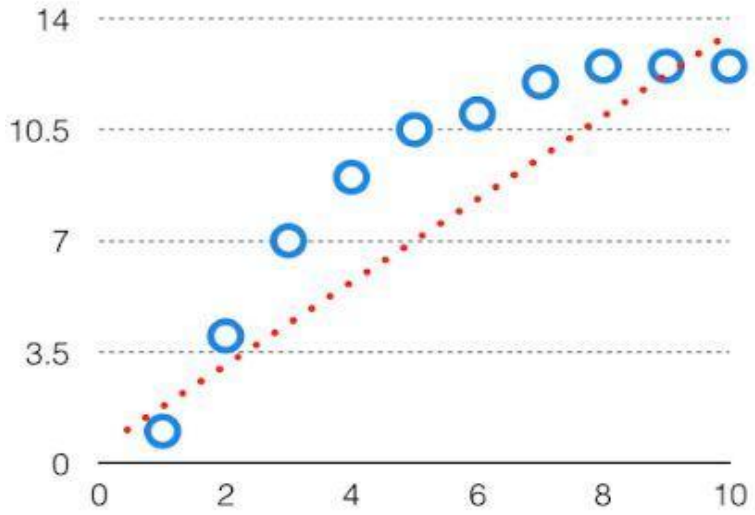
Machine Learning

1.What is the Bias-Variance Tradeoff?

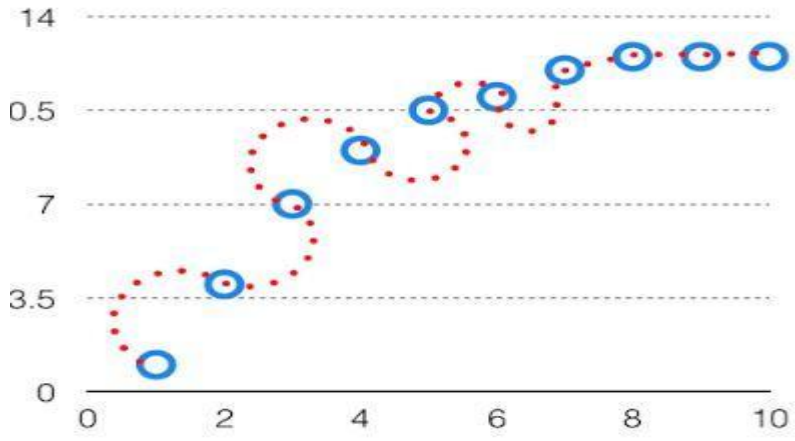
The bias is known as the difference between the prediction of the values by the Machine Learning model and the correct value.

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 the Underfitting of Data. This happens when the hypothesis is too simple or linear in nature.



The variability of model prediction for a given data point which tells us the spread of our data is called the variance of the model.

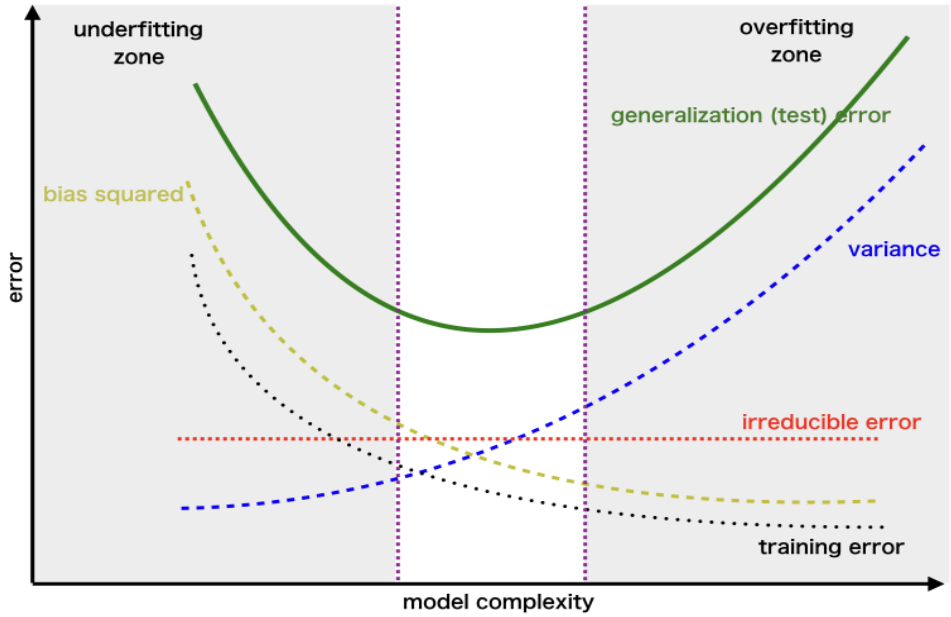
The models perform very well on training data but have high error rates on test data. When a model is high on variance, it is then said to as Overfitting of Data. Overfitting is fitting the training set accurately via complex curve and high order hypothesis but is not the solution as the error with unseen data is high. While training a data model variance should be kept low.



High Variance in the Model

Bias Variance Tradeoff

The bias-variance tradeoff is a fundamental machine learning concept, balancing two error types to minimize total prediction error: bias (simplifying assumptions, causing underfitting) and variance (sensitivity to data, causing overfitting). Increased model complexity decreases bias but increases variance, while simpler models increase bias and decrease variance.



2.What is the difference between a loss function and a cost function?

Loss' in Machine learning helps us understand the difference between the predicted value & the actual value.

The Function used to quantify this loss during the training phase in the form of a single real number is known as the "Loss Function".

These are used in those supervised learning algorithms that use optimization techniques. The terms cost function & loss function are analogous.

Loss function: Used when we refer to the error for a single training example.
Cost function: Used to refer to an average of the loss functions over an entire training data.

There are many cost functions in machine learning and each has its use cases depending on whether it is a regression problem or classification problem.

1. Regression cost Function
2. Binary Classification cost Functions
3. Multi-class Classification cost Functions

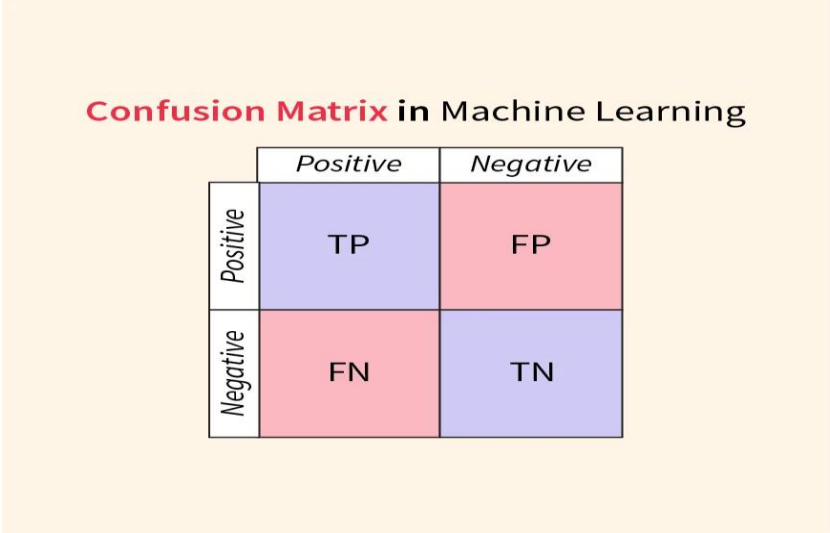
3. What is gradient descent?

Gradient Descent is an iterative optimization algorithm used to minimize a cost function by adjusting model parameters in the direction of the steepest descent of the function's gradient. In simple terms, it finds the optimal values of weights and biases by gradually reducing the error between predicted and actual outputs.

4. What is a confusion matrix?

A confusion matrix is a performance evaluation table for machine learning classification models, comparing predicted classes against actual ground truth. It displays counts of true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN), allowing for detailed analysis of model errors beyond simple accuracy.

- Correct Predictions: True Positives (TP) (correctly predicted positive) and True Negatives (TN)(correctly predicted negative).
- Incorrect Predictions: False Positives (FP) (Type I error: false alarm) and False Negatives (FN) (Type II error: missed detection).
- Metrics Derived: It is used to calculate vital performance metrics, including accuracy, precision, recall, and the F1 score.



5.What are L1 (Lasso) and L2 (Ridge) Regularization?

Ridge and Lasso Regression are two popular techniques in machine learning used for regularizing linear models to avoid overfitting and improve predictive performance. Both methods add a penalty term to the model's cost function to constrain the coefficients, but they differ in how they apply this penalty.

Ridge regression, also known as L2 regularization, is a technique used in linear regression to prevent overfitting by adding a penalty term to the loss function. This penalty is proportional to the square of the magnitude of the coefficients (weights).

Lasso regression, also known as L1 regularization, is a linear regression technique that adds a penalty to the loss function to prevent overfitting. This penalty is based on the absolute values of the coefficients.

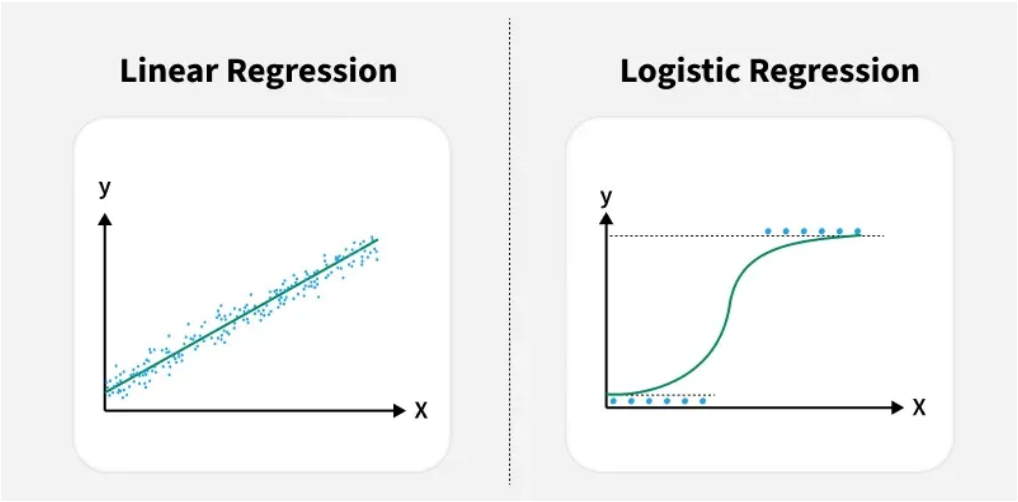
5. What is the "Curse of Dimensionality"?

Curse of Dimensionality in Machine Learning arises when working with high-dimensional data, leading to increased computational complexity, overfitting, and spurious correlations. Techniques like dimensionality reduction, feature selection, and careful model design are essential for mitigating its effects and improving algorithm performance. Navigating this challenge is crucial for unlocking the potential of high-dimensional datasets and ensuring robust machine-learning solutions.

6. How does Logistic Regression differ from Linear Regression?

Linear Regression and Logistic Regression are two widely used supervised machine learning algorithms. Although they sound similar, they are used for completely different purposes.

- Linear Regression is used for predicting continuous numerical values.
- Logistic Regression is used for predicting categorical outputs, mostly binary classification.



Linear Regression is used when the output is a continuous number. The model tries to find a straight line (or a plane in higher dimensions) that best fits the data. It predicts values by calculating a weighted sum of input features. It is mainly used to estimate quantities such as price, marks, salary or temperature.

Logistic Regression is used when the output is categorical, usually binary (0 or 1). Instead of predicting a straight-line value, it predicts the probability of an event happening using the sigmoid function which forms a S-shaped curve graph and it converts any number into a value between 0 and 1.

It is used for classification tasks such as spam detection or whether a student will pass or fail.

7. What is the difference between Parameters and Hyperparameters?

A model parameter is a variable of the selected model which can be estimated by fitting the given data to the model.
A model hyperparameter is the parameter whose value is set before the model start training. They cannot be learned by fitting the model to the data.

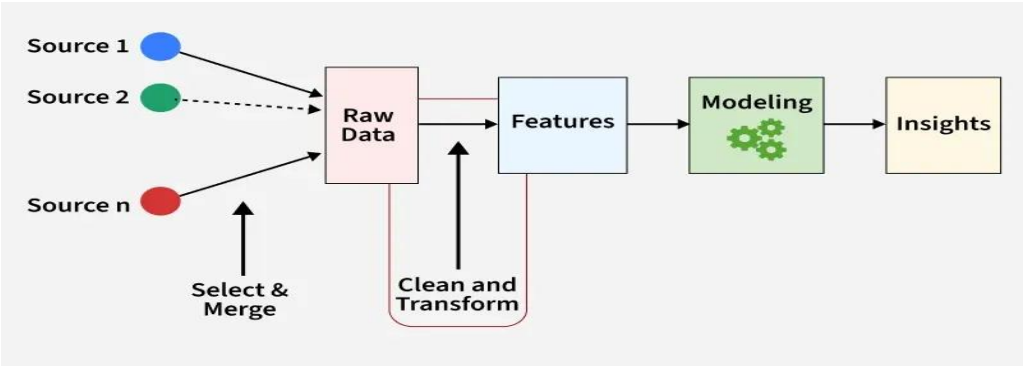
8. What is the difference between feature engineering and feature selection?

Feature Engineering is the process of selecting, creating or modifying features like input variables or data to help machine learning models learn patterns more effectively. It involves transforming raw data into meaningful inputs that improve model accuracy and performance.

This step may include handling missing values, encoding categories, scaling numbers, creating new features or combining existing ones. It helps turn messy real-world data into a form that models can understand and use for better predictions.

Importance of Feature Engineering

- Improve accuracy: Choosing the right features helps the model learn better, leading to more accurate predictions.
- Reduce overfitting: Using fewer, more important features helps the model avoid memorizing the data and perform better on new data.
- Boost interpretability: Well-chosen features make it easier to understand how the model makes its predictions.
- Enhance efficiency: Focusing on key features speeds up the model's training and prediction process, saving time and resources.



Feature selection is a important step in the machine learning pipeline. By identifying and retaining only the most relevant features, we can build models that generalize better, train faster, and are easier to interpret. Among the various approaches, filter methods are popular due to their simplicity, speed, and independence from specific machine learning models.

9. What are the key assumptions of linear regression?

Linear regression works reliably only when certain key assumptions about the data are met. These assumptions ensure that the model's estimates are accurate, unbiased, and suitable for prediction. Understanding and checking them is essential for building a valid regression model.

1. Linearity

The relationship between the independent and dependent variables is linear.

- The dependent variable should change proportionally with the independent variables, forming a straight-line trend.
- Curved or irregular patterns can cause underfitting and inaccurate predictions.
- When linearity fails, data transformations or non-linear models may be required.
- Linear Relationship**:** Increase in temperature results in a consistent increase in ice cream sales.
- Non-Linear Relationship**:** Increase in temperature leads to a more significant increase in ice cream sales at higher temperatures, indicating a non-linear relationship.

2. Homoscedasticity of Residuals

The variance of residuals remains constant across all levels of the independent variables.

- Residuals should appear evenly scattered, indicating uniform error spread.
- Patterns of increasing or decreasing variance lead to unreliable coefficient estimates.
- Severe heteroscedasticity may require transformations or weighted regression methods.

3. Multivariate Normality - Normal Distribution

The residuals follow a normal distribution when multiple predictors are involved.

- Normality supports valid confidence intervals, hypothesis tests and p-values.
- Skewed or peaked distributions weaken inference quality.
- Violations may be corrected through transformation or larger sample sizes.

4. Independence of Errors

Residuals must not correlate with each other across observations.

- Correlated errors suggest the model missed temporal or patterned structure.
- Autocorrelation can inflate significance and mislead conclusions.
- Time-series data often require specialized methods to resolve this.

5. Lack of Multicollinearity

The independent variables are not highly correlated with each other.

- Strong collinearity inflates coefficient variance and reduces interpretability.
- It becomes difficult to assess the true contribution of each predictor.
- Feature selection or regularization helps reduce the effect.