**MACHINE LEARNING WORK 5**

Answer sheet

**Answer1** there are Both R-squared and Residual Sum of Squares (RSS) are measures of goodness of fit in regression analysis, but they capture different aspects of the model's performance.

* R-squared measures the proportion of variation in the dependent variable that is explained by the independent variables in the model. In other words, it indicates how well the model fits the data, with values ranging from 0 to 1. Higher R-squared values indicate a better fit, as they mean that a larger proportion of the variation in the dependent variable is explained by the independent variables in the model.
* RSS measures the total sum of squared differences between the actual values of the dependent variable and the predicted values by the model. It represents the amount of unexplained variation in the data, and lower RSS values indicate a better fit, as they mean that the model is able to explain more of the variation in the data.

Because both measures are useful in evaluating the goodness of fit of a model, but they serve different purposes. R-squared is a useful measure to assess the overall fit of the model and to compare different models, while RSS is useful to identify the degree of the error in the model's predictions.

In general, a good model should have both a high R-squared value and a low RSS value, indicating that it explains a large proportion of the variation in the dependent variable and has a low degree of error in its predictions. However, in some cases, one measure may be more important than the other, depending on the research question and the nature of the data being analyzed.

Answer 2 In statistics and machine learning, TSS, RSS, and ESS are commonly used to evaluate the performance of a regression model.

The TSS (Total Sum of Squares) is the sum of the squared differences between the actual salary and the mean salary. It represents the total variation in the target variable that we want to explain:

∑TSS=∑i=1 →n​(yi​−y\_barˉ​)²

where y\_i is the actual salary of the i-th individual, y\_bar is the mean salary of all individuals, and n is the number of individuals.

The RSS (Residual Sum of Squares) is the sum of the squared differences between the predicted salary and the actual salary. It represents the unexplained variation in the target variable that is not explained by the model., which is also known as the residual error:

∑=RSS∑i=1 →n​(yi​−yi^​​)2

where hat {y}\_i is the predicted salary of the i-th individual. and y\_hat is the predicted salary

The ESS (Explained Sum of Squares) is the sum of the squared differences between the predicted salary and the mean salary. It represents the variation in the target variable that is explained by the independent variables and that is explained by the model (CGPA and IQ score):

∑ESS=∑i=1n​(y^​i​−yˉ​)2

The relationship between TSS, RSS, and ESS is as TSS=ESS+RSS In other words, the total variation in the target variable can be decomposed into the variation that is explained by the independent variables (ESS) and the variation that is unexplained by the model (RSS).

R-squared = Ess / Tss

Answer 3 Regression is basically a statistical approach to find the relationship between variables. In machine learning, this is used to predict the outcome of an event based on the relationship between variables obtained from the data-set.

Linear regression is one type regression used in Machine Learning. Any linear regression model will be represented as

**output = coefficient 1 + coefficient 2 \* input**.

For example, consider prediction of weight based on height.

Using linear regression, weight = a + b \* height

Having found a relationship between these two variables, the coefficients a and b can be found out, based on which the weight for a given height can be predicted.

Answer 4 The Gini Impurity is a loss function that describes the likelihood of misclassification for a single sample, according to the distribution of a certain set of labelled data. It is typically used within Decision Trees. More specifically, the Gini Impurity is used when training/growing a decision tree on a labelled training set. This metric can determine how best to split data at a given node in the tree, in order to create optimal child nodes. In general, a split that results in as little variation as possible, in the labels for each child node, is considered optimal.

Answer 5 Yes, its because there are unregularized in machine learning. there

There are basically 3 ways to avoid overfitting in decision trees-

1. **Bagging**
2. **Boosting**
3. **Pruning**

* **Bagging** -Here, you are expected to create multiple decision trees with various samples and get the output which is suggested by more number of decision trees.
* **Boosting -**Here, a single decision tree is initially created and updated multiple times by comparing with predicted and actual labels in each iteration. If any sample is found to be contributing to an error in prediction then those samples will be provided with more weights in the next iteration.Thereby in each iteration, we will train a sample that is producing wrong prediction more times and a sample that is producing correct prediction fewer times. When this process goes on for a long time then the tree gets saturated and avoids overfitting and is expected to produce more accurate results in the test data.
* **Prunin** This is a technique in which tree will be constructed only up to a certain level of depth.Either depth is initially defined or leaves will be removed from the tree from the lower end until the specified depth is reached after constructing the whole tree.The first method is called pre-pruning and later is called post-pruning.

**Answer 6** Ensemble methods is a machine learning technique that combines several base models in order to produce one optimal predictive model. To better understand this definition lets take a step back into ultimate goal of machine learning and model building. This is going to make more sense as I dive into specific examples and why Ensemble methods are used.I will largely utilize Decision Trees to outline the definition and practicality of Ensemble Methods (however it is important to note that Ensemble Methods do not only pertain to Decision Trees).

**Answer7** There are difference between bagging and boosting

**1. Bagging**

The Bagging ensemble technique is the acronym for “bootstrap aggregating” and is one of the earliest ensemble methods proposed. For this method, subsamples from a dataset are created and they are called “bootstrap sampling.” To put it simply, random subsets of a dataset are created using replacement, meaning that the same data point may be present in several subsets.

These subsets are now treated as independent datasets, on which several Machine Learning models will be fit. During test time, the predictions from all such models trained on different subsets of the same data are accounted for.There is an aggregation mechanism used to compute the final prediction (like averaging, weighted averaging, etc. discussed later). Bagging ensemble technique Bagging The image shown above exemplifies the Bagging ensemble mechanism.

Note :- that, in the bagging mechanism, a parallel stream of processing occurs. The main aim of the bagging method is to reduce variance in the ensemble predictions. Thus, the chosen ensemble classifiers usually have high variance and low bias (complex models with many trainable parameters). Popular ensemble methods based on this approach include:

* Bagged Decision Trees
* Random Forest Classifiers
* Extra Trees

**2. Boosting**

The boosting ensemble mechanism works in a way markedly different from the bagging mechanism. Here, instead of parallel processing of data, sequential processing of the dataset occurs. The first classifier is fed with the entire dataset, and the predictions are analyzed.The instances where Classifier-1 fails to produce correct predictions (that are samples near the decision boundary of the feature space) are fed to the second classifier. This is done so that Classifier-2 can specifically focus on the problematic areas of feature space and learn an appropriate decision boundary. Similarly, further steps of the same idea are employed, and then the ensemble of all these previous classifiers is computed to make the final prediction on the test data.

Boosting ensemble mechanism The main aim of the boosting method is to reduce bias in the ensemble decision. Thus, the classifiers are chosen for the ensemble usually need to have low variance and high bias, i.e., simpler models with less trainable parameters.

Other algorithms based on this approach include:

* Adaptive Boosting
* Stochastic Gradient Boosting
* Gradient Boosting Machines

Answer8 OOB (out-of-bag) errors are an estimate of the performance of a random forest classifier or regressor on unseen data. In scikit-learn, the OOB error can be obtained using the oob\_score\_ attribute of the random forest classifier or regressor.

The OOB error is computed using the samples that were not included in the training of the individual trees. This is different from the error computed using the usual training and validation sets, which are used to tune the hyperparameters of the random forest.

The OOB error can be useful for evaluating the performance of the random forest on unseen data. It is not always a reliable estimate of the generalization error of the model, but it can provide a useful indication of how well the model is performing.

Implementation of OOB Errors for Random Forests

To compute the OOB error, the samples that are not used in the training of an individual tree are known as “out-of-bag” samples. These samples are not used in the training of the tree, but they are used to compute the OOB error for that tree. The OOB error for the entire random forest is computed by averaging the OOB errors of the individual trees.

To install NumPy and scikit-learn, you can use the following commands:

* pip install numpy
* pip install scikit-learn

**Answer 9** Cross validation is an evaluation method used in machine learning to find out how well your machine learning model can predict the outcome of unseen data. It is a method that is easy to comprehend, works well for a limited data sample and also offers an evaluation that is less biased, making it a popular choice. The data sample is split into ‘k’ number of smaller samples, hence the name: K-fold Cross Validation. You may also hear terms like four fold cross validation, or ten fold cross validation, which essentially means that the sample data is being split into four or ten smaller samples respectively.

**Answer 10** In machine learning, we need to differentiate between parameters and hyper parameters. A learning algorithm learns or estimates model parameters for the given data set, then continues updating these values as it continues to learn. After learning is complete, these parameters become part of the model. For example, each weight and bias in a neural network is a parameter.

Hyper parameters, on the other hand, are specific to the algorithm itself, so we can’t calculate their values from the data. We use hyper parameters to calculate the model parameters. Different hype parameter values produce different model parameter values for a given data set.

Hyper parameter tuning consists of finding a set of optimal hyper parameter values for a learning algorithm while applying this optimized algorithm to any data set. That combination of hyper parameters maximizes the model’s performance, minimizing a predefined loss function to produce better results with fewer errors. Note that the learning algorithm optimizes the loss based on the input data and tries to find an optimal solution within the given setting. However, hyper parameters describe this setting exactly.

**Answer11**  The learning rate is an important hyper parameter that greatly affects the performance of gradient descent. It determines how quickly or slowly our model learns, and it plays an important role in controlling both convergence and divergence of the algorithm. When the learning rate is too large, gradient descent can suffer from divergence. This means that weights increase exponentially, resulting in exploding gradients which can cause problems such as instabilities and overly high loss values. On the other hand, if the learning rate is too small, then gradient descent can suffer from slow convergence or even stagnation—which means it may not reach a local minimum at all unless many iterations are performed on large datasets.In order to avoid these issues with different learning rates for each parameter/variable, we use adaptive techniques such as Adagrad and Adam which adjust their own learning rates throughout training based on real-time observations of parameters during optimization (i.e., they control exploration/exploitation trade-offs). These adaptive measures ensure better results than standard gradient descent while avoiding potential pitfalls in terms of either massive gains or slow losses due to misconfigured static global learning rates like those used with traditional gradient descent algorithms.

**Answer12** The short answer is: Logistic regression is considered a generalized linear model because the outcome always depends on the sum of the inputs and parameters. Or in other words, the output cannot depend on the product (or quotient, etc.) of its parameters So,that Let’s recapitulate the basics of logistic regression first, which hopefully makes things more clear. Logistic regression is an algorithm that learns a model for binary classification. A nice side-effect is that it gives us the probability that a sample belongs to class 1 (or vice versa: class 0). Our objective function is to minimize the so-called logistic function Φ (a certain kind of sigmoid function);

Answer 13 there are difference betwwen AdaBoost and Gradient Boosting

* AdaBoost or Adaptive Boosting is the first Boosting ensemble model. The method automatically adjusts its parameters to the data based on the actual performance in the current iteration. Meaning, both the weights for re-weighting the data and the weights for the final aggregation are re-computed iteratively. In practice, this boosting technique is used with simple classification trees or stumps as base-learners, which resulted in improved performance compared to the classification by one tree or other single base-learner.
* Gradient Boosting Gradient Boost is a robust machine learning algorithm made up of Gradient descent and Boosting. The word ‘gradient’ implies that you can have two or more derivatives of the same function. Gradient Boosting has three main components: additive model, loss function and a weak learner. The technique yields a direct interpretation of boosting methods from the perspective of numerical optimisation in a function space and generalises them by allowing optimisation of an arbitrary loss function.
* 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.
* Shortcomings In the case of Gradient Boosting, the shortcomings of the existing weak learners can be identified by gradients and with AdaBoost, it can be identified by high-weight data points.
* Wrapping Up Though there are several differences between the two boosting methods, both the algorithms follow the same path and share similar historic roots. Both the algorithms work for boosting the performance of a simple base-learner by iteratively shifting the focus towards problematic observations that are challenging to predict.

**Answer 14**- Bias in ML is an sort of mistake in which some aspects of a dataset are given more weight and/or representation than others. A skewed outcome, low accuracy levels, and analytical errors result from a dataset that is biased that does not represent a model’s use case accurately. ML projects require training data that is indicative of the real world because it is through this data that the model learns how to do what it was made for. From exclusion bias and recall bias to sample and association bias, machine learning bias can occur in a variety of ways. For any data project, it’s critical to be aware of the potential machine learning biased data. You can detect it before it becomes a problem or respond to it when it arises by putting the right systems in place early and staying on top of data collection, labeling, and implementation. This is why we’ll next discuss the different types of bias and then talk about how to help with reducing bias in machine learning.

**Answer15**  Kernel Function is a method used to take data as input and transform it into the required form of processing data. “Kernel” is used due to a set of mathematical functions used in Support Vector Machine providing the window to manipulate the data. So, Kernel Function generally transforms the training set of data so that a non-linear decision surface is able to transform to a linear equation in a higher number of dimension spaces. Basically, It returns the inner product between two points in a standard feature dimension.

* Standard Kernel Function Equation :

K (\bar{x}) = 1, if ||\bar{x}|| <= 1

K (\bar{x}) = 0, Otherwise

* Major Kernel Functions :-

For Implementing Kernel Functions, first of all, we have to install the “scikit-learn” library using the command prompt terminal: pip install scikit-learn

* Gaussian Kernel: = It is used to perform transformation when there is no prior knowledge about data.

K (x, y) = e ^ - (\frac{||x - y||^2} {2 \sigma^2})