BUDDHA SERIES

(Unit Wise Solved Question & Answers)

Course – B.Tech. [CSE(AIML/DS)]

College – Buddha Institute of Technology

(AKTU CODE-525)

Department: Computer Science & ALLIED

Subject: Machine Learning
(BCAI 601/BCDS 062)

Faculty Name: Chhiteesh Rai

Unit - 1

1. Compare supervised learning and unsupervised learning techniques with example. **(AKTU 2023-24)**

ANSWER:

Supervised Machine Learning:

Supervised learning is a machine learning method in which models are trained using labeled data. In supervised learning, models need to find the mapping function to map the input variable (X) with the output variable (Y). Y=f(x)

Supervised learning needs supervision to train the model, which is similar to as a student learns things in the presence of a teacher.

Unsupervised Machine Learning:

Unsupervised learning is another machine learning method in which patterns inferred from the unlabelled input data. The goal of unsupervised learning is to find the structure and patterns from the input data. Unsupervised learning does not need any supervision. Instead, it finds patterns from the data by its own.

Parameter	pervised Learning	supervised Learning
Input Data	Uses Known and Labeled Data as input	Uses Unknown Data as input
Computational Complexity	Less Computational Complexity	More Computational Complex
Real-Time	Uses off-line analysis	Uses Real-Time Analysis of Data
Number of Classes	The number of Classes is known	The number of Classes is not known
Accuracy of Results	Accurate and Reliable Results	Moderate Accurate and Reliable Results
Output data	The desired output is given.	The desired, output is not given.
Model	In supervised learning it is not possible to learn larger and more complex models than in unsupervised learning	In unsupervised learning it is possible to learn larger and more complex models than in supervised learning
Training data	In supervised learning training data is used to infer model	In unsupervised learning training data is not used.
Test of model	We can test our model.	We can not test our model.
Example	Optical Character Recognition	Find a face in an image.

2. Discuss the application of reinforcement learning. In which problems reinforcement learning is used? **(AKTU 2023-24)**

ANSWER:

In Reinforcement Learning (RL), agents are trained on a reward and punishment mechanism. The agent is rewarded for correct moves and punished for the wrong ones. In doing so, the agent tries to minimize wrong moves and maximize the right ones.

Appliction-

- **Industry automation-** In industry reinforcement, learning-based robots are used to perform various tasks. A great example is the use of AI agents by Deepmind to cool Google Data

Centers. This led to a 40% reduction in energy spending. The centers are now fully controlled with the AI system without the need for human intervention. There is obviously still supervision from data center experts. The system works in the following way:

- Taking snapshots of data from the data centers every five minutes and feeding this to deep neural networks
- It then predicts how different combinations will affect future energy consumptions
- Identifying actions that will lead to minimal power consumption while maintaining a set standard of safety criteria
- Sending and implement these actions at the data center

- **Trading and finance-** Supervised time series models can be used for predicting future sales as well as predicting stock prices. An RL agent can decide on such a task; whether to hold, buy, or sell. The RL model is evaluated using market benchmark standards in order to ensure that it's performing optimally. This automation brings consistency into the process, unlike previous methods where analysts would have to make every single decision. IBM for example has a sophisticated reinforcement learning based platform that has the ability to make financial trades. It computes the reward function based on the loss or profit of every financial transaction.
- **Image Processing -** mage Processing is another important method of enhancing the current version of an image to extract some useful information from it. And there are some steps associated like:
  - Capturing the image with machines like scanners.
  - Analyzing and manipulating it.
  - Using the output image obtained after analysis for representation, description-purposes.

Here, ML models like Deep Neural Networks (whose framework is Reinforcement Learning) can be leveraged for simplifying this trending image processing method. With Deep Neural Networks, you can either enhance the quality of a specific image or hide the info. of that image. Later, use it for any of your computer vision tasks.

- **Robotics -** Robotics without any doubt facilitates training a robot in such a way that a robot can perform tasks just like a human being can. But still, Robots aren't able to use common sense while making various moral, social decisions. Here, a combination of Deep Learning and Reinforcement Learning i.e. Deep Reinforcement Learning comes to the rescue to enable the robots with, "Learn How To Learn" model. With this, the robots can now: –
  - manipulate their decisions by grasping well various objects visible to them.
  - solve complicated tasks which even humans fail to do as robots now know what and how to learn from different levels of abstractions of the types of datasets available to them.
- **Healthcare -** Healthcare is an important part of our lives and through DTRs (a sequence-based use-case of RL), doctors can discover the treatment type, appropriate doses of drugs, and timings for taking such doses. DTRs are equipped with: –
  - a sequence of rules which confirm the current health status of a patient.
  - Then, they optimally propose treatments that can diagnose diseases like diabetes, HIV, Cancer, and mental illness too.
  - If required, these DTRs (i.e. Dynamic Treatment Regimes) can reduce or remove the delayed impact of treatments through their multi-objective healthcare optimization solutions.
- **Broadcast Journalism -** Recommending news that suits the frequently-changing preferences of readers and other online users can possibly be achieved since journalists can now be equipped with an RL-based system that keeps an eye on intuitive news content as well as the headlines.

- **Aviation-** Reinforcement Learning offers exciting opportunities in aviation, allowing autonomous systems to learn from real-time data and make informed decisions, leading to enhanced airspace management, improved efficiency, and reduced environmental impact.
    - Boeing's Autonomous Aerial Refueling: Boeing has been working on an RL-based system called the Autonomous Aerial Refueling (AAR) system. It uses RL algorithms to enable unmanned aircraft to autonomously perform aerial refueling operations, ensuring precise and safe refueling maneuvers.
    - NASA's Autonomous Systems: NASA has been actively researching RL for autonomous systems in aviation. They have developed RL algorithms to train autonomous drones and aerial vehicles for tasks such as collision avoidance, path planning, and autonomous landing.
    - Airbus Skywise Predictive Maintenance: Airbus has implemented RL techniques in their Skywise Predictive Maintenance platform. This platform utilizes RL algorithms to analyze aircraft sensor data, historical maintenance records, and operational data to predict component failures and optimize maintenance schedules, reducing maintenance costs and minimizing disruptions.

3. Discuss the important objectives of machine learning. (AKTU 2023-24)

ANSWER:

The important objectives are-

- To Reduced complexity and time - The essence of this technology is that it shortens the time required to complete the task. The time it takes to complete a task or job has been drastically reduced because everything is automated and streamlined using Machine Learning.
- To create Self-automated environments - Machine Learning is self-reliant and self-driven. It operates autonomously and does not require human intervention. Most of the time, we can train or teach them how to work.
- For Constant improvements - AI and Machine Learning are constantly evolving concepts. Their development is limitless. Experts in this field are continually improving. ML, on the other hand, is a dynamic concept. When exposed to new situations, ML programming creates something better.
- Proficiency level of pattern analysis - Machine Learning is based on the theory of computing various patterns. It means that when the machine is given data, it automatically begins evaluating patterns and checking the data flow.
- Wide Range of Applicability There are numerous applications for Machine Learning. It is essential in almost every field, including hospitality, education, medicine, science, banking, and business.

4. Illustrate the various area in which you can apply machine learning. (AKTU 2022-23)

ANSWER:

Here are a few examples of popular machine-learning applications from various area-

- Self-driving cars -Machine Learning techniques are used in self-driving cars. It uses the unsupervised learning system. Its algorithms guide data collection from cameras, sensors, and the environment, allowing it to select the best action option.
- Encountering spam Emails - One of the most well-known applications of Machine Learning is detecting spam emails. Email service providers create spam filters that use an ML algorithm to classify incoming emails as spam and route them to the spam folder.

- Detecting Malware - Malware detection in Machine Learning consists of two steps. The first step is to analyze suspicious activities in the Android environment to generate a suitable collection of features. The second step is to train the system to detect future cyberattacks in such environments using machine and deep learning techniques on the developed features.
- Image recognition -One of the most notable Machine Learning applications is image recognition, which detects and catalogs the object or feature in digital images. This is then used for analysis, such as pattern recognition, face detection, and face recognition. Speech recognition ML software can quantify spoken words using numbers representing the speech signal. Amazon's Alexa, Apple's Siri, and Google Maps are examples of some of the most popular speech recognition applications.
- Anticipate traffic patterns - Google Maps is one of the best examples of predicting traffic patterns. When we enter our location on the map, the application collects massive data, generates predictions about upcoming traffic, and recognizes the shortest route to the destination.

5. Explain the concept of machine learning. (AKTU 2020-21)

ANSWER:

- Machine Learning is defined as a technology that is used to train machines to perform various actions such as predictions, recommendations, estimations, etc., based on historical data or past experience. Machine Learning enables computers to behave like human beings by training them with the help of past experience and predicted data.

There are three key aspects of Machine Learning, which are as follows:

- Task: A task is defined as the main problem in which we are interested. This task/problem can be related to the predictions and recommendations and estimations, etc.
- Experience: It is defined as learning from historical or past data and used to estimate and resolve future tasks.
- Performance: It is defined as the capacity of any machine to resolve any machine learning task or problem and provide the best outcome for the same. However, performance is dependent on the type of machine learning problems.

How Does Machine Learning Work?

- Data Collection -This is where you gather the raw materials, the data, that your machine learning model will learn from. The quality and quantity of this data directly impact how well your model performs. Data can come from many sources, like databases, websites, sensors, or even manual creation.
- Data Preprocessing -

Real-world data isn't always perfect. This step involves cleaning the data (removing duplicates and errors), handling missing bits, and ensuring everything is formatted correctly for the machine learning algorithm to understand.

- Choosing the Right Model-

  There are many different machine learning models, like decision trees or neural networks, each with its strengths. Choosing the right one depends on the type of problem you're trying to solve and the characteristics of your data.
- Training the Model-

  The prepped data is fed into the chosen model, and it starts to learn patterns within that data. This process often involves multiple rounds of the model seeing the data and adjusting its internal settings to learn better.
- Evaluating the Model-

  Before using the model in the real world, we need to assess its performance. This involves testing it on a separate dataset it hasn't seen before.

Metrics like accuracy and precision help determine how well the model is learning.

- Hyperparameter Tuning and Optimization-
  Machine learning algorithms have adjustable settings called hyperparameters. Tweaking these settings can further improve the model's performance.
- Prediction and Deployment-
  Once the model is performing well, it's time to put it to use! The model can be integrated into a website, used to analyze new data, or even power a self-driving car.

6. Explain the "concept learning" task giving an example. (AKTU 2021-22)

ANSWER:

Concept learning is the task of inferring a Boolean-valued function from a set of training examples1. In this task, a human classifies objects by being shown a set of example objects along with their class labels. The learner simplifies what has been observed by condensing it in the form of an example. This simplified version of what has been learned is then applied to future examples2. Concept learning is based on a type of learning called inductive learning, where the learner learns by example.

Taking a very simple example, one possible target concept may be to Find the day when my friend Ramesh enjoys his favourite sport. We have some attributes/features of the day like, Sky, Air Temperature, Humidity, Wind, Water, Forecast and based on this we have a target Concept named EnjoySport. We have the following training example available:

Example	Sky	AirTemp	Humidity	Wind	Water	Forecast	EnjoySport
1	Sunny	Warm	Normal	Strong	Warm	Same	Yes
2	Sunny	Warm	High	Strong	Warm	Same	Yes
3	Rainy	Cold	High	Strong	Warm	Change	No
4	Sunny	Warm	High	Strong	Cool	Change	Yes

Let's Design the problem formally with TPE(Task, Performance, Experience):

Problem: Leaning the day when Ramesh enjoys the sport.

- Task T: Learn to predict the value of EnjoySport for an arbitrary day, based on the values of the attributes of the day.
- Performance measure P: Total percent of days (EnjoySport) correctly predicted.
- Training experience E: A set of days with given labels (EnjoySport: Yes/No)

Let us take a very simple hypothesis representation which consists of a conjunction of constraints in the instance attributes. We get a hypothesis h_i with the help of example i for our training set as below:

hi(x) := <x1, x2, x3, x4, x5, x6>

where x1, x2, x3, x4, x5 and x6 are the values of Sky, AirTemp, Humidity, Wind, Water and Forecast.

Hence h1 will look like(the first row of the table above):

h1(x=1): <Sunny, Warm, Normal, Strong, Warm, Same > Note: x=1 represents a positive hypothesis / Positive example

We want to find the most suitable hypothesis which can represent the concept. For example, Ramesh enjoys his favorite sport only on cold days with high humidity (This seems independent of the values of the other attributes present in the training examples).

h(x=1) = <?, Cold, High, ?, ?, ?>

Here ? indicates that any value of the attribute is acceptable. Note: The most generic hypothesis will be <

?, ?, ?, ?, ?, ?> where every day is a positive example and the most specific hypothesis will be <?,?,?,?,?,? > where no day is a positive example.

7. What are the five limitations of machine learning? Explain with example. (AKTU2022-23)

ANSWER:

Machine learning has its limitations, just like any other technology, and these must be considered before using it in practical situations. These limitations are-

- **Lack of Transparency and Interpretability -** One of its main drawbacks is more transparency and interpretability in machine learning. As they don't reveal how a judgment was made or how it came to be, machine learning algorithms are frequently called "black boxes." This makes it challenging to comprehend how a certain model concluded and might be problematic when explanations are required. For instance, understanding the reasoning behind a particular diagnosis in healthcare might be easier with transparency and interpretability.
- **Bias and Discrimination -**The possibility for bias and discrimination is a significant flaw in machine learning. Large datasets, which may have data biases, are used to train machine learning systems. If these biases are not addressed, the machine learning system may reinforce them, producing biased results.

  The algorithms used in facial recognition are one instance of bias in machine learning. According to research, facial recognition software performs worse on those with darker skin tones, which causes false positive and false negative rates to be higher for people of races. This bias may have significant consequences, particularly in law enforcement and security applications, where false positives may result in unjustified arrests or other undesirable results.
- **Overfitting and Underfitting -** Machine learning algorithms frequently have two limitations: overfitting and underfitting. Overfitting is a condition where a machine learning model performs poorly on new, unknown data because it needs to be simplified and has been trained too successfully on the training data. On the other side, underfitting happens when a machine learning model is overly simplistic and unable to recognize the underlying patterns in the data, resulting in subpar performance on both the training data and fresh data.
- **Limited Data Availability -** A major challenge for machine learning is the need for more available data. Machine learning algorithms need a lot of data to learn and produce precise predictions. Due to privacy considerations, it might be difficult to get medical data, while data from sporadic events, such as natural catastrophes, may be of restricted scope.
- **Computational Resources -** Machine learning algorithms can be computationally expensive, and they may require a lot of resources to be successfully trained. This may be a major barrier, particularly for people or smaller companies who want access to high-performance computing resources. Distributed and cloud computing can be used to get around this restriction, however the project's cost might go up.

  For huge datasets and complex models, machine learning approaches can be computationally expensive. The scalability and feasibility of machine learning algorithms may be hampered by the need for significant processing resources. The availability of computational resources like processor speed, memory, and storage is another limitation on machine learning.

8. Explain the difference between Find S and candidate elimination algorithm.

(AKTU 2022-23)

ANSWER:

Introduction -The find-S algorithm is a basic concept learning algorithm in machine learning. The find-S algorithm finds the most specific hypothesis that fits all the positive examples. We have to note here that the algorithm considers only those positive training example. The find-S algorithm starts with the most specific hypothesis and generalizes this hypothesis each time it fails to

classify an observed positive training data. Hence, the Find-S algorithm moves from the most specific hypothesis to the most general hypothesis.

Important Representation-

- ? indicates that any value is acceptable for the attribute.
- specify a single required value (e.g., Cold) for the attribute.
- ϕindicates that no value is acceptable.
- The most general hypothesis is represented by: {?, ?, ?, ?, ?, ?}
- The most specific hypothesis is represented by: {ϕ, ϕ, ϕ, ϕ, ϕ, ϕ}

Steps Involved In Find-S :

- Start with the most specific hypothesis.
- h = {ϕ, ϕ, ϕ, ϕ, ϕ, ϕ}
- Take the next example and if it is negative, then no changes occur to the hypothesis.
- If the example is positive and we find that our initial hypothesis is too specific then we update our current hypothesis to a general condition.
- Keep repeating the above steps till all the training examples are complete.
- After we have completed all the training examples we will have the final hypothesis when can use to classify the new examples.

Example -Consider the following data set having the data about which particular seeds are poisonous.

EXAMPLE	COLOR	TOUGHNESS	FUNGUS	APPEARANCE	POISONOUS
1.	GREEN	HARD	NO	WRINKELD	YES
2.	GREEN	HARD	YES	SMOOTH	NO
3.	BROWN	SOFT	NO	WRINKLED	NO
4.	ORANGE	HARD	NO	WRINKLED	YES
5.	GREEN	SOFT	YES	SMOOTH	YES
6.	GREEN	HARD	YES	WRINKLED	YES
7.	ORANGE	HARD	NO	WRINKLED	YES

ATTRIBUTES ON WHICH THE CONCEPT DEPENDS ON — CONCEPT

First, we consider the hypothesis to be a more specific hypothesis. Hence, our hypothesis would be :

h = {φ, φ, φ, φ, φ, φ}

Consider example 1 :

The data in example 1 is { GREEN, HARD, NO, WRINKLED }. We see that our initial hypothesis is more specific and we have to generalize it for this example. Hence, the hypothesis becomes :

h = { GREEN, HARD, NO, WRINKLED }

ML – Candidate Elimination Algorithm-

The candidate elimination algorithm incrementally builds the version space given a hypothesis space H and a set E of examples. The examples are added one by one; each example possibly shrinks the version space by removing the hypotheses that are inconsistent with the example. The candidate elimination algorithm does this by updating the general and specific boundary for each new example.

- You can consider this as an extended form of the Find-S algorithm.
- Consider both positive and negative examples.

- Actually, positive examples are used here as the Find-S algorithm (Basically they are generalizing from the specification).
- While the negative example is specified in the generalizing form.

Terms Used:

- Concept learning: Concept learning is basically the learning task of the machine (Learn by Train data)
- General Hypothesis: Not Specifying features to learn the machine.
- G = {'?', '?','?','?'…}: Number of attributes
- Specific Hypothesis: Specifying features to learn machine (Specific feature)
- S= {'pi','pi','pi'…}: The number of pi depends on a number of attributes.
- Version Space: It is an intermediate of general hypothesis and Specific hypothesis. It not only just writes one hypothesis but a set of all possible hypotheses based on training data-set.

Algorithm:

Step1: Load Data set
Step2: Initialize General Hypothesis and Specific Hypothesis.
Step3: For each training example.
Step4: If example is positive example

```
if attribute_value == hypothesis_value:
  Do nothing
else:
  replace attribute value with '?' (Basically generalizing it)
```

Step5: If example is Negative example

Advantages of CEA over Find-S:

- Improved accuracy: CEA considers both positive and negative examples to generate the hypothesis, which can result in higher accuracy when dealing with noisy or incomplete data
- Flexibility: CEA can handle more complex classification tasks, such as those with multiple classes or non-linear decision boundaries.
- More efficient: CEA reduces the number of hypotheses by generating a set of general hypotheses and then eliminating them one by one. This can result in faster processing and improved efficiency.
- Better handling of continuous attributes: CEA can handle continuous attributes by creating boundaries for each attribute, which makes it more suitable or a wider range of datasets.

9. Discuss about historical progress of machine learning. What is the concept of clustering in machine learning? (AKTU 2021-22)

ANSWER:

- historical progress of machine learning

1950s

- 1950: Alan Turing proposes the concept of a "learning machine" that could improve over time.
- 1957: Frank Rosenblatt develops the perceptron algorithm, an early type of artificial neuron.

1960s

- 1960s: Early machine learning algorithms, such as the nearest neighbor algorithm, are developed.
- 1967: The nearest neighbor algorithm is used for pattern recognition tasks.
- 1970s
- 1970s: The field of machine learning stagnates due to limited computational power and data.

- 1979: Kunihiko Fukushima proposes the neocognitron, a precursor to convolutional neural networks.

1980s

- 1980s: Resurgence of machine learning with the development of backpropagation for training neural networks.
- 1986: David Rumelhart, Geoffrey Hinton, and Ronald J. Williams popularize backpropagation.

1990s

- 1990s: Introduction of support vector machines (SVMs) by Vladimir Vapnik and colleagues.
- 1995: Leo Breiman develops the random forest algorithm, improving model accuracy through ensemble learning.

2000s

- 2000s: The rise of big data provides vast amounts of data for training machine learning models.
- 2006: Geoffrey Hinton and Ruslan Salakhutdinov publish a paper on deep belief networks, sparking interest in deep learning.

2010s

- 2010s: Deep learning gains prominence with the development of powerful neural networks and increased computational power.
- 2012: AlexNet, a deep convolutional neural network, wins the ImageNet Large Scale Visual Recognition Challenge.
- 2014: Ian Goodfellow introduces generative adversarial networks (GANs).
- 2015: The transformer model is introduced, revolutionizing natural language processing.
- 2017: Google's AlphaGo, powered by deep reinforcement learning, defeats the world champion in Go.

2020s

- 2020s: Continued advancements in reinforcement learning, natural language processing, and explainable AI.
- 2020: OpenAI releases GPT-3, a state-of-the-art language model with 175 billion parameters.
- 2021: AlphaFold, developed by DeepMind, makes significant breakthroughs in protein.

concept of clustering -Clustering techniques in machine learning is the task of dividing the unlabelled data or data points into different clusters such that similar data points fall in the same cluster than those which differ from the others. In simple words, the aim of the clustering process is to segregate groups with similar traits and assign them into clusters.

Types of Clustering Techniques in Machine Learning - Clustering broadly divides into two subgroups:

- Hard Clustering: Each input data point either fully belongs to a cluster or not. For instance, in the example above, every customer is assigned to one group out of the ten.
- Soft Clustering: Rather than assigning each input data point to a distinct cluster, it assigns a probability or likelihood of the data point being in those clusters. For example, in the given scenario, each customer receives a probability of being in any of the ten retail store clusters.

10. Discuss in details about the designing of appropriate machine learning algorithm.
(AKTU 2021-22)

ANSWER:

Steps to Choose Best Machine Learning Algorithm- Here is a step-by-step procedure to choose correct machine learning algorithm :

- Understand Your Problem : Begin by gaining a deep understanding on the problem you are trying to solve. What is your goal? What is the problem all about classification, regression , clustering, or something else? What kind of data you are working with?
- Process the Data: Ensure that your data is in the right format for your chosen algorithm. Process and prepare your data by cleaning, Clustering, Regression.
- Exploration of Data: Conduct data analysis to gain insights into your data. Visualizations and statistics helps you to understand the relationships within your data.
- Metrics Evaluation: Decide on the metrics that will measure the success of model. You must choose the metric that should align with your problem.
- Simple models: One should begin with the simple easy-to-learn algorithms. For classification, try regression, decision tree. Simple model provides a baseline for comparison.
- Use Multiple Algorithms: Try to use multiple algorithms to check that one performs on your dataset. That may include:
    - Decision Trees
    - Gradient Boosting(XGBoost, LightGBM)
    - Random Forest
    - k-Neasrest Neighbors(KNN)
    - Naive Bayes
    - Support Vector Machines(SVM)
    - Neural Networks(Deep Learning)
- Hyperparameter Tuning: Grid Search and Random Search can helps with adjusting parameters choose algorithm that find best combination.
- Cross- Validation: Use cross- validation to get assess the performance of your models. This helps prevent overfiting .
- Comparing Results: Evaluate the models's performance by using the metrics evaluation. Compare their performance and choose that best one that align with problem's goal.
- Consider Model Complexity: Balance complexity of model and their performance. Compare their performance and choose that one best algorithm to generalize better.

BUDDHA SERIES

(Unit Wise Solved Question & Answers)

Course – B.Tech. [CSE(AIML/DS)]

College – Buddha Institute of Technology

(AKTU CODE-525)

Department: Computer Science & Engineering

Subject: Machine Learning
(BCAI 601/BCDS 062)

Faculty Name: Chhiteesh Rai

Unit - 2

1. **Explain hyperplane (decision boundary) in SVM. Categories various popular kernels associated with SVM. (AKTU 2023-24)**

ANSWER:

Hyperplane - In SVMs, a hyperplane is a subspace of one dimension less than the original feature space. In two-dimensional space, a hyperplane is a line, while in three-dimensional space, it is a plane. In general, in n-dimensional space, a hyperplane is an (n-1)-dimensional flat affine subspace.

The hyperplane in SVMs is represented by the equation:

$w^T x+b=0$ where w is the weight vector, x is a data point, and b is the bias term. The weight vector determines the "shape" of the hyperplane, and the bias term determines the position of it.

The decision rule for SVMs is based on the sign of the equation above. If the result is positive, the data point is classified as one class, and if it is negative, the data point is classified as the other class.

Decision boundary is a crucial concept in machine learning and pattern recognition. It refers to the boundary or surface that separates different classes or categories in a classification problem. In simple terms, decision boundary is a line or curve that divides the data into two or more categories based on their features. The objective of decision boundary is to make accurate predictions on unseen data by identifying the correct class for a given input.

A hyperplane that partitions the feature space into distinct classes is known as a decision boundary. In binary classification problems, the decision boundary serves as the line of demarcation between positive and negative classes. The position and orientation of the decision boundary are determined by the model's training data and algorithm. The primary aim is to discover a decision boundary that can effectively generalize to new data, making it a reliable predictor.

Types of Decision boundaries-There are different types of decision boundaries based on the complexity of the classification problem. The most common types of decision boundaries are:

- Linear decision boundary- A linear decision boundary is a straight line that separates the data into two classes. It is the simplest form of decision boundary and is used when the classification problem is linearly separable. Linear decision boundary can be expressed in the form of a linear equation, $y = mx + b$, where m is the slope of the line and b is the y-intercept.
- Non Linear decision boundary- A non-linear decision boundary is a curved line that separates the data into two or more classes. Non-linear decision boundaries are used when the classification problem is not linearly separable. Non-linear decision boundaries can take different forms such as parabolas, circles, ellipses, etc.
- Decision Boundary with Margin-A decision boundary with margin is a line or curve that separates the data into two classes while maximizing the distance between the boundary and the closest data points. The margin is defined as the distance between the decision boundary and the closest data points of each class. The objective of decision boundary with margin is to improve the generalization performance of the classifier by reducing the risk of overfitting.
- Decision Boundary with Soft Margin- A decision boundary with soft margin is a line or curve that separates the data into two classes while allowing some misclassifications. Soft margin is used when the data is not linearly separable and when the classification problem has some noise or outliers. The objective of decision boundary with soft margin is to find a balance between the accuracy of the classifier and its ability to generalize to unseen data.

Types of Kernel in SVM - Here are some common types of kernels in support vector machine algorithms:

- Linear Kernel- The linear kernel is the simplest and is used when the data is linearly separable. It calculates the dot product between the feature vectors.

- Polynomial Kernel- The polynomial kernel is effective for non-linear data.
- It computes the similarity between two vectors in terms of the polynomial of the original variables.
- Radial Basis Function (RBF) Kernel - The RBF kernel is a common type of Kernel in SVM for handling non-linear decision boundaries. It maps the data into an infinite-dimensional space.
- Sigmoid Kernel- The sigmoid SVM kernel types can be used as an alternative to the RBF kernel. It is based on the hyperbolic tangent function and is suitable for neural networks and other non-linear classifiers.
- Custom Kernels - In addition to the standard kernels mentioned above, SVMs allow the use of custom kernels tailored to specific problems. Custom kernels can be designed based on domain knowledge or problem-specific requirements.

2. Explain concept learning task. Giving an example. (AKTU 2023-24)

ANSWER:

Concept learning in machine learning refers to the process of teaching a machine to identify and recognize patterns from specific examples or data points. In simple terms, concept learning involves learning a general rule from a set of observed instances. For example, if you show a machine many pictures of cats, it will learn to recognize the concept of a "cat" and apply that knowledge to identify new cat pictures.

Types of Concept Learning Tasks

There are two main types of concept learning tasks in machine learning: supervised concept learning and unsupervised concept learning.

- Supervised Concept Learning- In supervised concept learning, the machine is given labeled data, meaning each data point has a known output or target label. The goal is for the machine to learn the concept by identifying patterns between the inputs and the labeled outputs.

Example of Supervised Concept Learning:

Consider a dataset of images where each image is labelled as either a "cat" or "dog." By analyzing these labeled examples, the machine can learn to classify future images as either a cat or a dog based on what it has learned from the training data.

- Unsupervised Concept Learning- In unsupervised concept learning, the machine is given data without labels. The machine's task is to identify patterns or groupings within the data. This type of learning is more about discovering hidden structures in the data rather than learning specific labels.

Example of Unsupervised Concept Learning:

An e-commerce company might use unsupervised learning to group customers based on their purchasing history. The algorithm can identify clusters of customers who buy similar products, even though there are no predefined labels for these groups.

Concept learning helps machines generalize from data. Instead of memorizing each example, it creates a broader understanding that can be applied to unseen situations. This ability to generalize is what makes machine learning models so powerful.

Once a concept learning model has been trained, it can use what it has learned to make predictions on new, unseen data. Here's a simple explanation of how this process works:

- Input New Data- The model is provided with new data that it hasn't seen before. For example, if the model was trained to recognize cats and dogs, this new data might be an image of either a cat or a dog.

- Apply the Learned Concept- The model uses the concept it learned during training to analyze the new data. It looks for patterns or features in the input data that match the concept it has already learned. In our example, the model will look at features like fur, whiskers, or tail shapes to decide whether the new image is a cat or a dog.

- Make a Prediction - Based on the analysis, the model will make a prediction. This prediction could be a classification (e.g., "This is a cat") or another type of output depending on the task. The accuracy of the prediction depends on how well the model learned the concept and how closely the new data matches the patterns it learned from the training examples.

- Update or Adjust (Optional)- In some cases, after making predictions, models can be fine-tuned or adjusted with more data to improve their accuracy over time. This is known as continuous learning.

3. Differentiate between Naïve bayes classifier and Bayesian belief networks. Give an application of Bayesian belief networks. (AKTU 2023-24)

ANSWER:

Attribute	Bayesian Belief Network	Naive Bayes
Definition	A probabilistic graphical model representing a set of variables and their conditional dependencies.	A simple probabilistic classifier based on applying Bayes' theorem with strong independence assumptions.
Assumptions	Variables can have direct dependencies on each other.	Assumes strong independence between features.
Dependency Representation	Uses directed acyclic graphs (DAGs) to represent dependencies between variables.	Does not explicitly represent dependencies between features.
Complexity	Can handle complex dependencies between variables.	Simple and computationally efficient.
Training	Requires a large amount of training data and computational resources.	Can be trained quickly with limited data.
Feature Independence	Does not assume independence between features.	Assumes strong independence between features.
Scalability	Can be computationally expensive for large networks.	Highly scalable and efficient for large datasets.
Model Complexity	Can represent complex relationships between variables.	Assumes simple relationships between features.

Application of Bayesian belief networks-

- GRN or Gene Regulatory Network- It is basically a field of biomedical linked with machine learning. The main focus of this field is DNA segmentation in a cell. In this technique, the DNA also comes closer to other substances in the cell. The DNA basically interact with them

| Attribute | Bayesian Belief Network | Naive Bayes |

indirectly, here the word indirectly means it interacts through their interaction product. With the help of GRN, one can obtain the model behavior of the system using mathematical modeling. With the help of these mathematical modeling different predictions can be made and different weights can be assigned to them.

- Biomonitoring- is a field of biomedical instrumentation, in which the concentration of chemicals and their effects on the health of the person both short term and long term can be predicted. This can be explained by a simple example like there is an app that on the basis of a person's EMI tells him about the number of calories he should have burned in order to stay active and healthy. When a person provides all the information to the app the application then keeps a check on the different parameters of the human body and warns the user of upcoming threats. All of these are purely based on probability and predictions.
- Document Classification- is also an important application of the Bayesian networking system. In this technique, one can provide all the data to the system and the system on the basis of graphical probability tells us about the designated place for each book in the real-time library as keeping each book on the designated corner requires a lot of time and concentration.
- Information Retrieval- is basically a technique that is used mostly by interrogation teams, and law enforcement agencies. On the basis of this one can extract useful information from CCTV footage or an interrogation.
- Semantic Search- is basically improving the search result for a designated user. If a person watches Netflix more often than on his every search he has to find the ads related to different TV shows and movies.
- Image Processing- has also used this technique sometimes in order to extract useful information from the image by improving the pixel and making the color. With the help of Bayesian networking, the model can be so well trained that it can increase the pixel where necessary and even decrease the coloring where there is no need. With the help of this one can obtain useful information from low-quality images as well.

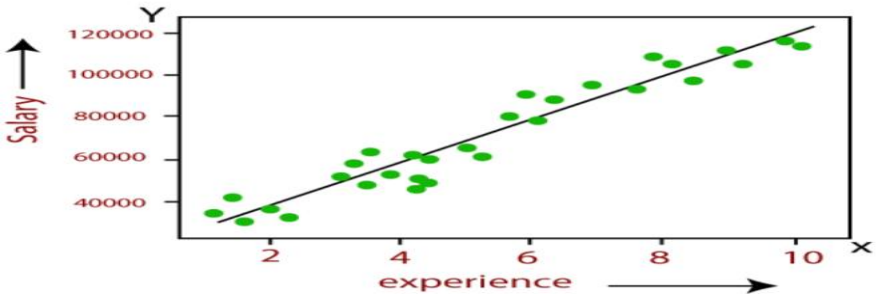
4. Discuss logistic regression and linear regression in detail. (AKTU 2022-23)

ANSWER:

Linear Regression:

- Linear Regression is one of the most simple Machine learning algorithm that comes under Supervised Learning technique and used for solving regression problems.
- It is used for predicting the continuous dependent variable with the help of independent variables.
- The goal of the Linear regression is to find the best fit line that can accurately predict the output for the continuous dependent variable.
- If single independent variable is used for prediction then it is called Simple Linear Regression and if there are more than two independent variables then such regression is called as Multiple Linear Regression.
- By finding the best fit line, algorithm establish the relationship between dependent variable and independent variable. And the relationship should be of linear nature.

The output for Linear regression should only be the continuous values such as price, age, salary, etc. The relationship between the dependent variable and independent variable can be shown in below image:



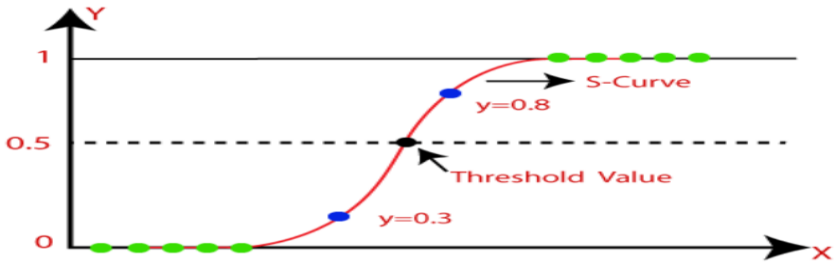
In above image the dependent variable is on Y-axis (salary) and independent variable is on x-axis(experience). The regression line can be written as:

$y= a0+a1x+ \varepsilon$

Where, a0 and a1 are the coefficients and ε is the error term.

Logistic Regression -

- Logistic regression is one of the most popular Machine learning algorithm that comes under Supervised Learning techniques.
- It can be used for Classification as well as for Regression problems, but mainly used for Classification problems.
- Logistic regression is used to predict the categorical dependent variable with the help of independent variables.
- The output of Logistic Regression problem can be only between the 0 and 1.
- Logistic regression can be used where the probabilities between two classes is required. Such as whether it will rain today or not, either 0 or 1, true or false etc.
- Logistic regression is based on the concept of Maximum Likelihood estimation. According to this estimation, the observed data should be most probable.
- In logistic regression, we pass the weighted sum of inputs through an activation function that can map values in between 0 and 1. Such activation function is known as sigmoid function and the curve obtained is called as sigmoid curve or S-curve. Consider the below image:



The equation for logistic regression is:

$$\log\left[\frac{y}{1-y}\right] = b_0 + b_1x_1 + b_2x_2 + b_3x_3 + \cdots + b_nx_n$$

5. What is naïve Bayes classifier and how does it work? Explain the advantages Naive Bayes Algorithm. (AKTU 2022-23

ANSWER:

Naïve Bayes Classifier Algorithm-

Naïve Bayes algorithm is a supervised learning algorithm, which is based on Bayes theorem and used

for solving classification problems. It is mainly used in text classification that includes a high-dimensional training dataset.

Naïve Bayes Classifier is one of the simple and most effective Classification algorithms which helps in building the fast machine learning models that can make quick predictions.

It is a probabilistic classifier, which means it predicts on the basis of the probability of an object. Some popular examples of Naïve Bayes Algorithm are spam filtration, Sentimental analysis, and classifying articles.

The Naïve Bayes algorithm is comprised of two words Naïve and Bayes, Which can be described as:

- Naïve: It is called Naïve because it assumes that the occurrence of a certain feature is independent of the occurrence of other features. Such as if the fruit is identified on the bases of color, shape, and taste, then red, spherical, and sweet fruit is recognized as an apple. Hence each feature individually contributes to identify that it is an apple without depending on each other.
- Bayes: It is called Bayes because it depends on the principle of Bayes' Theorem.

Working of Naïve Bayes' Classifier- Working of Naïve Bayes' Classifier can be understood with the help of the below example:

Suppose we have a dataset of weather conditions and corresponding target variable "Play". So using this dataset we need to decide that whether we should play or not on a particular day according to the weather conditions. So to solve this problem, we need to follow the below steps:

- Convert the given dataset into frequency tables.
- Generate Likelihood table by finding the probabilities of given features.
- Now, use Bayes theorem to calculate the posterior probability.

Problem: If the weather is sunny, then the Player should play or not?

Solution: To solve this, first consider the below dataset:

0	Rainy	Yes
1	Sunny	Yes
2	Overcast	Yes
3	Overcast	Yes
4	Sunny	No
5	Rainy	Yes
6	Sunny	Yes
7	Overcast	Yes
8	Rainy	No
9	Sunny	No
10	Sunny	Yes

11	Rainy	No
12	Overcast	Yes
13	Overcast	Yes

Frequency table for weather condition

Weather	Yes	No
Overcast	5	0
Rainy	2	2
Sunny	3	2
Total	10	5

Likelihood table weather condition-

Weather	No	Yes	
Overcast	0	5	5/14= 0.35
Rainy	2	2	4/14=0.29
Sunny	2	3	5/14=0.35
All	4/14=0.29	10/14=0.71	

Applying Bayes'theorem:
P(Yes|Sunny)= P(Sunny|Yes)*P(Yes)/P(Sunny)
P(Sunny|Yes)= 3/10= 0.3
P(Sunny)= 0.35
P(Yes)=0.71
So P(Yes|Sunny) = 0.3*0.71/0.35= 0.60
P(No|Sunny)= P(Sunny|No)*P(No)/P(Sunny)
P(Sunny|NO)= 2/4=0.5
P(No)= 0.29
P(Sunny)= 0.35
So P(No|Sunny)= 0.5*0.29/0.35 = 0.41
So as we can see from the above calculation that P(Yes|Sunny)>P(No|Sunny)
Hence on a Sunny day, Player can play the game.

Advantages of Naïve Bayes Classifier

- Naïve Bayes is one of the fast and easy ML algorithms to predict a class of datasets.
- It can be used for Binary as well as Multi-class Classifications.
- It performs well in Multi-class predictions as compared to the other Algorithms.

- It is the most popular choice for text classification problems.

6. What is advantages and disadvantages of locally weighted regression? Explain with suitable examples. (AKTU 2022-23)

ANSWER:

Locally Weighted Regression -is a type of non-parametric regression that creates a model for each query point by giving higher weights to nearby points. Unlike traditional regression techniques that fit a single model to the entire dataset, LWR fits models locally around each point, making it highly flexible for capturing non-linear patterns.

Locally Weighted Regression (LWR) is a non-parametric algorithm used in machine learning for regression analysis. LWR operates by assigning weights to data points based on their proximity to the query point. The closer a data point is to the query point, the higher its weight. This ensures that local patterns are given more importance in the prediction process. The weights are determined using a kernel function, typically the Gaussian kernel, which decreases the weight exponentially as the distance from the query point increases.

Advantages of Locally Weighted Regression

- Flexibility- Can model complex, non-linear relationships without specifying a global model.
- Local Adaptation- Makes predictions based on local data, leading to more accurate models in varying regions of the input space.

Challenges of Locally Weighted Regression

- Computationally Intensive- Calculating weights and fitting local models for each query point can be computationally expensive, especially for large datasets.
- Choice of Bandwidth- Selecting an appropriate bandwidth parameter is crucial and can be challenging.
- Data Sparsity- In regions with sparse data, predictions may be less reliable due to a lack of nearby points.

Example-Imagine you have data on housing prices based on various features like square footage and number of bedrooms. To predict the price of a new house using LWR:

- Choose the query point (new house features).
- Compute weights for all existing houses based on their distance to the query point.
- Fit a local regression model using the weighted houses.
- Predict the price of the new house using the fitted local model.

7. What is the assumption in naïve Bayesian algorithm that makes it different Bayesian theorem? (AKTU 2021-22)

ANSWER:

The Fundamental Naive Bayes assumption that make it different from Bayesian theorem are-

- Feature independence: The features of the data are conditionally independent of each other, given the class label.
- Continuous features are normally distributed: If a feature is continuous, then it is assumed to be normally distributed within each class.
- Discrete features have multinomial distributions: If a feature is discrete, then it is assumed to have a multinomial distribution within each class.
- Features are equally important: All features are assumed to contribute equally to the prediction of the class label.

- No missing data: The data should not contain any missing values.

8. Write short notes on Bayesian belief networks. (AKTU 2021-22)

ANSWER:

A Bayesian Belief Network (BBN) is a type of probabilistic graphical model that represents a set of variables and their conditional dependencies through a directed acyclic graph (DAG). These networks are powerful tools for reasoning under uncertainty and are widely used in artificial intelligence (AI) applications.

BBNs are built using Bayes' theorem, which allows the model to update probabilities dynamically when new evidence is introduced. This makes them essential for decision-making processes in fields like healthcare, finance, and robotics, where conditions are uncertain or incomplete.

A Bayesian Belief Network (BBN) is represented as a directed acyclic graph (DAG), where each node corresponds to a random variable, and directed edges indicate the dependencies between variables. The variables can represent real-world entities such as symptoms, weather conditions, or financial factors, with each node capturing the probability distribution for the corresponding variable.

The edges in the network define conditional dependencies, showing how the state of one variable influences another. For instance, if one node represents "Rain" and another represents "Wet Grass," a directed edge from "Rain" to "Wet Grass" indicates that the grass being wet depends on whether it rained.

Relationships between nodes are established by identifying causal dependencies or statistical correlations between variables. Each node in the network has an associated Conditional Probability Table (CPT), which stores the probability of the variable's states given the states of its parent nodes. Nodes without parents are assigned prior probabilities, while child nodes rely on their parent nodes' outcomes.

This structure ensures that the network efficiently represents joint probability distributions by breaking them down into smaller components. The absence of cycles ensures that the network can propagate probabilities without loops, maintaining consistency. BBNs offer a scalable way to model dependencies and reason under uncertainty in complex systems.

9. Discuss the concept of regression. (AKTU 2021-22)

ANSWER:

Regression is a statistical approach used to analyze the relationship between a dependent variable (target variable) and one or more independent variables (predictor variables). The primary goal is to determine the most suitable function that characterizes the connection between these variables, enabling predictions or drawing conclusions. In Regression, we plot a graph between the variables which best fits the given data points, using this plot, the machine learning model can make predictions about the data. In simple words, "**Regression shows a line or curve that passes through all the data points on target-predictor graph in such a way that the vertical distance between the data points and the regression line is minimum**." The distance between data points and line tells whether a model has captured a strong relationship or not.

Terminologies Related to the Regression Analysis-

- Dependent Variable: The main factor in Regression analysis which we want to predict or understand is called the dependent variable. It is also called target variable.
- Independent Variable: The factors which affect the dependent variables or which are used to predict the values of the dependent variables are called independent variable, also called as a predictor.
- Outliers: Outlier is an observation which contains either very low value or very high value in comparison to other observed values. An outlier may hamper the result, so it should be avoided.

- Multicollinearity: If the independent variables are highly correlated with each other than other variables, then such condition is called Multicollinearity. It should not be present in the dataset, because it creates problem while ranking the most affecting variable.
- Underfitting and Overfitting: If our algorithm works well with the training dataset but not well with test dataset, then such problem is called Overfitting. And if our algorithm does not perform well even with training dataset, then such problem is called underfitting.

Types of Regression - There are various types of regressions which are used in data Science and machine learning. Here we are discussing some important types of regression which are given below-

- Linear Regression- Linear regression is a statistical regression method which is used for predictive analysis. It is one of the very simple and easy algorithms which works on regression and shows the relationship between the continuous variables. It is used for solving the regression problem in machine learning. Linear regression shows the linear relationship between the independent variable (X-axis) and the dependent variable (Y-axis), hence called linear regression.
- Logistic Regression- Logistic regression is another supervised learning algorithm which is used to solve the classification problems. In classification problems, we have dependent variables in a binary or discrete format such as 0 or 1. Logistic regression algorithm works with the categorical variable such as 0 or 1, Yes or No, True or False, Spam or not spam, etc. It is a predictive analysis algorithm which works on the concept of probability.
- Polynomial Regression- Polynomial Regression is a type of regression which models the non-linear dataset using a linear model. It is similar to multiple linear regression, but it fits a non-linear curve between the value of x and corresponding conditional values of y.
- Decision Tree Regression- Decision Tree is a supervised learning algorithm which can be used for solving both classification and regression problems. It can solve problems for both categorical and numerical data. Decision Tree regression builds a tree-like structure in which each internal node represents the "test" for an attribute, each branch represent the result of the test, and each leaf node represents the final decision or result.
- Support Vector Regression- Support Vector Machine is a supervised learning algorithm which can be used for regression as well as classification problems. So if we use it for regression problems, then it is termed as Support Vector Regression. Support Vector Regression is a regression algorithm which works for continuous variables.

10. What is the need of Bayesian networks? Discuss about the type of support vector kernel. (AKTU 2021-22)

ANSWER:

Need of Bayesian Networks- Bayesian Networks are used across various fields for their ability to model complex relationships and make predictions. Some of the most common needs are-

- Medicine - Bayesian Networks are used for diagnostic purposes in the medical field. They help in understanding the relationships between various symptoms and diseases.
- Finance -In finance, these networks are used for risk assessment and portfolio management. They can analyze various economic indicators and their interdependencies to predict stock market trends or assess the risk of investments.
- Machine Learning - Bayesian Networks are fundamental to machine learning, particularly in areas requiring probabilistic inference. They are used for tasks like anomaly detection, where the network helps identify unusual patterns that deviate from the norm.

BUDDHA SERIES

(Unit Wise Solved Question & Answers)

Course – B.Tech. [CSE(AIML/DS)]

College – Buddha Institute of Technology

(AKTU CODE-525)

Department: Computer Science & Engineering

Subject: Machine Learning

(BCAI 601/BCDS 062)

Faculty Name: Chhiteesh Rai

Unit - 3

1. Compare and contrast Information Gain, Gain Ratio, and Gini Index in detail.
(AKTU 2023-24)

ANSWER:

Information gain, **entropy**, and **Gini index** are commonly used metrics in decision tree algorithms to determine the best split when building a tree.

Entropy is a measure of the impurity or uncertainty of a set of data. It ranges from 0 (completely pure) to 1 (completely impure). When building a decision tree, the entropy of a set is calculated before and after a split, and the change in entropy is used to determine the information gain.

$$Entropy(x) = -\sum (P(x=k) * log_2(P(x=k))$$

Information gain is a measure of the difference in entropy between the set before and after a split. The attribute that provides the highest information gain is chosen as the split attribute.

$$InformationGain(feature) = Entropy(Dataset) - Entropy(feature)$$

Gini index is another measure of impurity or uncertainty. It ranges from 0 (completely pure) to 1 (completely impure). The Gini index measures the probability of a random sample being incorrectly labeled when it is randomly labeled according to the distribution of the labels in the set. When building a decision tree, the Gini index of a set is calculated before and after a split, and the change in Gini index is used to determine the split attribute.

$$Gini\ Index = 1 - \sum (P(x=k))^2$$

In general, all three metrics can be used in decision tree algorithms to determine the best split attribute. However, some situations may favor one metric over the others. For example, when **dealing with binary classification problems, Gini index is preferred over entropy** because it tends to be more computationally efficient. On the other hand, **entropy is preferred when the data set is imbalanced,** meaning there is a significant difference in the number of instances belonging to different classes. **Information gain is a popular metric that is often used because it is easy to understand and generally works well in a variety of situations**.

2. Discuss Decision Tree and explain its working in detail. **(AKTU 2023-24)**

ANSWER: A decision tree is a type of supervised learning algorithm that is commonly used in machine learning to model and predict outcomes based on input data. It is a tree-like structure where each internal node tests on attribute, each branch corresponds to attribute value and each leaf node represents the final decision or prediction. The decision tree algorithm falls under the category of supervised learning. They can be used to solve both **regression** and **classification problems**.

Decision Tree Terminologies

There are specialized terms associated with decision trees that denote various components and facets of the tree structure and decision-making procedure. :

- **Root Node:** A decision tree's root node, which represents the original choice or feature from which the tree branches, is the highest node.
- **Internal Nodes (Decision Nodes)**: Nodes in the tree whose choices are determined by the values of particular attributes. There are branches on these nodes that go to other nodes.

- **Leaf Nodes (Terminal Nodes)**: The branches' termini, when choices or forecasts are decided upon. There are no more branches on leaf nodes.
- **Branches (Edges)**: Links between nodes that show how decisions are made in response to particular circumstances.
- **Splitting**: The process of dividing a node into two or more sub-nodes based on a decision criterion. It involves selecting a feature and a threshold to create subsets of data.
- **Parent Node**: A node that is split into child nodes. The original node from which a split originates.
- **Child Node**: Nodes created as a result of a split from a parent node.
- **Decision Criterion**: The rule or condition used to determine how the data should be split at a decision node. It involves comparing feature values against a threshold.
- **Pruning**: The process of removing branches or nodes from a decision tree to improve its generalization and prevent overfitting.

Decision Tree algorithm works in simpler steps

- **Starting at the Root**: The algorithm begins at the top, called the "root node," representing the entire dataset.
- **Asking the Best Questions:** It looks for the most important feature or question that splits the data into the most distinct groups. This is like asking a question at a fork in the tree.
- **Branching Out**: Based on the answer to that question, it divides the data into smaller subsets, creating new branches. Each branch represents a possible route through the tree.
- **Repeating the Process**: The algorithm continues asking questions and splitting the data at each branch until it reaches the final "leaf nodes," representing the predicted outcomes or classifications.

Below are some assumptions that we made while using the decision tree:

- At the beginning, we consider the whole training set as the root.
- Feature values are preferred to be categorical. If the values are continuous then they are discretized prior to building the model.
- On the basis of attribute values, records are distributed recursively.
- We use statistical methods for ordering attributes as root or the internal node.

3. Demonstrate K-Nearest Neighbors algorithm for classification with the help of an example. (AKTU-2023-24)

ANSWER: The K-NN algorithm compares a new data entry to the values in a given data set (with different classes or categories).

Based on its closeness or similarities in a given range (**K**) of neighbors, the algorithm assigns the new data to a class or category in the data set (training data).

Let's break that down into steps:

- Assign a value to **K**.
- Calculate the distance between the new data entry and all other existing data entries (you'll learn how to do this shortly). Arrange them in ascending order.
- Find the **K** nearest neighbors to the new entry based on the calculated distances.
- Assign the new data entry to the majority class in the nearest neighbors.

Distance Metric used is Euclidean Distance - This is nothing but the cartesian distance between the two points which are in the plane/hyperplane. Euclidean distance can also be visualized as the length of the straight line that joins the two points which are into consideration. This metric helps us calculate the net displacement done between the two states of an object.

distance(x,Xi)=∑j=1d(xj–Xij)2]distance(x,Xi)=∑j=1d(xj–Xij)2]

For Example- Apply K nearest neighbor classifier to predict the diabetic patient with the given features BMI, Age. If the training examples are, Assume K=3,

BMI	Age	Sugar
33.6	50	1
26.6	30	O
23.4	40	O
43.1	67	O
35.3	23	1
35.9	67	1
36.7	45	1
25.7	46	O
23.3	29	O
31	56	1

Test Example BMI=43.6, Age=40, Sugar=1

Given the dataset and new test instance, we need to find the distance from the new test instance to every training example. Here we use the euclidean distance formula to find the distance.

$$Distance = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

BMI	Age	Sugar	Formula	Distance
33.6	50	1	√((43.6-33.6)^2+(40-50)^2)	14.14
26.6	30	O	√((43.6-26.6)^2+(40-30)^2)	19.72
23.4	40	O	√((43.6-23.4)^2+(40-40)^2)	20.2
43.1	67	O	√((43.6-43.1)^2+(40-67)^2)	27
35.3	23	1	√((43.6-35.3)^2+(40-23)^2)	18.92
35.9	67	1	√((43.6-35.9)^2+(40-67)^2)	28.08
36.7	45	1	√((43.6-36.7)^2+(40-45)^2)	8.52
25.7	46	O	√((43.6-25.7)^2+(40-46)^2)	18.88
23.3	29	O	√((43.6-23.3)^2+(40-29)^2)	23.09
31	56	1	√((43.6-31)^2+(40-56)^2)	20.37

Once you calculate the distance, the next step is to find the nearest neighbors based on the value of k. In this case, the value of k is 3. Hence we need to find 3 nearest neighbors.]

BMI	Age	Sugar	Formula	Distance	Rank
33.6	50	1	√((43.6-33.6)^2+(40-50)^2)	14.14	2
26.6	30	O	√((43.6-26.6)^2+(40-30)^2)	19.72	
23.4	40	O	√((43.6-23.4)^2+(40-40)^2)	20.2	
43.1	67	O	√((43.6-43.1)^2+(40-67)^2)	27	
35.3	23	1	√((43.6-35.3)^2+(40-23)^2)	18.92	3
35.9	67	1	√((43.6-35.9)^2+(40-67)^2)	28.08	
36.7	45	1	√((43.6-36.7)^2+(40-45)^2)	8.52	1
25.7	46	O	√((43.6-25.7)^2+(40-46)^2)	18.88	
23.3	29	O	√((43.6-23.3)^2+(40-29)^2)	23.09	
31	56	1	√((43.6-31)^2+(40-56)^2)	20.37	

Now, we need to apply the majority voting technique to decide the resulting label fro the new example. Here the 1st and 2nd nearest neighbors have target label 1 and the 3rd nearest neighbor has target label 0. Target label 1 has the majority. Hence the new example is classified as 1, that is the diabetic patient has **Sugar**.

4. Describe the following concepts in decision tree in detail: (AKTU 2022-23)

I. Avoiding over fitting in decision tree.

II. Incorporating continuous valued attributes

ANSWER

I. **Avoiding over fitting in decision tree -** To avoid overfitting in a decision tree, you can use pruning, regularization, or ensemble methods. You can also use cross-validation to evaluate the model's performance on multiple subsets of the data

- **Pruning**
  - Pre-pruning
  - Stops the tree from growing before it perfectly classifies the training set. This method involves adjusting hyperparameters in the model.
  - Post-pruning
  - Allows the tree to fully grow, then removes parts of it. This method is more successful than pre-pruning because it's difficult to estimate when to stop growing the tree
- **Regularization**
  - Adds constraints or penalties to the tree growth. For example, you can limit the depth, number of nodes, or the minimum samples required for a split.
- **Ensemble methods**

- Combines multiple decision trees into a single model. This can improve the accuracy and robustness of the model.

- **Cross-validation**
  - Splits the data into training and validation sets multiple times. This helps ensure that the model generalizes well to unseen data.

II. **Incorporating continuous valued attributes -** Continuous valued attributes are numerical values that can take any value in a given range. To incorporate continuous valued attributes into a decision tree, you can discretize them into subranges. Steps for incorporating continuous valued attributes

- **Identify partitions**
  - Test the information gain of different partitions of the continuous attribute. For example, if the temperature is a continuous attribute, you can test the information gain of partitions like "temperature > 42.5".
- **Select the best partition**
  - Choose the partition with the greatest information gain.
- **Remove the attribute**
  - If the partition has the greatest information gain, remove the attribute from the list of potential attributes to split on.
- **Build the tree**
  - Build the decision tree top-down from a root node, partitioning the data into subsets with similar values. Example of a continuous attribute temperature, weight, and heigh

5. Define the following in decision tree algorithm: (AKTU 2022-23)

(i) Entropy (ii) Information gain (iii) Gini index (iv) Gain Ratio (v) Chi-Square

ANSWER:

(i) Entropy Entropy is nothing but the uncertainty in our dataset or measure of disorder. Suppose you have a group of friends who decides which movie they can watch together on Sunday. There are 2 choices for movies, one is ***"Lucy"*** and the second is ***"Titanic"*** and now everyone has to tell their choice. After everyone gives their answer we see that *"Lucy" gets 4 votes* and *"Titanic" gets 5 votes*. Which movie do we watch now? Isn't it hard to choose 1 movie now because the votes for both the movies are somewhat equal.

In a decision tree, the output is mostly "yes" or "no"

The formula for Entropy is shown below:

$$E(S) = -p_{(+)}\log p_{(+)} - p_{(-)}\log p_{(-)}$$

Here,

- p_+ is the probability of positive class
- p_- is the probability of negative class
- S is the subset of the training example

(ii) Information Gain Information gain measures the reduction of uncertainty given some feature and it is also a deciding factor for which attribute should be selected as a decision node or root node.

$$Information\ Gain = E(Y) - E(Y|X)$$

It is just entropy of the full dataset – entropy of the dataset given some feature.

(iii) Gini Index The gini index, or gini coefficient, or gini impurity computes the degree of probability of a specific variable that is wrongly being classified when chosen randomly and a variation of gini coefficient. It works on categorical variables, provides outcomes either be "successful" or "failure" and hence conducts binary splitting only.

The degree of gini index varies from 0 to 1,

- Where 0 depicts that all the elements be allied to a certain class, or only one class exists there.
- The gini index of value as 1 signifies that all the elements are randomly zdistributed across various classes, and
- A value of 0.5 denotes the elements are uniformly distributed into some classes.

(iv) Gain Ratio Proposed by John Ross Quinlan, Gain Ratio or Uncertainty Coefficient is used to normalize the information gain of an attribute against how much entropy that attribute has. Formula of gini ratio is given by

Gain Ratio=Information Gain/Entropy

(v)Chi-Square Chi-square is a statistical method used to find the difference between child and parent nodes in a decision tree. It's used to split nodes in datasets with categorical target values.

6. Explain the steps of decision tree making, with following data set, also calculate the Calculate data set entropy and information gain. (AKTU 2022-23)

ANSWER:

DAY	Outlook	Temperature	Humidity	Sun light	Play
D1	Rainy	Hot	High	Weak	No
D2	Rainy	Hot	High	strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Sunny	Mild	High	Weak	Yes
D5	Sunny	Cool	Normal	Weak	Yes

Entropy of the data set is given by-

$$E(S) = - [(3/5) \log (3/5) + (2/5) \log (2/15)] =$$

To find the information gain. It is the difference between parent entropy and average weighted entropy we found above.

7. What is the advantages & disadvantages of locally weighted regression? Explain with suitable example. (AKTU 2022-23)

ANSWER:

Advantages

- **Models Non-Linear Relationships-** LWLR captures complex, non-linear patterns in data that global linear regression often misses. **Example**: Predicting sales where seasonal trends create non-linear variations.
- **Localized Adaptability -** Fits models to subsets of data near the query point, making it highly flexible in handling datasets with varying relationships across different regions.
- **No Fixed Global Formula -** Unlike parametric models, LWLR does not assume a global form, making it robust for datasets where no clear global pattern exists.
- **Intuitive Understanding of Local Effects -** By focusing on local data points, LWLR provides insights into how individual data regions contribute to predictions.

Disadvantages

- **Computationally Intensive -** Requires fitting a model for each query point, leading to high computational costs, especially with large datasets. **Example**: Running LWLR on a dataset with millions of data points can be time-consuming.
- **Sensitivity to Bandwidth (τ\tauτ)-** The performance heavily depends on the choice of the bandwidth parameter. Poor selection can result in either overfitting or underfitting.
- **Requires Normalized Data -** Data normalization is critical to ensure accurate weight calculation. Without it, features with larger scales may dominate the model.
- **Not Scalable for Large Datasets -** Due to its memory-based nature, LWLR struggles with scalability when applied to big data.

8. Explain Instance based learning. Compare locally weighted regression and radial basis function networks. (AKTU 2022-23)

ANSWER:

Instance-based Learning, also known as Memory-based Learning, is a family of learning algorithms that, instead of performing explicit generalization, compares new problem instances with instances seen in training. It is an integral part of Machine Learning, used to classify new instances based on a similarity measure. Instance-based Learning models employ an approach where the instances are used during the learning process.

The core features of these models include the following:

The worst-case time complexity of this algorithm is **O (n)**, where n is the number of training instances. For example, If we were to create a spam filter with an instance-based learning algorithm, instead of just flagging emails that are already marked as spam emails, our spam filter would be programmed to also flag emails that are very similar to them. This requires a measure of resemblance between two emails. A similarity measure between two emails could be the same sender or the repetitive use of the same keywords or something else.

- **Classification:** Instance-based Learning excels in classifying unseen data based on the proximity to the known instances.
    - **Regression:** These models can also perform regression tasks, predicting continuous outputs.
    - **Lazy Learning:** Being a type of lazy learning, Instance-based Learning does not create general models but uses training data during the testing phase too.
- **Benefits and Use Cases -** Instance-based Learning provides various benefits:

- **Flexibility:** It can adapt quickly to changes as it does not rely on a previously built model.
- **Ease of implementation:** The algorithm is easy to implement and understand.
- **No Training Phase:** It does not require a training phase as each instance represents itself.
- Instance-based Learning has found applications in many areas such as recommendation systems, image recognition, and **computer vision**.

- **Challenges and Limitations -** Despite its benefits, Instance-based Learning also has its limitations:
  - **Performance:** The performance heavily depends on the quality of the dataset.
  - **Time and Space Intensive:** The algorithm can be computationally expensive and require significant storage space, especially for larger datasets.
  - **Sensitivity to Irrelevant Features:** It is affected by irrelevant features, which may cause misclassification.

Comparison between locally weighted regression and radial basis function

- Both of them are the techniques of instance based learning.
- LWR uses local approximation while RBF uses global approximation.
- RBF is used in neural networks.
- LWR is lazy learning while RBF is eager learning.
- Both of them are model less learning.
- LWR and RBF are parametric methods, where we do not need to store data in memory during learning phase.

9. Discuss the following issues in Decision Tree Learning: (AKTU 2021-22)

a) Overfitting the data
b) Guarding against bad attribute choices
c) Handling continuous valued attributes
d) Handling missing attribute values
e) Handling attributes with differing costs

ANSWER:

a. Overfitting the data- When we are designing a machine learning model, a model is said to be a good machine learning model, if it generalizes any new input data from the problem domain in a proper way. This helps us to make predictions in the future data, that data model has never seen. Now, suppose we want to check how well our machine learning model learns and generalizes to the new data. For that we have overfitting and underfitting, which are majorly responsible for the poor performances of the machine learning algorithms.

- **Underfitting -** A machine learning algorithm is said to have underfitting when it cannot capture the underlying trend of the data. Underfitting destroys the accuracy of our machine learning model.
- **Overfitting -** A machine learning algorithm 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 much of details and noise.

b. Guarding against bad attribute choices - There is a natural bias in the information gain measure that favors attributes with many values over those with few values.

- As an extreme example, consider the attribute Date, which has a very large number of possible values. What is wrong with the attribute Date? Simply put, it has so many possible values that it is bound to separate the training examples into very small subsets. Because of this, it will have a very high information gain relative to the training examples.
- How ever, having very high information gain, its a very poor predictor of the target function over unseen instances.

c. Handling continuous valued attributes - Our initial definition of ID3 is restricted to attributes that take on a discrete set of values.

- The target attribute whose value is predicted by learned tree must be discrete valued.
- The attributes tested in the decision nodes of the tree must also be discrete valued.

This second restriction can easily be removed so that continuous-valued decision attributes can be incorporated into the learned tree. For an attribute A that is continuous-valued, the algorithm can dynamically create a new boolean attribute A, that is true if A < c and false otherwise. The only question is how to select the best value for the threshold c.

d. Handling missing attribute values- In certain cases, the available data may be missing values for some attributes. For example, in a medical domain in which we wish to predict patient outcome based on various laboratory tests, it may be that the Blood-Test-Result is available only for a subset of the patients. In such cases, it is common to estimate the missing attribute value based on other examples for which this attribute has a known value.

e. Handling attributes with differing costs - In some learning tasks the instance attributes may have associated costs. For example, in learning to classify medical diseases we might describe patients in terms of attributes such as Temperature, BiopsyResult, Pulse, BloodTestResults, etc. These attributes vary significantly in their costs, both in terms of monetary cost and cost to patient comfort.

In such tasks, we would prefer decision trees that use low-cost attributes where possible, relying on high-cost attributes only when needed to produce reliable classifications.

10. Differentiate between Lazy and Eager Learning. (AKTU 2021-22)

ANSWER:

Aspect	Lazy Learning	Eager Learning
Timing of Model Building	The model is built during prediction.	The model is built before prediction.
Data Dependency	Relies heavily on the training data during prediction.	Less dependent on training data during prediction.
Computational Efficiency	Faster during training, but slower during prediction due	Slower during training, but faster during prediction due to pre-built

	to real-time model building.	model.
Example	k-Nearest Neighbors (KNN)	Decision Trees, Support Vector Machines (SVM), Neural Networks
Memory Usage	Less memory usage during training, but more during prediction.	More memory usage during training, but less during prediction.

11. For which problem decision tree is best suitable. (AKTU 2023-24)

ANSWER:

Decision Trees are suitable for the following cases:

- Decision Trees are most suitable for tabular data.
- The outputs are discrete.
- Explanations for Decisions are required.
- The training data may contain errors and noisy data(outliers).
- The training data may contain missing feature values.

12. Illustrate the advantages of instance-based learning techniques over other machine learning techniques. (AKTU 2022-23)

ANSWER:

- **Flexibility:** It can adapt quickly to changes as it does not rely on a previously built model. This algorithm can adapt to new data easily, one which is collected as we go .
- **Ease of implementation:** The algorithm is easy to implement and understand.
- **No Training Phase:** It does not require a training phase as each instance represents itself.
- **Interpretability**: Since the predictions are made based on known instances, it is easier to interpret why a decision was made, which is valuable in fields like medicine and finance.
- **Local approximations**: Instead of estimating for the entire instance set, local approximations can be made to the target function.
- **Handling complex relationships**: Instance-based learning can handle complex and non-linear relationships in the data without explicitly modeling them.

BUDDHA SERIES

(Unit Wise Solved Question & Answers)

Course – B.Tech. [CSE(AIML/DS)]

College – Buddha Institute of Technology

(AKTU CODE-525)

Department: Computer Science & Engineering

Subject: Machine Learning

(BCAI 601/BCDS 062)

Faculty Name: Chhiteesh Rai

Unit - 4

1. **Illustrate backpropagation algorithm by assuming the training rules for output unit weights and Hidden Unit weights. (AKTU 2023-24)**

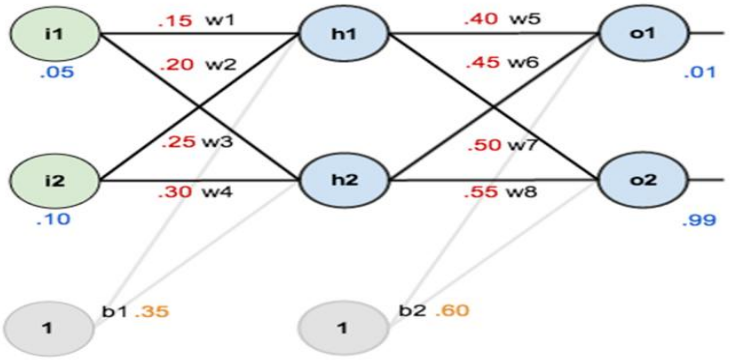
ANSWER:

Backpropagation **(**short for "***Backward Propagation of Errors***") is a method used to train artificial neural networks. Its goal is to reduce the difference between the model's predicted output and the actual output by adjusting the weights and biases in the network.

Backpropagation is a powerful algorithm in deep learning, primarily used to train artificial neural networks, particularly **feed-forward networks**. It works iteratively, minimizing the cost function by adjusting weights and biases.

In each epoch, the model adapts these parameters, reducing loss by following the error gradient. Backpropagation often utilizes optimization algorithms like **gradient descent** or **stochastic gradient descent**. The algorithm computes the gradient using the chain rule from calculus, allowing it to effectively navigate complex layers in the neural network to minimize the cost function.

Example :we're going to use a neural network with two inputs, two hidden neurons, two output neurons. Additionally, the hidden and output neurons will include a bias. With initial weights, the biases, and training inputs/ouputs:



The goal of backpropagation is to optimize the weights so that the neural network can learn how to correctly map arbitrary inputs to outputs. For the rest of this tutorial we're going to work with a single training set: given inputs 0.05 and 0.10, we want the neural network to output 0.01 and 0.99.

The Forward Pass

To begin, lets see what the neural network currently predicts given the weights and biases above and inputs of 0.05 and 0.10. To do this we'll feed those inputs forward though the network.

We figure out the total net input to each hidden layer neuron, squash the total net input using an activation function (here we use the logistic function), then repeat the process with the output layer neurons.

Here's how we calculate the total net input for :h_1:

$$net_{h1} = w_1 * i_1 + w_2 * i_2 + b_1 * 1$$
$$net_{h1} = 0.15 * 0.05 + 0.2 * 0.1 + 0.35 * 1 = 0.3775$$

We then squash it using the logistic function to get the output of h_1:

$$out_{h1} = \frac{1}{1+e^{-net_{h1}}} = \frac{1}{1+e^{-0.3775}} = 0.593269992$$

Carrying out the same process for h_2 we get:

$$out_{h2} = 0.596884378$$

We repeat this process for the output layer neurons, using the output from the hidden layer neurons as inputs.

Here's the output for o_1:

$$net_{o1} = w_5 * out_{h1} + w_6 * out_{h2} + b_2 * 1$$

$$net_{o1} = 0.4 * 0.593269992 + 0.45 * 0.596884378 + 0.6 * 1 = 1.105905967$$

$$out_{o1} = \frac{1}{1+e^{-net_{o1}}} = \frac{1}{1+e^{-1.105905967}} = 0.75136507$$

And carrying out the same process for o_2 we get:

$$out_{o2} = 0.772928465$$

Calculating the Total Error

We can now calculate the error for each output neuron using the squared error function and sum them to get the total error:

$$E_{total} = \sum \frac{1}{2}(target - output)^2$$

For example, the target output for o_1 is 0.01 but the neural network output 0.75136507, therefore its error is:

$$E_{o1} = \frac{1}{2}(target_{o1} - out_{o1})^2 = \frac{1}{2}(0.01 - 0.75136507)^2 = 0.274811083$$

Repeating this process for o_2 (remembering that the target is 0.99) we get:

$$E_{o2} = 0.023560026$$

The total error for the neural network is the sum of these errors:

$$E_{total} = E_{o1} + E_{o2} = 0.274811083 + 0.023560026 = 0.298371109$$

The Backwards Pass

Our goal with backpropagation is to update each of the weights in the network so that they cause the actual output to be closer the target output, thereby minimizing the error for each output neuron and the network as a whole.

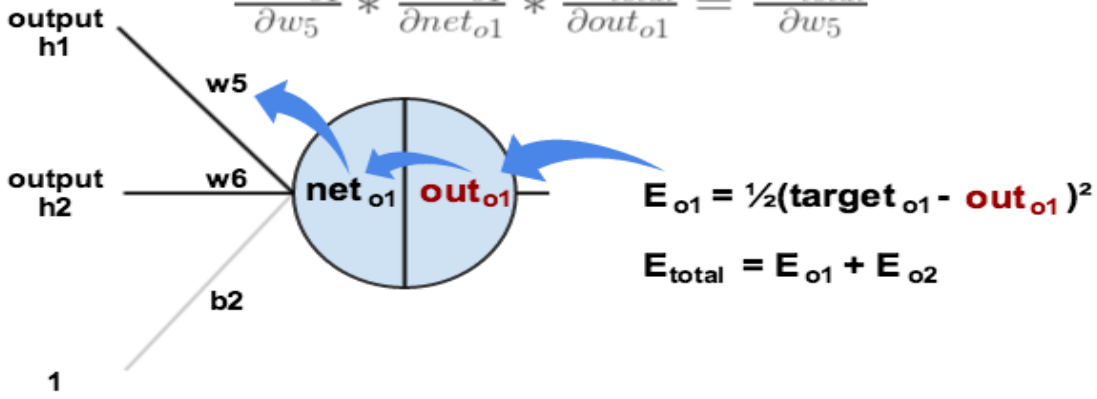
Output Layer

Consider w_5. We want to know how much a change in w_5 affects the total error, aka $\frac{\partial E_{total}}{\partial w_5}$.

y applying the chain rule we know that:

Visually, here's what we're doing:

$$\frac{\partial net_{o1}}{\partial w_5} * \frac{\partial out_{o1}}{\partial net_{o1}} * \frac{\partial E_{total}}{\partial out_{o1}} = \frac{\partial E_{total}}{\partial w_5}$$



much does the total error change with respect to the output?

$$E_{total} = \frac{1}{2}(target_{o1} - out_{o1})^2 + \frac{1}{2}(target_{o2} - out_{o2})^2$$
$$\frac{\partial E_{total}}{\partial out_{o1}} = 2 * \frac{1}{2}(target_{o1} - out_{o1})^{2-1} * -1 + 0$$
$$\frac{\partial E_{total}}{\partial out_{o1}} = -(target_{o1} - out_{o1}) = -(0.01 - 0.75136507) = 0.74136507$$

Next, how much does the output of o_1 change with respect to its total net input?

The partial derivative of the logistic function is the output multiplied by 1 minus the output:

$$out_{o1} = \frac{1}{1+e^{-net_{o1}}}$$
$$\frac{\partial out_{o1}}{\partial net_{o1}} = out_{o1}(1 - out_{o1}) = 0.75136507(1 - 0.75136507) = 0.186815602$$

Finally, how much does the total net input of $o1$ change with respect to w_5?

$$net_{o1} = w_5 * out_{h1} + w_6 * out_{h2} + b_2 * 1$$
$$\frac{\partial net_{o1}}{\partial w_5} = 1 * out_{h1} * w_5^{(1-1)} + 0 + 0 = out_{h1} = 0.593269992$$

Putting it all together:

$$\frac{\partial E_{total}}{\partial w_5} = 0.74136507 * 0.186815602 * 0.593269992 = 0.082167041$$

To decrease the error, we then subtract this value from the current weight (optionally multiplied by some learning rate, eta, which we'll set to 0.5):

$$w_5^+ = w_5 - \eta * \frac{\partial E_{total}}{\partial w_5} = 0.4 - 0.5 * 0.082167041 = 0.35891648$$

We can repeat this process to get the new weights w_6, w_7, and w_8:

$$w_6^+ = 0.408666186$$
$$w_7^+ = 0.511301270$$
$$w_8^+ = 0.561370121$$

Hidden Layer

Next, we'll continue the backwards pass by calculating new values for w_1, w_2, w_3, and w_4.

2. What is the difference between forward propagation and backward propagation in neural networks? Also explain weight calculation for forward pass network? (AKTU 2022-23)

Key Differences Between Feed-Forward and Back-Propagation		
Aspect	**Feed-Forward Neural Network (FFNN)**	**Back-Propagation Neural Network**
Definition	A type of neural network where data flows in one direction, from input to output.	A training algorithm used to adjust weights and minimize error in neural networks.
Data Flow Direction	Unidirectional, from input layer to output layer.	Bidirectional, with forward pass for predictions and backward pass for error correction.
Purpose	Used to make predictions or classifications based on input data.	Used to train the network by optimizing weights to reduce prediction errors.
Process	Involves processing inputs through layers to generate an output.	Involves calculating the error, finding gradients, and updating weights iteratively.
Complexity	Conceptually simpler, as it only requires forward data flow.	More complex, involving gradient descent and weight adjustments over multiple iterations.

Usage	Defines the structure and prediction mechanism of the network.	Applied during training to improve the network's accuracy.
Role in Learning	Does not involve learning; simply forwards inputs to produce outputs.	Essential for learning, as it adjusts weights to minimize errors over time.
Error Handling	Does not handle errors directly; outputs are generated without feedback.	Directly handles errors by propagating them backward to update weights.
Iteration	A single pass of data through the network.	Requires multiple iterations (epochs) for effective training.
Network Layers	Involves input, hidden, and output layers for data flow.	Interacts with all layers to update weights during training.

Weight calculation- Each connection between neurons is assigned a weight, representing its relative importance. When input passes through a neuron, it is multiplied by this weight. Weights can be positive (excitatory) or negative (inhibitory) and are initialized randomly at the beginning of training.

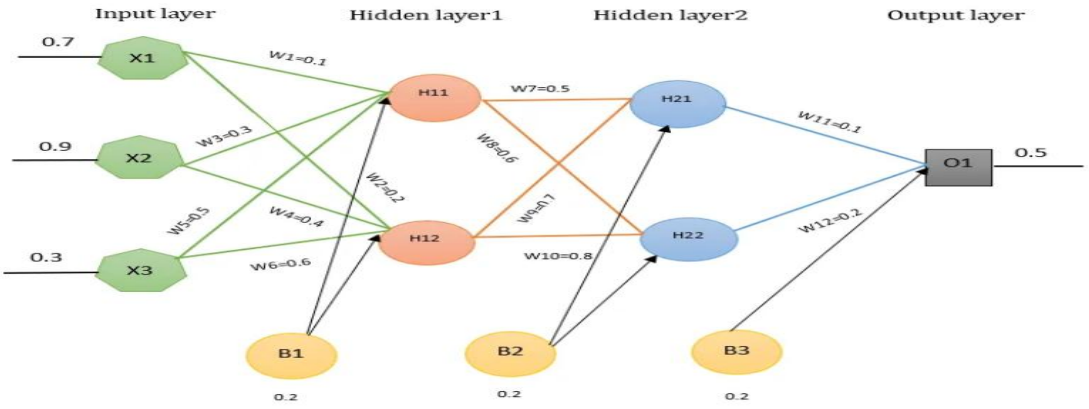
Bias: Bias is an extra input to neurons, and it is always 1, and has its own connection weight. This bias weight gives the ability to shift the activation function to the left or right, which may be critical for successful learning.

Activation Function - A neural network is composed of layers of nodes (also known as neurons). Each node in a layer receives input from multiple nodes from the previous layer. Each of these inputs is multiplied by a weight, which is a value that the network learns during training. So, for each node, we sum up the product of the input values and their associated weights. This sum is also known as the "weighted sum". We can write it as:

Weighted Sum = (input_1 * weight1) + (input2 * weight2) + … + (inputN * weightN) + bias

Forward Propagation -Forward propagation is the first step in training a neural network, where the input data is passed through the network layers to produce a prediction.

The first step in forward propagation involves calculating a weighted sum of the inputs and the associated weights, and then adding the bias. This is also known as a linear transformation. The equation is usually as follows: z = weights * inputs + bias.



Step1: Input layer to Hidden layer

$y_{h11} = W_1X_1 + W_3X_2 + W_5X_3 + B_1$

$= 0.1(0.7) + 0.3(0.9) + 0.5(0.3) + 0.2 = 0.69$

$y_{h12} = W_2X_1 + W_4X_2 + W_6X_3 + B1$

$= 0.2(0.7) + 0.4(0.9) + 0.6(0.3) + 0.2 = 0.88$

We use sigmoid Activation function

$$\sigma(Z) = \frac{1}{1 + e^{-Z}}$$

$h_{11} = \sigma(0.69) = \frac{1}{1+e^{-0.69}} \approx 0.665$

$h_{12} = \sigma(0.88) = \frac{1}{1+e^{-0.88}} \approx 0.707$

Step2: Hidden Layer1 to Hidden layer2:

$y_{h21} = W_7h_{11} + W_9h_{12} + B_1$

$= 0.5(0.665) + 0.7(0.707) + 0.2 = 1.0274$

$y_{h22} = W_8h_{11} + W_{10}h_{12} + B1$

$= 0.6(0.665) + 0.8(0.707) + 0.2 = 1.1646$

We use sigmoid Activation function

$$\sigma(Z) = \frac{1}{1 + e^{-Z}}$$

$h_{21} = \sigma(1.0274) = \frac{1}{1+e^{-1.0274}} \approx 0.736$

$h_{22} = \sigma(1.1646) = \frac{1}{1+e^{-1.1646}} \approx 0.762$

Step3: Hidden Layer2 to Output layer

$y_{o1} = W_{11}h_{21} + W_{12}h_{22} + B_1$

$= 0.1(0.736) + 0.2(0.762) + 0.2 = 0.426$

We use sigmoid Activation function

$$\sigma(Z) = \frac{1}{1 + e^{-Z}}$$

$O_1 = \sigma(0.426) = \frac{1}{1+e^{-0.426}} \approx 0.6049$

The final output of the forward pass, using the sigmoid activation function, is approximately 0.6049.

3. Write short notes on Probably Approximately Correct (PAC) learning model.

(AKTU 2023-24)

ANSWER

Probably Approximately Correct (PAC) learning is a theoretical framework introduced by Leslie Valiant in 1984. It addresses the problem of learning a function from a set of samples in a way that is both probably correct and approximately correct. In simpler terms, PAC learning formalizes the conditions under which a learning algorithm can be expected to perform well on new, unseen data after being trained on a finite set of examples.

PAC learning is concerned with the feasibility of learning in a probabilistic sense. It asks whether there exists an algorithm that, given enough examples, will find a hypothesis that is approximately

correct with high probability. The "probably" aspect refers to the confidence level of the algorithm, while the "approximately correct" aspect refers to the accuracy of the hypothesis.

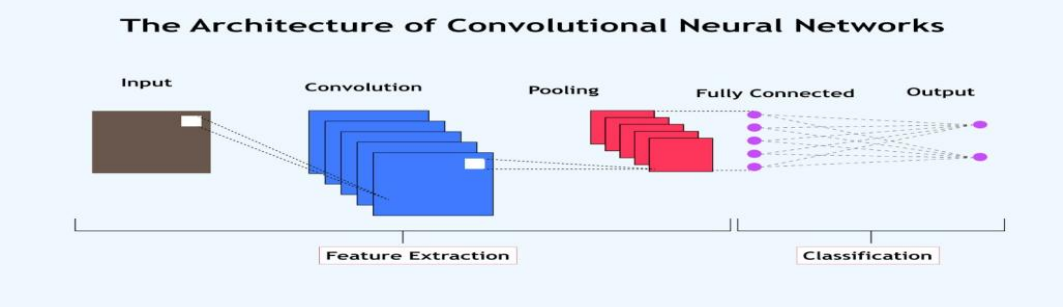
Core Concepts of PAC Learning

- Sample Complexity - Sample complexity refers to the number of samples required for a learning algorithm to achieve a specified level of accuracy and confidence. In PAC learning, sample complexity is a key measure of the efficiency of a learning algorithm. It helps determine how much data is needed to ensure that the learned hypothesis will generalize well to unseen instances.
- Hypothesis Space - The hypothesis space is the set of all possible hypotheses (or models) that a learning algorithm can choose from. In PAC learning, the size and structure of the hypothesis space play a crucial role in determining the sample complexity and the generalization ability of the algorithm.
- Generalization -Generalization is the ability of a learning algorithm to perform well on unseen data. In the PAC framework, generalization is quantified by the probability that the chosen hypothesis will have an error rate within an acceptable range on new samples. Generalization is a fundamental goal of machine learning, as it determines the practical usefulness of the learned hypothesis.
- Probably Approximately Correct (PAC) learning is a theoretical framework introduced by Leslie Valiant in 1984. It addresses the problem of learning a function from a set of samples in a way that is both probably correct and approximately correct. In simpler terms, PAC learning formalizes the conditions under which a learning algorithm can be expected to perform well on new, unseen data after being trained on a finite set of examples.
- PAC learning is concerned with the feasibility of learning in a probabilistic sense. It asks whether there exists an algorithm that, given enough examples, will find a hypothesis that is approximately correct with high probability. The "probably" aspect refers to the confidence level of the algorithm, while the "approximately correct" aspect refers to the accuracy of the hypothesis.

4. Explain the different layers used in convolutional neural network with suitable examples.
(AKTU 2023-24)

ANSWER:

Convolutional Neural Networks (CNNs) are a type of deep learning model used for image recognition, processing, and classification. Using CNNs, you can automatically and efficiently extract features from input data. Convolutional Neural Network (CNN) is the extended version of artificial neural networks (ANN) which is predominantly used to extract the feature from the grid-like matrix dataset. For example visual datasets like images or videos where data patterns play an extensive role.



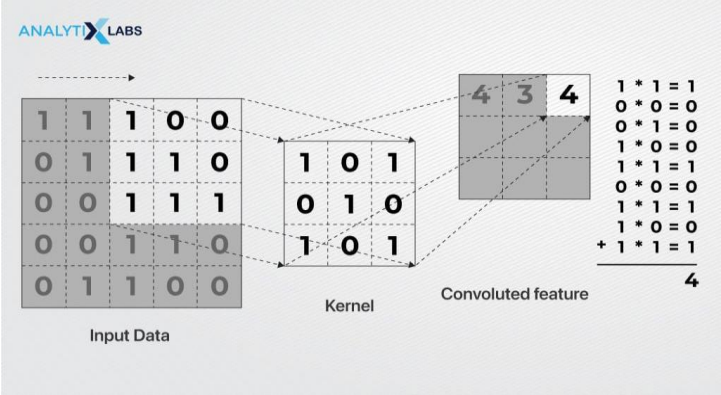
The working of CNNs is like solving a puzzle. It first identifies individual pieces (comparable to identifying features like edges or shapes in an image) and then puts them to get the full picture (similar to classification or output).

In today's world, CNNs are widely used for purposes such as video recognition (e.g., facial recognition), medical imaging (e.g., detecting cancerous tumors), self-driving cars (e.g., identifying road signs), and natural language processing (e.g., text classification).

The CNN architecture can be divided into five components. Here's an overview of the five layers.

Feature Extraction through Convolutional Layers

Of the many layers of CNN, the first layer (after the input layer) is the convolutional layer that extracts the features from the data. Convolutional layers scan the input data using filters (kernels) to detect patterns like edges, textures, or shapes.



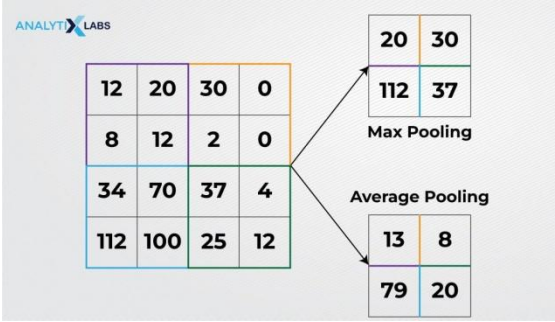
In convolution, the initial matrix is a filter, also known as a kernel, with a size of MxN. This filter comprises learnable parameters (kernel weights) that modify its functions. The second matrix is a confined portion of the image.

This process produces an outcome known as a feature map or a convolved feature, highlighting key features such as corners, edges, and other essential information in an image.

Pooling Layers

Pooling layers preserve key features while reducing computational complexity. This helps reduce multiple dimensions. This extraction can happen in two ways – average pooling or max pooling. In average pooling, the average of all the values in the pooling region is calculated, while in max pooling, the maximum value within the pooling region is considered.

By performing such a process, the resultant matrix comprises the main features of the input matrix (i.e., the input image) in fewer dimensions or, in other words, summarizes the features extracted by the convolution layer.

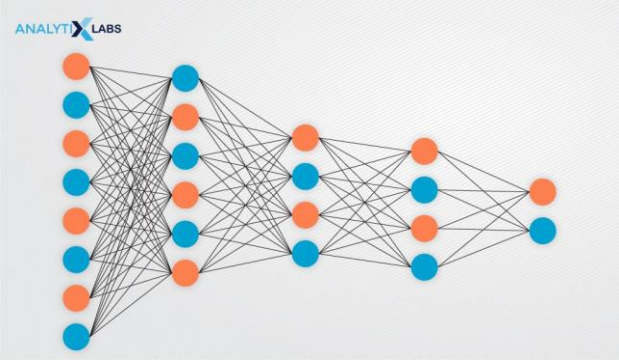


Activation Layers

Activation layers apply non-linear functions like ReLU to introduce non-linearity. This enables the network to learn complex patterns.

Flattening and Fully Connected Layers

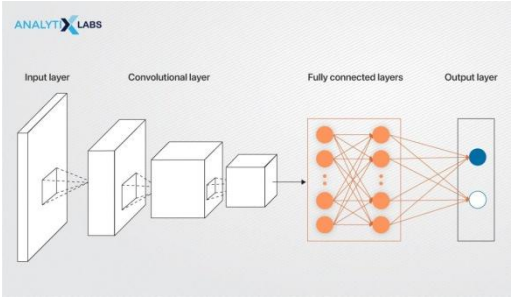
Once the feature has been extracted, the data is converted into a vector and passed through fully connected layers for classification.



Output Layer

The output layer provides the final prediction using a Softmax function for classification tasks.

Among all CNN operations, convolution is the core operation that allows the model to extract meaningful features from the input data. It applies filters to the input to detect patterns like edges, textures, and shapes in an image.



5. Define the term ANN, and CNN. (AKTU 2023-24)

ANSWER:

	ANN	**CNN**
Type of Data	Tabular Data, Text Data	Image Data
Parameter Sharing	No	Yes
Fixed Length input	Yes	Yes
Recurrent Connections	No	No
Vanishing and Exploding Gradient	Yes	Yes

Spatial Relationship	No	Yes
Performance	ANN is considered to be less powerful than CNN, RNN.	CNN is considered to be more powerful than ANN, RNN.
Application	Facial recognition and Computer vision.	Facial recognition, text digitization and Natural language processing.
Main advantages	Having fault tolerance, Ability to work with incomplete knowledge.	High accuracy in image recognition problems, Weight sharing.
Disadvantages	Hardware dependence, Unexplained behavior of the network.	Large training data needed, don't encode the position and orientation of object.

6. What is gradient descent delta rule? (AKTU 2023-24)

ANSWER:

A gradient is nothing but a derivative that defines the effects on outputs of the function with a little bit of variation in inputs.

Gradient Descent stands as a cornerstone orchestrating the intricate dance of model optimization. At its core, it is a numerical optimization algorithm that aims to find the optimal parameters—weights and biases—of a neural network by minimizing a defined cost function.

Gradient Descent (GD) is a widely used optimization algorithm in machine learning and deep learning that minimizes the cost function of a neural network model during training. It works by iteratively adjusting the weights or parameters of the model in the direction of the negative gradient of the cost function until the minimum of the cost function is reached.

7. Differentiate between Gradient Descent and Stochastic Gradient Descent. (AKTU2022-23)

ANSWER:

Batch Gradient Descent	Stochastic Gradient Descent
Computes gradient using the whole Training sample	Computes gradient using a single Training sample
Slow and computationally expensive algorithm	Faster and less computationally expensive than Batch GD
Not suggested for huge training samples.	Can be used for large training samples.
Deterministic in nature.	Stochastic in nature.
Gives optimal solution given sufficient time to converge.	Gives good solution but not optimal.
No random shuffling of points are required.	The data sample should be in a random order, and this is why we want to shuffle the training set for every epoch.
Can't escape shallow local minima easily.	SGD can escape shallow local minima more easily.
Convergence is slow.	Reaches the convergence much faster.

It updates the model parameters only after processing the entire training set.	It updates the parameters after each individual data point.
The learning rate is fixed and cannot be changed during training.	The learning rate can be adjusted dynamically.
It typically converges to the global minimum for convex loss functions.	It may converge to a local minimum or saddle point.
It may suffer from overfitting if the model is too complex for the dataset.	It can help reduce overfitting by updating the model parameters more frequently.

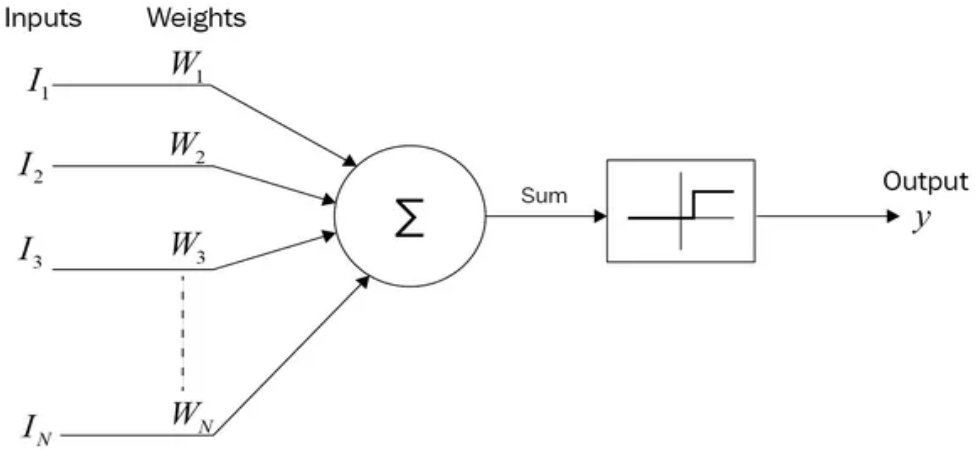
8. Discuss model representation of artificial neuron. (AKTU 2022-23)

ANSWER:

Biological neurons are the fundamental units of the brain. They consist of:

- Dendrite: Receives signals from other neurons.
- Soma: Processes the information.
- Axon: Transmits the output to other neurons.
- Synapse: Connection points to other neurons.

A neuron functions like a tiny biological computer, taking input signals, processing them, and passing on the output.



The McCulloch-Pitts Neuron is the first computational model of a neuron. It can be divided into two parts:

- Aggregation: The neuron aggregates multiple boolean inputs (0 or 1).
- Threshold Decision: Based on the aggregated value, the neuron makes a decision using a threshold function.

9. What is the role of Inductive Bias in ANN?Explain the image identification with CNN.

(AKTU 2022-23)

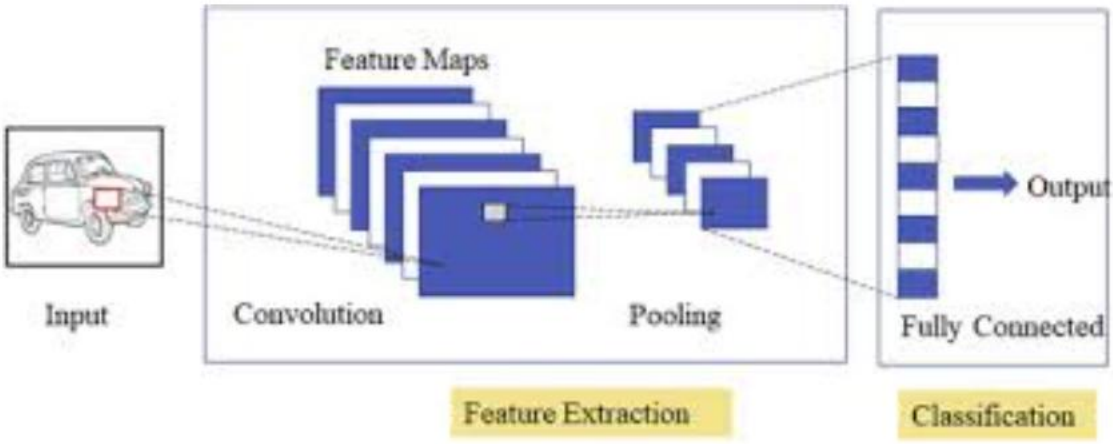
ANSWER:

Inductive bias is crucial in machine learning as it helps algorithms generalize from limited training data to unseen data. Without a well-defined inductive bias, algorithms may struggle to make accurate predictions or may overfit the training data, leading to poor performance on new data.

Understanding the inductive bias of an algorithm is essential for model selection, as different biases may be more suitable for different types of data or tasks. It also provides insights into how the algorithm is learning and what assumptions it is making about the data, which can aid in interpreting its predictions and results.

CNN is a powerful algorithm for image processing.Convolutional Neural Networks specialized for applications in image & video recognition. CNN is mainly used in image analysis tasks like Image recognition, Object detection & Segmentation.There are three types of layers in Convolutional Neural Networks:

- Convolutional Layer - In a typical neural network each input neuron is connected to the next hidden layer. In CNN, only a small region of the input layer neurons connect to the neuron hidden layer.
- Pooling Layer - The pooling layer is used to reduce the dimensionality of the feature map. There will be multiple activation & pooling layers inside the hidden layer of the CNN.
- Fully-Connected layer - Fully Connected Layers form the last few layers in the network. The input to the fully connected layer is the output from the final Pooling or Convolutional Layer, which is flattened and then fed into the fully connected layer.



10. Describe the Kohonen self-organizing maps and its algorithm. (AKTU 2022-23)

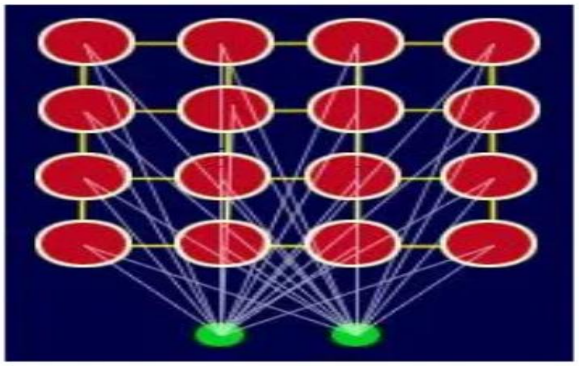
ANSWER:

The Self Organizing Map is one of the most popular neural models. It belongs to the category of the competitive learning network. The SOM is based on unsupervised learning.

The Self Organized Map was developed by professor kohenen which is used in many applications. The purpose of SOM is that it's providing a data visualization technique that helps to understand high dimensional data by reducing the dimension of data to map. SOM also represents the clustering concept by grouping similar data together. We can say that Self Organizing Map reduces data dimension and

displays similarly among data.

For example consider a two –dimensional SOM.The network is created from a 2D lattice of 'nodes', each of which is fully connected to the input layer. The below Figure shows a very small Kohonen network of 4 X 4 nodes connected to the input layer (shown in green) representing a two-dimensional vector.Each node has a specific topological position (an x, y coordinate in the lattice) and contains a vector of weights of the same dimension as the input vectors. That is to say, if the training data consists of vectors, V, of n dimensions:



A simple Kohonen network

V1, V2, V3…Vn
Then each node will contain a corresponding weight vector W, of n dimensions:
W1, W2, W3…Wn
The lines connecting the nodes in the above Figure are only there to represent adjacency and do not signify a connection as normally indicated when discussing a neural network. There are no lateral connections between nodes within the lattice.

Training occurs in several steps and over many iterations:

- Each node's weights are initialized.
- A vector is chosen at random from the set of training data and presented to the lattice.
- Every node is examined to calculate which ones weights are most like the input vector. The winning node is commonly known as the Best Matching Unit (BMU).
- The radius of the neighborhood of the BMU is now calculated. This is a value that starts large, typically set to the 'radius' of the lattice, but diminishes each time-step. Any nodes found within this radius are deemed to be inside the BMU's neighborhood.
- Each neighboring node's (the nodes found in step 4) weights are adjusted to make them more like the input vector. The closer a node is to the BMU; the more its weights get altered.
- Repeat step 2 for N iterations.

BUDDHA SERIES

(Unit Wise Solved Question & Answers)

Course – B.Tech. [CSE(AIML/DS)]

College – Buddha Institute of Technology

(AKTU CODE-525)

Department: Computer Science & Engineering

Subject: Machine Learning

(BCAI 601/BCDS 062)

Faculty Name: Chhiteesh Rai

Unit - 5

1. **Explain Q-learning with its key terms, key feature and elements. Discuss its applications used in real life. (AKTU 2023-24)**

ANSWER:

Q-learning is a model-free reinforcement learning algorithm that helps an agent learn the optimal action-selection policy by iteratively updating Q-values, which represent the expected rewards of actions in specific states. Q-learning is a basic form of Reinforcement Learning that uses Q-values (also called action values) to iteratively improve the behavior of the learning agent.

Key Terminologies in Q-learning

Before we jump into how Q-learning works, we need to learn a few useful terminologies to understand Q-learning's fundamentals.

- States(s): the current position of the agent in the environment.
- Action(a): a step taken by the agent in a particular state.
- Rewards: for every action, the agent receives a reward and penalty.
- Episodes: the end of the stage, where agents can't take new action. It happens when the agent has achieved the goal or failed.
- Q(St+1, a): expected optimal Q-value of doing the action in a particular state.
- Q(St, At): it is the current estimation of Q(St+1, a).
- Q-Table: the agent maintains the Q-table of sets of states and actions.
- Temporal Differences(TD): used to estimate the expected value of Q(St+1, a) by using the current state and action and previous state and action.

Elements of Q Learning

Q-Values or Action-Values

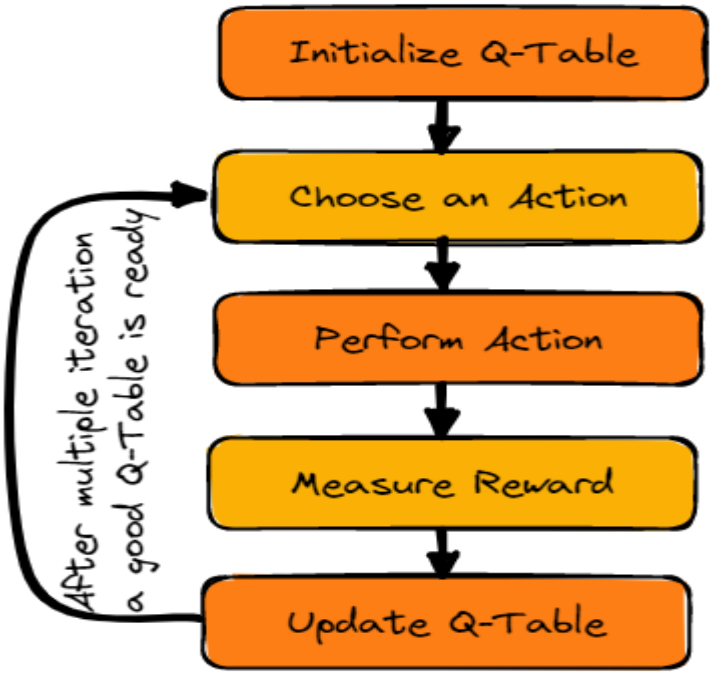
- Q-values are defined for states and actions. Q(S,A)Q(S,A) is an estimation of how good is it to take the action A at the state S . This estimation of Q(S,A)Q(S,A) will be iteratively computed using the TD- Update rule which we will see in the upcoming sections.
- Rewards and Episodes
- An agent throughout its lifetime starts from a start state, and makes several transitions from its current state to a next state based on its choice of action and also the environment the agent is interacting in. At every step of transition, the agent from a state takes an action, observes a reward from the environment, and then transits to another state. If at any point in time, the agent ends up in one of the terminating states that means there are no further transitions possible. This is said to be the completion of an episode.
- Temporal Difference or TD-Update
  The Temporal Difference or TD-Update rule can be represented as follows:
  $Q(S,A) \leftarrow Q(S,A)+\alpha(R+\gamma Q(S',A')-Q(S,A))Q(S,A) \leftarrow Q(S,A)+\alpha(R+\gamma Q(S',A')-Q(S,A))$
  This update rule to estimate the value of Q is applied at every time step of the agent's interaction with the environment. The terms used are explained below:
  - S: Current State of the agent.
  - A: Current Action Picked according to some policy.
  - S': Next State where the agent ends up.
  - A': Next best action to be picked using current Q-value estimation, i.e. pick the action with the maximum Q-value in the next state.
  - R: Current Reward observed from the environment in Response of current action.
  - $\gamma\gamma$(>0 and <=1) : Discounting Factor for Future Rewards. Future rewards are less valuable than current rewards so they must be discounted. Since Q-value is an estimation of expected rewards from a state, discounting rule applies here as well.
  - $\alpha\alpha$: Step length taken to update the estimation of Q(S, A).
- Selecting the Course of Action with ϵ-greedy policy

A simple method for selecting an action to take based on the current estimates of the Q-value is the ϵ-greedy policy. This is how it operates:

- Superior Q-Value Action (Exploitation):
  - With a probability of $1-\epsilon$, representing the majority of cases,
  - Select the action with the highest Q-value at the moment.
  - In this instance of exploitation, the agent chooses the course of action that, given its current understanding, it feels is optimal.
- Exploration through Random Action:
  - With probability ϵ, occasionally,
  - Rather than selecting the course of action with the highest Q-value,
  - Select any action at random, irrespective of Q-values.
  - In order to learn about the possible benefits of new actions, the agent engages in a type of exploration.

Q-learning models can improve processes in various scenarios. Here are a few examples of Q-learning uses:

- Energy management. Q-learning models help manage energy for different resources such as electricity, gas and water utilities. A 2022 report from IEEE provides a precise approach for integrating a Q-learning model for energy management.
- Finance. A Q-learning-based training model can build models for decision-making assistance, such as determining optimal moments to buy or sell assets.
- Gaming. Q-learning models can train gaming systems to achieve an expert level of proficiency in playing a wide range of games as the model learns the optimal strategy to advance.
- Recommendation systems. Q-learning models can help optimize recommendation systems, such as advertising platforms. For example, an ad system that recommends products commonly bought together can be optimized based on what users select.
- Robotics. Q-learning models can help train robots to execute various tasks, such as object manipulation, obstacle avoidance and transportation.
- Self-driving cars. Autonomous vehicles use many different models, and Q-learning models help train models to make driving decisions, such as when to switch lanes or stop.
- Supply chain management. The flow of goods and services as part of supply chain management can be improved with Q-learning models to help find the optimized path for products to market.



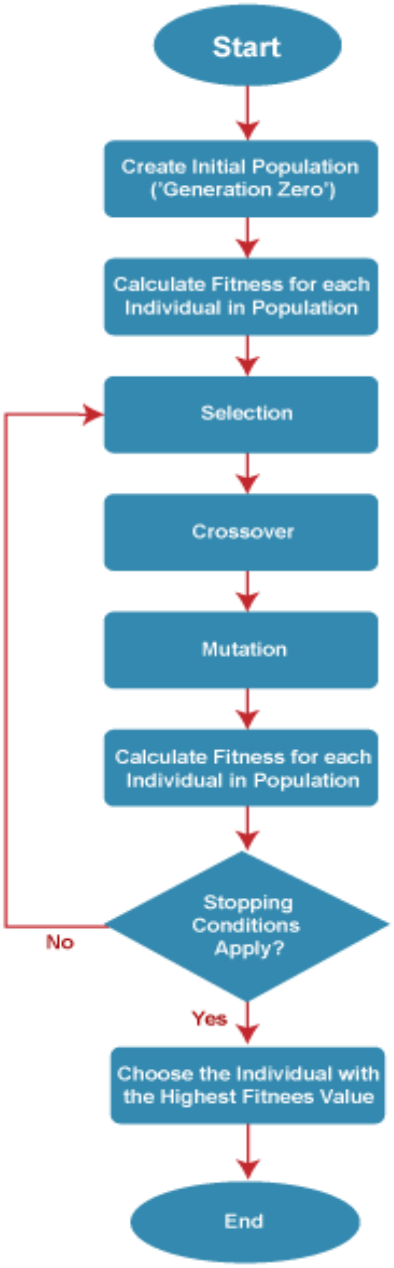
2. Define the term Genetic Algorithm. Discuss the working of Genetic algorithm with the help of flowchart. (AKTU 2023-24)

ANSWER:

A genetic algorithm (GA) is a computer science technique that uses natural selection and genetic operators to find solutions to problems. GAs are often used to solve optimization problems, and are particularly useful when the problem space is large or complex.

- Step-01- Randomly generate a set of possible solutions to a problem. Represent each solution as a fixed length character string.
- Step-02 - Using a fitness function, test each possible solution against the problem to evaluate them.
- Step-03 - Keep the best solutions. Use best solutions to generate new possible solutions.
- Step-04- Repeat the previous two steps until-
    - Either an acceptable solution is found
    - Or until the algorithm has completed its iterations through a given number of cycles / generations.
- Step-05 Basic Operators- The basic operators of Genetic Algorithm are-
- Selection (Reproduction)- It is the first operator applied on the population. It selects the chromosomes from the population of parents to cross over and produce offspring. It is based on evolution theory of "Survival of the fittest" given by Darwin. There are many techniques for reproduction or selection operator such as-
    - Tournament selection
    - Ranked position selection
    - Steady state selection etc.
- Cross Over- Population gets enriched with better individuals after reproduction phase. Then crossover operator is applied to the mating pool to create better strings. Crossover operator makes clones of good strings but does not create new ones. By recombining good individuals, the process is likely to create even better individuals.
- Mutation- Mutation is a background operator. Mutation of a bit includes flipping it by changing 0 to 1 and vice-versa. After crossover, the mutation operator subjects the strings to mutation. It facilitates a sudden change in a gene within a chromosome. Thus, it allows the algorithm to see for the solution far away from the current ones. It guarantees that the search algorithm is not trapped on a local optimum. Its purpose is to prevent premature convergence and maintain diversity within the population.

Flow Chart- The following flowchart represents how a genetic algorithm works-



3. Discuss the applications of reinforcement learning. In which problems reinforcement learning is used? (AKTU 2023-24)

ANSWER:

Applications of reinforcement learning are as follow:

- RL in Marketing
  Marketing is all about promoting and then, selling the products or services either of your brand or someone else's. In the process of marketing, finding the right audience which yields larger returns on investment you or your company is making is a challenge in itself. And, it is one of the reasons companies are investing dollars in managing digitally various marketing campaigns. Through real-time bidding supporting well the fundamental capabilities of RL, your and other companies, smaller or larger, can expect: –
  - more display ad impressions in real-time.

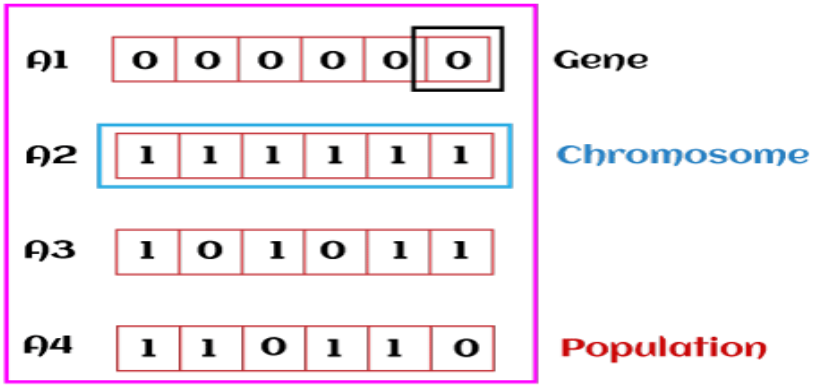
- increased ROI, profit margins.
- predicting the choices, reactions, and behavior of customers towards your products/services.

- RL in Broadcast Journalism
  Through different types of Reinforcement Learning, attracting likes and views along with tracking the reader's behavior is much simpler. Besides, recommending news that suits the frequently-changing preferences of readers and other online users can possibly be achieved since journalists can now be equipped with an RL-based system that keeps an eye on intuitive news content as well as the headlines. Take a look at other advantages too which Reinforcement Learning is offering to readers all around the world.
  - News producers are now able to receive the feedback of their users instantaneously.
  - Increased communication, as users are more expressive now.
  - No space for disinformation, hatred.
- RL in Healthcare
  Healthcare is an important part of our lives and through DTRs (a sequence-based use-case of RL), doctors can discover the treatment type, appropriate doses of drugs, and timings for taking such doses. Curious to know how is this possible!! See, DTRs are equipped with: –
  - a sequence of rules which confirm the current health status of a patient.
  - Then, they optimally propose treatments that can diagnose diseases like diabetes, HIV, Cancer, and mental illness too.
  - If required, these DTRs (i.e. Dynamic Treatment Regimes) can reduce or remove the delayed impact of treatments through their multi-objective healthcare optimization solutions.
- RL in Robotics
  Robotics without any doubt facilitates training a robot in such a way that a robot can perform tasks – just like a human being can. But still, there is a bigger challenge the robotics industry is facing today – Robots aren't able to use common sense while making various moral, social decisions. Here, a combination of Deep Learning and Reinforcement Learning i.e. Deep Reinforcement Learning comes to the rescue to enable the robots with, "Learn How To Learn" model. With this, the robots can now:
  - manipulate their decisions by grasping well various objects visible to them.
  - solve complicated tasks which even humans fail to do as robots now know what and how to learn from different levels of abstractions of the types of datasets available to them.
- RL in Gaming
  Gaming is something nowadays without which you, me, or a huge chunk of people can't live. With games optimization through Reinforcement Learning algorithms, we may expect better performances of our favorite games related to adventure, action, or mystery.
  To prove it right, the Alpha Go example can be considered. This is a computer program that defeated the strongest Go (a challenging classical game) Player in October 2015 and itself became the strongest Go player. The trick of Alpha Go to defeat the player was Reinforcement Learning which kept on developing stronger as the game is constantly exposed to unexpected gaming challenges. Like Alpha Go, there are many other games available. Even you can also optimize your favorite games by applying appropriately prediction models which learn how to win in even complex situations through RL-enabled strategies.
- RL in Image Processing
  Image Processing is another important method of enhancing the current version of an image to extract some useful information from it. And there are some steps associated like:
  - Capturing the image with machines like scanners.
  - Analyzing and manipulating it.
  - Using the output image obtained after analysis for representation, description-

purposes.
  - Here, ML models like Deep Neural Networks (whose framework is Reinforcement Learning) can be leveraged for simplifying this trending image processing method. With Deep Neural Networks, you can either enhance the quality of a specific image or hide the info. of that image. Later, use it for any of your computer vision tasks.
- RL in Manufacturing
  Manufacturing is all about producing goods that can satisfy our basic needs and essential wants. Cobot Manufacturers (or Manufacturers of Collaborative Robots that can perform various manufacturing tasks with a workforce of more than 100 people) are helping a lot of businesses with their own RL solutions for packaging and quality testing. Undoubtedly, their use is making the process of manufacturing quality products faster that can say a big no to negative customer feedback. And the lesser negative feedbacks are, the better is the product's performance and also, sales margin too.
  The overall aim is to maximize the reward as much as possible. Reinforcement learning can be used in a variety of applications, such as game playing, robotics, and autonomous driving. It is also used to solve complex problems in areas like finance, healthcare, and energy management.

4. Define the term Offspring, Chromosome and Genes are used in GA. (AKTU 2023-24)
ANSWER:

- Population: Population is the subset of all possible or probable solutions, which can solve the given problem.
- Chromosomes: A chromosome is one of the solutions in the population for the given problem, and the collection of gene generate a chromosome.
- Gene: A chromosome is divided into a different gene, or it is an element of the chromosome.
- Allele: Allele is the value provided to the gene within a particular chromosome.
- Offsping: Offspring are created by exchanging the genes of parents among themselves until the crossover point is reached. Exchanging genes among parents. The new offspring are added to the population.



5. Explain various types of reinforcement learning techniques with suitable examples. (AKTU 2022-23)

ANSWER:

Reinforcement Learning (RL) is a branch of machine learning that focuses on how agents should act in an environment to maximize cumulative rewards. It is inspired by behavioral psychology, where

Attribute	Bayesian Belief Network	Naive Bayes

agents learn through interaction with the environment and feedback. There are mainly two types of reinforcement learning:

- Positive Reinforcement: It can be understood as an event, which occurs due to specific behaviour. Positive Reinforcement Learning gives a positive impact on the action, which is taken by the agent, and it increases the two factors of the behaviour:
  - Strength
  - Frequency

  Positive reinforcement can sustain the change for a long-interval. But more than limit, it may lead overloading of states that can reduce the consequences.
- Negative Reinforcement: In terms of nature of process, it is contrary to or opposite of positive Reinforcement Learning. It can be said as the event, which aims to strengthen the behaviour that occurs due to a negative condition which have to be stopped, or avoided. It is defined to have minimum performance.

Approaches of Reinforcement Machine Learning -There are mainly three approaches of reinforcement learning:

- Value Based: Within Value-based method, we try to maximize the value function V(s). In this approach method, the agent can be dreamt of a long-term return at any state under policy (π) at any state.
- Policy Based: Within Policy-based approach, we focus on choosing the best policy in which the performed action in every state, maximizes the reward in the future. There are two types of policy-based approaches:
  - Deterministic: For every state, the same will be taken under a policy (π).
    N {a,s} = P/A = a/S = S]
  - Stochastic: All of the actions have their own certain probabilities, which are determined by the given stochastic function:
- Model Based: In Model-based approach, we do create a virtual model for each environment, and the agent try learn from specific environments.

6. How to identify the reproduction cycle of genetic algorithm? Explain with suitable example? (AKTU 2022-23)

ANSWER:

New generations are created using different methods such as stochastic universal sampling, random selection and genetic operators (crossover and mutation).

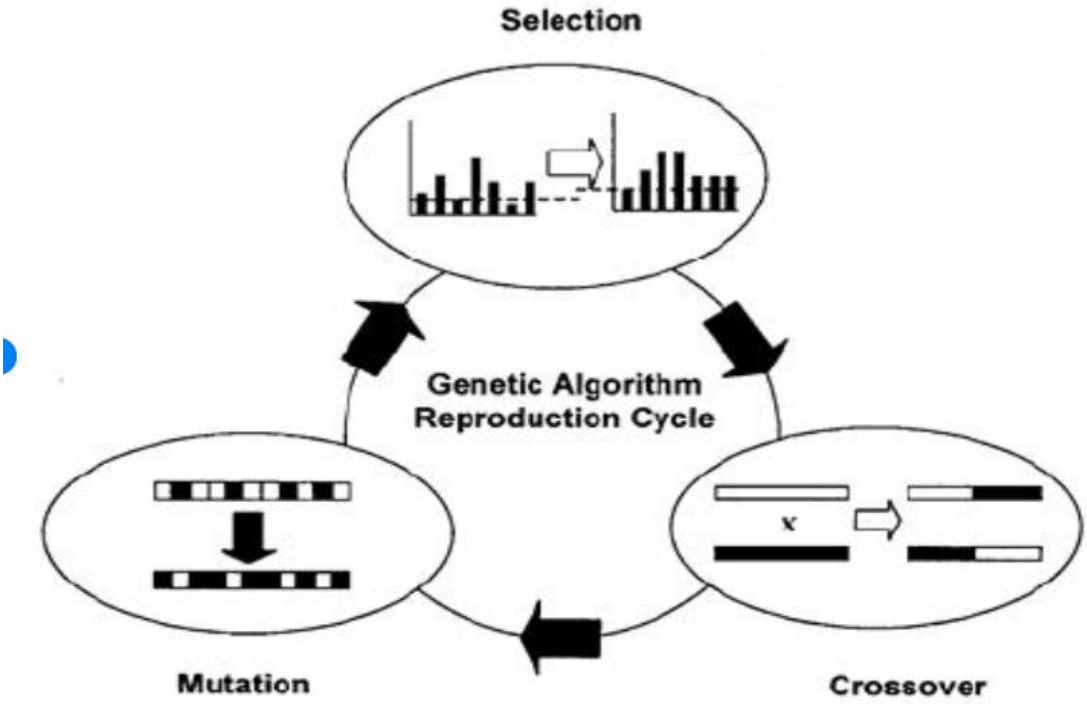
A typical GA can be divided into five stages: initial population, fitness function, selection, crossover and mutation. Below are the different phases of the Genetic Algorithm:

- Initialization of Population(Coding) - Every gene represents a parameter (variables) in the solution. This collection of parameters that forms the solution is the chromosome. So, the population is a collection of chromosomes. Order of genes on the chromosome matters. Chromosomes are often depicted in binary as 0's and 1's, but other encodings are also possible.
- Fitness Function - We have to select the best ones to reproduce offspring out of the available chromosomes, so each chromosome is given a fitness value. The fitness score helps to select the individuals who will be used for reproduction.
- Selection - This phase's main goal is to find the region where getting the best solution is more. Inspiration for this is from the survival of the fittest. It should be a balance between exploration and exploitation of search space. GA tries to move the genotype to higher fitness in the search space. Too strong fitness selection bias can lead to sub-optimal solutions. Too little fitness bias selection results in an unfocused search. Thus,

Fitness proportionate selection is used, also known as roulette wheel selection, as a genetic operator used in genetic algorithms to select potentially useful recombination solutions.

- Reproduction - Generation of offsprings happen in 2 ways:
  - Crossover - Crossover is the most vital stage in the genetic algorithm. During crossover, a random point is selected while mating a pair of parents to generate offsprings. There are 3 major types of crossover –
    - Single Point Crossover: A point on both parents' chromosomes is picked randomly and designated a 'crossover point'. Bits to the right of that point are exchanged between the two parent chromosomes.
    - Two-Point Crossover: Two crossover points are picked randomly from the parent chromosomes. The bits in between the two points are swapped between the parent organisms.
    - Uniform Crossover: In a uniform crossover, typically, each bit is chosen from either parent with equal probability.

    The new offspring are added to the population.
  - Mutation - In a few new offspring formed, some of their genes can be subjected to a low random probability mutation. This indicates that some of the bits in the bit chromosome can be flipped. Mutation happens to take care of diversity among the population and stop premature convergence.
- Convergence (when to stop) - Few rules which are followed which tell when to stop is as follows:
  - When there is no improvement in the solution quality after completing a certain number of generations set beforehand.
  - When a hard and fast range of generations and time is reached.
  - Till an acceptable solution is obtained.



7. Is reinforcement learning Artificial intelligence or Machine Learning? Also explain the

data mining in ML. **(AKTU 2022-23)**

ANSWER:

Reinforcement learning (RL) is a type of machine learning (ML), which is a branch of artificial intelligence (AI).

Data mining is a popular and multidisciplinary field that mainly focuses on finding useful information from a large volume of data. Machine learning (ML), on the other hand, is a subset of data science. ML primarily focuses on creating algorithms that can learn and predict from given data. Machine learning and data mining can be combined to deliver results that can help make better business decisions and boost the profit margins of an organization.

Machine learning use in data mining - In spite of all the differences, machine learning and data mining have many similarities as well. Both use analytical processes and are good at recognizing patterns. Sometimes, machine learning techniques can be used in data mining to get accurate outputs.

Here are some of the scenarios where machine learning can help in tackling the challenges of data mining.

- The quality of the output of data mining tools depends on the data quality. It sometimes may not even address the data quality issues. This leads to wrong results as the tool analyzes faulty data. So, it is important to clean the data before processing it.
- In such situations, machine learning algorithms are recommended as they can be incorporated with data mining tools to automate the data entry process and get quality data. This combination can easily identify any duplicate data and eliminate it. After this, a random forest algorithm can be used to classify the data.
- Data mining tools can be used to identify process-related issues, but they cannot find the root cause of the issues. Machine learning algorithms, on the contrary, can help in solving the problem. We can also introduce software with root cause analysis and data mining tools that can tackle these kinds of issues.
- Real-time data can be structured and unstructured. Some traditional data mining tools can process only structured data and, therefore, are not applicable to unstructured data. This can be solved by using these two machine learning algorithms - Optical Character Recognition (OCR) and Natural Processing Language (NLP).
- Machine learning techniques help in converting unstructured data to a machine-readable format so that the data mining tool can do a better analysis and make decisions. Note that developers need to pay attention while converting unstructured data into the machine-readable format as they can result in imperfect data and produce errors.
- Sometimes, data mining tools provide less clarity when processing a large number of variables. The addition of data increases the complexity of the data mining outputs which is hard for humans to understand. Data mining tools integrated with machine learning algorithms and computer vision help to overcome this. Hence processed data can be captured and the relevant output can be generated.
- Data mining tools analyze the past performance of the process rather than analyzing the ongoing process. They cannot guarantee predicting performance in the future. Using machine learning applications with data mining can predict the final results and future events. They also send an alert message to users if there are any shortcomings and if any improvements are required.

8. Differentiate between Q learning and Deep Learning. **(AKTU 2022-23)**

ANSWER:

The main difference between deep and regular Q-learning is the implementation of the Q-table. In deep Q-learning, this is replaced with two neural networks that handle the learning process. While these networks have the same overarching architectures, they have different weights.

	Q-learning	Deep Q-learning	Deep Q-network
Approach	Tabular learning using Q-table	Function approximation with neural networks	Function approximation with neural networks
Input	(state, action) pairs	Raw State input	Raw State input
Output	Q-values for each (state, action) pair	Q-values for each (state, action) pair	Q-values for each (state, action) pair
Training data	Q-table entries	Experience Replay buffer	Experience Replay buffer
Training time	Fast	Slow	Slow
Complexity	Limited by the number of states and actions	More complex due to the use of neural networks	More complex due to the use of neural networks
Generalization	Limited to states in Q-table	Can generalize to unseen states	Can generalize to unseen states
Scalability	Struggles with large state and action spaces	Handles large spaces well	Handles large spaces well
Stability	Prone to overfitting	More stable than Q-learning, but can	More stable than Q-learning and

9. What is the problem of crowding in GA. Comparison of purely analytical and purely inductive learning. (AKTU 2023-24)

ANSWER:

The replacement phase of crowding is usually carried out through deterministic or probabilistic crowding, which have the limitations that they apply the same selective pressure regardless of the problem being solved and the stage of genetic algorithm search.

The two main differences between analytical and inductive learning methods are the use of prior knowledge and data requirements.

	Analytical learning	Inductive learning
Prior Knowledge	Incorporates prior knowledge and assumptions to guide the learning process	Relies solely on data to derive patterns and rules.
Data Requirements	Can work with smaller datasets due to the incorporation of prior knowledge.	Requires a large amount of data to derive accurate rules.

	Analytical learning	Inductive learning
Prior Knowledge	Incorporates prior knowledge and assumptions to guide the learning process	Relies solely on data to derive patterns and rules.
Data Requirements	Can work with smaller datasets due to the incorporation of prior knowledge.	Requires a large amount of data to derive accurate rules.

10. Illustrate the process of Q-learning and discuss the following terms:

(i) Q-values or action value

(ii) Rewards and Episode

(iii)Temporal difference or TD update. **(AKTU 2022-23)**

ANSWER:

(i) Q-Values or Action-Values

Q-values are defined for states and actions. Q(S,A)Q(S,A) is an estimation of how good is it to take the action A at the state S . This estimation of Q(S,A)Q(S,A) will be iteratively computed using the TD- Update rule which we will see in the upcoming sections.

(ii) Rewards and Episodes

An agent throughout its lifetime starts from a start state, and makes several transitions from its current state to a next state based on its choice of action and also the environment the agent is interacting in. At every step of transition, the agent from a state takes an action, observes a reward from the environment, and then transits to another state. If at any point in time, the agent ends up in one of the terminating states that means there are no further transitions possible. This is said to be the completion of an episode.

(iii) Temporal Difference or TD-Update

The Temporal Difference or TD-Update rule can be represented as follows:

$Q(S,A) \leftarrow Q(S,A)+\alpha(R+\gamma Q(S',A')-Q(S,A))Q(S,A) \leftarrow Q(S,A)+\alpha(R+\gamma Q(S',A')-Q(S,A))$

This update rule to estimate the value of Q is applied at every time step of the agent's interaction with the environment. The terms used are explained below:

- S: Current State of the agent.
- A: Current Action Picked according to some policy.
- S': Next State where the agent ends up.
- A': Next best action to be picked using current Q-value estimation, i.e. pick the action with the maximum Q-value in the next state.
- R: Current Reward observed from the environment in Response of current action.
- $\gamma\gamma$(>0 and <=1) : Discounting Factor for Future Rewards. Future rewards are less valuable than current rewards so they must be discounted. Since Q-value is an estimation of expected rewards from a state, discounting rule applies here as well.
- $\alpha\alpha$: Step length taken to update the estimation of Q(S, A).