**ML Tearms**

**Scaler:** A object which contains only magnitude. We call it scalar value.

Ex: 4kgs 🡪 4 is magnitude here

**Vector**: An object which contains both magnitude and direction.

Ex: 2x+5y 🡪 2 is magnitude and in x-axis direction; 5 magnitude in y direction

With the help of vector differential operator, we will define some quantities.

1. Gradient
2. Divergent
3. Curl

Appling Gradient to a scalar to make it vector. Or

Converting a scalar into a vector we use gradient.

**Bias:** Bias refers to a systematic error or assumption made by a machine learning model. It represents the model's tendency to consistently predict values that are different from the true values. For example, if a model is biased towards predicting higher values for all inputs, it may consistently overestimate outcomes.

**Variance:** Variance refers to the variability or inconsistency of a machine learning model's predictions for different inputs. A model with high variance may give very different predictions for similar inputs, leading to less reliable results. It can be caused by models that are too complex and have learned specific details from the training data that may not generalize well to new data.

**Underfitting:** Underfitting occurs when a machine learning model is not able to capture the underlying patterns and relationships in the data. It happens when the model is too simple or lacks complexity, resulting in poor performance on both the training and testing data. An underfit model may produce overly generalized predictions that do not accurately represent the data.

**Overfitting:** Overfitting happens when a machine learning model becomes too complex and learns the training data too well, including noise and random fluctuations. As a result, the model may perform very well on the training data but fails to generalize to new, unseen data. Overfitting can lead to poor performance and unreliable predictions.

Normalization: Normalization is a technique used to scale and transform the features or inputs of a machine learning model. It ensures that all features have similar ranges and helps the model to learn more effectively. Normalization typically involves adjusting the values of the features to a specific range, such as between 0 and 1, or making their mean zero and standard deviation one.

Standardization: Standardization is another technique used to transform the features of a machine learning model. It involves subtracting the mean value of a feature and dividing it by the standard deviation. This process helps to center the data around zero and scale it, making it easier for the model to learn patterns.

Standard Deviation: Standard deviation is a measure of how spread out the values in a dataset are from the mean or average. It provides information about the variability or dispersion of the data points. A small standard deviation means the data points are close to the mean, while a large standard deviation indicates that the data points are more spread out.

Scalar: In machine learning, a scalar refers to a single value, such as a number, that represents a quantity. It can be a real number or an integer. Scalars are often used to represent features or labels in a dataset, and they can be manipulated and used in mathematical operations.

**Level 2:**

Data Preprocessing: Data preprocessing involves preparing and cleaning the data before feeding it into a machine learning model. It includes tasks like removing missing values, handling outliers, and transforming the data to make it suitable for the model to learn from.

Feature Selection: Feature selection is the process of choosing the most relevant and informative features from a dataset to train a machine learning model. It helps in improving the model's performance by reducing noise and eliminating redundant or irrelevant information.

Model Evaluation: Model evaluation is the process of assessing the performance of a machine learning model. It involves using various metrics and techniques to measure how well the model is able to make predictions. Common evaluation metrics include accuracy (how often the model predicts correctly) and mean squared error (a measure of how close the predictions are to the actual values).

Optimization Algorithms: Optimization algorithms are mathematical techniques used to find the best values for the parameters of a machine learning model. These algorithms help the model to learn and improve its predictions by minimizing errors or maximizing performance. Examples of optimization algorithms include gradient descent and stochastic gradient descent.

Linear Algebra: Linear algebra is a branch of mathematics that deals with vectors (directional quantities) and matrices (collections of numbers arranged in a grid). It is used extensively in machine learning to represent and manipulate data, as well as perform calculations and transformations on features and parameters.

Statistics: Statistics is the study of data collection, analysis, interpretation, presentation, and organization. In machine learning, understanding basic statistical concepts like mean, median, and mode helps in summarizing and interpreting data, identifying patterns, and making informed decisions about Modeling and predictions.

Interpretation and Visualization: Interpreting and visualizing data is the process of understanding and communicating patterns, trends, and relationships within the data. It involves using charts, graphs, and other visual representations to make the data more understandable and meaningful.

Critical Thinking and Asking Questions: In machine learning, critical thinking involves being curious, asking questions, and exploring different approaches. It means questioning assumptions, considering alternative solutions, and evaluating the strengths and weaknesses of different models and techniques.

**Level 3:**

Supervised Learning: Supervised learning is a type of machine learning where the model learns from labeled examples, meaning it is provided with input-output pairs. It learns to map inputs to outputs based on the provided labels and can make predictions for new, unseen inputs.

Unsupervised Learning: Unsupervised learning is another type of machine learning where the model learns from unlabeled data. The goal is to discover patterns, relationships, and structures within the data without explicit labels or predefined outputs.

Deep Learning: Deep learning is a subfield of machine learning that focuses on using artificial neural networks to learn and make predictions. Deep learning models are capable of automatically learning hierarchical representations of data and have achieved remarkable success in various domains, including image recognition and natural language processing.

Neural Networks: Neural networks are computational models inspired by the human brain. They consist of interconnected layers of artificial neurons (also called nodes) and are designed to learn from data. Neural networks are the building blocks of deep learning models.

Reinforcement Learning: Reinforcement learning is a type of machine learning where an agent learns to make decisions by interacting with an environment. The agent receives feedback in the form of rewards or penalties based on its actions and learns to maximize rewards over time through trial and error.

Natural Language Processing (NLP): NLP is a field of study that focuses on enabling computers to understand, interpret, and generate human language. It involves techniques like text processing, sentiment analysis, language translation, and chatbots.

Computer Vision: Computer vision is an area of study that deals with enabling machines to understand and interpret visual information from images or videos. It involves tasks like object detection, image classification, and image segmentation.

Transfer Learning: Transfer learning is a technique where a pre-trained model, trained on a large dataset, is used as a starting point for a new task or dataset. By leveraging the learned features, transfer learning allows models to achieve good performance with less training data.

**Deep Learning**

Weights: In a neural network, weights represent the strength of connections between neurons. They determine the influence of one neuron's output on the input of another neuron. Weights are adjusted during the training process to improve the accuracy of predictions.

Hidden Layers: Hidden layers are layers of neurons in a neural network that sit between the input layer and the output layer. They perform computations on the input data by applying weights and activation functions. Hidden layers help the network learn and extract relevant features from the data.

Activation Function: An activation function is a mathematical function applied to the output of a neuron. It determines whether the neuron should be activated or not based on the input it receives. Activation functions introduce non-linearity into the neural network, allowing it to learn complex relationships between inputs and outputs.

Backpropagation: Backpropagation is a technique used to train neural networks by adjusting the weights based on the prediction error. It involves calculating the gradient of the loss function with respect to the weights and propagating this information backward through the network to update the weights.

Convolutional Neural Networks (CNNs): Convolutional Neural Networks are a type of deep learning model specifically designed for analyzing visual data, such as images. CNNs use convolutional layers to scan and extract local patterns and features from the input data, enabling them to recognize objects and patterns in images.

Recurrent Neural Networks (RNNs): Recurrent Neural Networks are a type of deep learning model that can process sequential or time-series data. RNNs have connections between neurons that form a directed cycle, allowing them to capture and learn patterns from sequences of data, such as natural language or speech.

Generative Adversarial Networks (GANs): Generative Adversarial Networks are a type of deep learning model consisting of two components: a generator and a discriminator. GANs are used to generate new data that resembles the training data by having the generator and discriminator compete against each other in a game-like setting.

**Level 2:**

Dropout: Dropout is a regularization technique used in deep learning to prevent overfitting. During training, dropout randomly sets a portion of the neurons' outputs to zero, which helps in reducing the interdependencies between neurons and encourages the network to learn more robust and generalizable representations.

Loss Function: The loss function measures how well the predictions made by a deep learning model match the true values or labels in the training data. It quantifies the error between the predicted output and the actual output. The goal is to minimize the loss function during training to improve the model's accuracy.

Batch Size: In deep learning, the batch size refers to the number of training examples used in each iteration of the training process. Instead of updating the weights after every single example, training is often done in batches for efficiency. The batch size can influence the training speed and the quality of the learned representations.

Learning Rate: The learning rate is a hyperparameter that controls the step size at which the weights of a deep learning model are updated during training. It determines how quickly or slowly the model learns from the data. A high learning rate may result in unstable training, while a low learning rate may slow down the convergence.

Gradient Descent: Gradient descent is an optimization algorithm commonly used in deep learning to update the weights of the model during training. It works by calculating the gradient (derivative) of the loss function with respect to the weights and adjusting the weights in the direction that minimizes the loss.

Transfer Learning: Transfer learning, as mentioned earlier, is a technique where a pre-trained deep learning model is used as a starting point for a new task or dataset. By leveraging the knowledge and representations learned from a large dataset, transfer learning can significantly speed up training and improve performance on smaller datasets.

Hyperparameters: Hyperparameters are the settings or configuration choices made by the developer before training a deep learning model. They are not learned from the data but are set externally. Examples of hyperparameters include the learning rate, batch size, number of layers, and activation functions.

Model Architecture: The model architecture refers to the design and structure of a deep learning model. It specifies the number and arrangement of layers, the activation functions used, the connectivity between neurons, and any other architectural choices. The model architecture determines the model's capacity to learn and represent complex patterns in the data.

Epoch: An epoch represents a complete pass through the entire training dataset during the training process. In deep learning, training is typically done over multiple epochs, where each epoch consists of one or more iterations (batches) of updating the weights.

GPU (Graphics Processing Unit): A GPU is a specialized hardware device that is highly efficient in performing parallel computations. Deep learning models often require significant computational power, and using GPUs can greatly accelerate the training process, allowing for faster experimentation and model development.

**Easy: level:**

Multilayer Perceptron: A multilayer perceptron is like a machine with many layers. Each layer has small parts called neurons that work together to understand and learn from the data.

Multi-layer Inputs: Sometimes, we have different types of information or features for a problem. With multi-layer inputs, we can give the machine multiple sets of information to help it understand and make better predictions.

Normal Distribution: Normal distribution is like a shape or pattern that many things in nature follow. It looks like a bell and helps us understand how likely something is to happen. In deep learning, we sometimes use this pattern to start with random numbers for weights or biases.

Uniform Distribution: Uniform distribution means that all the numbers in a range are equally likely to be chosen. It helps us set starting values for weights or biases in a way that is fair and spreads them out evenly.

Initialization of Weights: Initialization of weights means setting the starting values for the connections between neurons in a deep learning model. We want to start with good values so that the model can learn well.

Sigmoid Activation Function: The sigmoid activation function helps us decide whether something is true or false, like a light switch that can be on (1) or off (0). It helps the machine make decisions based on the input it receives.

Softmax Activation Function: The softmax activation function is used to make choices when we have many options. It helps the machine decide which option is the most likely or most probable.

Tanh Activation Function: The tanh activation function is similar to the sigmoid function but gives results between -1 and 1. It helps the machine understand negative values and make decisions based on them.

Rectified Linear Unit (ReLU): The Rectified Linear Unit (ReLU) is an activation function that turns off negative numbers and keeps positive numbers as they are. It helps the machine understand and work with positive values effectively.

Rule of Derivation: The rule of derivation is a math rule that helps us find the slope or rate of change of something. In deep learning, we use this rule to adjust the weights and biases in the right direction to improve the model's predictions.

Forward Propagation: Forward propagation means passing the input data through the layers of a deep learning model to make predictions. It's like going forward through a machine to get an answer.

Backward Propagation (Backpropagation): Backward propagation is the process of going backward through the machine to adjust the weights and biases based on the difference between the predicted output and the correct output. This helps the machine learn from its mistakes and make better predictions.

Full Factorization Machine (FFM): A Full Factorization Machine is a special technique used in deep learning to handle large and sparse datasets efficiently. It helps the machine find useful patterns and relationships in the data.

Gradient Descent: Gradient descent is like a guide that helps the machine learn and make better predictions. It adjusts the weights and biases in small steps, following the slope of the hill, until it finds the best values for accurate predictions.

**Clear: level:**

Multilayer Perceptron: The multilayer perceptron (MLP) is a type of feedforward neural network consisting of multiple layers of artificial neurons (also called perceptron’s). It typically includes an input layer, one or more hidden layers, and an output layer. MLPs are used for various tasks, including classification and regression.

Multi-layer Inputs: In some cases, deep learning models can have multiple inputs, where each input represents a different type of data or feature. For example, a model for image classification might have one input for the image pixels and another input for additional metadata associated with the image.

Normal Distribution: Normal distribution, also known as the Gaussian distribution, is a probability distribution that is symmetric and bell-shaped. In deep learning, random variables, such as weights or biases, are often initialized following a normal distribution. It is a common assumption that many phenomena in nature and statistics can be modelled using a normal distribution.

Uniform Distribution: Uniform distribution is a probability distribution where all values in a given range have an equal likelihood of being sampled. It is often used for random initialization of weights or biases in deep learning models, ensuring that initial values are spread uniformly across a given range.

Initialization of Weights: Initialization of weights refers to the process of assigning initial values to the weights in a deep learning model. Proper weight initialization is important for effective training. Common initialization methods include random initialization, where weights are randomly assigned, and Xavier or He initialization, which take into account the size of the previous layer to set initial weights.

Sigmoid Activation Function: The sigmoid activation function is a mathematical function that maps the input to a value between 0 and 1. It is commonly used in deep learning models to introduce non-linearity and produce probability-like outputs.

Softmax Activation Function: The softmax activation function is used in the output layer of a deep learning model for multi-class classification tasks. It converts a vector of real numbers into a probability distribution, ensuring that the predicted class probabilities sum up to 1.

Tanh Activation Function: The tanh (hyperbolic tangent) activation function is similar to the sigmoid function but maps the input to a value between -1 and 1. It is useful in deep learning models for its ability to introduce non-linearity and capture negative values in the data.

Rectified Linear Unit (ReLU): The Rectified Linear Unit (ReLU) is an activation function that sets all negative values in the input to zero and leaves positive values unchanged. ReLU is widely used in deep learning due to its simplicity and computational efficiency.

Rule of Derivation: The rule of derivation, also known as the chain rule, is a fundamental principle in calculus used to compute the derivative of a composite function. In deep learning, the chain rule is used during backpropagation to calculate gradients and update the weights of the neural network.

Forward Propagation: Forward propagation is the process of moving the input data through the layers of a deep learning model to generate predictions. It involves computing the weighted sums of inputs, applying activation functions, and passing the outputs to the next layer.

Backward Propagation (Backpropagation): Backward propagation, or backpropagation, is the process of calculating gradients (derivatives) of the loss function with respect to the weights and biases in a deep learning model. It enables the model to learn and update its parameters by propagating the error from the output layer back to the input layer.

Full Factorization Machine (FFM): Full Factorization Machine is an extension of the Factorization Machine algorithm used in deep learning. It is designed to handle large and sparse datasets efficiently by utilizing feature interactions through factorization techniques.

Gradient Descent: Gradient descent is an optimization algorithm used in deep learning to update the weights of the model based on the gradients (derivatives) of the loss function. It iteratively adjusts the weights in the direction that minimizes the loss, allowing the model to converge to an optimal solution.

**Few More:**

Loss Function: A loss function measures how well a deep learning model is performing by quantifying the difference between predicted outputs and actual outputs. The goal is to minimize the loss function to improve the accuracy of the model's predictions.

Cost Function: The cost function is another term for the loss function. It represents the cost or error of the model's predictions.

Optimizer: An optimizer is an algorithm used to adjust the weights and biases of a deep learning model during training. It determines how the model learns from the data and updates its parameters to minimize the loss function.

Mini-Batch: In deep learning, data is often divided into smaller groups called mini-batches for training. Mini-batch training involves processing a small subset of the data at a time, making the training process more efficient and allowing the model to learn from different examples in each batch.

Learning Rate: The learning rate is a hyperparameter that determines the step size at which the optimizer adjusts the model's weights during training. It controls how quickly or slowly the model learns from the data.

Epoch: An epoch represents a complete pass through the entire training dataset during the training process. In deep learning, the model is trained iteratively over multiple epochs, with each epoch consisting of one or more mini-batches.

Validation Set: A validation set is a portion of the dataset that is held out during training to evaluate the model's performance. It is used to monitor the model's generalization ability and make decisions regarding hyperparameter tuning or early stopping.

Early Stopping: Early stopping is a technique used to prevent overfitting in deep learning models. It involves monitoring the model's performance on a validation set and stopping the training process early if the performance starts to degrade.

Hyperparameters: Hyperparameters are settings or configurations chosen by the developer before training a deep learning model. Examples include the learning rate, batch size, number of hidden layers, and number of neurons in each layer.

Model Evaluation: Model evaluation involves assessing the performance of a trained deep learning model on unseen data. It includes metrics such as accuracy, precision, recall, and F1-score, depending on the specific task and problem domain.