1. What are the three stages to build the hypotheses or model in machine learning?

The three stages to build a model are:

1. Data Selection – Choosing how much and what features of the data to train the model on.
2. Data Preprocessing – This includes converting data to a format that is standard, cleaning data with outliers and missing values and sampling the data instead of including all the available data to build the model.
3. Data Transformation – Applying scaling, attribution decompositions and attribute aggregations – limiting the range of values, combining features where applicable and splitting them as applicable
4. What is the standard approach to supervised learning?

Supervised learning is used when it is known in advance what the desired output is. The learning algorithm receives inputs along with the corresponding correct outputs and the algorithm learns by comparing its prediction with the correct output and making adjustments to the model parameters. It is used when past history is used to predict likely future events. It might be used in classification – where the task is to classify inputs into categories based on features, regression – where the task is to learn the underlying relationship between the input and target variables and anomaly detection – where the task is to learn how usual behavior is, in order to detect differences from the usual and expected.

1. What is Training set and Test set?

Training set is the set of data used to train the model on – the data using which the model updates its parameters to better understand the underlying relationship or nature that makes one data point differ from the other in case of classification, or how the data points reinforce the relationship between input and target in regression.

The test set is used to check whether the characteristics of the data that the model has captured reflect the reality. If the model has learned the important features and their relationship to the output, then the performance of the model on the test set will be good.

1. What is the general principle of an ensemble method and what is bagging and boosting in ensemble method?

Ensemble methods are methods that combine multiple machine learning techniques to give one predictive model. They may be sequential – which make use of the similarity in the underlying techniques or parallel – which make use of the different models learned by the different techniques to produce the final model.

Bagging: Stands for bootstrap aggregation. It attempts to reduce the variance in an estimate by averaging multiple estimates. For example, M different trees are trained on M different subsets of data and aggregated using voting for classification and averaging for regression. Random forests are an example of this method.

Boosting: Family of algorithms that are capable of converting weak learners to strong learners. The principle is to fit a sequence of weak learners such as small decision trees on weighted versions of the data. Higher weights are given to the data points that were previously misclassified. The predictions are combined using weighted vote for classification , or weighted sum for regression. The sequential method of building the model is a feature of this technique. Gradient boosting and adaboost are examples of bagging.

1. How can you avoid overfitting?

Overfitting occurs when the model learns the individual data points in the training set, rather than the relationship between the input and output parameters. Some ways to avoid overfitting are:

1. Cross-validation: Dividing the training data into k folds and training using only k-1 of the folds while using the kth fold for improving the generalization of the relationship that the model learns.
2. Use more data: Use data with data points representative of the nature of the data and the possible variations it can have. With more data, the model can learn better.
3. Remove features: Use only features that are pertinent to reveal the underlying nature of the relationship between input and output variable. Remove features that add no value to modeling the relationship or pattern.
4. Stop early: Instead of training on all of the training data, stop training when the performance of the model does not improve across training iterations.
5. Regularization: Add penalty terms to your model that penalize the presence of more features, thereby preferring simpler models with fewer variables.
6. Ensembling: Combine multiple learning techniques to come up with a model that captures different aspects of the underlying data.