1. Define Machine Learning. Why machine learning is important? Differentiate traditional programming vs machine learning?

- ❖ The term Machine Learning was first coined by Arthur Samuel in the year 1959.

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

- ❖ "A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P if its performance at tasks in T, as measured by P, improves with experience E."
  --by Tom M. Mitchell, 1997

- ❖ In simple terms, Machine learning is a subset of Artificial Intelligence (AI) which provides machines the ability to learn automatically & improve from experience without being explicitly programmed to do so. In the sense, it is the practice of getting Machines to solve problems by gaining the ability to think.

- ❖ There are three main types of machine learning:
  1. **Supervised Learning:** Trains models with labelled dataset, making predictions or classifications based on known outcomes.
  2. **Unsupervised Learning:** Deals with unlabelled data to discover patterns, clusters, or associations without predefined targets.
  3. **Reinforcement Learning:** Involves an agent interacts with an environment and learns to make decisions by receiving feedback in the form of rewards or penalties.

Here's a list of reasons **why Machine Learning is so important**:

1. **Increase in Data Generation:** Due to excessive production of data, we need a method that can be used to structure, analyse and draw useful insights from data. This is where Machine Learning comes in. It uses data to solve problems and find solutions to the most complex tasks faced by organizations.
2. **Improve Decision Making:** By making use of various algorithms, Machine Learning can be used to make better business decisions. For example, Machine Learning is used to forecast sales, predict downfalls in the stock market, identify risks and anomalies, etc.
3. **Uncover patterns & trends in data:** Finding hidden patterns and extracting key insights from data is the most essential part of Machine Learning. By building predictive models and using statistical techniques, Machine Learning allows you to dig beneath the surface and explore the data at a minute scale. Understanding data and extracting patterns manually will take days, whereas Machine Learning algorithms can perform such computations in less than a second.
4. **Solve complex problems:** From detecting the genes linked to the deadly ALS disease to building self-driving cars, Machine Learning can be used to solve the most complex problems.

Traditional programming vs machine learning

Aspect	Traditional Programming	Machine Learning
Approach	In traditional programming, a programmer writes explicit rules or instructions for the computer to follow.	In machine learning, instead of writing explicit rules, a programmer trains a model using a large dataset.
Data Dependency	Relies less on data. The quality of the output depends mainly on the logic defined by the programmer.	Heavily reliant on data. The quality and quantity of the training data significantly impact the performance and accuracy of the model.
Flexibility and Adaptability	Has limited flexibility. Changes in the problem domain require manual updates to the code.	Offers higher adaptability to new scenarios, especially if the model is retrained with updated data.
Problem Complexity	Suitable for tasks with well-defined rules.	Suitable for tasks with complex, non-linear patterns.
Input	Explicitly provided by the programmer.	Derived from data; the system learns from examples.
Rule Definition	Rules are explicitly defined by the programmer.	Rules are learned from data through training.
Human Involvement	High level of human involvement in rule specification.	Lesser human involvement in specifying detailed rules; more emphasis on data and algorithms.
Error Handling	Error handling is explicitly programmed.	The model learns from errors and adjusts its behaviour.
Development Process	Development process is generally linear and predictable, focusing on implementing and debugging predefined logic.	Involves an iterative process where models are trained, evaluated, and fine-tuned. This Process can be less predictable and more experimental
Primary Objective	The primary Objective is to meet functional and non-functional requirements.	The primary goal is to optimize the metric(accuracy, precision/recall, RMSE etc.) of the model
Examples	Basic algorithms, scripts, software applications.	Image recognition, natural language processing, recommendation systems.

2. Discuss some applications of machine learning with examples.

1. **Image Recognition:**
   - Image recognition involves training algorithms to identify and classify objects or patterns within visual data.
   - Example: Facial recognition systems in smartphones, object detection in self-driving cars, or image categorization in social media.
2. **Speech Recognition:**
   - It involves training models to understand and interpret spoken words.
   - Example: Voice assistants like Siri or Google Assistant, transcription services, or voice-activated commands in smart devices.
3. **Traffic Prediction:**
   - Predicting and optimizing traffic flow to improve navigation and reduce congestion.
   - Example: Navigation apps like Google Maps predicting travel time based on current and historical traffic conditions.
4. **Product Recommendation:**
   - Product recommendation systems analyze user behavior and preferences to suggest items that users might be interested in.
   - Example: Amazon's product recommendations, Netflix suggesting movies, or personalized recommendations on e-commerce websites.
5. **Autonomous Vehicles:**
   - Autonomous vehicles use a combination of sensors and machine learning algorithms to navigate and make decisions without human intervention.
   - Example: Self-driving cars from companies like Tesla, Waymo, or autonomous delivery robots.
6. **Email Spam Filtering:**
   - E-mail spam filtering involves classifying emails as either spam or not spam based on their content and characteristics.
   - Example: Gmail's spam filter, which uses machine learning to identify and filter out unwanted emails.
7. **Virtual Personal Assistant:**
   - Virtual personal assistants use natural language processing and machine learning to understand and respond to user queries or commands
   - Example: Virtual personal assistants use natural language processing and machine learning to understand and respond to user queries or commands.
8. **Online Fraud Detection:**
   - Online fraud detection systems analyze patterns of user behavior to identify and prevent fraudulent activities.
   - Example: Credit card fraud detection, where machine learning models detect unusual spending patterns or transactions.
9. **Stock Market Trading:**
   - Machine learning is used in stock market trading to analyze historical data, predict market trends, and make investment decisions.
   - Example: Algorithmic trading systems using machine learning to predict market trends and execute trades.
10. **Medical Diagnosis:**
    - Machine learning in medical diagnosis involves analyzing patient data to assist in the identification and prediction of diseases.
    - Example: Image analysis for diagnosing medical images (e.g., X-rays, MRIs), predicting disease risk based on patient data.
11. **Automatic Language Translation:**
    - Application: Translating text or speech from one language to another.
    - Example: Google Translate, which employs neural machine translation to provide translations between numerous languages.

3. Explain different perspectives and issues in machine learning.

Machine Learning involves searching a very large space of possible hypotheses that fits observed data and any prior knowledge held by the observer

1. **Algorithmic Perspective:**
   - Issue: Bias and Fairness
   - Machine learning algorithms can inherit biases present in the training data, leading to unfair or discriminatory outcomes. Ensuring fairness and mitigating bias in algorithms is a crucial concern.
2. **Data Perspective:**
   - Issue: Data Quality and Quantity
   - The success of machine learning models heavily depends on the quality and quantity of the training data. Incomplete, biased, or unrepresentative data can lead to inaccurate or unreliable predictions.
3. **Ethical Perspective:**
   - Issue: Ethical Use of AI
   - Decisions made by machine learning models can have ethical implications. Issues like privacy, consent, and the responsible use of AI raise concerns about the societal impact of machine learning applications.
4. **Interpretability Perspective:**
   - Issue: Lack of Model Interpretability
   - Many machine learning models, especially complex ones like deep neural networks, are considered "black boxes" because their decision-making process is not easily interpretable. Understanding and interpreting model decisions is important for user trust and accountability.
5. **Deployment Perspective:**
   - Issue: Model Deployment Challenges
   - Transitioning from a trained model to a deployed and operational system can be challenging. Integration with existing systems, scalability, and real-world implementation pose various deployment-related issues.
6. **Data Quality Perspectives:**
   - Issue: Garbage In, Garbage Out
   - The performance of machine learning models heavily depends on the quality and representativeness of the training data. Poor-quality or biased data can lead to inaccurate predictions and unreliable models.
7. **Security Perspective:**
   - Issue: Adversarial Attacks
   - Machine learning models are vulnerable to adversarial attacks, where malicious actors manipulate input data to mislead the model's predictions. Developing robust models that can withstand such attacks is a significant challenge.
8. **Human-Machine Interaction Perspective:**
   - Issue: Human Understanding of AI Decisions
   - Ensuring that users and stakeholders can understand and trust the decisions made by machine learning models is crucial for widespread acceptance and successful integration into various applications.
9. **Resource Perspective:**
   - Issue: Computing Resources and Energy Consumption
   - Training complex machine learning models, especially deep neural networks, requires significant computing resources. The environmental impact and energy consumption of large-scale machine learning operations are growing concerns.
10. **Educational Perspective:**
    - Issue: Skill Gap and Education

- The rapid evolution of machine learning technologies has created a demand for skilled professionals. There is a need for comprehensive educational programs to bridge the gap between the demand for ML expertise and the available workforce.

11. Environmental Perspectives:

- Issue: Energy Consumption
- Some complex machine learning models, particularly deep neural networks, can be computationally expensive and energy-intensive. Optimizing models for efficiency and exploring green AI solutions are environmental considerations.

Issues in Machine Learning

During the development phase our focus is to select a learning algorithm and train it on some data, the two things that might be a problem are a bad algorithm or bad data, or perhaps both of them.

The following are some of the Issues in ML

1. Not enough training data.

- Machine Learning is not quite there yet; it takes a lot of data for most Machine Learning algorithms to work properly.
- Even for very simple problems you typically need thousands of examples, and for complex problems such as image or speech recognition you may need millions of examples.

2. Poor Quality of data:

- Obviously, if your training data has lots of errors, outliers, and noise, it will make it impossible for your
- machine learning model to detect a proper underlying pattern.
- Hence, it will not perform well.
- So put in every ounce of effort in cleaning up your training data.
- No matter how good you are in selecting and hyper tuning the model, this part plays a major role in helping us make an accurate machine learning model.
- "Most Data Scientists spend a significant part of their time in cleaning data".
- There are a couple of examples when you'd want to clean up the data :
    - If you see some of the instances are clear outliers just discard them or fix them manually.
    - If some of the instances are missing a feature like (E.g., 2% of user did not specify their age), you can either ignore these instances, or fill the missing values by median age, or train one model with the feature and train one without it to come up with a conclusion.

3. Irrelevant Features:

"Garbage in, garbage out (GIGO)."

- In the above image, we can see that even if our model is “AWESOME” and we feed it with garbage data, the result will also be garbage(output). Our training data must always contain more relevant and less to none irrelevant features.
- The credit for a successful machine learning project goes to coming up with a good set of features on which it has been trained (often referred to as feature engineering), which includes feature selection, extraction, and creating new features which are other interesting topics to be covered in upcoming blogs.

4. Nonrepresentative training data:

- To make sure that our model generalizes well, we have to make sure that our training data should be representative of the new cases that we want to generalize to.
- If train our model by using a nonrepresentative training set, it won’t be accurate in predictions it will be biased against one class or a group.

For E.G., Let us say you are trying to build a model that recognizes the genre of music. One way to build your training set is to search it on youtube and use the resulting data. Here we assume that youtube’s search engine is providing representative data but in reality, the search will be biased towards popular artists and maybe even the artists that are popular in your location(if you live in India you will be getting the music of Arijit Singh, Sonu Nigam or etc).

- So use representative data during training, so your model won’t be biased among one or two classes when it works on testing data.

5. Overfitting the Training Data

Overfitting happens when the model is too complex relative to the amount and noisiness of the training data.

The possible solutions are:

To simplify the model by selecting one with fewer parameters (e.g., a linear model rather than a high-degree polynomial model), by reducing the number of attributes in the training data or by constraining the model

- To gather more training data

- To reduce the noise in the training data (e.g., fix data errors and remove outliers)

6. Underfitting the Training Data

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

For example, a linear model of life satisfaction is prone to underfit; reality is just more complex than the model, so its predictions are bound to be inaccurate, even on the training examples.

The main options to fix this problem are:

- Selecting a more powerful model, with more parameters

- Feeding better features to the learning algorithm (feature engineering)

- Reducing the constraints on the model (e.g., reducing the regularization hyperparameter)

4. What do you mean by well posed learning problem? Differentiate supervised, unsupervised and reinforcement learning?

WELL-POSED LEARNING

Definition: A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks T, as measured by P, improves with experience E.

To have a well-defined learning problem, three features need to be identified:

1. The class of tasks
2. The measure of performance to be improved
3. The source of experience

In a well-posed learning problem, the problem statement is clearly defined, and the solution is expected to be stable, unique, and have a meaningful interpretation.

Clear problem formulation is crucial for the success of machine learning tasks, and well-posed problems provide a solid foundation for developing effective and reliable models.

Examples:

1. **A robot driving learning problem:**
    - Task T: driving on public, 4-lane highway using vision sensors
    - Performance measure P: average distance travelled before an error (as judged by human overseer)
    - Training experience E: a sequence of images and steering commands recorded while observing a human driver

2. **A Handwriting Recognition Learning Problem**
    - Task T: recognizing and classifying handwritten words within images
    - Performance measure P: percent of words correctly classified
    - Training experience E: a database of handwritten words with given classifications

3. **Text Categorization Problem**
    - Task T: assign a document to its content category
    - Performance measure P: Precision and Recall
    - Training experience E: Example pre-classified documents

4. **A checkers learning problem:**
    - Task T: playing checkers
    - Performance measure P: percent of games won against opponents
    - Training experience E: playing practice games against itself

5. **Spam Filter:**
    - Task-T: Classifying e-mails into spam or non-spam
    - Training experience E: Training dataset (e-mails with labels)
    - Performance-P: Accuracy

Differences between supervised, unsupervised and reinforcement learning:

Feature	Supervised Learning	Unsupervised Learning	Reinforcement Learning
Definition	The Machine learns by using labelled data	Machine is trained on unlabelled data without any guidance	An agent interacts with its environment by performing actions and learning from trail and error
Types of Problems	Regression & Classification	Association & clustering	Reward based
Type of data	Labelled data	Unlabelled data	No – predefined data
Training	External supervision	No supervision	No supervision
Approach	Maps the labelled inputs to the known outputs	Understands patterns & discover outputs	Follow the trail and error method
Aim	Calculate outcomes	Discover underlying patterns	Learns a series of action
Common Algorithms	Linear Regression, Decision Trees, Neural Networks	K-Means Clustering, Principal Component Analysis (PCA)	Q-Learning, Deep Q Networks (DQN), Policy Gradient methods
Model Building	Model is built and trained prior to testing	Model is built and trained prior to testing	The model is trained and tested simultaneously.
Examples	Email spam filtering, image recognition	Customer behavior analysis, market basket analysis	Autonomous vehicles, robotics, AlphaGo

5. Differentiate Artificial Intelligence vs Deep Learning vs Machine Learning with Suitable Examples?

ARTIFICIAL INTELLIGENT:

- "Artificial Intelligence is defined as a field of science and engineering that deals with making intelligent machines or computers to perform human-like activities"
- It encompasses various approaches, including rule-based systems, expert systems, and learning-based systems.

DEEP LEARNING:

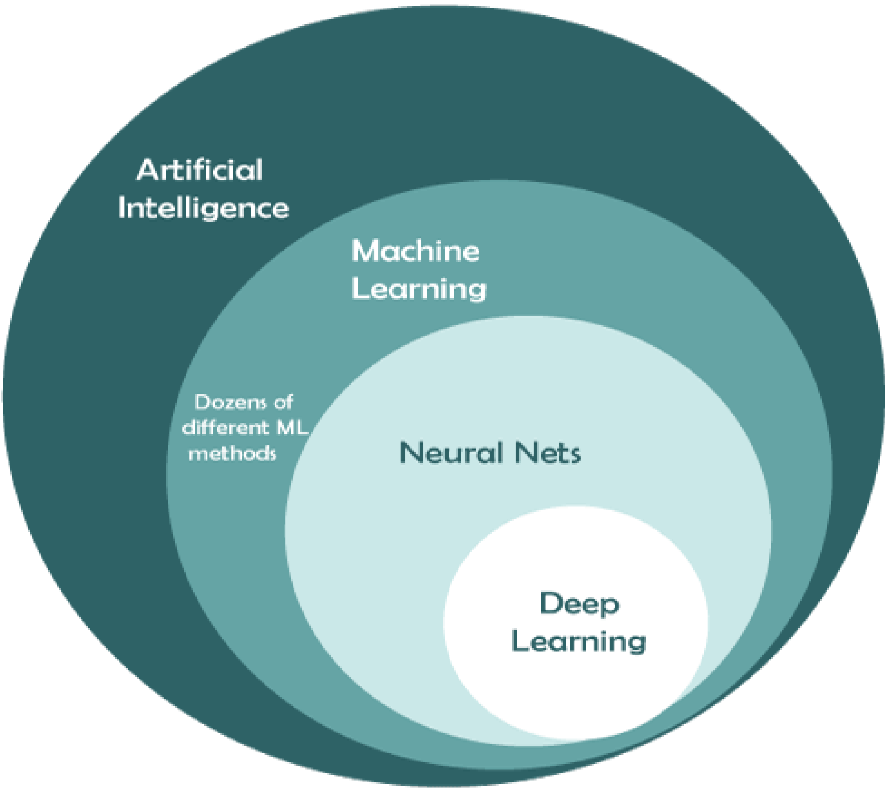
- "Deep learning is defined as the subset of machine learning and artificial intelligence that is based on artificial neural networks".
- In deep learning, the deep word refers to the number of layers in a neural network.
- Deep Learning is a set of algorithms inspired by the structure and function of the human brain.

MACHINE LEARNING:

- "A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P if its performance at tasks in T, as measured by P, improves with experience E."
- "Machine Learning is the field of study that gives computers the ability to learn without being explicitly programmed"

Artificial Intelligence, Machine Learning, Deep learning

- ❖ Deep Learning, Machine Learning, and Artificial Intelligence are the most used terms on the internet for IT folks. However, all these three technologies are connected with each other. Artificial Intelligence (AI) can be understood as an umbrella that consists of both Machine learning and deep learning. Or We can say deep learning and machine learning both are subsets of artificial intelligence.
- ❖ As these technologies look similar, most of the persons have misconceptions about 'Deep Learning, Machine learning, and Artificial Intelligence' that all three are similar to each other. But in reality, although all these technologies are used to build intelligent machines or applications that behave like a human, still, they differ by their functionalities and scope.
- ❖ It means these three terms are often used interchangeably, but they do not quite refer to the same things. Let's understand the fundamental difference between deep learning, machine learning, and Artificial Intelligence with the below image.



With the above image, you can understand Artificial Intelligence is a branch of computer science that helps us to create smart, intelligent machines. Further, ML is a subfield of AI that helps to teach machines and build AI-driven applications. On the other hand, Deep learning is the sub-branch of ML that helps to train ML models with a huge amount of input and complex algorithms and mainly works with neural networks.

Feature	AI	DL	ML
Definition	Development of systems with human-like intelligence	Subset of ML focusing on neural networks with multiple layers	Development of algorithms enabling systems to learn and make decisions
Approach	Rule-based, expert systems, and learning-based approaches	Primarily neural network architectures with multiple layers	Algorithms that learn from data without explicit programming
Data Dependency	Can be data-dependent, may also rely on rules and expert knowledge	Heavily reliant on large datasets for training	Requires labelled data for training; performance improves with more data
Problem Complexity	Can handle a broad range of complexities	Particularly effective for handling complex problems	Suitable for a wide range of problem complexities
Input	Can come from various sources, including structured and unstructured data	Deals with raw input data (images, audio, text)	Processes structured or unstructured data
Rule Definition	Rules often explicitly defined by humans or knowledge engineers	Learns complex rules from data during training	Rules can be explicitly defined or learned from data
Human Involvement	Substantial involvement in rule definition and system design	Mainly in designing architecture, selecting parameters, and preparing data	Needed for feature engineering, algorithm selection, and providing labelled data
Error Handling	Depends on implementation, handled through rule refinement or feedback	Integral part of training process; model learns from mistakes	Based on algorithmic performance; adjustments may be made
Flexibility	Generally, more flexible due to various AI approaches	Flexible in learning complex hierarchical features from data	Adaptable based on the learning algorithms and data
Training Data	May involve labelled or unlabelled data, rule-based knowledge	Requires labelled data for supervised learning, large datasets for effective training	Requires labelled data for supervised learning, may use unlabelled data for unsupervised learning
Real-World Example	Smart personal assistants (Siri, Google Assistant) using natural language processing	Image recognition, Speech recognition, Language translation	Email spam filters that learn to identify spam based on examples

6. What is Statistical Learning? What is the objective of statistical learning?

Statistical learning:

- Statistical Learning theory is a framework for machine learning drawing from the fields of statistics and functional analysis.
- When the goal is to interpret the model and quantify the uncertainty in the data, this analysis is usually referred to as statistical learning.
- It focuses on developing methods and algorithms for making predictions or decisions based on data.
- Statistical learning theory deals with the statistical inference problem at finding predictive function based on data.
- Statistical theory has lead to successful applications in fields such as computer vision,speech recognition and Bioinformatics.
- From perspective of statistical learning, Supervised learning is best understood(from training set of data). we get to predict or estimate an outcome based on previously present output
- with unsupervised statistical learning, we find various patterns present within the data by clustering them into similar groups.
- Statistical Learning helps us understand why a system behaves the way it does.
- It reduces ambiguity and produces results that matter in the real world.
- Statistical Learning provides accurate results that can find medical, business, banking, and government applications.

Objectives of Statistical Learning:

- The primary objective of statistical learning is to understand the underlying patterns and relationships in data, allowing for accurate prediction, classification, or decision-making in new, unseen instances, minimizing the loss or empherical risk.
- Gaining knowledge, making predictions, making decisions or constructing models from a set of data.

Some of the objectives are:

1. **Prediction:**
    - **Objective:** Minimize the prediction error or loss function, ensuring that the model generalizes well to new data.
2. **Inference:**
    - **Objective:** Understand underlying data patterns and relationships between variables to gain insights into the factors that influence the outcomes.
3. **Decision Making:**
    - **Objective:** Develop models that aid decision-making processes, such as in business, healthcare, finance, and other fields.
4. **Pattern Recognition:**
    - **Objective**: Identify and leverage patterns, trends, and structures in the data.
5. **Classification and Regression:**
    - **Objective:** Categorize data (classification) or predict continuous variables (regression).
6. **Feature Selection and Dimensionality Reduction:**
    - **Objective:** Improve model efficiency by selecting relevant features and reducing data dimensionality.

7. **Model Interpretability:**
    - **Objective:** Enhance understanding by creating interpretable models for stakeholders.
8. **Scalability and Efficiency:**
    - **Objective:** Ensure models handle large datasets and are computationally efficient.
9. **Robustness and Generalization:**
    - **Objective:** Develop models that generalize well to new data and remain robust.
10. **Ethical Considerations:**
    - **Objective:** Address biases, fairness, and transparency in model predictions, considering ethical implications.

7.How Regression and Classification are different? Explain in detail with real time examples?

Regression and classification are two types of supervised learning techniques in machine learning, each addressing different types of prediction problems.

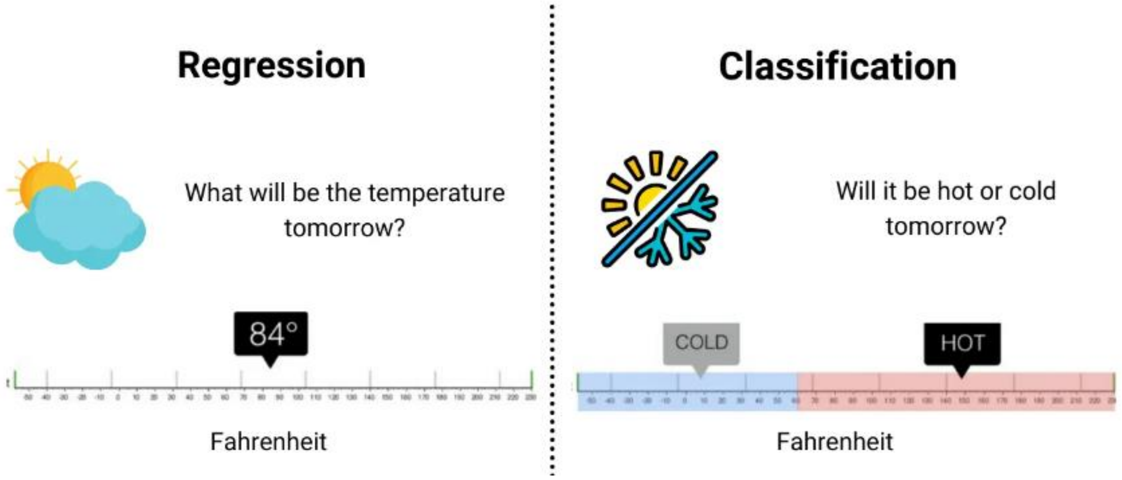
Regression:

- Regression involves predicting a continuous output or numeric value based on input features.
- The goal is to establish a relationship between the independent variables (features) and the dependent variable (output) to make accurate predictions.
- Regression algorithms are used to solve regression problems in which there is a linear relationship between input and output variables.
- These are used to predict continuous output variables, such as market trends, weather prediction, etc.
- Some popular Regression algorithms are given below:
    - Simple Linear Regression Algorithm
    - Multivariate Regression Algorithm
    - Decision Tree Algorithm
    - Lasso Regression

Classification:

- Classification deals with predicting the category or class of an observation based on input features.
- It involves assigning instances to predefined classes or labels.
- Classification algorithms are used to solve the classification problems in which the output variable is categorical, such as "Yes" or No, Male or Female, Red or Blue, etc.
- The classification algorithms predict the categories present in the dataset.
- Some real-world examples of classification algorithms are Spam Detection, Email filtering, etc.
- Some popular classification algorithms are given below:
    - Random Forest Algorithm
    - Decision Tree Algorithm
    - Logistic Regression Algorithm
    - Support Vector Machine Algorithm

Aspect	Regression	Classification
Task	The regression algorithm's task is mapping input value (x) with continuous output variable (y).	The classification algorithm's task mapping the input value of x with the discrete output variable of y.
Output Type	Continuous numerical values.	Discrete categorical labels or classes.
Nature of Output	Real number within a range	Class label or category
Evaluation Metrics	Uses metrics like Mean Squared Error (MSE), Mean Absolute Error (MAE)	Uses metrics like Accuracy, Precision, Recall, F1 Score, etc.
Model Output	Numeric value	Probability scores or predicted class labels
Decision Boundary	No clear decision boundary in the output space	Decision boundary separates different classes
Division	We can further divide Regression algorithms into Linear and Non-linear Regression.	We can further divide Classification algorithms into Binary Classifiers and Multi-class Classifiers.
Use Cases	Financial forecasting, predicting sales, predicting temperature, etc.	Email spam detection, image recognition, sentiment analysis, etc.
Examples	Predicting house prices, stock prices, temperature, etc.	Spam detection, image recognition, sentiment analysis, etc.
Graph		



Detailed Explanation with Real-Time Examples:

Regression:

1. **Housing Price Prediction:**
    - **Objective:** Predict the price of a house based on features like square footage, number of bedrooms, and location.
    - **Output:** Continuous numerical values representing the predicted price.
    - **Example:** Given a dataset of houses with known features and prices, a regression model can learn the relationship between the features and the house prices. For a new house with specified features, the model can predict its price.
2. **Temperature Forecast:**
    - **Objective:** Predict the temperature for the next day based on historical weather data and other relevant features.

- **Output:** Continuous numerical values representing the predicted temperature.
- **Example:** Using historical weather data (temperature, humidity, wind speed, etc.), a regression model can learn to predict the temperature for the following day.

<u>Classification:</u>

1. **Spam Email Detection:**
    - **Objective:** Classify emails as either spam or not spam based on their content and features.
    - **Output:** Discrete categorical labels (spam or not spam).
    - **Example:** A classification model trained on a dataset of labelled emails can learn to distinguish between spam and non-spam emails. It can then be used to classify new, unseen emails as spam or not spam.
2. **Image Classification (Cat vs. Dog):**
    - **Objective:** Identify whether an image contains a cat or a dog.
    - **Output:** Discrete categorical labels (cat or dog).
    - **Example:** Using a dataset of labelled images of cats and dogs, a classification model can learn to distinguish between the two. Given a new image, the model can predict whether it contains a cat or a dog.

8.How to estimate the loss function in statistical learning? Discuss how to assess model accuracy?

In statistical learning, estimating the loss function and assessing model accuracy are crucial steps in evaluating the performance of a predictive model.

<u>Loss function:</u>

- Loss is a value that represents the summation of errors in our model.
- To calculate the loss, a loss or cost function is used.
- The loss function, also referred to as the error function, is a crucial component in machine learning that quantifies the difference between the predicted outputs of a machine learning algorithm and the actual target values. If your predictions are totally off, your loss function will output a higher number. If they're pretty good, it'll output a lower number.
- For each prediction that we make, our loss function will simply measure the absolute difference between our prediction and the actual value.
- In mathematical notation, it might look something like

  abs(y_predicted – y_actual).

Estimating the Loss Function:

The choice of a loss function depends on the nature of the problem (regression or classification). Common loss functions include:

1. **Mean Squared Error (MSE):**

- MSE is commonly used to solve regression problems.
- Mean squared error (MSE) is the workhorse of basic loss functions.
- It's easy to understand and implement and generally works pretty well.
- To calculate MSE, you take the difference between your predictions and the ground truth, square it, and average it out across the whole dataset.
- Formula:

$$MSE = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$$

Where,

- n is the number of observations.
- Y_i is the actual value for the i-th observation.
- $y\hat{}_i$ is the predicted value for the i-th observation.

- A lower MSE indicates better performance.

2. **Likelihood loss:**

- The likelihood function is also relatively simple, and is commonly used in classification problems.
- The function takes the predicted probability for each input example and multiplies them.
- And although the output isn't exactly human-interpretable, it's useful for comparing models.

3. **Cross-Entropy Loss (Log Loss):**

- Log loss is a loss function also used frequently in classification problems,It's just a straightforward modification of the likelihood function with logarithms.
- Measures the performance of a classification model whose output is a probability value.
- Formula:

$$\text{Log Loss} = -\frac{1}{n} \sum_{i=1}^{n} [y_i \cdot \log(\hat{y}_i) + (1 - y_i) \cdot \log(1 - \hat{y}_i)]$$

- Lower log loss indicates better performance.

<u>Accuracy:</u>

- Accuracy is a method for measuring a classification model's performance.
- It is typically expressed as a percentage.
- Accuracy score in machine learning is an evaluation metric that measures the number of correct predictions made by a model in relation to the total number of predictions made.
- We calculate it by dividing the number of correct predictions by the total number of predictions.

$$Accuracy = \frac{Number\ of\ Correct\ Predictions}{Total\ Number\ of\ Predictions}$$

This formula provides an easy-to-understand definition that assumes a binary classification problem.

- Accuracy is often graphed and monitored during the training phase though the value is often associated with the overall or final model accuracy.
- Accuracy is easier to interpret than loss.

Assess Model Accuracy:

- Assessing model accuracy involves various metrics that provide insights into the model's overall performance.

Assessing Model Accuracy:

1. **Accuracy:**
   - **Formula:**

$$\text{Accuracy} = \frac{\text{Number of Correct Predictions}}{\text{Total Number of Predictions}}$$

   - **Interpretation:** The proportion of correctly predicted instances. Commonly used in classification problems.

2. **Precision:**
   - precision is defined as the percentage of the correct predictions among the total prediction of a class between all of the classes present in the dataset.
   - **Formula:**

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}}$$

   - **Interpretation:** The ability of the model to correctly identify positive instances among instances predicted as positive.

3. **Recall (Sensitivity or True Positive Rate):**
   - Recall is defined as the proportion of the correct predictions among the total prediction of a class between all of the classes present in the dataset.
   - **Formula:**

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}}$$

   - **Interpretation:** The ability of the model to correctly identify all positive instances.

4. **F1 Score:**
   - F-score is defined as a metric combination of precision and also the recall.
   - **Formula:**

$$F1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

   - **Interpretation:** The harmonic mean of precision and recall, providing a balance between the two metrics.

5. **Receiver Operating Characteristic (ROC) Curve:**
   - ROC is defined as the binary classification of the diagnostic plot of the curve.
   - Plots the true positive rate against the false positive rate at various threshold settings.

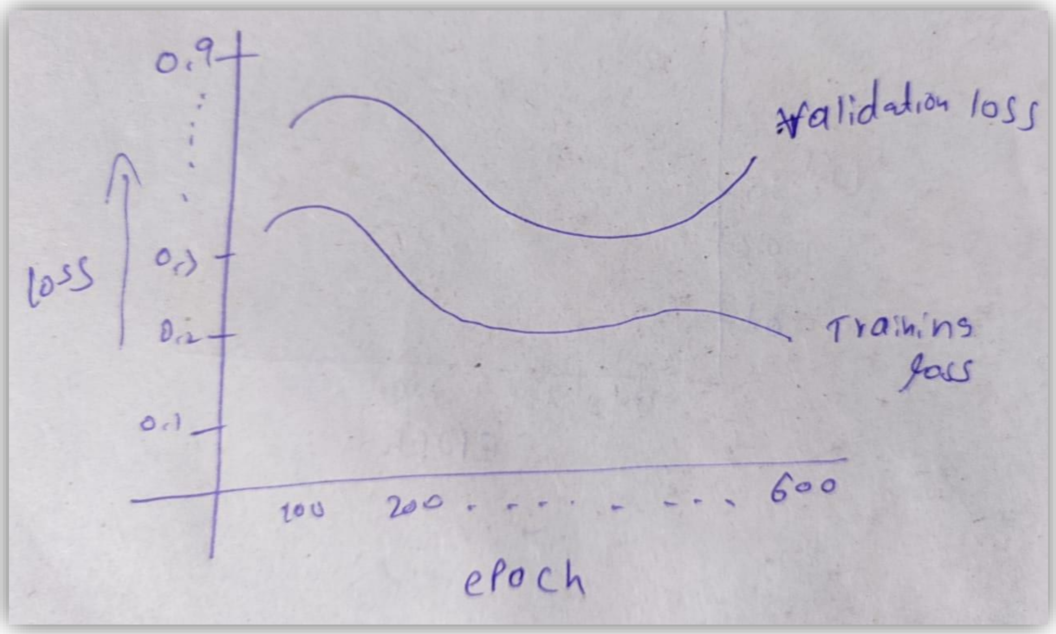
6. **Area Under the Curve (AUC):**
   - Measures the area under the ROC curve. AUC close to 1 indicates a good model.

9.Define the terms over fitting and under fitting? How can we resolve these issues?

Overfitting and underfitting are common issues in machine learning where the model's performance is affected negatively.

Overfitting:

- If the model performs well on the training data but poorly on the validation set, then this scenario is called Overfitting.



- Here,the validation loss decreases till it reaches a particular point then it increases whereas, training loss keeps decreasing.
- Overfitting happens when the model is too complex relative to the amount and noisiness of the training data or the model was trained for a long period.
- When a model gets trained with so much data, it starts learning from the noise and inaccurate data entries in our data set. And when testing with test data results in High variance.

Reasons for Overfitting:

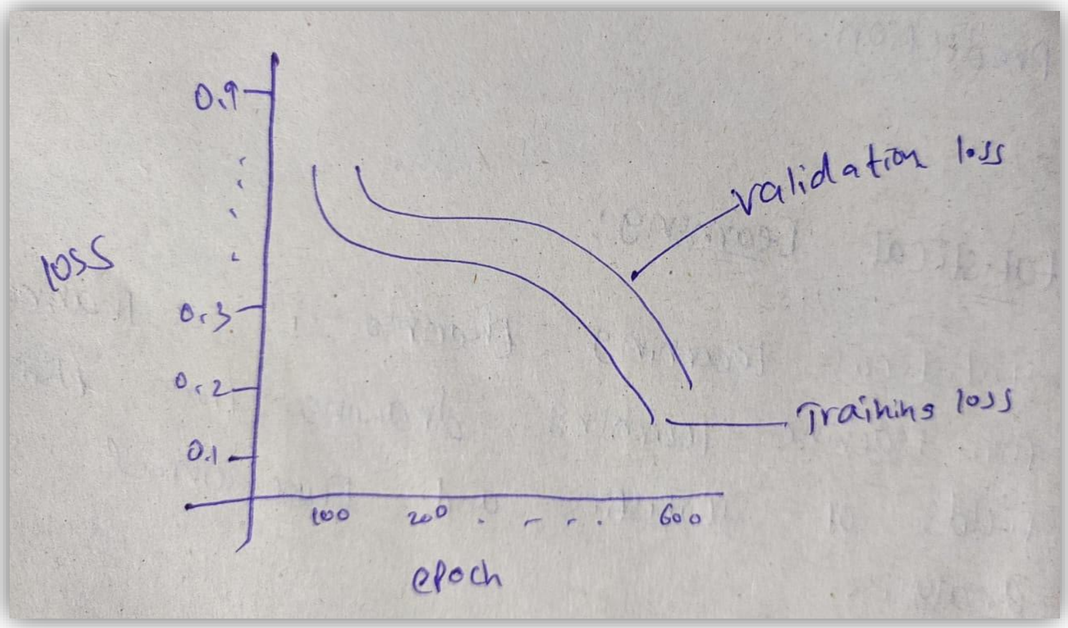
- High variance and low bias.
- The model is too complex.
- The size of the training data.

Techniques to Reduce Overfitting

- Increase training data.
- Reduce model complexity.
- Early stopping during the training phase (training can be halted when the loss is low and stable).
- By reducing the number of attributes in the training data or by constraining the model
- Ridge Regularization and Lasso Regularization.
- Use dropout for neural networks to tackle overfitting.

Underfitting:

- At times, the validation loss is greater than training loss. This may indicate that the model is underfitting.



- Here, both validation loss and training loss decreases.
- Underfitting occurs when the model is unable to accurately model the training data and hence generate errors.
- Underfitting is the opposite of overfitting.
- It occurs when your model is too simple to learn the underlying structure of the data.
- It represents the inability of the model to learn the training data effectively result in poor performance both on the training and testing data.
- In simple terms, an underfit model's are inaccurate, especially when applied to new, unseen examples.

Reasons for Underfitting

- The model is too simple, So it may be not capable to represent the complexities in the data.
- The input features which is used to train the model is not the adequate representations of underlying factors influencing the target variable.
- The size of the training dataset used is not enough.
- Excessive regularization are used to prevent the overfitting, which constraint the model to capture the data well.
- Features are not scaled.

Techniques to Reduce Underfitting

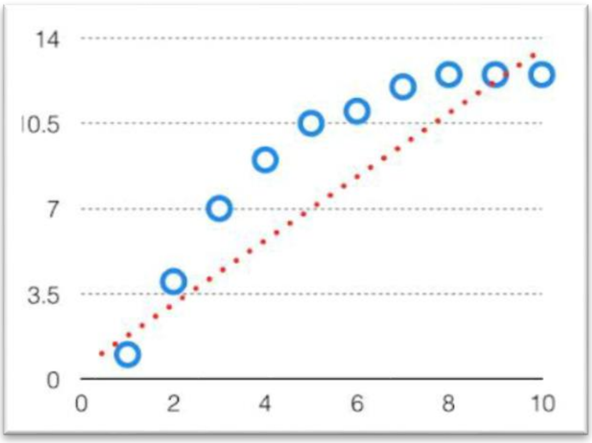
- Increase model complexity.
- Increase the number of features, performing feature engineering.
- Remove noise from the data.
- Increase the number of epochs or increase the duration of training to get better results.
- Increasing the training data.
- Furthur training.

10.Describe bias and variance? Explain Bias variance trade-offs in statistical learning?

It is important to understand prediction errors (bias and variance) when it comes to accuracy in any machine learning algorithm. There is a trade-off between a model's ability to minimize bias and variance which is referred to as the best solution for selecting a value of **Regularization**(Constraining a model to make it simpler and reduce the risk of overfitting is called regularization.) constant. Proper understanding of these errors would help to avoid the overfitting and underfitting of a data set while training the algorithm.

Bias

The bias is known as the difference between the prediction of the values by the ML model and the correct value. Being high in biasing gives a large error in training as well as testing data. Its recommended that an algorithm should always be low biased to avoid the problem of underfitting.
By high bias, the data predicted is in a straight line format, thus not fitting accurately in the data in the data set. Such fitting is known as **Underfitting of Data**. This happens when the hypothesis is too simple or linear in nature. Refer to the graph given below for an example of such a situation.



High Bias

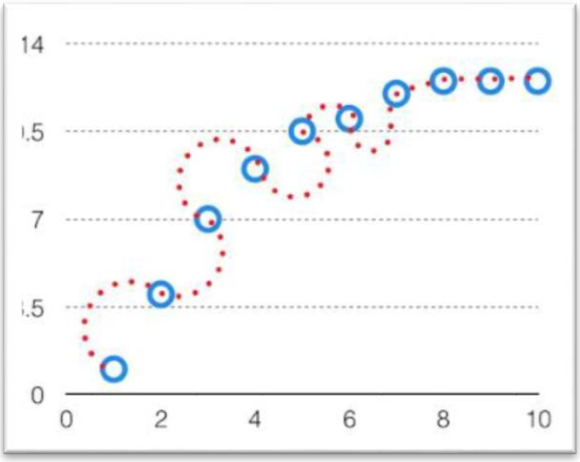
In such a problem, a hypothesis looks like follows.

$$h_\theta(x) = g(\theta_0 + \theta_1 x_1 + \theta_2 x_2)$$

Variance

The variability of model prediction for a given data point which tells us spread of our data is called the variance of the model. The model with high variance has a very complex fit to the training data and thus is not able to fit accurately on the data which it hasn't seen before. As a result, such models perform very well on training data but has high error rates on test data.
When a model is high on variance, it is then said to as **Overfitting of Data**. Overfitting is fitting the training set accurately via complex curve and high order hypothesis but is not the solution as the error with unseen data is high.
While training a data model variance should be kept low.

The high variance data looks like follows.



High Variance

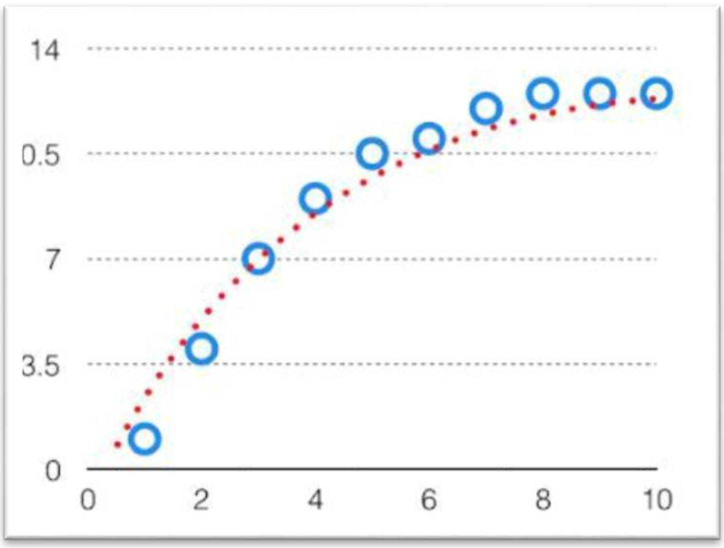
In such a problem, a hypothesis looks like follows.

$$h_\theta(x) = \theta_0 + \theta_1 x + \theta_2 x^2 + \theta_3 x^3 + \theta_4 x^4$$

Bias Variance Tradeoff

If the algorithm is too simple (hypothesis with linear eq.) then it may be on high bias and low variance condition and thus is error-prone. If algorithms fit too complex (hypothesis with high degree eq.) then it may be on high variance and low bias. In the latter condition, the new entries will not perform well. Well, there is something between both of these conditions, known as Trade-off or Bias Variance Trade-off.

This tradeoff in complexity is why there is a tradeoff between bias and variance. An algorithm can't be more complex and less complex at the same time. For the graph, the perfect tradeoff will be like.



We try to optimize the value of the total error for the model by using the Bias-Variance Tradeoff.

$$\text{Total Error} = \text{Bias}^2 + \text{Variance} + \text{Irreducible Error}$$

The best fit will be given by the hypothesis on the tradeoff point.
This is referred to as the best point chosen for the training of the algorithm which gives low error in training as well as testing data.