**Statistics**

1. D
2. C
3. C
4. B
5. C
6. B
7. A
8. A
9. B
10. A

**SQL**

1. Select \* from movie;
2. Select title from movie order by runtime desc limit 1;
3. Select title from movie order by revenue desc limit 1;
4. Select title from movie order by budget desc limit 1;
5. Select title , person\_name, gender, character\_name, cast\_order from movie m inner join movie\_cast mc on m.movie\_id= mc.movie\_id inner join person p on p.person\_id=mc.person\_id inner join gender g on g.gender\_id=mc.gender\_id;
6. Select country\_name , count(c.country\_id) as number\_of\_movies\_produced from movie m inner join production\_country pc on m.movie\_id =pc.movie\_id inner join country c on c.country\_id = pc.country\_id group by c.country\_id;
7. Select \* from genre;
8. Select language\_name , count(l.language\_id) as number\_of\_movies from language l inner join movie\_languages ml on l.language\_id = ml.language\_id group by l.language\_id;
9. Select m.movie\_id, mcast.cast\_count,mcrew.crew\_count from movie m inner join (select movie\_id, count(movie\_id) as cast\_count from movie\_cast group by movie\_id) mcast on m.movie\_id=mcast.movie\_id inner join (select movie\_id , count(movie\_id) as crew\_count from movie\_crew group by movie\_id) mcrew on mcrew.movie\_id = m .movie\_id ;
10. Select title from movie order by popularity desc limit 10;
11. Select title, revenue from movie order by revenue desc limit 2,1;
12. Select title from movie where movie\_status = ‘rumoured’;
13. Select title, revenue from movie m inner join production\_country pc on pc.movie\_id = m.movie\_id inner join country c on c.country\_id = pc. country\_id where country\_name= ‘United State of America’;
14. Select m.movie\_id, pc.company\_name from movie m inner join movie\_company mc on mc.movie\_id = m.movie\_id inner join production\_company pc on pc.company\_id =mc.company\_id;
15. Select title from movie order by budget desc limit 10;

**Machine Learning**

1. **RSS (**residual sum of squares) only gives the estimation of total variation given by the model. It does not gives a measure to what extent the line fits our data. In order to get this measure of goodness of fit ,R2 is used . So if R2 is more we say that our regression line fitted our model closely and vice versa if R2 is low. R2 make use of RSS in its equation to get measure of goodness of fit.
2. **TSS (total sum of squares)**is the squared sum of difference between the actual value and average value of target variable.

**TSS=** = ∑(yi – Ӯ)2 , here yi=Actual, Ӯ average value of y.(also known as variance of target)

**RSS** is the residual sum of squares taken to get the variance of the target value around the best fit line.

**RSS=** ∑(yi – ŷ)2. here yi=actual ,ŷ predicted value

**ESS** explained sum of squares is the squares of deviation of predicted values from the mean value.

ESS= ∑( ŷ - Ӯ)2

R^2 = 1 - RSS / SST = 1- RSS/ (ESS+RSS)

1. Regularization reduces the error by fitting the function appropriately on the dataset and reduces overfitting problem. Two main regularization techniques are L1 and L2 regularization techniques in machine learning.
2. **Gini** impurity index is used an impurity measure which helps in making decision and build the Decision Tree. It is an alternative to Information gain.

It can also be defined as probability of a variable being wrongly classified when it is chosen randomly.

1. Unregularized decision tree may be grown to full depth since the depth will not defined then and hence can lead to overfitting problem.
2. Ensemble techniques are the techiques which make use of multiple model and then combine them to get improved results. Ensemble technique usually produces more accurate results than single model. In ML there are two types of ensemble techniques Boosting and Bagging.
3. **Bagging**: Bagging is also known as bootstrap aggregating and works on majority vote principle. Different Random sample of data set is used by each estimators to predict and validate the predictions. Each sample fed to n-estimators is different. Finally the majority output of all n-estimators is considered as final output.

**Boosting**: This technique uses concept of correcting previous classifier mistakes.

Each classifier gets trained on the sample set and learns to predict .The misclassification errors are then fed to the next classifier in chain to correct the mistakes till the final model predicts accurate results

1. Out of bag error is the measure of prediction error of decision tree in random forest models. It is obtained by averaging the number of errors for out of bag samples in each estimator.
2. K in cross validation refers to the number of groups the data needs to be split into. Cross validation is a technique to evaluate machine learning models. The performance of each group is noted and finally the mean of all metric outcomes is taken as the final score of the model. This techniques helps to check and remove overfitting problem in the models.
3. Hyperparamter tuning is used to get the best set of parameters of any algorithm for the particular data being used. If no hyperparameter tuning is done then algorithm chooses default parameter for all type of data which of course might not be giving the best possible outcome of the model. In short the hyperparameter tuning brings out the best possible version of model by giving the right parameters for the data being used.
4. Large learning rate may introduce the issue of fast jumping of the derivates such that it did not reach the global minima. It is advised to choose the learning rate appropriately so that the gradient descent operation can be smooth and global minima point can be obtained easily. Large learning rate causes problem for the derivate to converge on the global minima.
5. Logistic regression is meant to be used in linear data problems where the classes are linearly separable by a line (plane or hyperplane). Non linear data will not give a linear decision boundary and hence errors will be more and more prediction will be incorrect.

In logistic regression the output is taken as **sigmoid(best fit line)** OR **sigmoid(mx+c).**

Since sigmoid function is used , there need to be a threshold value for x after which sig(mx+c) results to 1 else 0. This threshold value cannot be defined in case of non linear data. Even if it is defined we will get most wrong predictions.

1. Both Ada Boost and gradient boosting converts set of weak learners into a single strong learner. The difference is in the way they create weak learners during the iterative process.

Ada Boost changes the sample distribution by modifying the weights attached to each instances. It increases the weights of the incorrect instances and decreases the weight of correctly predicted instances. This was the weak learners will focus more on the difficult instances . After getting trained the weak learners gets added to strong learner based on its performance. The higher it performs the more it contributes to strong learner.

Gradient Boost do not change the sample distribution instead the weak learners get trained on only the remaining errors (incorrect predictions) made by the strong learner.

At each iteration the remaining errors are computed and a weak learner is fitted to these remaining errors. Finally contribution of weak learner to strong learner is not decided by its performance (as in Ada grad) instead a gradient optimization process is used for this purpose.

1. Bias error occurs when model undergoes underfitting . Due to underfitting ,model try to make assumptions about the target function. Hence model gets biased.

**Low Bias:** Less assumptions are made about the form of target function

**High bias:** More assumptions are made about the form of target function

Variance error is the amount that the estimated target function by the model will change with respect to new data.

Low variance: small changes to the estimated target function on occurrence of new data

High variance: large changes to the estimated target function on occurrence of new data

To get good prediction results a model should have low bias and low variance. Process of achieving this low bias and low variance by changing or choosing right parameters is known as **bias variance Trade off.**

1. Kernels are mathematical functions used in SVM to manipulate the training data. Data which has non linear decision boundary is manipulated to in a way to get linear decision boundary. This transformation is done by mapping the original data to a higher dimensional space. There are different ways of doing this transformation which are given by **linear , RBF, poly.**

**Linear-** Used for linear model where data is linearly separable.

**Polynomial-** Decision boundary is of a polynomial function like   


**RBF-** RBF kernel is a function whose value depends on the distance from the origin or from some point.



||X1 — X2 || = Euclidean distance between X1 & X2

γ : Gamma (used only for RBF kernel)  *As the value of ‘* γ*’ increases the model gets overfits.* *As the value of ‘* γ*’ decreases the model underfits.*