MACHINE LEARNING ASSIGNMENT - 8

In Q1 to Q7, only one option is correct, Choose the correct option:

1. What is the advantage of hierarchical clustering over K-means clustering?
    D) None of these

2. Which of the following hyper parameter(s), when increased may cause random forest to over fit the data?
    A) max_depth

3. Which of the following is the least preferable resampling method in handling imbalance datasets?
    D) ADASYN

4. Which of the following statements is/are true about “Type-1” and “Type-2” errors? 1. Type1 is known as false positive and Type2 is known as false negative. 2. Type1 is known as false negative and Type2 is known as false positive. 3. Type1 error occurs when we reject a null hypothesis when it is actually true.
    C) 1 and 3

5. Arrange the steps of k-means algorithm in the order in which they occur: 1. Randomly selecting the cluster centroids 2. Updating the cluster centroids iteratively 3. Assigning the cluster points to their nearest center
    D) 1-3-2

6. Which of the following algorithms is not advisable to use when you have limited CPU resources and time, and when the data set is relatively large?
    B) Support Vector Machines

7. What is the main difference between CART (Classification and Regression Trees) and CHAID (Chi Square Automatic Interaction Detection) Trees?
    C) CART can only create binary trees (a maximum of two children for a node), and CHAID can create multiway trees (more than two children for a node)

In Q8 to Q10, more than one options are correct, Choose all the correct options:

8. In Ridge and Lasso regularization if you take a large value of regularization constant(lambda), which of the following things may occur?
    B) Lasso will lead to some of the coefficients to be very close to 0

D) Lasso will cause some of the coefficients to become 0.

9. Which of the following methods can be used to treat two multi-collinear features?

C) Use ridge regularization

D) use Lasso regularization

10. After using linear regression, we find that the bias is very low, while the variance is very high. What are the possible reasons for this?

A) Overfitting

Q11 to Q15 are subjective answer type questions, Answer them briefly.

11. In which situation One-hot encoding must be avoided? Which encoding technique can be used in such a case?
→ For categorical variables where no such ordinal relationship exists, the integer encoding is not enough.
In fact, using this encoding and allowing the model to assume a natural ordering between categories may result in poor performance or unexpected results (predictions halfway between categories).

In this case, a one-hot encoding can be applied to the integer representation. This is where the integer encoded variable is removed and a new binary variable is added for each unique integer value.

In the "*color*" variable example, there are 3 categories and therefore 3 binary variables are needed. A "1" value is placed in the binary variable for the color and "0" values for the other colors.

12. In case of data imbalance problem in classification, what techniques can be used to balance the dataset? Explain them briefly.
→ This problem is predominant in scenarios where anomaly detection is crucial like electricity pilferage, fraudulent transactions in banks, identification of rare diseases, etc. In this situation, the predictive model developed using conventional machine learning algorithms could be biased and inaccurate.

This happens because Machine Learning Algorithms are usually designed to improve accuracy by reducing the error. Thus, they do not take into account the class distribution / proportion or balance of classes.

13. What is the difference between SMOTE and ADASYN sampling techniques?
→ The key difference between ADASYN and SMOTE is that the former uses a density distribution, as a criterion to automatically decide the number of synthetic samples that must be generated for each minority sample by adaptively changing the weights of the different minority samples to compensate for the skewed distributions. The latter generates the same number of synthetic samples for each original minority sample.

14. What is the purpose of using GridSearchCV? Is it preferable to use in case of large datasets? Why or why not?
→ **GridSearchCV** tries all the combinations of the values passed in the dictionary and evaluates the model for each combination **using** the Cross-Validation method. Hence after **using** this function we get accuracy/loss for every combination of hyperparameters and we can choose the one **with** the **best** performance.

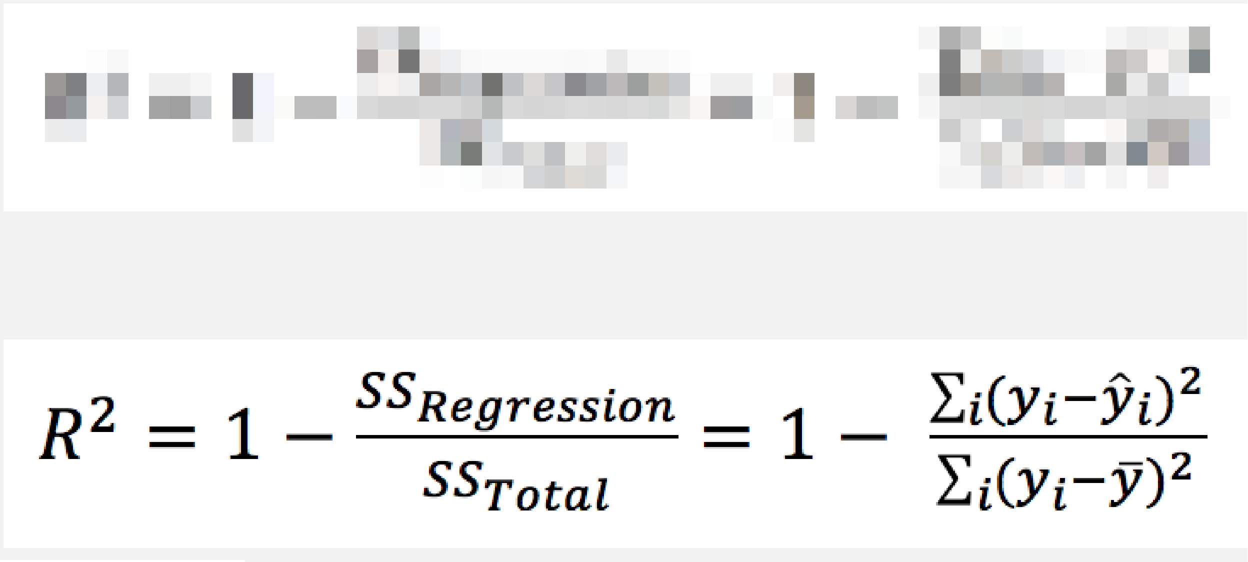
15. List down some of the evaluation metric used to evaluate a regression model. Explain each of them in brief.
→ *1. R Square/Adjusted R Square*

2. Mean Square Error(MSE)/Root Mean Square Error(RMSE)

3. Mean Absolute Error(MAE)

R Square/Adjusted R Square

R Square measures how much variability in dependent variable can be explained by the model. It is the square of the Correlation Coefficient(R) and that is why it is called R Square.



$$R^2 = 1 - \frac{SS_{Regression}}{SS_{Total}} = 1 - \frac{\sum_i (y_i - \hat{y}_i)^2}{\sum_i (y_i - \bar{y})^2}$$

R square formula

R Square is calculated by the sum of squared of prediction error divided by the total sum of the square which replaces the calculated prediction with mean. R Square value is between 0 to 1 and a bigger value indicates a better fit between prediction and actual value.

R Square is a good measure to determine how well the model fits the dependent variables. **However, it does not take into consideration of overfitting problem**. If your regression model has many independent variables, because the model is too complicated, it may fit very well to the training data but performs badly for testing data. That is why Adjusted R Square is introduced because it will penalize additional independent variables added to the model and adjust the metric to prevent overfitting issues.

```
#Example on R_Square and Adjusted R Square

import statsmodels.api as sm

X_addC = sm.add_constant(X)

result = sm.OLS(Y, X_addC).fit()

print(result.rsquared, result.rsquared_adj)

# 0.79180307318 0.790545085707
```

In Python, you can calculate R Square using Statsmodel or Sklearn Package

From the sample model, we can interpret that around 79% of dependent variability can be explained by the model, and adjusted R Square is roughly the same as R Square meaning the model is quite robust.

Mean Square Error(MSE)/Root Mean Square Error(RMSE)

While R Square is a relative measure of how well the model fits dependent variables, Mean Square Error is an absolute measure of the goodness for the fit.

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2$$

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2$$

Mean Square Error formula

MSE is calculated by the sum of square of prediction error which is real output minus predicted output and then divide by the number of data points. It gives you an absolute number on how much your predicted results deviate from the actual number. You cannot interpret many insights from one single result but it gives you a real number to compare against other model results and help you select the best regression model.

Root Mean Square Error(RMSE) is the square root of MSE. It is used more commonly than MSE because firstly sometimes MSE value can be too big to compare easily. Secondly, MSE is calculated by the square of error, and thus square root brings it back to the same level of prediction error and makes it easier for interpretation.

```
from sklearn.metrics import mean_squared_error

import math

print(mean_squared_error(Y_test, Y_predicted))

print(math.sqrt(mean_squared_error(Y_test, Y_predicted)))

# MSE: 2017904593.23

# RMSE: 44921.092965684235
```

MSE can be calculated in Python using Sklearn Package

Mean Absolute Error(MAE)

Mean Absolute Error(MAE) is similar to Mean Square Error(MSE). However, instead of the sum of square of error in MSE, MAE is taking the sum of the absolute value of error.

$$MAE = \frac{1}{n}\sum_{j=1}^{n}|y_j - \hat{y}_j|$$

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|y_i - \hat{y}_i|$$

Mean Absolute Error formula

Compare to MSE or RMSE, MAE is a more direct representation of sum of error terms.

MSE gives larger penalization to big prediction error by square it while MAE treats all errors the same.

```
from sklearn.metrics import mean_absolute_error

print(mean_absolute_error(Y_test, Y_predicted))

#MAE: 26745.1109986
```

MAE can be calculated in Python using Sklearn Package