Artificial Intelligence

Interview Questions & Answers

Q1. You are given a train data set having 1000 columns and 1 million rows. The data set is based on a classification problem. Your manager has asked you to reduce the dimension of this data so that model computation time can be reduced. Your machine has memory constraints. What would you do? (You are free to make practical assumptions.)

Answer: Processing a high dimensional data on a limited memory machine is a strenuous task, your interviewer would be fully aware of that. Following are the methods you can use to tackle such situation:

1. Since we have lower RAM, we should close all other applications in our machine, including the web browser, so that most of the memory can be put to use.
2. We can randomly sample the data set. This means, we can create a smaller data set, let's say, having 1000 variables and 300000 rows and do the computations.
3. To reduce dimensionality, we can separate the numerical and categorical variables and remove the correlated variables. For numerical variables, we'll use correlation. For categorical variables, we'll use chi-square test.
4. Also, we can use PCA and pick the components which can explain the maximum variance in the data set.
5. Building a linear model using Stochastic Gradient Descent is also helpful.
6. We can also apply our business understanding to estimate which all predictors can impact the response variable. But, this is an intuitive approach, failing to identify useful predictors might result in significant loss of information.

Q2. You are given a data set. The data set has missing values which spread along 1 standard deviation from the median. What percentage of data would remain unaffected? Why?

Answer: This question has enough hints for you to start thinking! Since, the data is spread across median, let's assume it's a normal distribution. We know, in a normal distribution, ~68% of the data lies in 1 standard deviation from mean (or mode, median), which leaves ~32% of the data unaffected. Therefore, ~32% of the data would remain unaffected by missing values.

Q3. You are given a data set on cancer detection. You've build a classification model and achieved an accuracy of 96%. Why shouldn't you be happy with your model performance? What can you do about it?

Answer: If you have worked on enough data sets, you should deduce that cancer detection results in imbalanced data. In an imbalanced data set, accuracy should not be used as a

measure of performance because 96% (as given) might only be predicting majority class correctly, but our class of interest is minority class (4%) which is the people who actually got diagnosed with cancer. Hence, in order to evaluate model performance, we should use Sensitivity (True Positive Rate), Specificity (True Negative Rate), F measure to determine class wise performance of the classifier. If the minority class performance is found to to be poor, we can undertake the following steps:

1. We can use undersampling, oversampling or SMOTE to make the data balanced.
2. We can alter the prediction threshold value by doing probability caliberation and finding a optimal threshold using AUC-ROC curve.
3. We can assign weight to classes such that the minority classes gets larger weight.
4. We can also use anomaly detection.

Q4. You came to know that your model is suffering from low bias and high variance. Which algorithm should you use to tackle it? Why?

Answer: Low bias occurs when the model's predicted values are near to actual values. In other words, the model becomes flexible enough to mimic the training data distribution. While it sounds like great achievement, but not to forget, a flexible model has no generalization capabilities. It means, when this model is tested on an unseen data, it gives disappointing results.

In such situations, we can use bagging algorithm (like random forest) to tackle high variance problem. Bagging algorithms divides a data set into subsets made with repeated randomized sampling. Then, these samples are used to generate a set of models using a single learning algorithm. Later, the model predictions are combined using voting (classification) or averaging (regression).

Also, to combat high variance, we can:

1. Use regularization technique, where higher model coefficients get penalized, hence lowering model complexity.
2. Use top n features from variable importance chart. May be, with all the variable in the data set, the algorithm is having difficulty in finding the meaningful signal.

Q5. How is kNN different from kmeans clustering?

Answer: Don't get mislead by 'k' in their names. You should know that the fundamental difference between both these algorithms is, kmeans is unsupervised in nature and kNN is supervised in nature. kmeans is a clustering algorithm. kNN is a classification (or regression) algorithm.

kmeans algorithm partitions a data set into clusters such that a cluster formed is homogeneous and the points in each cluster are close to each other. The algorithm tries to maintain enough separability between these clusters. Due to unsupervised nature, the clusters have no labels.

kNN algorithm tries to classify an unlabeled observation based on its k (can be any number) surrounding neighbors. It is also known as lazy learner because it involves minimal training of

model. Hence, it doesn't use training data to make generalization on unseen data set.

Q6. What is the difference between covariance and correlation?

Answer: Correlation is the standardized form of covariance.

Covariances are difficult to compare. For example: if we calculate the covariances of salary ($) and age (years), we'll get different covariances which can't be compared because of having unequal scales. To combat such situation, we calculate correlation to get a value between -1 and 1, irrespective of their respective scale.

Q7. Both being tree based algorithm, how is random forest different from Gradient boosting algorithm (GBM)?

Answer: The fundamental difference is, random forest uses bagging technique to make predictions. GBM uses boosting techniques to make predictions.

In bagging technique, a data set is divided into n samples using randomized sampling. Then, using a single learning algorithm a model is build on all samples. Later, the resultant predictions are combined using voting or averaging. Bagging is done is parallel. In boosting, after the first round of predictions, the algorithm weighs misclassified predictions higher, such that they can be corrected in the succeeding round. This sequential process of giving higher weights to misclassified predictions continue until a stopping criterion is reached.

Random forest improves model accuracy by reducing variance (mainly). The trees grown are uncorrelated to maximize the decrease in variance. On the other hand, GBM improves accuracy my reducing both bias and variance in a model.

Q8. 'People who bought this, also bought…' recommendations seen on amazon is a result of which algorithm?

Answer: The basic idea for this kind of recommendation engine comes from collaborative filtering.

Collaborative Filtering algorithm considers "User Behavior" for recommending items. They exploit behavior of other users and items in terms of transaction history, ratings, selection and purchase information. Other users behaviour and preferences over the items are used to recommend items to the new users. In this case, features of the items are not known.

Q9. What do you understand by Type I vs Type II error ?

Answer: Type I error is committed when the null hypothesis is true and we reject it, also known as a 'False Positive'. Type II error is committed when the null hypothesis is false and we accept it, also known as 'False Negative'.

In the context of confusion matrix, we can say Type I error occurs when we classify a value as positive (1) when it is actually negative (0). Type II error occurs when we classify a value

as negative (0) when it is actually positive(1).

Q10. In k-means or kNN, we use euclidean distance to calculate the distance between nearest neighbors. Why not manhattan distance ?

Answer: We don't use manhattan distance because it calculates distance horizontally or vertically only. It has dimension restrictions. On the other hand, euclidean metric can be used in any space to calculate distance. Since, the data points can be present in any dimension, euclidean distance is a more viable option.

Example: Think of a chess board, the movement made by a bishop or a rook is calculated by manhattan distance because of their respective vertical & horizontal movements.

Q11. Considering the long list of machine learning algorithm, given a data set, how do you decide which one to use?

Answer: You should say, the choice of machine learning algorithm solely depends of the type of data. If you are given a data set which is exhibits linearity, then linear regression would be the best algorithm to use. If you given to work on images, audios, then neural network would help you to build a robust model.

If the data comprises of non linear interactions, then a boosting or bagging algorithm should be the choice. If the business requirement is to build a model which can be deployed, then we'll use regression or a decision tree model (easy to interpret and explain) instead of black box algorithms like SVM, GBM etc.

In short, there is no one master algorithm for all situations. We must be scrupulous enough to understand which algorithm to use.

Q12. List the supervised and unsupervised tasks in Deep Learning.

Answer: Now, this can be one tricky question. There might be a misconception that deep learning can only solve unsupervised learning problems. This is not the case. Some example of Supervised Learning and Deep learning include:

- Image classification
- Text classification
- Sequence tagging

On the other hand, there are some unsupervised deep learning techniques as well:

- Word embeddings (like Skip-gram and Continuous Bag of Words): Understanding Word Embeddings: From Word2Vec to Count Vectors
- Autoencoders: Learn How to Enhance a Blurred Image using an Autoencoder!

Q13. What's the difference between valid and same padding in a CNN?

Answer: This question has more chances of being a follow-up question to the previous one. Or if you have explained how you used CNNs in a computer vision task, the interviewer might ask this question along with the details of the padding parameters.

- Valid Padding: When we do not use any padding. The resultant matrix after convolution will have dimensions (n – f + 1) X (n – f + 1)
- Same padding: Adding padded elements all around the edges such that the output matrix will have the same dimensions as that of the input matrix.

Q14. How does LSTM solve the vanishing gradient challenge?

Answer: The LSTM model is considered a special case of RNNs. The problems of vanishing gradients and exploding gradients we saw earlier are a disadvantage while using the plain RNN model.

In LSTMs, we add a forget gate, which is basically a memory unit that retains information that is retained across timesteps and discards the other information that is not needed. This also necessitates the need for input and output gates to include the results of the forget gate as well.

Q15. How is the transformer architecture better than RNN?

Answer: Advancements in deep learning have made it possible to solve many tasks in Natural Language Processing. Networks/Sequence models like RNNs, LSTMs, etc. are specifically used for this purpose – so as to capture all possible information from a given sentence, or a paragraph.
However, sequential processing comes with its caveats:

- It requires high processing power
- It is difficult to execute in parallel because of its sequential nature

This gave rise to the Transformer architecture. Transformers use what is called the attention mechanism. This basically means mapping dependencies between all parts of a sentence.