 ASSIGNMENT - 5

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 of goodness of fit model in regression and why?
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.
3. What is the need of regularization in machine learning?
4. What is Gini–impurity index?
5. Are unregularized decision-trees prone to overfitting? If yes, why?
6. What is an ensemble technique in machine learning?
7. What is the difference between Bagging and Boosting techniques?
8. What is out-of-bag error in random forests?
9. What is K-fold cross-validation?
10. What is hyper parameter tuning in machine learning and why it is done?
11. What issues can occur if we have a large learning rate in Gradient Descent?
12. Can we use Logistic Regression for classification of Non-Linear Data? If not, why?
13. Differentiate between Adaboost and Gradient Boosting.
14. What is bias-variance trade off in machine learning?
15. Give short description each of Linear, RBF, Polynomial kernels used in SVM.
16. Using the residual values, one can determine the sum of squares of the residuals known as **Residual sum of squares** or RSS.

Residual Sum of Squares

* The lower the value of **Residual sum of squares the** better the model can make predictions **or in other words,** a regression line is a best fit if it minimizes the RSS value. However, this view can be challenged as it is a scale variant statistic i.e. the value depends on the scale of the target variable

### Example: Consider your target variable is the revenue generated by selling a product. The residuals would depend on the scale of this target. If the revenue scale was taken in “Hundreds of Rupees” (i.e. target would be 1, 2, 3, etc.) then we might get an RSS of about 0.54 (hypothetically speaking).

### If the target variable is the revenue generated after sales, the residuals will depend on the scale of this target. If the scale is in hundreds or thousands of rupees, the target would be 1, 2, 3, etc. The RSS can be about 0.54 whereas if we take the same for rupees, the target could be 100, 200, 300, etc. The RSS could go upto 5400

### Even though the data did not change, the value of RSS varies depending on the scale of the target which makes it difficult to determine the right RSS value.

### Problems with R-squared statistic: The R-squared statistic is not prefect either. Its value does not decreases no matter how many variables we add to the regression model ie, even if we are adding redundant variables to the data, the value of R-squared does not decrease. Adjusted R-squared deals with this issue.

### The Adjusted R-squared takes into account the number of independent variables used for predicting the target variable which helps in determining whether adding new variables to the model actually increases the model fit.

### It better to use Adjusted R-squared when there are multiple variables in the regression model as compared to RSS or R-squared

### ****Total Sum of Squares****

Total variation is the sum of squares of the difference between the actual values and their mean in a given target variable

Total Sum of Squares

TSS or Total sum of squares gives the total variation in Y. While the variance is the average of the squared sums of difference between actual values and data points, TSS is the total of the squared sums.

## Explained Sum of Squares

The Explained SS shows the amount of variation in the dependent variable in the model explained.  
Explained SS = Σ(Y-Hat – mean of Y)2.

## Residual Sum of Squares?

The residual sum of squares shows how much of the dependent variable’s variation your model **did not explain**. It is the sum of the squared differences between the actual Y and the predicted Y:  
Residual Sum of Squares = Σ e2

1. Regularization takes the guess work out .Regularization refers to techniques that are used in machine learning models to minimize the adjusted loss function and prevent overfitting or underfitting. It assists with fitting the machine learning model appropriately on a given test set , thereby reducing the errors in it. There are two main types of regularization techniques: Ridge Regularization and Lasso

Ridge: Ridge Regression, modifies the over-fitted or under fitted models by adding the penalty. The penalty is equivalent to the sum of the squares of the magnitude of coefficients. It I useful when there are many variables with relatively small data samples. It prevents overfitting my getting the convergence close to zero.

Lasso: It is sed to fit a linear model with few variables. It encourages the coefficients of the variables to go towards zero because of the shape of the constraints which is an absolute value.

 Regularization is all about finding the right balance between the simple and too complex. In an underfit model, the model is unable to perform well both on the training and test data set but it is not complex enough to display the patterns in the given data. On the other hand, an overfit model learns the data patterns so thoroughly that it even includes the noise in the data set, ultimately losing the vision to predict unseen data correctly. This is why moderation is required while finding patterns in the given data set and regularization techniques assist with this.

When the bias is high, the training and testing error will be high, and when the variance is high, the training error will be low and the testing error high. There will be a point where both the training and test error will be reasonably low and somewhere between these two extremes. The goal of regularization techniques is to find this right balance.

1. It is a measurement used to build Decision Trees to determine how the features of a dataset should split nodes to form the tree. Gini impurity involves measuring the diversity of the data set. Gini Index is the probability of picking two distinct elements. To be more precise, Gini Impurity is a number between 0-0.5, which indicates any likelihood of new, random data being misclassified.
2. Yes, unregularized decision-trees prone to overfitting. Up until a certain number of iterations, new iterations can improve the model. However, the model’s ability to generalize tends to weaken as it begins to overfit the training data. Hence, it is important to prune and ensemble the unregulated data. Unregulated data is prone to overfitting. Especially , in case of decision tree’s, it can learn a training set to a point of high granularity which makes them prone to overfitting. Allowing a decision tree to split to a granular degree, is the behavior of this model which makes it susceptible to learning every point with great efficiency — to the point of overfitting. Regularization refers to a broad range of techniques that artificially force the model to be simpler.
3. What is an ensemble technique in machine learning?
4. What is the difference between Bagging and Boosting techniques?

Ensembles are machine learning techniques for combining predictions from many separate models. Ensemble techniques combine individual models together to improve the stability and predictive power of the model. This technique permits higher predictive performance as it combines multiple machine learning models into one predictive model.

Certain models do well in modeling one aspect of the data, while others do well in modeling another. It learns several simple models and combines their output to produce the final decision. The combined strength of the models offsets individual model variances and biases

The two most common are: bagging and boosting.

Boosting reduces bias by training weak learners sequentially, each trying to correct its predecessor. For example, x causes headache, but x does not cause back pain. Y causes back pain. These tw variables are not responsibe for stomach pain. Z causes stomach pain. If h1 is a claaaifier that classifies data wth accuracy. It trains H1 and identifies regions where H1 produces errors, adds weights to them amd produces H2 classier. Aggregate those samples for which H1 gives a different result from H2 and produces H3 classifier

*Boosting* aims to improve the predictive flexibility of simple models. It trains a large number of “weak” learners in sequence. A weak learner is a constrained model (i.e. you could limit the max depth of each decision tree). Each one in the sequence focuses on learning from the mistakes of the one before it. it then combines all the weak learners into a single strong learner.

Bagging uses complex base models and tries to “smooth out” their predictions, while boosting uses simple base models and tries to “boost” their aggregate complexity.

Bagging is also known as Bootstrap aggregating. In bagging, we reduce the variance of the model by generating additional test data. Once the size of the data set is increased, we can tune the model to be more immune to variance.

Boosting: Boosting is a two step algorithm for ensemble learning. In boosting, we use subsets of the dataset to create an average performance model. Then we tune the model on a larger data set to boost the performance of the model.

Data partition: In bagging data partition is random. In boosting mis-classified data is given higher importance.

Goal: In bagging the goal is to reduce the variance in model. Boosting aims for increasing the prediction accuracy of the model.

Method: We can use random subspace in bagging. Bosting uses gradient descent method.

1. OOB\_Score is a a powerful Validation Technique used in Random Forest algorithm for least Variance results. OOB Error is the number of wrongly classifying the OOB Sample.

The RandomForestClassifier is trained using bootstrap aggregation, where each new tree is fit from a bootstrap sample of the training observations zi=(xi,yi). The  (OOB) error is the average error for each zi calculated using predictions from the trees that do not contain zi in their respective bootstrap sample OOB Error allows to test the data simultaneously with training, but the overall process is time-consuming as compared to other validation techniques. In case the data is huge it is not recommended to use this technique. It is best for the medium or small sized, data set. OOB\_Score is a preferred method over other techniques for a predictive model.

1. What is K-fold cross-validation?

Following steps are involved with K-fold cross-validation : Reserve some portion of sample data-set, use the rest data-set train the model and test the model using the reserve portion of the data-set. In K-fold cross-validation, the data-set is split into k number of subsets (known as folds) then we perform training on the all the subsets but leave one(k-1) subset for the evaluation of the trained model In this method, we iterate k times with a different subset reserved for testing purpose each time.

1. A model parameter is a variable of the selected model which can be estimated by fitting the given data to the model. A model hyperparameter is the parameter whose value is set before the model start training. They cannot be learned by fitting the model to the data. They are completely tuned out and can be set manually.

In grid search we define the combinations and do training of the model.

Random search: This model selects the combinations randomly

Hyperparameters are supplied as arguments to the model algorithm during initializing them as key,value.

The values are picked by the data scientist,who is building the model in iterative mode. It is used for optimizing the model and decides the quality of the model.

Hyperparameters are influencing the below factors while designing your model.

Linear Model - What degree of polynomial features should use?

Decision Tree - What is the maximum allowed depth? And What is the minimum number of samples required at a leaf node in the decision tree?

Random forest - How many trees we should include?

Neural Network - How many neurons we should keep in a layer?/ How many layers should keep in a layer?

Gradient Descent - What learning rate should we use?

1. **Learning Rate is** the hyperparameter that determines the steps the gradient descent algorithm takes. If the step is too big, the algorithm may bypass the local minimum and overshoot and it too small, it might increase the total computation time significantly.It is important to note that the step a gradient descent takes is a function of step size ηη as well as the gradient values gg. If the local minimum has zero gradient , the algorithm will not update the parameters pp because the gradient is zero, similarly if pp is in a "steep slope", even a small change in syep size - ηη will lead to a large update in pp's values.

Gradient descent can converge to a local minimum,even with the learning rate a fixed. As we approach a local minimum gradient descent will automatically take smaller steps. So,no need to decrease a over time.

1. Generally, this is not possible under normal circumstances.

If we have to classify malignant tumour as No, value is 0. if it is malignant, the value is 1

Ifh(x)>0.5:

Predict Malignant tumor

Else:

Predict benignly

Now, ifh(x)>0.5

Malignant doesn't work

Now change h(x)to 0.2

But no algorithm works like that. Hypothesis can't change every time for a new dataset. Linear Regression is unbounded

However, with feature transformation this is possible.

1. Boosting techniques uses various loss functions. In case of AdaBoost, it minimizes the exponential loss function which makes the algorithm sensitive to the outliers. Any differential loss is utilized. Gradient Boosting algorithm is more robust to outliers than AdaBoost.

Gradient boosting is more flexible than Adaboost as Adaboost is limited to a particular loss deduction whereas Gradient boosting is a generic algorithm that assists with searching the approximate solutions to additive modelling problem

AdaBoost is best used with weak learners, is designed for binary classification problems and can be utilized to boost the decision trees performance. Gradient Boosting can be used for classification and regression problems.

The drawback of gradient boosting is that the existing weak learners can be identified by gradients and with Adaboost , it can be identified by heavy weight data points

In the case of AdaBoost, the shifting is done by up-weighting observations that were misclassified before, while Gradient Boosting identifies the difficult observations by large residuals computed in the previous iterations.

1. Finding the right balance between bias and variance of the model is called the bias variance trade off. It’s a way to ensure that the model is overfitted or underfitted in any case. If the model is too simple and has very few parameters it will suffer from high bias and low variance. On the other hand, if the model has a large number of parameters, it will have a high variance and low bias. This trade off should result in a perfectly balanced relationship between the two. Ideally the low bias and low variance is the target for any ML model that you have. High variance will overfit the model
2. Linear KERNEL: When the data is linearly separable:

In 2D the classifying function is a line

In 3D it's a Plane. The similar function above

3D is called hyperplane.

Z= x.y

Polynomial Kernel: When the data is not linearly separable, one can make it linearly separable by mapping the data in higher dimensional space

RBF/Gaussian Kernel: When the data is not linearly separable, one can make it linearly separable by mapping the data in infinite dimensional space. We often use RBF Kernel when we have no previous information about the data

|  |  |  |  |
| --- | --- | --- | --- |
| Property | Linear | RBF/Gaussian | Polynomial |
| Learning time | Worst of all | In between | Best |
| Fitting Data | Worst of all | In between | Best |
| Overfitting | Lowest possibility | In between | Highest possibility |
| Underfitting | Highest possibility | In between | Lowest possibility |