**What is difference between Euclidian distance and Manhattan distance. Explain in simple words.**

Manhattan Distance: It is also called the Taxicab distance or the City Block distance. It is the distance between two points measured along axes at right angles.

* ManhattanDistance = sum for i to N sum |x[i] – y[i]|

Euclidean Distance: It is the straight line distance between two data points in a plane. It is calculated using Minkowski Distance formula by setting ‘p’ value to 2, also known as the ‘L2’ norm distance metric.

* EuclideanDistance = sqrt(sum for i to N (x[i] – y[i])^2)

**What do you understand by feature selection, transformation, engineering and EDA & What are the steps**

**that you have performed in each of these in detail with example.**

Feature Selection: Selection of features with the highest influence on the target variable, from a set of existing features.

This can be done with various techniques: Linear Regression, Decision Trees, calculation of "importance" weights

Feature Transformation: Transformation of features in order to create new ones based on the old ones to improve the accuracy of the algorithm

A popular technique is Principal Component Analysis

Feature Engineering: Generation of features which is in a format that is difficult to analyse directly and are not directly comparable. Ex: images, time-series, etc.

**What is difference between single values decomposition (SVD) and PCA? (hint: SVD is one of the way to do**

**PCA)**

PCA: Principal Component Analysis (PCA) is a statistical procedure that uses an orthogonal transformation that converts a set of correlated variables to a set of uncorrelated variables. PCA is the most widely used tool in exploratory data analysis and in machine learning for predictive models.

SVD: The Singular-Value Decomposition, is a matrix decomposition method for reducing a matrix to its constituent parts in order to make certain subsequent matrix calculations simpler.

A = U . Sigma . V^T

Where A is the real m x n matrix that we wish to decompose, U is an m x m matrix, Sigma (often represented by the uppercase Greek letter Sigma) is an m x n diagonal matrix, and V^T is the transpose of an n x n matrix where T is a superscript.

**What kind of feature transformations have you done in your last project?**

Feature transformation is the process of modifying data but keeping the information.

Few thing we have done is:

Data Smoothing

Data Aggregation

Generalization

Normalization

**Have you taken any external features in any of the projects from any 3rd party data? If yes, explain that scenario.**

Yes, while predicting the covid cases from different states, we had to measure the oxygen production capacity of that state. If not then the nearest oxygen producer state. For this purpose we had used 3rd party data.

**If your model is overfitted, what will you do next?**

Overfitting of model means Low bias with High variance

There are numbers of techniques available to handle overfitting:

Cross-validation: This is done by splitting your dataset into ‘test’ data and ‘train’ data. Build the model using the ‘train’ set. The ‘test’ set is used for in-time validation.

Regularization: It regularizes or shrinks the coefficient estimates towards zero.

Early stopping: This prevents the model from memorizing the dataset.

Pruning: This technique applies to decision trees.

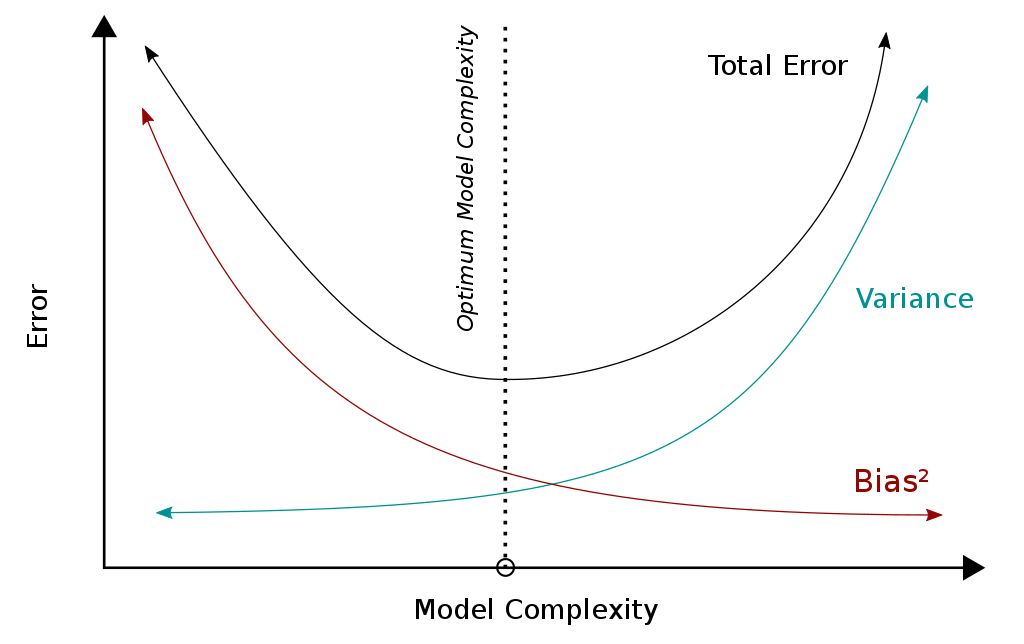
Pre-pruning: Stop ‘growing’ the tree earlier before it perfectly classifies the training set.

Post-pruning: Allows the tree to ‘grow’, perfectly classify the training set and then post prune the tree.

Dropout: This is a technique where randomly selected neurons are ignored during training.

**Explain me bias variance trade-off.**

If our model is too simple and has very few parameters then it may have high bias and low variance. On the other hand if our model has large number of parameters then it’s going to have high variance and low bias. So we need to find the good balance without overfitting and underfitting the data.



**What steps would you take to improve accuracy of your model? At-least mention 5 approach. And justify**

**why would you choose those approach**

Few steps to improve accuracy of a model:

Add more data: Adding more data will help to learn the model in a good way instead of relying on assumption and weak correlations.

Handle missing and outlier values: Presence of missing and outlier values often reduces the accuracy of a model or leads to a biased model, because we don’t analyse the relationship with other variable correctly. So it is important to handle missing and outlier value.

Feature Engineering: It will help to extract more information from existing data. It may have high ability to explain the variance in the training data.

Feature Selection: It will help to find the best attribute which better explains the relationship of independent variable with target variable.

Multiple algorithm: Some algorithm are better suited to a particular type of dataset than other. Hence we should apply all relevant model and check the performance.

Algorithm Tuning: The objective of parameter tuning is to find the optimum value for each parameter to improve the accuracy of the model.

**Explain process of feature engineering in context of text categorization.**

* Language detection: Understand which natural language data is in.
* Text preprocessing: Preparing raw data to make it suitable for machine learning model. Ex: text cleaning, stopword removal, stemming and lemmatization
* Length analysis: It’s important to have a look at the length of the text because it’s an easy calculation that can give a lot of insights.
* Sentiment analysis: determine whether a text is positive or negative.
* Named-Entity recognition: It is a process to tag text with pre-defined categories such as person names, organizations, locations.
* Word frequency: find the importance of single word by computing the n-gram frequency.
* Word vectors: transform a word into numbers.
* Topic modeling: extract the main topics from corpus.

**Explain vectorization and hamming distance.**

Vectorization: It is a technique by which we can make our code execute very fast. It takes multiple iterative operations among data points and turn them into matrix operation. Matrix operations are fast, they can be parallelize to some extent

Hamming distance: Hamming distance is a metric for comparing two binary data strings. While comparing two binary strings of equal length, Hamming distance is the number of bit positions in which the two bits are different.

The Hamming distance between two strings, a and b is denoted as d(a,b).

**Can you please explain chain rule and its use?**

Suppose cost is calculated as follows, the input is *x* and the target value is *y*,

f’= f(x)

g’=g(x)

y’= k(g’)

cost= criterion(y,y’)

If you want to calculate *d(cost) / d(x), x* can be a number, a vector, or a matrix. You can calculate ***d(f’) / d(x)*** *⨉* ***d(g’) / d(f’)*** *⨉* ***d(cost) / y’*** to get *d(cost) / d(x).* In machine learning, the three functions here, *f, g, k* represent different mappings, and the criterion is also understood as a mapping, except that the input here adds the target value y. The x here represents the input data, but the meaning of the input value is not significant, because we can’t change the data to make our target cost smaller. The actual situation is to change the variable contained in each map. The variables are derived.

For example, if you use *w𝚏* to represent the variable in function *f*, you can now calculate the derivative of cost to *w𝚏*. You can calculate it as follows. Before displaying the calculation method, rewrite the previous expression here. Include *w𝚏,*

f’= f(x, *w𝚏* )

g’= g(f’)

y’= k(g’)

cost= criterion(y,y’)

d(cost)/d(*w𝚏* )= d{f’)/d(*w𝚏* ) \* d(g’)/d(g’) \* d(y’)/d(g’) \* d(cost)/y’

This is from the chain rule of calculus.

**What is difference between correlation and covariance?**

| **Covariance** | **Correlation** |
| --- | --- |
| Covariance is a measure to indicate the extent to which two random variables change in tandem. | Correlation is a measure used to represent how strongly two random variables are related to each other. |
| Covariance is nothing but a measure of correlation. | Correlation refers to the scaled form of covariance. |
| Covariance indicates the direction of the linear relationship between variables. | Correlation on the other hand measures both the strength and direction of the linear relationship between two variables. |
| Covariance can vary between -∞ and +∞ | Correlation ranges between -1 and +1 |
| Covariance is affected by the change in scale. | Correlation is not influenced by the change in scale. |
| Covariance assumes the units from the product of the units of the two variables. | Correlation is dimensionless, i.e. It’s a unit-free measure of the relationship between variables. |
| Covariance of two dependent variables measures how much in real quantity (i.e. cm, kg, liters) on average they co-vary. | Correlation of two dependent variables measures the proportion of how much on average these variables vary w.r.t one another. |
| Covariance is zero in case of independent variables | Independent movements do not contribute to the total correlation. |

**What are the sampling techniques you have used in your project?**

Probability Sampling:

Simple random sampling

Stratified sampling

Systematic sampling

Cluster sampling

Multi stage sampling

Non-Probability Sampling:

Convenience Sampling

Purposive Sampling

Quota sampling

Referral/Snowball sampling

**Have you ever used Hypothesis testing in your last project, if yes, explain How?**

Hypothesis testing is a statistical method that is used in making statistical decision using experimental data. It is basically an assumption that we make about population parameter.

Ex: In a heart disease prediction project we took the hypothesis that Depression increases the risk for coronary heart disease in established diabetes.

**In which case you will use naïve Bayes classifier and decision tree separately?**

**What is the adv & disadvantage of naïve Bayes classifier, explain**

**In case of numerical data what is naïve Bayes classification equation you will use?**

Naive Bayes is used a lot in robotics and computer vision, and does quite well with those tasks. Decision trees perform very poorly in those situations.

Decision trees are neat because they tell you what inputs are the best predicators of the outputs so often decision trees can guide you to find if there is a statistical relationship between a given input to the output and how strong that relationship is.

### **Advantages**

* This algorithm works quickly and can save a lot of time.
* Naive Bayes is suitable for solving multi-class prediction problems.
* If its assumption of the independence of features holds true, it can perform better than other models and requires much less training data.
* Naive Bayes is better suited for categorical input variables than numerical variables.

### **Disadvantages**

* Naive Bayes assumes that all predictors (or features) are independent, rarely happening in real life. This limits the applicability of this algorithm in real-world use cases.
* This algorithm faces the ‘zero-frequency problem’ where it assigns zero probability to a categorical variable whose category in the test data set wasn’t available in the training dataset. It would be best if you used a smoothing technique to overcome this issue.
* Its estimations can be wrong in some cases, so you shouldn’t take its probability outputs very seriously.

**Give me scenario where I will be able to use a boosting classifier and regressor?**

Boosting can be used for regression as well as for classification problems. Being mainly focused at reducing bias, the base models that are often considered for boosting are models with low variance but high bias.

Regression analysis is used when you want to predict a continuous dependent variable from a number of independent variables.

**In case of Bayesian classifier what exactly it tries to learn. Define its learning procedure.**

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 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:

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

Bayes theorem provides a way of calculating posterior probability P(c|x) from P(c), P(x) and P(x|c).

P(c|x)= P(x|c)P(c)/P(x)

Above,

* *P*(*c|x*) is the posterior probability of *class* (c, *target*) given *predictor* (x, *attributes*).
* *P*(*c*) is the prior probability of *class*.
* *P*(*x|c*) is the likelihood which is the probability of the predictor given *class*.
* *P*(*x*) is the prior probability of the predictor.

**Give me a situation where I will be able to use SVM instead of Logistic regression.**

If the use case is we have to find if the student has passed or not,

or in a supermarket if a customer will purchase a product or not,

in such cases we can use Logistic regression over SVM.

**What do you understand by rbf kernel in SVM?**

RBF kernels are the most generalized form of kernelization and is one of the most widely used kernels due to its similarity to the Gaussian distribution. The RBF kernel function for two points X₁ and X₂ computes the similarity or how close they are to each other. This kernel can be mathematically represented as follows:



Where,

‘σ’ is the variance and our hyperparameter

||*X₁ - X₂||* is the Euclidean (L*₂*-norm) Distance between two points X₁ and X₂

**Give me 2 scenarios where AI can be used to increase revenue of travel industry.**

1. Smarter traffic light algorithms & real-time tracking can control higher and lower traffic patterns effectively, This can be applied to public transport for optimal scheduling & routing.
2. Autonomous Rail Rapid Transit is a train system that runs without rails, operating instead on a virtual painted track which the train’s computer system detects and follows

**What do you understand by leaf node in decision tree?**

A decision tree is a structure that includes a root node, branches, and leaf nodes. Each internal node denotes a test on an attribute, each branch denotes the outcome of a test, and each leaf node holds a class label.

**What is information gain & Entropy in decision tree?**

Entropy is the measures of impurity, disorder or uncertainty in a bunch of examples.

Entropy= -∑p(X)logp(x)

Information gain (IG) measures how much “information” a feature gives us about the class.

IG= entropy(parent) - [weighted average] \* entropy(children)

**Give disadvantages of using Decision tree**

1. A small change in the data can cause a large change in the structure of the decision tree causing instability.
2. For a Decision tree sometimes calculation can go far more complex compared to other algorithms.
3. Decision tree often involves higher time to train the model.
4. Decision tree training is relatively expensive as the complexity and time has taken are more.
5. The Decision Tree algorithm is inadequate for applying regression and predicting continuous values.

**List some of the features of random forest.**

* It reduces overfitting in decision trees and helps to improve the accuracy.
* It is flexible to both classification and regression problems.
* It works well with both categorical and continuous values.
* It automates missing values present in the data.

**How can you avoid overfitting in decision tree?**

Pruning: Pruning refers to a technique to remove the parts of the decision tree to prevent growing to its full depth. By tuning the hyperparameters of the decision tree model one can prune the trees and prevent them from overfitting.

Pre-Pruning: The pre-pruning technique refers to the early stopping of the growth of the decision tree.

Post-Pruning: The Post-pruning technique allows the decision tree model to grow to its full depth, then removes the tree branches to prevent the model from overfitting.

**Explain polynomial regression in your own way.**

Polynomial Regression is a regression algorithm that models the relationship between a dependent(y) and independent variable(x) as nth degree polynomial.

y= b0+b1x1+ b2x12+ b2x13+...... bnx1n

It is also called the special case of Multiple Linear Regression in ML. Because we add some polynomial terms to the Multiple Linear regression equation to convert it into Polynomial Regression.

**Explain learning mechanism of linear regression.**

* Regression is a supervised machine learning technique which is used to predict continuous values.
* The ultimate goal of the regression algorithm is to plot a best-fit line or a curve between the data.
* The three main metrics that are used for evaluating the trained regression model are variance, bias and error. If the variance is high, it leads to overfitting and when the bias is high, it leads to underfitting.
* Based on the number of input features and output labels, regression is classified as linear (one input and one output), multiple (many inputs and one output) and multivariate (many outputs).
* Linear regression allows us to plot a linear equation, i.e., a straight line. We need to tune the coefficient and bias of the linear equation over the training data for accurate predictions.
* The tuning of coefficient and bias is achieved through gradient descent or a cost function — least squares method.

**What is the cost function in logistic regression?**

The cost function in Logistic Regression is Log Loss:

Log Loss is the most important classification metric based on probabilities. It’s hard to interpret raw log-loss values, but log-loss is still a good metric for comparing models. For any given problem, a lower log loss value means better predictions.

Log Loss = ∑ ( x , y ) ∈ D − y log ⁡ ( y ′ ) − ( 1 − y ) log ⁡(1-y’)

**What is the error function in linear regression?**

Mean squared error (MSE) is the most commonly used loss function for regression. The loss is the mean overseen data of the squared differences between true and predicted values

MSE is calculated by: measuring the **distance of the** observed y-values from the predicted y-values at each value of x; squaring each of these distances; calculating the mean of each of the squared distances.

**What is the use of implementing OLS technique wrt dataset?**

In statistics, **ordinary least squares** (**OLS**) is a type of linear least squares method for estimating the unknown parameters in a linear regression model. OLS chooses the parameters of a linear function of a set of explanatory variables by the principle of least squares: minimizing the sum of the squares of the differences between the observed dependent variable in the given dataset

**Explain dendrogram in your own way.**

A dendrogram is a diagram that shows the hierarchical relationship between objects. It is most commonly created as an output from hierarchical clustering. The main use of a dendrogram is to work out the best way to allocate objects to clusters.