MACHINE LEARNING

Q1 to Q15 are subjective answer type questions, Answer them briefly.

1. **R-squared or Residual Sum of Squares (RSS) which one of these two is a better measure ofgoodness of fit model in regression and why?**

Ans: -

R-squared evaluates the scatter of the data points around the fitted regression line. It is also called the coefficient of determination, or the coefficient of multiple determination for multiple regression. For the same data set, higher R-squared values represent smaller differences between the observed data and the fitted values.

R-squared is the percentage of the dependent variable variation that a linear model explains.

$$R^2 = \frac{\text{Variance explained by the model}}{\text{Total variance}}$$

R-squared is always between 0 and 100%:

- 0% represents a model that does not explain any of the variation in the response variable around its mean. The mean of the dependent variable predicts the dependent variable as well as the regression model.
- 100% represents a model that explains all the variation in the response variable around its mean.

Usually, the larger the R^2, the better the regression model fits your observations. However, this guideline has important caveats that I'll discuss in both this post and the next post.

2. **What are TSS (Total Sum of Squares), ESS (Explained Sum of Squares) and RSS (Residual Sumof Squares) in regression. Also mention the equation relating these three metrics with each other.**

Ans: -

1. Total sum of squares

The total sum of squares is a variation of the values of a dependent variable from the sample means of the dependent variable. Essentially, the total sum of squares quantifies the total variation in a sample.

It can be determined using the following formula:

$$\text{TSS} = \sum_{i=1}^{n} (y_i - \bar{y})^2$$

Where:

- y_i – the value in a sample
- $\bar{y}$ – the mean value of a sample

2. Explained Sum of Squares

The explained sum of squares (ESS) is the sum of the squares of the deviations of the predicted values from the mean value of a response variable, in a standard regression model.

for example,

$$y_i = a + b_1x_{1i} + b_2x_{2i} + \ldots + \varepsilon_i$$

$$\mathrm{ESS} = \sum_{i=1}^{n} (\hat{y}_i - \bar{y})^2.$$

where:

- y_i is the i^{th} observation of the response variable
- x_{ji} is the i^{th} observation of the j^{th} explanatory variable
- a and b_j are coefficients
- i indexes the observations from 1 to n
- ε_i is the i^{th} value of the error term.

3. Residual sum of squares

The residual sum of squares (RSS) is a statistical technique used to measure the amount of variance in a data set that is not explained by a regression model itself. Instead, it estimates the variance in the residuals, or error term.

Linear regression is a measurement that helps determine the strength of the relationship between a dependent variable and one or more other factors, known as independent or explanatory variables.

It can be determined using the following formula:

$$RSS = \sum^{n}_{i=1} (y^i - f(x_i))^2$$

Where:

- y_i = the i^{th} value of the variable to be predicted
- $f(x_i)$ = predicted value of y_i
- n = upper limit of summation

3. What is the need of regularization in machine learning?

Ans: -

- Regularization is one of the most important concepts of machine learning. It is a technique to prevent the model from overfitting by adding extra information to it.
- Sometimes the machine learning model performs well with the training data but does not perform well with the test data. It means the model is not able to predict the output when deals with unseen data by introducing noise in the output, and hence the model is called overfitted. This problem can be deal with the help of a regularization technique.
- This technique can be used in such a way that it will allow to maintain all variables or features in the model by reducing the magnitude of the variables. Hence, it maintains accuracy as well as a generalization of the model.
- It mainly regularizes or reduces the coefficient of features toward zero.
- There are mainly two types of regularization techniques, which are given below:
    i. Ridge Regression

ii. Lasso Regression

4. **What is Gini–impurity index?**

Ans: -

Gini Impurity is a measurement used to build Decision Trees to determine how the features of a dataset should split nodes to form the tree. More precisely, the Gini Impurity of a dataset is a number between 0-0.5, which indicates the likelihood of new, random data being misclassified if it were given a random class label according to the class distribution in the dataset.

For example

say you want to build a classifier that determines if someone will default on their credit card. You have some labeled data with features, such as bins for age, income, credit rating, and whether or not each person is a student. To find the best feature for the first split of the tree – the root node – you could calculate how poorly each feature divided the data into the correct class, default ("yes") or didn't default ("no"). This calculation would measure the **impurity** of the split, and the feature with the lowest impurity would determine the best feature for splitting the current node. This process would continue for each subsequent node using the remaining features.

5. **Are unregularized decision-trees prone to overfitting? If yes, why?**

Ans: -

Overfitting refers to the condition when the model completely fits the training data but fails to generalize the testing unseen data. Overfit condition arises when the model memorizes the noise of the training data and fails to capture important patterns. A perfectly fit decision tree performs well for training data but performs poorly for unseen test data.

If the decision tree is allowed to train to its full strength, the model will overfit the training data. There are various techniques to prevent the decision tree model from overfitting.

6. **What is an ensemble technique in machine learning?**

Ans: -

Ensemble methods are techniques that create multiple models and then combine them to produce improved results. Ensemble methods usually produces more accurate solutions than a single model would. This has been the case in a number of machine learning competitions, where the winning solutions used ensemble methods.

7. **What is the difference between Bagging and Boosting techniques?**

Ans: -

S.NO	Bagging	Boosting
1.	The simplest way of combining predictions that belong to the same type.	A way of combining predictions that belong to the different types.
2.	Aim to decrease variance, not bias.	Aim to decrease bias, not variance.
3.	Each model receives equal weight.	Models are weighted according to their performance.

4.	Each model is built independently.	New models are influenced by the performance of previously built models.
5.	Different training data subsets are selected using row sampling with replacement and random sampling methods from the entire training dataset.	Every new subset contains the elements that were misclassified by previous models.
6.	Bagging tries to solve the over-fitting problem.	Boosting tries to reduce bias.
7.	If the classifier is unstable (high variance), then apply bagging.	If the classifier is stable and simple (high bias) the apply boosting.
8.	In this base classifier are trained parallelly.	In these base classifiers are trained sequentially.
9	Example: The Random Forest model uses Bagging.	Example: The AdaBoost uses Boosting techniques

8. **What is out-of-bag error in random forests?**

Ans: -

Multiple trees are built on the bootstrap samples, and the resulting predictions are averaged. This ensemble method, known as a random forest, often outperforms using a single tree. During the bootstrap process, random resamples of variables and records are often taken. The prediction error on each of the bootstrap samples is known as the out of bag score. It is used to fine-tune the model's parameters. With classification and regression trees.

9. **What is K-fold cross-validation?**

Ans: -

K-Fold CV is where a given data set is split into a ***K*** number of sections/folds where each fold is used as a testing set at some point. Let's take the scenario of 5-Fold cross validation(K=5). Here, the data set is split into 5 folds. In the first iteration, the first fold is used to test the model and the rest are used to train the model. In the second iteration, 2nd fold is used as the testing set while the rest serve as the training set. This process is repeated until each fold of the 5 folds have been used as the testing set.

10. **What is hyper parameter tuning in machine learning and why it is done?**

Ans: -

Machine Learning/Deep Learning, a model is represented by its parameters. In contrast, a training process involves selecting the best/optimal hyperparameters that are used by learning algorithms to provide the best result. So, what are these hyperparameters? The answer is, Hyperparameters are defined as the parameters that are explicitly defined by the user to control the learning process.

Here the prefix "hyper" suggests that the parameters are top-level parameters that are used in controlling the learning process. The value of the Hyperparameter is selected and set by the machine learning engineer before the learning algorithm begins training the model. **Hence, these are external to the model, and their values cannot be changed during the training process**.

11. What issues can occur if we have a large learning rate in Gradient Descent?

Ans: -

In order for Gradient Descent to work, we must set the learning rate to an appropriate value. This parameter **determines how fast or slow we will move towards the optimal weights**. If the learning rate is very large, we will skip the optimal solution. If it is too small, we will need too many iterations to converge to the best values. So, using a good learning rate is crucial.

In simple language, we can define learning rate as how quickly our network abandons the concepts it has learned up until now for new ones.

12. Can we use Logistic Regression for classification of Non-Linear Data? If not, why?

Ans: -

Non-linear problems can't be solved with logistic regression because it has a linear decision surface. Linearly separable data is rarely found in real-world scenarios.

13. Differentiate between Adaboost and Gradient Boosting.

Ans: -

S.No	Adaboost	Gradient Boost
1	An additive model where shortcomings of previous models are identified by high-weight data points.	An additive model where shortcomings of previous models are identified by the gradient.
2	The trees are usually grown as decision stumps.	The trees are grown to a greater depth usually ranging from 8 to 32 terminal nodes.
3	Each classifier has different weights assigned to the final prediction based on its performance.	All classifiers are weighed equally and their predictive capacity is restricted with learning rate to increase accuracy.
4	It gives weights to both classifiers and observations thus capturing maximum variance within data.	It builds trees on previous classifier's residuals thus capturing variance in data.

14. What is bias-variance trade off in machine learning?

Ans: -

The goal of any supervised machine learning algorithm is to achieve low bias and low variance. In turn the algorithm should achieve good prediction performance.

You can see a general trend in the examples above:

- **Linear** machine learning algorithms often have a high bias but a low variance.
- **Nonlinear** machine learning algorithms often have a low bias but a high variance.

The parameterization of machine learning algorithms is often a battle to balance out bias and variance.

Below are two examples of configuring the bias-variance trade-off for specific algorithms:

- The k-nearest neighbors' algorithm has low bias and high variance, but the trade-off can be changed by increasing the value of k which increases the number of neighbors that contribute to the prediction and in turn increases the bias of the model.
- The support vector machine algorithm has low bias and high variance, but the

trade-off can be changed by increasing the C parameter that influences the number of violations of the margin allowed in the training data which increases the bias but decreases the variance.

There is no escaping the relationship between bias and variance in machine learning.

- Increasing the bias will decrease the variance.
- Increasing the variance will decrease the bias.

There is a trade-off at play between these two concerns and the algorithms you choose and the way you choose to configure them are finding different balances in this trade-off for your problem

In reality, we cannot calculate the real bias and variance error terms because we do not know the actual underlying target function. Nevertheless, as a framework, bias and variance provide the tools to understand the behavior of machine learning algorithms in the pursuit of predictive performance.

15. **Give short description each of Linear, RBF, Polynomial kernels used in SVM.**

Ans: -

Linear SVM: Linear SVM is used for linearly separable data, which means if a dataset can be classified into two classes by using a single straight line, then such data is termed as linearly separable data, and classifier is used called as Linear SVM classifier.

Radial Basis Function: When the **data set is linearly inseparable** or in other words, the data set is non-linear, it is **recommended** to use **kernel functions** such as **RBF**. For a linearly separable dataset (linear dataset) one could use linear kernel function (kernel="linear"). Getting a good understanding of when to use kernel functions will help train the most optimal model using the SVM algorithm. We will use **Sklearn Breast Cancer** data set to understand SVM RBF kernel concepts in this post. The scatter plot given below represents the fact that the dataset is linearly inseparable and it may be a good idea to apply the **kernel method** for training the model.

Polynomial kernel: - In machine learning, the **polynomial kernel** is a kernel function commonly used with support vector machines (SVMs) and other kernelized models, that represents the similarity of vectors (training samples) in a feature space over polynomials of the original variables, allowing learning of non-linear models.

The polynomial kernel looks not only at the given features of input samples to determine their similarity, but also combinations of these. In the context of regression analysis, such combinations are known as interaction features. The (implicit) feature space of a polynomial kernel is equivalent to that of polynomial regression, but without the combinatorial blowup in the number of parameters to be learned. When the input features are binary-valued (Booleans), then the features correspond to logical conjunctions of input features.