**MACHINE LEARNING**

# **WORKSHEET – 3 : ANSWERS**

* **Polynomial kernel:-**  represents the similarity of vectors (training samples) in a feature space over polynomials of the original variables, allowing learning of non-linear models.Intuitively, the polynomial kernel looks not only at the given features of input samples to determine their similarity, but also combinations of these. In the polynomial kernel, we simply calculate the dot product by increasing the power of the kernel.
* **Radial basis function kernel (RBF)/ Gaussian Kernel:** Radial Basis Function is another popular Kernel method used in SVM models for more. RBF kernel is a function whose value depends on the distance from the origin or from some point.
* **Linear Kernel**:- is used when the data is Linearly separable, that is, it can be separated using a single Line. It is one of the most common kernels to be used. It is mostly used when there are a large number of Features in a particular Data Set. One of the examples where there are a lot of features, is Text Classification, as each alphabet is a new feature. So we mostly use Linear Kernel in Text Classification.

\*\* The main differences between these three kernels under few measures are:-

* time of SVM learning: linear < poly < rbf
* ability to fit any data: linear < poly < rbf
* risk of overfitting: linear < poly < rbf
* risk of underfitting: rbf < poly < linear
* number of hyperparameters: linear (0) < rbf (2) < poly (3)
* how "local" is particular kernel: linear < poly < rbf

1. **Residual Sum of Squares (RSS)** is a better measure of goodness of fit of model in regression because Rsquare can measure the goodness of out-of-sample prediction. If we split our data into training and testing parts and fit a regression model on the training one, we can get a valid R squared value on training part, but we can't legitimately compute an R squared on the test part.
2. **Explained sum of squares (ESS):** It is the portion of total variation that measures how well the regression equation explains the relationship between X and Y.

image2.png

**Residual sum of squares (RSS):** This expression is also known as unexplained variation and is the portion of total variation that measures discrepancies (errors) between the actual values of Y and those estimated by the regression equation. You compute the RSS with the formula

image3.png

**Total sum of squares (TSS):** It is the sum of RSS and ESS

image4.png

Above equation also represents relation between the three metrics

1. **Gini**  Index is an indicator of how the classification split is with respect to the classes. It is used to minimize misclassification.Lower the gini better the split
2. **Yes**, because unregularized tress grow to a depth until they don’t get pure leaf nodes. Sometimes, it may happen that there is only one point in each leaf nodes. Now, changing a single point in dataset will cause a change in the entire model, which is nothing but over-fitting.
3. **Ensemble technique** refers to the process of combining various weak learners to result into a strong learner. Bagging, boosting and stacking are examples of ensmebling techniques used widely.
4. **Bagging:** Bagging is also known as bootstrap aggregating sits on top of the majority voting principle. The samples are bootstrapped each time when the model is trained. When the samples are chosen, they are used to train and validate the predictions. The samples are then replaced back into the training set. The samples are selected at random. This technique is known as bagging. To sum up, base classifiers such as decision trees are fitted on random subsets of the original training set. Subsequently, the individual predictions are aggregated (voting or averaging etc.). The final results are then used as predictions. It reduces the variance of a black box estimator. Due to this the chances of overfitting is ruled out.

**Boosting:** The concept of Adaptive Boost revolves around correcting previous classifier mistakes. Each classifier gets trained on the sample set and learns to predict. The misclassification errors are then fed into the next classifier in the chain and are used to correct the mistakes until the final model predicts accurate results. When a weak-classifier misclassifies a training sample, the algorithm then uses these very samples to improve the performance of the ensemble.

1. In random forest algorithm, each tree is trained on about 2/3 of the total training data. As the forest is built, each tree can thus be tested (similar to leave one out cross validation) on the samples not used in building that tree. This is the **out of bag error** estimate - an internal error estimate of a random forest as it is being constructed.
2. **K-fold Cross validation** is a technique to fit a model on data set. In cross validation the data set is divided into ‘k’ number of sets where ‘k-1’ sets are used for training and 1 set is used as validation set. And this is done for all the set one by one and the final score of model is taken as average score of all the ‘k’ number of fits.

Advantage of using Cross validation is that, there is no need of separate validation data, cross validation reduces chances of overfitting and gives a more generic model. Cross validation has a disadvantage that it takes more time to fit the model over a large dataset and the model built is more complex than the basic model.

1. **Hyperparameter tuning** is the problem of choosing a set of optimal hyperparameters for a learning algorithm. A hyperparameter is a parameter whose value is used to control the learning process. By contrast, the values of other parameters (typically node weights) are learned.
2. When the learning rate is too large, gradient descent can inadvertently increase rather than decrease the training error. We might overshoot the minima, and keep bouncing along the ridges of the "valley" without ever reaching the minima.
3. If the bias of a model is decreased, the variance of the model automatically increases. The vice-versa is also true, that is if the variance of a model decreases, bias starts to increase. Hence, it can be concluded that it is nearly impossible to have a model with no bias or no variance since decreasing one increases the other. This phenomenon is known as the **Bias-Variance Trade off**.
4. **Regularisation** is a technique used to reduce the errors by fitting the function appropriately on the given training set and avoid overfitting. The commonly used regularisation techniques are: L1 regularisation L2 regularisation Dropout regularisation.
5. **Adaboost** is more about ‘voting weights’.It increases the accuracy by giving more weightage to the target which is misclassified by the model. At each iteration, Adaptive boosting algorithm changes the sample distribution by modifying the weights attached to each of the instances. It increases the weights of the wrongly predicted instances and decreases the ones of the correctly predicted instances.

**Gradient** is more about ‘adding gradient optimization’. It calculates the gradient (derivative) of the Loss Function with respect to the prediction (instead of the features). Gradient boosting increases the accuracy by minimizing the Loss Function (error which is difference of actual and predicted value) and having this loss as target for the next iteration. Gradient boosting algorithm builds first weak learner and calculates the Loss Function. It then builds a second learner to predict the loss after the first step. The step continues for third learner and then for fourth learner and so on until a certain threshold is reached.

1. We can’t use Logistic Regression for classification of Non-Linear Data because in linear regression the decision boundary is linear. It is used to come up with a hyperplane in feature space to separate observations that belong to a class from all the other observations that do not belong to that class. The decision boundary is thus linear.