Machine Learning Worksheet 5

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

R-squared is a statistical measure that represents the proportion of the variance for a dependent variable that's explained by an independent variable or variables in a regression model. In other words, R-squared measures the extent to which changes in the dependent variable can be predicted by changes in the independent variable(s). Higher R-squared values indicate a better fit of the regression model to the data. Therefore, R-squared is often used to compare different models and select the best one.

On the other hand, Residual Sum of Squares (RSS) measures the difference between the observed values of the dependent variable and the predicted values by the model. It represents the sum of the squared differences between the actual and predicted values of the dependent variable. The goal is to minimize the residual sum of squares to obtain a better model fit.

In terms of determining the goodness of fit of a model, R-squared is generally considered a better measure than RSS. This is because R-squared provides an overall measure of the proportion of variance in the dependent variable that is explained by the model, whereas RSS only measures the magnitude of the residuals. Additionally, R-squared is a standardized measure and ranges from 0 to 1, making it easy to compare the fit of different models. In contrast, the magnitude of the RSS value depends on the scale of the dependent variable and can't be easily compared across models.

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

Total Sum of Squares (TSS) measures the difference between the observed values of the dependent variable and the mean values by the model. It represents the sum of the squared differences between the actual and mean values of the dependent variable.

Residual Sum of Squares (RSS) measures the difference between the observed values of the dependent variable and the predicted values by the model. It represents the sum of the squared differences between the actual and predicted values of the dependent variable.

Explained Sum of Squares (ESS) measures the difference between the mean of the dependent variable and the predicted values by the model. It represents the sum of the squared differences between the mean and predicted values of the dependent variable.

These three metrics can be related by the equation: ***TSS= RSS + ESS***

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

In Machine Learning we divide our dataset into train and test data. Where our machine learns from our train data. While doing so, there are chances that our algorithm learns the data too well including the noise which is **Overfitting**. To avoid this overfitting, Regularization comes into picture.

Regularization is a form of regression that shrinks the coefficient estimates towards zero. The result is that the model generalizes well on the unseen data once overfitting is minimized.

Consider the RSS equation,

$$\mathrm{RSS} = \sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 .$$

Here, y is the dependent variable,
x_1, x_2, x_3.... x_n are independent variables.
b_0, b_1, b_2..... b_n, are the coefficients estimates for different variables of x.

Regularization will shrink these coefficients towards zero.

4. What is Gini–impurity index?

Decision trees algorithm divides the entire dataset into a tree-like vertical information structure so as to divide the different sections of the information with root nodes at the top.

The Gini impurity measure is one of the methods used in decision tree algorithms to decide the optimal split from a root node, and subsequent splits. Gini Impurity tells us what is the probability of misclassifying an observation.

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

Decision tree splits your data set by applying some functions on it, in such a way that data set finally ends up in different homogeneous buckets, i.e a bucket belongs to only one single class. If unregularized, decision tree becomes too large such that they tend to overfit. Here by regularizing, we restrict the decision tree to grow to its full potential.

6. What is an ensemble technique in machine learning?

All machine learning algorithms follows similar behaviour. Models process given inputs and process the output. The output is a prediction base don the patterns a model sees during the training process. Every model has some advantages and disadvantages. There are chances that a single model might not be sufficient to get accurate result. To overcome this ensemble technique is used.

Ensemble is a Machine Learning technique that combines several models in order to produce one optimal predictive model.

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

Both are ensemble methods to get N learners from 1 learner but , while each model is built differently in bagging, in boosting, new models are affected by the previously build models.

Bagging tries to reduce overfitting, boosting tries to reduce bias.

Several training data subsets are randomly drawn with replacement from the whole training dataset in bagging, Whereas in boosting, Each new subset includes the components that were misclassified by previous model.

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

This approach utilizes the usage of bootstrapping in the random forest. Since the bootstrapping samples the data with the possibility of selecting one sample multiple times, it is very likely that we won't select all the samples from the original data set. Therefore, one smart decision would be to exploit somehow these unselected samples, called out-of-bag samples. Correspondingly, the error achieved on these samples is called out-of-bag error. What we can do is to use out-of-bag samples for each decision tree to measure its performance. This strategy provides reliable results in comparison to other validation techniques such as train-test split or cross-validation.

9. What is K-fold cross-validation?

It refers to a method for estimating the performance of a model on unseen data. This technique is recommended to be used when the data is scarce and there is an ask to get a good estimate of training and generalization error thereby understanding the aspects such as underfitting and overfitting. This technique is used for hyperparameter tuning such that the model with the most optimal value of hyperparameters can be trained. It is a resampling technique without replacement. K-fold Cross-Validation is when the dataset is split into a K number of folds and is used to evaluate the model's ability when given new data. K refers to the number of groups the data sample is split into.

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

Hyperparameters in Machine learning are those parameters that are explicitly defined by the user to control the learning process. These hyperparameters are used to improve the learning of the model, and their values are set before starting the learning process of the model. 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?

Gradient descent is the popular optimization algorithm used in machine learning to estimate the model parameters. During training a model, the value of each parameter is guessed or assigned random values initially. The cost function is calculated based on the initial values and the parameter estimates are improved over several steps such that the cost function assumes a minimum value eventually.

In machine learning, we deal with two types of parameters:- 1) Machine learnable parameters And 2) Hyper-parameters.

The Machine learnable parameters are the one which the algorithms learn on their own during the training for a given dataset.

The Hyper-parameters are the one which the machine learning engineers or data scientists will assign specific values to, to control the way the algorithms learn and also to tune the performance of the model.

Learning rate is used to scale the magnitude of parameter updates during gradient descent. The choice of the value for learning rate can impact two things: 1) how fast the algorithm learns and 2) whether the cost function is minimized or not. So, 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.

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

No, we use Logistic Regression for classification of Non-Linear Data because Logistic regression is neither linear nor is it a classifier. The idea of a "decision boundary" has little to do with logistic regression, which is instead a direct probability estimation method that separates predictions from decision.

13. Differentiate between Adaboost and Gradient Boosting.

Loss Function:
The technique of Boosting uses various loss functions. In case of Adaptive Boosting or AdaBoost, it minimises the exponential loss function that can make the algorithm sensitive to the outliers. With Gradient Boosting, any differentiable loss function can be utilised. Gradient Boosting algorithm is more robust to outliers than AdaBoost.

Flexibility:
AdaBoost is the first designed boosting algorithm with a particular loss function. On the other hand, Gradient Boosting is a generic algorithm that assists in searching the approximate solutions to the additive modelling problem. This makes Gradient Boosting more flexible than AdaBoost.

Benefits:
AdaBoost minimises loss function related to any classification error and is best used with weak learners. The method was mainly designed for binary classification problems and can be utilised to boost the performance of decision trees. Gradient Boosting is used to solve the differentiable loss function problem. The technique can be used for both classification and regression problems.

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

Bias is the difference between the average prediction of our model and the correct value which we are trying to predict. Variance is the variability of model prediction for a given data point or a value which tells us spread of our data.

Bias–variance trade-off is the property of a model that the variance of the parameter estimated across samples can be reduced by increasing the bias in the estimated parameters.

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 right/good balance without overfitting and underfitting the data.

This trade-off in complexity is why there is a trade-off between bias and variance. An algorithm can’t be more complex and less complex at the same time.

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

• When we can easily separate data with hyperplane by drawing a straight line is Linear SVM

• The radial basis function kernel, or RBF kernel, is a popular kernel function used in various kernelized learning algorithms. In particular, it is commonly used in support vector machine classification.

• 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.