining and user modeling.
- The TD-IDF value increases proportionally to the number of times a word appears in the document but it is often offset by the frequency of the word in the corpus, which helps to adjust for the fact that some words appear more frequently.

1.16.3 Unmanned Vehicles

- An Unmanned Aerial Vehicle (UAV), sometimes known as a drone, is an aircraft or airborne system that is controlled remotely by an onboard computer or a human operator. The ground control station, aircraft components and various types of sensors make up the UAV system.
- UAVs are categorized depending on their endurance, weight and altitude range. They can be used for multiple commercial and military applications.
- Machine learning is the process of using, storing and finding patterns within massive amounts of data, which can eventually be fed into algorithms. It's basically a process of using the data accumulated by the machine or device that allows computers to develop their own algorithm so that humans won't have to create challenging algorithms manually.
- Unmanned ground vehicles are classified into two broad types, remotely operated and autonomous.
- Autonomous unmanned ground vehicles comprise several technologies that allow the machine to be self - acting and self - regulating, sans human intervention. The

technology was initially developed to aid ground forces in the transfer of heavy equipment.

- However, the technology has witnessed significant evolution over the years, giving rise to more tactical vehicles designed to assist in surveillance or IED search-and-destroy missions.
- For example, unmanned ships in the course of the voyage, the default route is to ensure the obstruction of the premise of a straight line navigation.

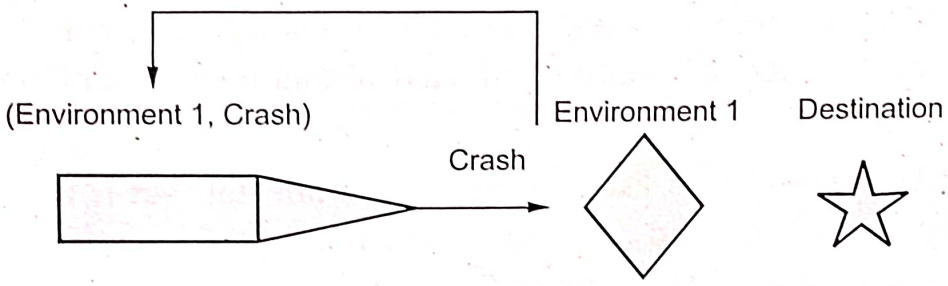


Fig. 1.16.1 First time lane

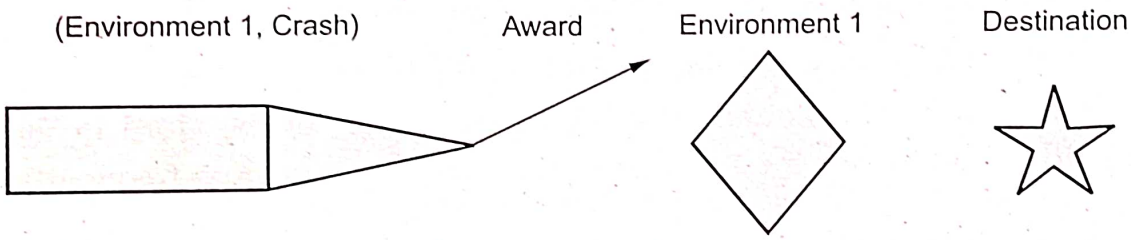


Fig. 1.16.2 The second time

- During the course of the voyage, the hull is changed by the intensity and direction of the waves and is unpredictable. It is clear for the unmanned boat itself.
- Therefore, unmanned ships in the process of navigation, continue to train the perception of the surrounding environment and make the appropriate strategy, if the results of the implementation of the strategy in line with the default route to be rewarded.

Review Question

1. *Describe the role of machine learning in the following applications :*
   a) Google home or Alexa b) Unmanned vehicles.

Unit II

2 Regression

Syllabus

Introduction - *Regression, Need of Regression, Difference between Regression and Correlation, Types of Regression : Univariate vs. Multivariate, Linear vs. Nonlinear, Simple Linear vs. Multiple Linear, Bias-Variance tradeoff, Overfitting and Underfitting.*

Regression Techniques - *Polynomial Regression, Stepwise Regression, Decision Tree Regression, Random Forest Regression, Support Vector Regression, Ridge Regression, Lasso Regression, ElasticNet Regression, Bayesian Linear Regression.*

Evaluation Metrics : *Mean Squared Error (MSE), Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), R-squared , Adjusted R-squared.*

Contents

2.1 Introduction
2.2 Regression
2.3 Types of Regression
2.4 Overfitting and Underfitting
2.5 Regression Techniques : Polynomial Regression
2.6 Support Vector Regression
2.7 Ridge Regression
2.8 Lasso Regression
2.9 ElasticNet Regression
2.10 Bayesian Linear Regression
2.11 Evaluation Metrics

2.1 Introduction

- Regression analysis is a set of statistical methods used for the estimation of relationships between a dependent variable and one or more independent variables. It can be utilized to assess the strength of the relationship between variables and for modelling the future relationship between them.
- The two basic types of regression are linear regression and multiple linear regression.
- Regression analysis includes several variations, such as linear, multiple linear and nonlinear. The most common models are simple linear and multiple linear. Nonlinear regression analysis is commonly used for more complicated data sets in which the dependent and independent variables show a nonlinear relationship.

Review Question

1. *How the performance of regression is assessed ? Write various performance metrics used for it.*

2.2 Regression

- For an input x, if the output is continuous, this is called a regression problem. For example, based on historical information of demand for tooth paste in your supermarket, you are asked to predict the demand for the next month.
- Regression is concerned with the prediction of continuous quantities. Linear regression is the oldest and most widely used predictive model in the field of machine learning. The goal is to minimize the sum of the squared errors to fit a straight line to a set of data points.

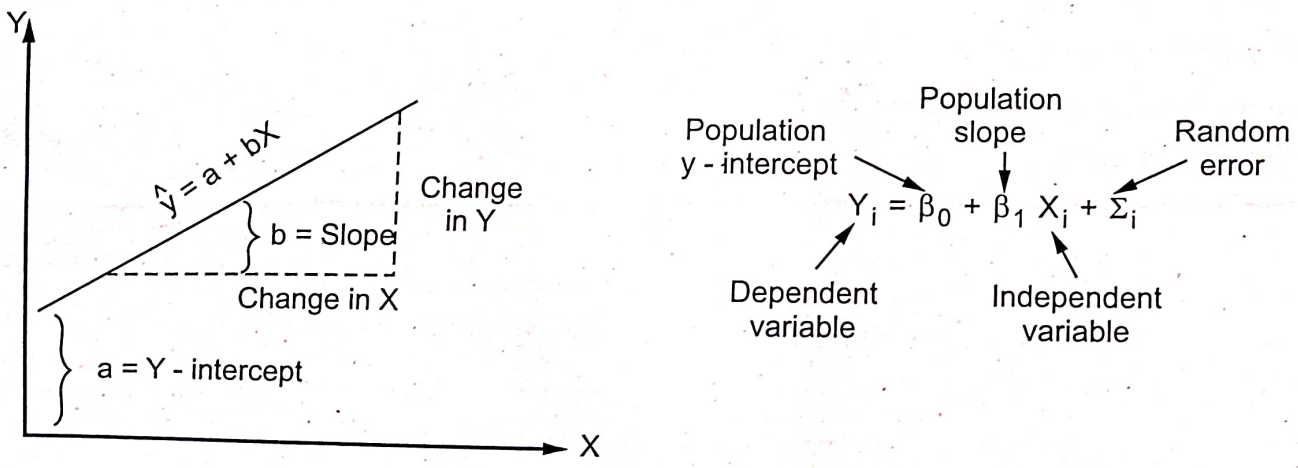


Fig. 2.2.1 Regression

- For regression tasks, the typical accuracy metrics are Root Mean Square Error (RMSE) and Mean Absolute Percentage Error (MAPE). These metrics measure the distance between the predicted numeric target and the actual numeric answer.

Regression Line

- **Least squares :** The least squares regression line is the line that makes the sum of squared residuals as small as possible. Linear means "straight line".
- **Regression line** is the line which gives the best estimate of one variable from the value of any other given variable.
- **The regression line** gives the average relationship between the two variables in mathematical form.
- For two variables X and Y, there are always two lines of regression.
- **Regression line of X on Y** gives the best estimate for the value of X for any specific given values of Y :

$$\mathbf{X = a + b\,Y}$$

where

a = X - intercept

b = Slope of the line

X = Dependent variable

Y = Independent variable

- **Regression line of Y on X :** gives the best estimate for the value of Y for any specific given values of X :

$$\mathbf{Y = a + bx}$$

where

a = Y - intercept

b = Slope of the line

Y = Dependent variable

x = Independent variable

- By using the least squares method (a procedure that minimizes the vertical deviations of plotted points surrounding a straight line) we are able to construct a best fitting straight line to the scatter diagram points and then formulate a regression equation in the form of :

$$\hat{y} = a + bX$$

$$\hat{y} = \overline{y} + b\,(x - \overline{x})$$

- Regression analysis is the art and science of fitting straight lines to patterns of data. In a linear regression model, the variable of interest ("dependent" variable) is predicted from k other variables ("independent" variables) using a linear equation. If Y denotes the dependent variable, and $X_1, \ldots, X_k$, are the independent variables, then the assumption is that the value of Y at time t in the data sample is determined by the linear equation :

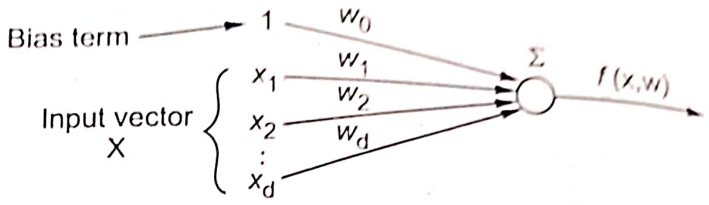


Fig. 2.2.2

$$Y_1 = \beta_0 + \beta_1 X_{1t} + \beta_2 X_{2t} + \ldots + \beta_k X_{kt} + \varepsilon_t$$

where the betas are constants and the epsilons are independent and identically distributed normal random variables with mean zero.

- In a regression tree the idea is this : since the target variable does not have classes, we fit a regression model to the target variable using each of the independent variables. Then for each independent variable, the data is split at several split points.
- At each split point, the "error" between the predicted value and the actual values is squared to get a "Sum of Squared Errors (SSE)". The split point errors across the variables are compared and the variable/point yielding the lowest SSE is chosen as the root node/split point. This process is recursively continued.
- Error function measures how much our predictions deviate from the desired answers.

$$\text{Mean-squared error } J_n = \frac{1}{n} \sum_{i=1 \ldots n} (y_i - f(x_i))^2$$

- **Multiple linear regression** is an extension of linear regression, which allows a response variable, y, to be modeled as a linear function of two or more predictor variables

Evaluating a Regression Model

- Assume we want to predict a car's price using some features such as dimensions, horsepower, engine specification, mileage etc. This is a typical regression problem, where the target variable (price) is a continuous numeric value.
- We can fit a simple linear regression model that, given the feature values of a certain car, can predict the price of that car. This regression model can be used to score the same dataset we trained on. Once we have the predicted prices for all of the cars, we can evaluate the performance of the model by looking at how much the predictions deviate from the actual prices on average.

Advantages :

a. Training a linear regression model is usually much faster than methods such as neural networks.

b. Linear regression models are simple and require minimum memory to implement.

c. By examining the magnitude and sign of the regression coefficients you can infer how predictor variables affect the target outcome.

2.2.1 Need of Regression

- Regression is a technique for investigating the relationship between independent variables or features and a dependent variable or outcome. It's used as a method for predictive modelling in machine learning, in which an algorithm is used to predict continuous outcomes.
- Regression models are used to predict a continuous value. Predicting prices of a house given the features of house like size, price is one of the common examples of Regression. It is a supervised technique.
- Regression analysis is a way to find trends in data. For example, we might guess that there's a connection between how much we eat and how much is the weigh; regression analysis can help us quantify that equation. Regression analysis will provide with an equation for a graph so that we can make predictions about our data.
- Regression is essentially the "best guess" at utilizing a collection of data to generate some form of forecast. It is the process of fitting a set of points to a graph.
- Briefly, the goal of regression model is to build a mathematical equation that defines y as a function of the x variables. Next, this equation can be used to predict the outcome (y) on the basis of new values of the predictor variables (x).
- Linear regression is the most simple and popular technique for predicting a continuous variable. It assumes a linear relationship between the outcome and the predictor variables.
- In some cases, the relationship between the outcome and the predictor variables is not linear. In these situations, we need to build a non-linear regression, such as polynomial and spline regression.
- When we have multiple predictors in the regression model, we might want to select the best combination of predictor variables to build an optimal predictive model. This process called model selection, consists of comparing multiple models containing different sets of predictors in order to select the best performing model

that minimize the prediction error. Linear model selection approaches include best subsets regression and stepwise regression.

2.2.2 Difference between Regression and Correlation

Regression	Correlation
Regression tells us how to draw the straight line described by the correlation.	Correlation describes the strength of a linear relationship between two variables.
For regression only the dependent variable Y must be random.	For correlation, both variables should be random variables.
Main goal is use the measure of relation to predict values of the random variable based on values of the fixed variable.	Main goal is simply to find a number that expresses the relation between the variables.

2.3 Types of Regression

- The most common regression algorithms are,
  a) Simple linear regression
  b) Multiple linear regression
  c) Polynomial regression
  d) Multivariate adaptive regression splines
  e) Logistic regression
  f) Maximum likelihood estimation (Least squares)

2.3.1 Univariate Vs. Multivariate

Sr. No.	Univariate	Multivariate
1.	Univariate analysis refers to the analysis of one variable.	Multivariate analysis refers to the analysis of more than one variable.
2.	It does not deal with causes and relationships	It deal with causes and relationships
3.	It does not contain any dependent variable.	It contains more than one dependent variable
4.	Equation : $Y = A + BX$	Equation : $Y = A + BX + CX_1$

2.3.2 Linear Vs. Nonlinear

- Linear regression is an approach for modelling dependent variable (y) and one or more explanatory variables (x).

- Equation : $y = \beta_0 + \beta_1 x + \varepsilon$
- Nonlinear regression arises when predictors and response follows particular function form.
- Equation : $y = f\ \beta, x + \varepsilon$
- The nonlinear model is more flexible and accurate. Although both models can accommodate curvature, the nonlinear model is significantly more versatile in terms of the forms of the curves it can accept.

2.3.3 Simple Linear Vs. Multiple Linear

Simple regression	Multiple regression
One dependent variable Y predicted from one independent variable X.	One dependent variable Y predicted from a set of independent variables ($X_1, X_2 \ldots X_k$).
One regression coefficient.	One regression coefficient for each independent variable.
r^2 : Proportion of variation in d.ependent variable Y predictable from X.	R^2 : Proportion of variation in dependent variable Y predictable by set of independent variables (X's).

2.4 Overfitting and Underfitting

- In addition to using models for prediction, the ability to interpret what a model has learned is receiving an increasing amount of attention.
- Interpretability has to do with how accurate a machine learning model can associate a cause to an effect.
- If a model can take the inputs, and routinely get the same outputs, the model is interpretable :
  1. If you overeat your magi at dinnertime and you always have troubles sleeping, the situation is interpretable.
  2. If all 2019 polls showed " ABC party" win and the "XYZ party" candidate took office, all those models showed low interpretability.
- Interpretability poses no issue in low-risk scenarios. If a model is recommending movies to watch, that can be a low-risk task
- 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.
- Training error can be reduced by making the hypothesis more sensitive to training data, but this may lead to overfitting and poor generalization.

- Overfitting occurs when a statistical model describes random error or noise instead of the underlying relationship. Overfitting is when a classifier fits the training data too tightly. Such a classifier works well on the training data but not on independent test data. It is a general problem that plagues all machine learning methods.
- Underfitting : If we put too few variables in the model, leaving out variables that could help explain the response, we are **underfitting.** Consequences :
  1. Fitted model is not good for prediction of new data - prediction is biased
  2. Regression coefficients are biased
  3. Estimate of error variance is too large
- Because of overfitting, low error on training data and high error on test data. Overfitting occurs when a model begins to memorize training data rather than learning to generalize from trend.
- The more difficult a criterion is to predict, the more noise exists in past information that need to be ignored. The problem is determining which part to ignore.
- Overfitting generally occurs when a model is excessively complex, such as having too many parameters relative to the number of observations. We can determine whether a predictive model is underfitting or overfitting the training data by looking at the prediction error on the training data and the evaluation data. Fig. 2.4.1 shows underfiting and overfiting.

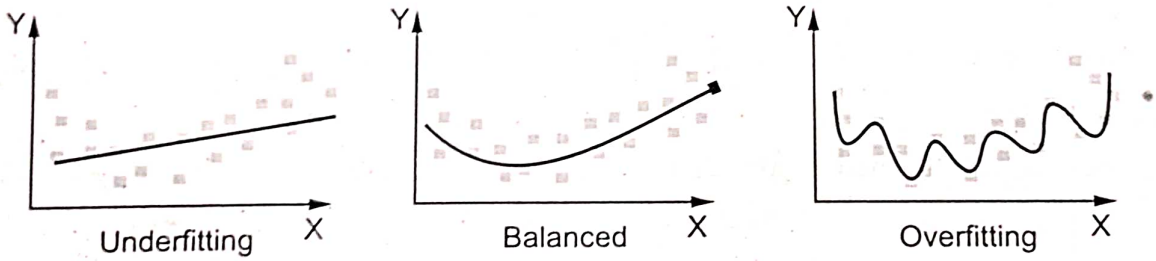


Fig. 2.4.1

- Reasons for overfitting
  1. Noisy data
  2. Training set is too small
  3. Large number of features

- In the machine learning the more complex model is said to show signs of overfitting, while the simpler model underfitting. Often several heuristic are developed in order to avoid overfitting, for example, when designing neural networks one may :
  1. Limit the number of hidden nodes.
  2. Stop training early to avoid a perfect explanation of the training set, and
  3. Apply weight decay to limit the size of the weights, and thus of the function class implemented by the network.

2.4.1 Bias Vs Variance

- In the experimental practice we observe an important phenomenon called the bias variance dilemma.
- 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'.
- Fig. 2.4.2 shows

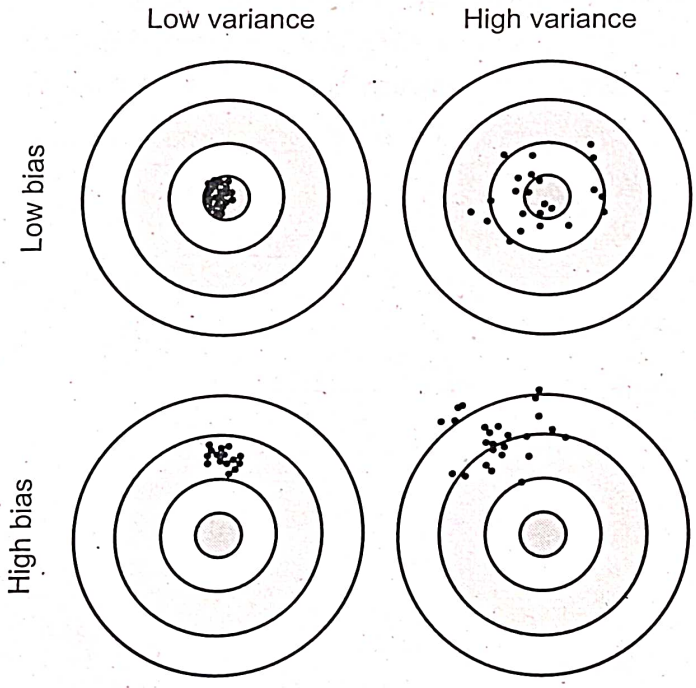


Fig. 2.4.2 Bias-variance trade off

- Give two classes of hypothesis (e.g. linear models and k-NNs) to fit to some training data set, we observe that the more flexible hypothesis class has a low bias term but a higher variance term. If we have parametric family of hypothesis, then we can increases the flexibility of the hypothesis but we still observe the increase of variance.

- The bias-variance-dilemma is the problem of simultaneously minimizing two sources of error that prevent supervised learning algorithm from generalizing beyond their training set :
  1. The bias is error from erroneous assumptions in the learning algorithm. High bias can cause an algorithm to miss the relevant relations between features and target outputs.
  2. The variance is error from sensitivity to small fluctuations in the training set. High variance can cause overfitting : modeling the random noise in the training data, rather than the intended outputs.
- In order to reduce the model error, the designer can aim at reducing either the bias or the variance, as the noise components is irreducible.
- As the model increases in complexity, its bias is likely to diminish. However, as the number of training examples is kept fixed, the parametric identification of the model may strongly vary from one DN to another. This will increase the variance term.
- At one stage, the decrease in bias will be inferior to the increase in variance, warning that the model should not be too complex. Conversely, to decrease the variance term, the designer has to simplify its model so that it is less sensitive to a specific training set. This simplification will lead to a higher bias.

Example 2.4.1

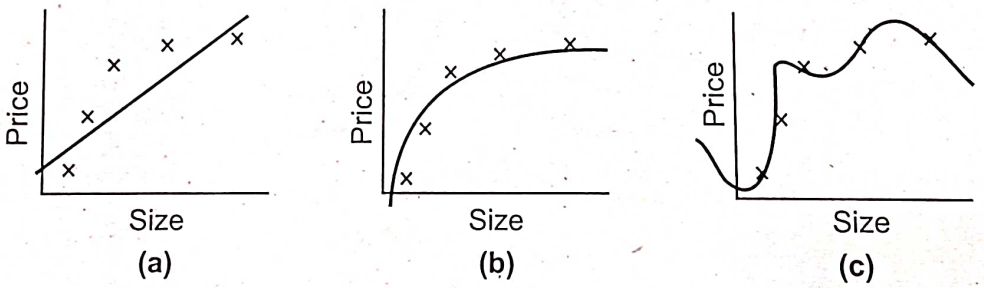


Fig. 2.4.3

Explain the aboe Fig. 2.4.3 (a), (b) and (c).

Solution :

- Given Fig. 2.4.3 is related to overfitting and underfitting.

Underfitting (High bias and low variance) :

- A statistical model or a machine learning algorithm is said to have underfitting when it cannot capture the underlying trend of the data.
- It usually happens when we have less data to build an accurate model and also when we try to build a linear model with a non-linear data.

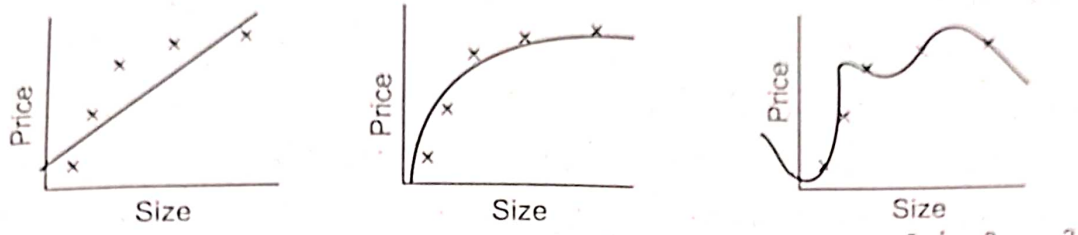


Fig. 2.4.4

- In such cases the rules of the machine learning model are too easy and flexible to be applied on such minimal data and therefore the model will probably make a lot of wrong predictions.
- Underfitting can be avoided by using more data and also reducing the features by feature selection.

Overfitting (High variance and low bias) :

- A statistical model is said to be overfitted, when we train it with a lot of data.
- When a model gets trained with so much of data, it starts learning from the noise and inaccurate data entries in our data set.
- Then the model does not categorize the data correctly, because of too many details and noise.
- The causes of overfitting are the non-parametric and non-linear methods because these types of machine learning algorithms have more freedom in building the model based on the dataset and therefore they can really build unrealistic models.
- A solution to avoid overfitting is using a linear algorithm if we have linear data or using the parameters like the maximal depth if we are using decision trees.

2.4.2 Difference between Overfitting and Underfitting

Sr. No	Parameter	Overfitting	Underfitting
1.	Complexity	It is too complex	Model is too simple
2.	Reason	Low bias, High variance	High bias, low variance
3.	Quantity of features	Smaller quantity of features.	A larger quantity of feature.
4.	Regularization	More regularization	Less regularization

Review Questions

1. *Explain the term bias-variance dilemma.*
2. *What is overfitting and underfitting ? What are the catalysts of overfitting ?*
3. *Elaborate bias variance dilemma.*
4. *Explain with example K-fold cross validation.*
5.

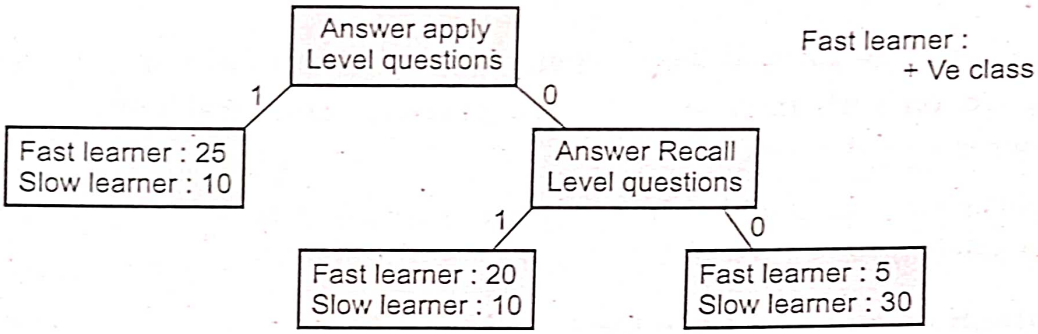


Fig. 2.4.5

i) Find contingency table

ii) Find recall *iii) Precision*

iv) Negative recall *v) False positive rate*

6. *Difference between overfitting and underfitting.*

2.5 Regression Techniques : Polynomial Regression

- Polynomial Regression is a regression algorithm that models the relationship between a dependent(y) and independent variable(x) as n^{th} degree polynomial.
- The Polynomial Regression equation is given below :

$$y = b_0 + b_1x_1 + b_2x_1^2 + b_3x_1^3 + \ldots + b_nx_1^n$$

- Polynomial regression is needed when there is no linear correlation fitting all the variables. So instead of looking like a line, it looks like a nonlinear function. Fig. 2.5.1 shows polynomial regression.
- If the datasets are arranged in a non-linear fashion, then we should use the Polynomial

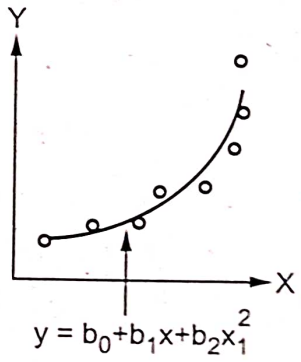


Fig. 2.5.1 Polynomial regression

Regression model instead of Simple Linear Regression. Polynomial models are very powerful to handle nonlinearity, because polynomials can approximate continuous functions within any given precision.

- There are two standard procedures for building a polynomial model :
  1. **Forward selection :** Successively fit models of increasing order until the t test for the highest order term is non-significant.
  2. **Backward elimination :** Appropriately fit the highest order model and then delete terms one at a time, starting with the highest order, until the highest order remaining term has a significant t statistic.

Advantages :

- We can model non-linear relationships between variables.
- There is a large range of different functions that you can use for fitting.
- Good for exploration purposes : You can test for the presence of curvature and its inflections.

2.5.1 Stepwise Regression

- Stepwise regression is a step-by-step process of constructing a model by introducing or eliminating predictor variables. First, the variables undergo T-tests and F-tests. Then, predictor variables are individually tested to fit a linear regression model.
- Stepwise regression drops insignificant variables one by one. This is particularly useful if we have many potential explanatory variables.
- Stepwise regression is used to design a regression model to introduce only relevant and statistically significant variables. Other variables are discarded. However, every regression calculation contains unwanted variables. These variables are predictive and complicate the process unnecessarily.
- The stepwise regression consists of iteratively adding and removing predictors, in the predictive model, in order to find the subset of variables in the data set resulting in the best performing model, that is a model that lowers prediction error.
- There are three strategies of stepwise regression :
  1. Forward selection, which starts with no predictors in the model, iteratively adds the most contributive predictors and stops when the improvement is no longer statistically significant.

2. Backward selection which starts with all predictors in the model (full model), iteratively removes the least contributive predictors and stops when you have a model where all predictors are statistically significant.
3. Stepwise selection which is a combination of forward and backward selections. We start with no predictors, then sequentially add the most contributive predictors. After adding each new variable, remove any variables that no longer provide an improvement in the model fit.

2.5.2 Decision Tree Regression

- Decision Tree is a decision-making tool that uses a flowchart-like tree structure or is a model of decisions and all of their possible results, including outcomes, input costs and utility.
- Decision-tree algorithm falls under the category of supervised learning algorithms. It works for both continuous as well as categorical output variables.
- Decision tree regression observes features of an object and trains a model in the structure of a tree to predict data in the future to produce meaningful continuous output. Continuous output means that the output/result is not discrete, i.e., it is not represented just by a discrete, known set of numbers or values.
- Discrete output example : A weather prediction model that predicts whether or not there'll be rain in a particular day.
- Continuous output example : A profit prediction model that states the probable profit that can be generated from the sale of a product.
- Here, continuous values are predicted with the help of a decision tree regression model.
- A decision tree is able to make a prediction by running through the entire tree, asking true/false questions, until it reaches a leaf node. The final prediction is given by the average of the value of the dependent variable in that leaf node.

2.5.3 Random Forest Regression

- Random forest is a famous system learning set of rules that belongs to the supervised getting to know method. It may be used for both classification and regression issues in ML. It is based totally on the concept of ensemble studying, that's a process of combining multiple classifiers to solve a complex problem and to enhance the overall performance of the model.

- As the call indicates, "Random forest is a classifier that incorporates some of choice timber on diverse subsets of the given dataset and takes the average to improve the predictive accuracy of that dataset." Instead of relying on one decision tree, the random forest takes the prediction from each tree and primarily based on most of the people's votes of predictions, and it predicts the very last output.
- The more wider variety of trees within the forest results in better accuracy and prevents the hassle of overfitting.

2.5.3.1 How does Random Forest Algorithm Work ?

- Random forest works in two-section first is to create the random woodland by combining N selection trees and second is to make predictions for each tree created inside the first segment.
- The working technique may be explained within the below steps and diagram :

Step - 1 : Select random K statistics points from the schooling set.

Step - 2 : Build the selection trees associated with the selected information points (Subsets).

Step - 3 : Choose the wide variety N for selection trees which we want to build.

Step - 4 : Repeat step 1 and 2.

Step - 5 : For new factors, locate the predictions of each choice tree and assign the new records factors to the category that wins most people's votes.

- The working of the set of rules may be higher understood by the underneath example :
- Example : Suppose there may be a dataset that includes more than one fruit photo. So, this dataset is given to the random wooded area classifier. The dataset is divided into subsets and given to every decision tree. During the training section, each decision tree produces a prediction end result and while a brand new statistics point occurs, then primarily based on the majority of consequences, the random forest classifier predicts the final decision. Consider the underneath picture :

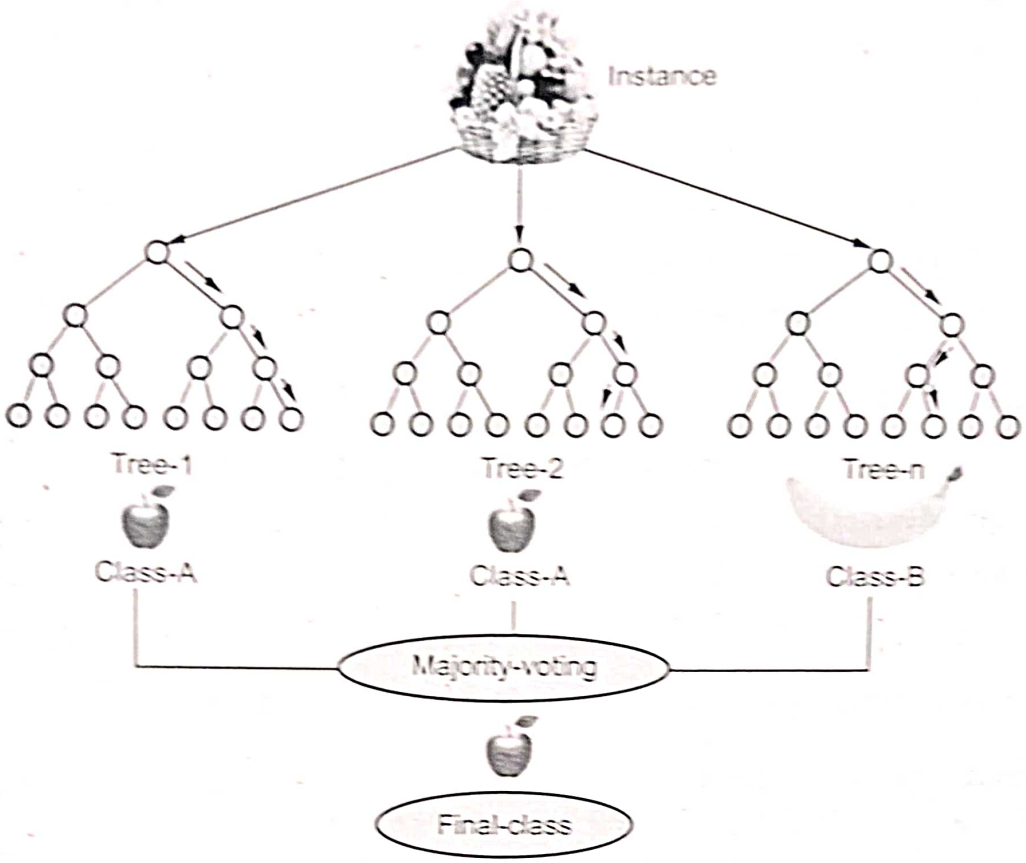


Fig. 2.5.2 Example of random forest

2.5.3.2 Applications of Random Forest

There are specifically 4 sectors where random forest normally used :

1. **Banking :** Banking zone in general uses this algorithm for the identification of loan danger.
2. **Medicine :** With the assistance of this set of rules, disorder traits and risks of the disorder may be recognized.
3. **Land use :** We can perceive the areas of comparable land use with the aid of this algorithm.
4. **Marketing :** Marketing tendencies can be recognized by the usage of this algorithm.

2.5.3.3 Advantages of Random Forest

- Random forest is able to appearing both classification and regression responsibilities.
- It is capable of managing large datasets with high dimensionality.
- It enhances the accuracy of the version and forestalls the overfitting trouble.

2.5.3.4 Disadvantages of Random Forest

- Although random forest can be used for both class and regression responsibilities, it isn't extra appropriate for regression obligations.

2.6 Support Vector Regression

- Support Vector Machines (SVMs) are a set of supervised learning methods which learn from the dataset and used for classification. SVM is a classifier derived from statistical learning theory by Vapnik and Chervonenkis.

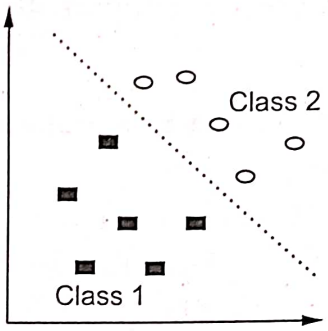


Fig. 2.6.1 Two class problem

- An SVM is a kind of large-margin classifier: it is a vector space based machine learning method where the goal is to find a decision boundary between two classes that is maximally far from any point in the training data

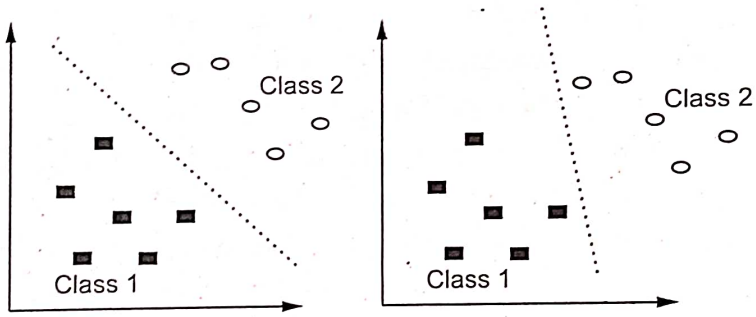


Fig. 2.6.2 Bad decision boundary of SVM

- Given a set of training examples, each marked as belonging to one of two classes, an SVM algorithm builds a model that predicts whether a new example falls into one class or the other. Simply speaking, we can think of an SVM model as representing the examples as points in space, mapped so that each of the examples of the separate classes are divided by a gap that is as wide as possible.
- New examples are then mapped into the same space and classified to belong to the class based on which side of the gap they fall on.

Two Class Problems

- Many decision boundaries can separate these two classes. Which one should we choose ?

- Perceptron learning rule can be used to find any decision boundary between class 1 and class 2.
- The line that maximizes the minimum margin is a good bet. The model class of "hyper-planes with a margin of m" has a low VC dimension if m is big.
- This maximum-margin separator is determined by a subset of the data points. Data points in this subset are called "support vectors". It will be useful computationally if only a small fraction of the data points are support vectors, because we use the support vectors to decide which side of the separator a test case is on.

Example of Bad Decision Boundaries

- SVM are primarily two-class classifiers with the distinct characteristic that they aim to find the optimal hyperplane such that the expected generalization error is minimized. Instead of directly minimizing the empirical risk calculated from the training data, SVMs perform structural risk minimization to achieve good generalization.
- The empirical risk is the average loss of an estimator for a finite set of data drawn from P. The idea of risk minimization is not only measure the performance of an estimator by its risk, but to actually search for the estimator that minimizes risk over distribution P. Because we don't know distribution P we instead minimize empirical risk over a training dataset drawn from P. This general learning technique is called **empirical risk minimization.**

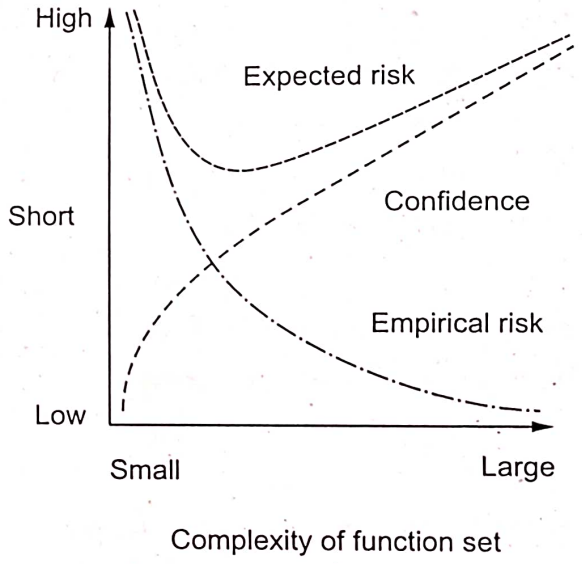


Fig. 2.6.3 Empirical risk

- Fig. 2.6.3 shows empirical risk.

Good Decision Boundary : Margin Should Be Large

- The decision boundary should be as far away from the data of both classes as possible. If data points lie very close to the boundary, the classifier may be consistent but is more "likely" to make errors on new instances from the distribution. Hence, we prefer classifiers that maximize the minimal distance of data points to the separator.

1. **Margin (m)** : the gap between data points & the classifier boundary. The Margin is the minimum distance of any sample to the decision boundary. If this hyperplane is in the canonical form, the margin can be measured by the length of the weight vector. The margin is given by the projection of the distance between these two points on the direction perpendicular to the hyperplane. **Margin** of the separator is the distance between support vectors.

$$\text{Margin (m)} = \frac{2}{||w||}$$

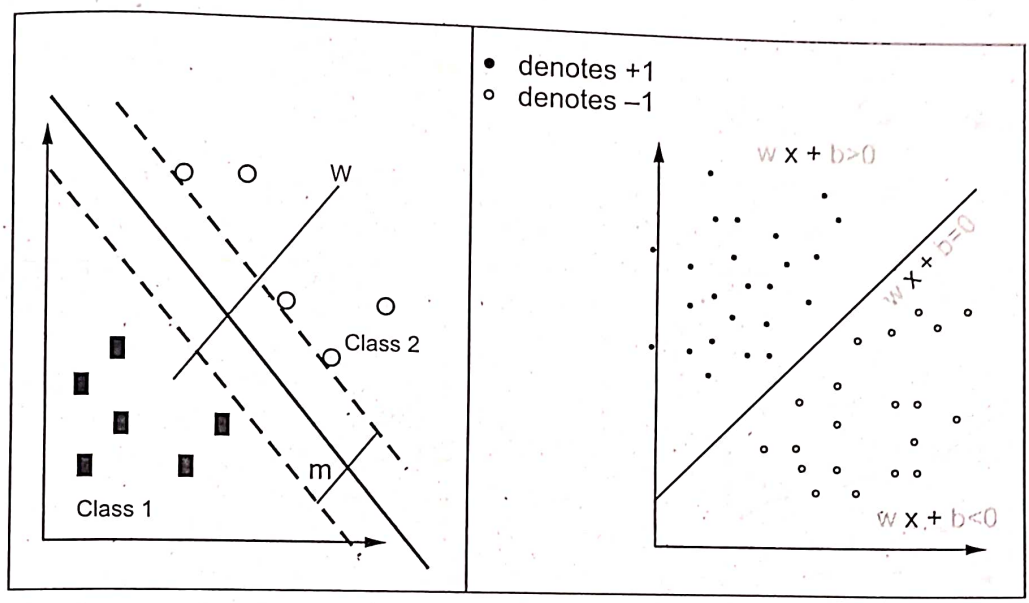


Fig. 2.6.4 Good decision boundary

2. **Maximal margin classifier :** a classifier in the family F that maximizes the margin. Maximizing the margin is good according to intuition and PAC theory. Implies that only support vectors matter; other training examples are ignorable.

Example 2.6.1 *For the following figure find a linear hyperplane (decision boundary) that will separate the data.*

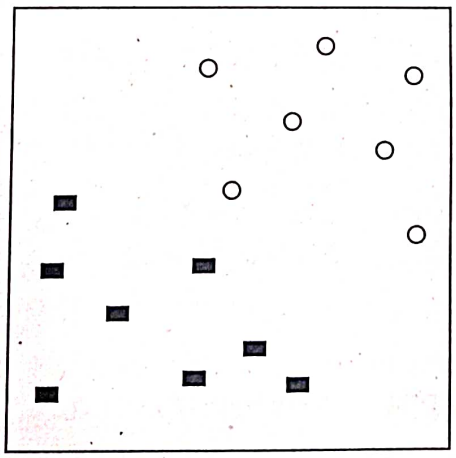


Fig. 2.6.5

Solution :

1. Define what an optimal hyperplane is : **maximize margin**
2. Extend the above definition for non-linearly separable problems : **have a penal term for misclassifications**

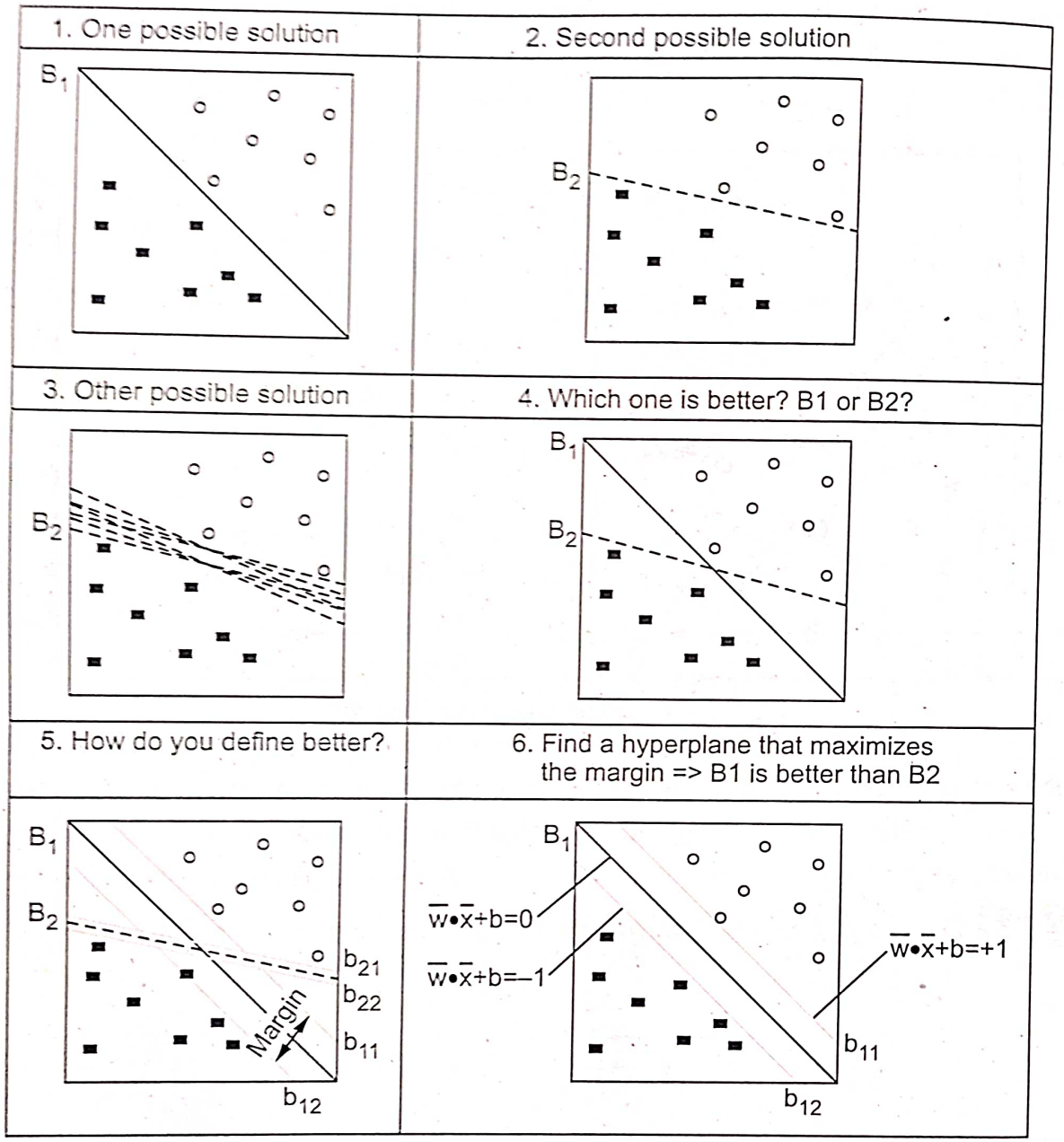


Fig. 2.6.6

3. Map data to high dimensional space where it is easier to classify with linear decision surfaces : **reformulate problem so that data is mapped implicitly to this space**

2.6.1 Key Properties of Support Vector Machines

1. Use a single hyperplane which subdivides the space into two half-spaces, one which is occupied by Class 1 and the other by Class 2.

2. They maximize the margin of the decision boundary using quadratic optimization techniques which find the optimal hyperplane.
3. Ability to handle large feature spaces.
4. Overfitting can be controlled by soft margin approach.
5. When used in practice, SVM approaches frequently map the examples to a higher dimensional space and find margin maximal hyperplanes in the mapped space, obtaining decision boundaries which are not hyperplanes in the original space.
6. The most popular versions of SVMs use non-linear kernel functions and map the attribute space into a higher dimensional space to facilitate finding "good" linear decision boundaries in the modified space.

2.6.2 SVM Applications

- SVM has been used successfully in many real-world problems,
  1. Text (and hypertext) categorization
  2. Image classification
  3. Bioinformatics (Protein classification, Cancer classification)
  4. Hand-written character recognition
  5. Determination of SPAM email.

2.6.3 Limitations of SVM

1. It is sensitive to noise.
2. The biggest limitation of SVM lies in the choice of the kernel.
3. Another limitation is speed and size.
4. The optimal design for multiclass SVM classifiers is also a research area.

2.6.4 Soft Margin SVM

- For the very high dimensional problems common in text classification, sometimes the data are linearly separable. But in the general case they are not, and even if they are, we might prefer a solution that better separates the bulk of the data while ignoring a few weird noise documents.
- What if the training set is not linearly separable ? *Slack variables* can be added to allow misclassification of difficult or noisy examples, resulting margin called *soft*.
- A *soft-margin* allows a few variables to cross into the margin or over the hyperplane, allowing misclassification.
- We penalize the crossover by looking at the number and distance of the misclassifications. This is a trade off between the hyperplane violations and the

margin size. The slack variables are bounded by some set cost. The farther they are from the soft margin, the less influence they have on the prediction.

- All observations have an associated slack variable,
  1. Slack variable = 0 then all points on the margin.
  2. Slack variable > 0 then a point in the margin or on the wrong side of the hyperplane
  3. C is the tradeoff between the slack variable penalty and the margin.

2.6.5 Comparison of SVM and Neural Networks

Support Vector Machine	Neural Network
Kernel maps to a very-high dimensional space	Hidden Layers map to lower dimensional spaces
Search space has a unique minimum	Search space has multiple local minima
	Classification extremely efficient
Very good accuracy in typical domains	Very good accuracy in typical domains
Kernel and cost the two parameters to select	Requires number of hidden units and layers
Training is extremely efficient	Training is expensive

Example 2.6.2 *From the following diagram, identify which data points (1, 2, 3, 4, 5) are support vectors (if any), slack variables on correct side of classifier (if any) and slack variables on wrong side of classifier (if any). Mention which point will have maximum penalty and why ?*

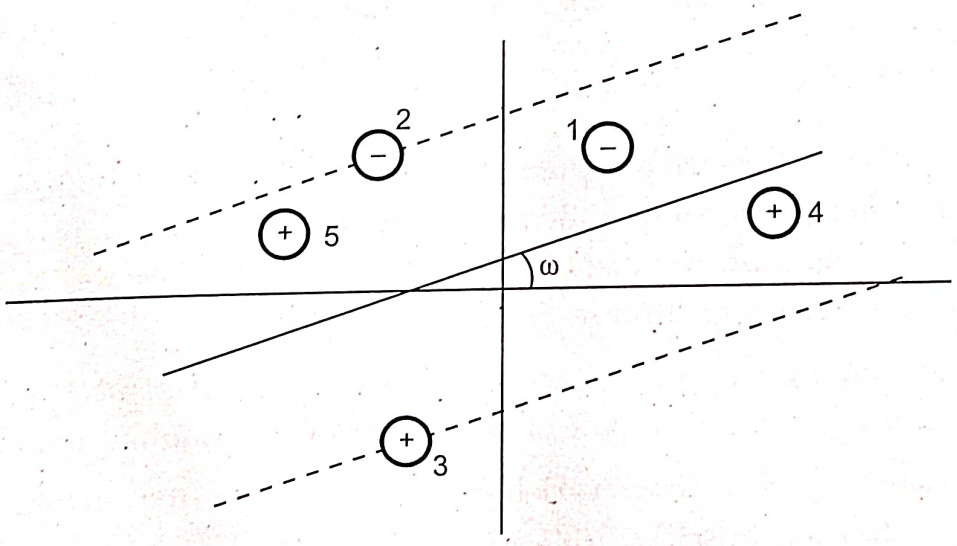


Fig. 2.6.7

Solution :

- Data points 1 and 5 will have maximum penalty.
- Margin (m) is the gap between data points & the classifier boundary. The margin is the minimum distance of any sample to the decision boundary. If this hyperplane is in the canonical form, the margin can be measured by the length of the weight vector.
- Maximal margin classifier : A classifier in the family F that maximizes the margin. Maximizing the margin is good according to intuition and PAC theory. Implies that only support vectors matter; other training examples are ignorable.
- What if the training set is not linearly separable ? Slack variables can be added to allow misclassification of difficult or noisy examples, resulting margin called soft.
- A soft-margin allows a few variables to cross into the margin or over the hyperplane, allowing misclassification.
- We penalize the crossover by looking at the number and distance of the misclassifications. This is a trade off between the hyperplane violations and the margin size. The slack variables are bounded by some set cost. The farther they are from the soft margin, the less influence they have on the prediction.
- All observations have an associated slack variable

1. Slack variable = 0 then all points on the margin.
2. Slack variable > 0 then a point in the margin or on the wrong side of the hyperplane.
3. C is the tradeoff between the slack variable penalty and the margin.

Review Questions

1. *What are support vectors and margins ? Also explain soft margin SVM.*
2. *What is slack variable ? Discuss margin errors.*
3. *Explain support vector machine.*
4. *What are support vectors and margins ? Also explain soft margin SVM.*
5. *What is slack variable ? Discuss margin errors.*
6. *Explain support vector machine.*
7. *What are the support vectors and margins ? Explain soft SVM and hard SVM.*

2.7 Ridge Regression

- Ridge regression is one of the maximum sturdy versions of linear regression wherein a small quantity of bias is delivered so that we will get higher long term predictions.

- The quantity of bias added to the version is called ridge regression penalty. We can compute this penalty term with the aid of multiplying with the lambda to the squared weight of each individual features.
- The equation for ridge regression might be :

$$L(x, y) = \text{Min}\left(\sum_{i=1}^{n} (y_i - w_i x_i)^2 + \lambda \sum_{i=1}^{n} (w_i)^2 \right)$$

- A standard linear or polynomial regression will fail if there may be high collinearity among the unbiased variables, to be able to solve such problems, ridge regression may be used.
- Ridge regression is a regularization method, which is used to reduce the complexity of the version. It is likewise referred to as L2 regularization.
- It helps to clear up the troubles if we have greater parameters than samples.

2.7.1 Scikit Learn Code for Ridge Regression

```
from sklearn.linear_model import Ridge
import numpy as np
n_samples, n_features = 24, 19
rng = np.random.RandomState(0)
y = rng.randn(n_samples)
X = rng.randn(n_samples, n_features)
rdg = Ridge(alpha = 0.5)
rdg.fit(X, y)
rdg.score(X,y)
```

2.8 Lasso Regression

- Lasso regression is any other regularization technique to lessen the complexity of the version.
- It is similar to the ridge regression except that penalty time period includes only the absolute weights instead of a rectangular of weights.
- Since it takes absolute values, consequently, it is able to shrink the slope to 0, while ridge regression can simplest cut back it close to zero.

- It is likewise known as as L1 regularization. The equation for Lasso regression may be :

$$L(x, y) = \text{Min}\left(\sum_{i=1}^{n} (y_i - w_i x_i)^2 + \lambda \sum_{i=1}^{n} |w_i| \right)$$

2.9 ElasticNet Regression

- Elastic net is a combination of the two most popular regularized variants of linear regression : Ridge and Lasso. Ridge utilizes an L2 penalty and lasso uses an L1 penalty.
- This allows it to balance between feature selection and feature preservation, and to deal with situations where lasso and ridge regression may fail.
- For example, when there are more features than observations, lasso regression may select only one feature from a group of correlated features, while ridge regression may keep them all. Elastic net regression can select a subset of correlated features and avoid the instability of lasso regression.

Advantages

1. It can reduce model complexity by eliminating irrelevant features, which is more effective than ridge regression.
2. Elastic net regression can achieve a better trade-off between bias and variance than lasso and ridge regression by tuning the regularization parameters.
3. This type of regression can be applied to various types of data, such as linear, logistic or Cox regression models.

Disadvantages

1. It requires more computational resources and time due to two regularization parameters and a cross-validation process.
2. It may not be easily interpretable, as it could select a large number of features with small coefficients or a small number of features with large coefficients.

2.10 Bayesian Linear Regression

- Bayesian linear regression allows a useful mechanism to deal with insufficient data, or poor distributed data. It allows user to put a prior on the coefficients and on the noise so that in the absence of data, the priors can take over. A prior is a distribution on a parameter.
- If we could flip the coin an infinite number of times, inferring its bias would be easy by the law of large numbers. However, what if we could only flip the coin a handful of times? Would we guess that a coin is biased if we saw three heads in

three flips, an event that happens one out of eight times with unbiased coins? The MLE would overfit these data, inferring a coin bias of p = 1.

- A Bayesian approach avoids overfitting by quantifying our prior knowledge that most coins are unbiased, that the prior on the bias parameter is peaked around one-half. The data must overwhelm this prior belief about coins.
- Bayesian methods allow us to estimate model parameters, to construct model forecasts and to conduct model comparisons. Bayesian learning algorithms can calculate explicit probabilities for hypotheses.
- Bayesian classifiers use a simple idea that the training data are utilized to calculate an observed probability of each class based on feature values.
- When Bayesian classifier is used for unclassified data, it uses the observed probabilities to predict the most likely class for the new features.
- Each observed training example can incrementally decrease or increase the estimated probability that a hypothesis is correct.
- Prior knowledge can be combined with observed data to determine the final probability of a hypothesis. In Bayesian learning, prior knowledge is provided by asserting a prior probability for each candidate hypotheses and a probability distribution over observed data for each possible hypothesis.
- Bayesian methods can accommodate hypotheses that make probabilistic predictions. New instances can be classified by combining the predictions of multiple hypotheses, weighted by their probabilities.
- Even in cases where Bayesian methods prove computationally intractable, they can provide a standard of optimal decision making against which other practical methods can be measured.
- Uses of Bayesian classifiers are as follows :
  1. Used in text-based classification for finding spam or junk mail filtering.
  2. Medical diagnosis.
  3. Network security such as detecting illegal intrusion.
- The basic procedure for implementing Bayesian Linear Regression is :
  i) Specify priors for the model parameter.
  ii) Create a model mapping the training inputs to the training outputs.
  iii) Have a Markov Chain Monte Carlo (MCMC) algorithm draw samples from the posterior distributions for the parameters.

2.11 Evaluation Metrics

- Mean Squared Error (MSE), and Mean Absolute Error (MAE) are used to evaluate the regression problem's accuracy.

2.11.1 Mean Squared Error

- Mean Squared Error (MSE) is calculated by taking the average of the square of the difference between the original and predicted values of the data. It can also be called the quadratic cost function or sum of squared errors.
- The value of MSE is always positive or greater than zero. A value close to zero will represent better quality of the estimator/predictor. An MSE of zero (0) represents the fact that the predictor is a perfect predictor.

$$MSE = \frac{1}{N}\sum_{i=1}^{n}(\text{Actual values} - \text{Predicted values})^2$$

- Here N is the total number of observations/rows in the dataset. The sigma symbol denotes that the difference between actual and predicted values taken on every i value ranging from 1 to n.

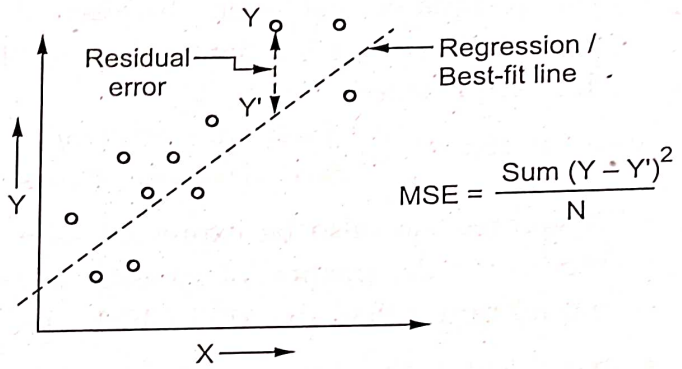


Fig. 2.11.1 Representation of MSE

- Mean squared error is the most commonly used loss function for regression. MSE is sensitive towards outliers and given several examples with the same input feature values, the optimal prediction will be their mean target value. This should be compared with Mean Absolute Error, where the optimal prediction is the median. MSE is thus good to use if you believe that your target data, conditioned on the input, is normally distributed around a mean value, and when it's important to penalize outliers extra much.
- MSE incorporates both the variance and the bias of the predictor. MSE also gives more weight to larger differences. The bigger the error, the more it is penalized.
- Example : You want to predict future house prices. The price is a continuous value, and therefore we want to do regression. MSE can here be used as the loss function.

2.11.2 Mean Absolute Error

- MAE is the sum of absolute differences between our target and predicted variables. So it measures the average magnitude of errors in a set of predictions, without considering their directions.

- The loss is the mean overseen data of the absolute differences between true and predicted values

$$MAE = \frac{1}{n}\sum_{j=1}^{n}|y_i - \hat{y}_j|$$

- Use mean absolute error when you are doing regression and don't want outliers to play a big role. It can also be useful if you know that your distribution is multimodal. MAE loss is useful if the training data is corrupted with outliers.

2.11.3 R-square

- R-squared is also known as the coefficient of determination. This metric gives an indication of how good a model fits a given dataset. It indicates how close the regression line is to the actual data values.
- The R squared value lies between 0 and 1 where 0 indicates that this model doesn't fit the given data and 1 indicates that the model fits perfectly to the dataset provided.

$$\text{R-squared} = 1 - \frac{\text{First sum of errors}}{\text{Second sum of errors}}$$

- R-squared can also be expressed as a function of mean squared error. R-squared represents the fraction of variance of response variable captured by the regression model rather than the MSE which captures the residual error.
- Specifically, this linear regression is used to determine how well a line fits' to a data set of observations, especially when comparing models. Also, it is the fraction of the total variation in y that is captured by a model. Or, how well does a line follow the variations within a set of data.
- The R_2 value varies between 0 and 1 where 0 represents no correlation between the predicted and actual value and 1 represents complete correlation.
- **R-squared** is a good measure to evaluate the model fitness. It is also known as the coefficient of determination. R-squared is the fraction by which the variance of the errors is less than the variance of the dependent variable.
- It is called R-squared because in a simple regression model it is just the square of the correlation between the dependent and independent variables, which is commonly denoted by "r".
- In a multiple regression model R-squared is determined by pairwise correlations among all the variables, including correlations of the independent variables with each other as well as with the dependent variable.

Example 2.11.1 *Suppose you have been given a set of training examples* $\{(x_1, y_1)\}, (x_2, y_2), \ldots, (x_n, y_n)\}$. *Find the equation of the line that best fits the data in that minimizes the squared error.*

Solution : Fit the regression line $y = \beta_0 + \beta_1 x$ to the data.

$$(x_1, y_1), \ldots, (x_n, y_n)$$

by finding the "best" match between the line and the data. The "best" choice of β_0, β_1 will be chosen to minimize.

$$Q = \sum_{i=1}^{n} (y_i - (\beta_0 + \beta_1 x_i)^2 = \sum_{i=1}^{n} \varepsilon_i^2$$

This is called the least square fit. Let's solve ...

$$\frac{\partial Q}{\partial \beta_0} = -2 \sum (y_i - (\beta_0 + \beta_1 x_i)) = 0$$

$$\frac{\partial Q}{\partial \beta_1} = -2 \sum x_i (y_i - (\beta_0 + \beta_1 x_i)) = 0$$

$$\Leftrightarrow \quad \sum y_i = n\beta_0 + \beta_1 \sum x_i$$

$$\sum x_i y_i = -2 \sum x_i (y_i - (\beta_0 + \beta_1 x_i)) = 0$$

After a little algebra, get

$$\hat{\beta}_1 = \frac{n \sum x_i y_i - (\sum x_i)(\sum y_i)}{n(\sum x_i^2) - (\sum x_i)^2}$$

$$\hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x}, \text{ where } \bar{y} = \frac{1}{n} \sum y_i \text{ and } \bar{x} \equiv \frac{1}{n} \sum x_i$$

$$S_{xx} = \sum (x_i - \bar{x})^2 = \sum x_i^2 - n\bar{x}^2$$

$$= \sum x_i^2 - \frac{(\sum x_i)^2}{n}$$

$$S_{xy} = \sum (x_i - \bar{x})^2 (y_i - \bar{y})^2 = \sum x_i y_i - n\overline{xy}$$

$$= \sum x_i y_i - \frac{(\sum x_i)(\sum y_i)}{n}$$

Review Question

1. *What do you mean by coefficient of regression? Explain SST, SSE, SSR, MSE in the context of regression.*

2.11.4 Root Mean Squared Error (RMSE)

- It measures the average difference between values predicted by a model and the actual values. It provides an estimation of how well the model is able to predict the target value (accuracy).
- The lower the value of the Root Mean Squared Error, the better the model is. A perfect model would have a Root Mean Squared Error value of 0.
- Root Mean Square Error (RMSE) is the standard deviation of the residuals (prediction errors). Residuals are a measure of how far from the regression line data points are; RMSE is a measure of how spread out these residuals are.
- In other words, it tells us how concentrated the data is around the line of best fit. Root mean square error is commonly used in climatology, forecasting and regression analysis to verify experimental results.
- The RMSE represents the square root of the second sample moment of the differences between predicted values and observed values or the quadratic mean of these differences. These deviations are called residuals when the calculations are performed over the data sample that was used for estimation and are called errors (or prediction errors) when computed out-of-sample.
- The RMSE serves to aggregate the magnitudes of the errors in predictions for various data points into a single measure of predictive power.
- RMSE is calculated y using following formula :

$$RMSE = \sqrt{\frac{SSE_W}{W}} = \sqrt{\frac{1}{W}\sum_{i=1}^{N} w_i u_i^2}$$

where :

SSE_W = Weighted sum of squares

W = Total weight of the population

N = Number of observations

w_i = Weight of the i^{th} observation

u_i = Error associated with the i^{th} observation

- RMSE is a measure of accuracy, to compare forecasting errors of different models for a particular dataset and not between datasets, as it is scale-dependent.
- RMSE is always non-negative and a value of 0 would indicate a perfect fit to the data. In general, a lower RMSE is better than a higher one. However, comparisons across different types of data would be invalid because the measure is dependent on the scale of the numbers used.

- RMSE is the square root of the average of squared errors. The effect of each error on RMSD is proportional to the size of the squared error; thus larger errors have a disproportionately large effect on RMSD. Consequently, RMSE is sensitive to outliers.

2.11.5 Adjusted R-squared

- Adjusted R-squared is a modified form of R-squared whose value increases if new predictors tend to improve models performance and decreases if new predictors does not improve performance as expected.
- Adjusted R^2 is a corrected goodness-of-fit (model accuracy) measure for linear models. It identifies the percentage of variance in the target field that is explained by the input or inputs.
- Adjusted R squared is calculated by dividing the residual mean square error by the total mean square error. The result is then subtracted from 1.
- Adjusted R^2 is always less than or equal to R^2. A value of 1 indicates a model that perfectly predicts values in the target field. A value that is less than or equal to 0 indicates a model that has no predictive value. In the real world, adjusted R^2 lies between these values.
- Adjusted R-squared can be negative when R-squared is close to zero. Adjusted R-squared value always be less than or equal to R-squared value.

❑❑❑

SOLVED MODEL QUESTION PAPER (In Sem)
Machine Learning

B.E. (AI & DS) Semester - VII (As Per 2020 Pattern)

Time : 1 Hour] [Maximum Marks : 30

N. B. :

i) Attempt Q.1 or Q.2, Q.3 or Q.4.

ii) Neat diagrams must be drawn wherever necessary.

iii) Figures to the right side indicate full marks.

iv) Assume suitable data, if necessary.

Q.1 **a)** *Compare machine learning vs artificial intelligence.* **(Refer section 1.3.1)** [5]

b) *Describe parametric and non-parametric machine learning models.* **(Refer sections 1.11 and 1.12)** [5]

c) *Explain supervised learning with its advantages and disadvantages.* **(Refer section 1.5)** [5]

OR

Q.2 **a)** *Explain PCA and LDA. What is difference between LDA and PCA.* **(Refer sections 1.14 and 1.15)** [7]

b) *What is reinforcement learning ? Explain elements of reinforcement learning.* **(Refer section 1.8)** [8]

Q.3 **a)** *Explain following evaluation matrix :* *MSE, MAE, RMSE, R-Square* **(Refer section 2.11)** [8]

b) *What is SVM ? Explain key properties of SVM. Compare SVM with neural network.* **(Refer section 2.6)** [7]

OR

Q.4 **a)** *What is regression ? Explain need of regression. Discuss types of regression.* **(Refer sections 2.2 and 2.3)** [8]

b) *Explain Lasso and ElasticNet regression.* **(Refer sections 2.8 and 2.9)** [7]