STATISTICS WORKSHEET-5

Q1 to Q10 are MCQs with only one correct answer. Choose the correct option.

1. Using a goodness of fit, we can assess whether a set of obtained frequencies differ from a set of frequencies.

a) Mean b) Actual
c) Predicted **d) Expected**

2. Chisquare is used to analyse

a) Score b) Rank
c) Frequencies **d) All of these**

3. What is the mean of a Chi Square distribution with 6 degrees of freedom?

a) 4 b) 12
c) 6 d) 8

4. Which of these distributions is used for a goodness of fit testing?

a) Normal distribution **b) Chisqared distribution**
c) Gamma distribution d) Poission distribution

5. Which of the following distributions is Continuous

a) Binomial Distribution b) Hypergeometric Distribution
c) F Distribution d) Poisson Distribution

6. A statement made about a population for testing purpose is called?

a) Statistic **b) Hypothesis**
c) Level of Significance d) TestStatistic

7. If the assumed hypothesis is tested for rejection considering it to be true is called?

a) Null Hypothesis b) Statistical Hypothesis
c) Simple Hypothesis d) Composite Hypothesis

8. If the Critical region is evenly distributed then the test is referred as?

a) Two tailed b) One tailed
c) Three tailed d) Zero tailed

9. Alternative Hypothesis is also called as?

a) Composite hypothesis **b) Research Hypothesis**
c) Simple Hypothesis d) Null Hypothesis

10. In a Binomial Distribution, if 'n' is the number of trials and 'p' is the probability of success, then the mean value is given by

a) np b)n

MACHINE LEARNING

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?

R-squared is a better measure of goodness of fit model in regression because it measures the variance of the data that is explained by the model. R-squared values range from 0 to 1, with higher values indicating a better fit.

Residual sum of squares (RSS) is a measure of the difference between the actual data and the predicted data, and is used to assess the accuracy of a model.

RSS values can be any number, with lower values indicating a better fit.

R-squared is a better measure of goodness of fit model in regression because it measures the variance of the data that is explained by the model. R-squared values range from 0 to 1, with higher values indicating a better fit. Residual sum of squares (RSS) is a measure of the difference between the actual data and the predicted data, and is used to assess the accuracy of a model. RSS values can be any number, with lower values indicating a better fit.

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.

TSS (Total Sum of Squares) is the sum of the squared differences between each observation in a dataset and the mean of the dataset. It is a measure of the total variation in the dataset.

ESS (Explained Sum of Squares) is the sum of the squared differences between each observation in the dataset and the corresponding regression line. It is a measure of the variation explained by the regression model.

RSS (Residual Sum of Squares) is the sum of the squared differences between each observation in the dataset and the corresponding predicted value from the regression line. It is a measure of the variation not explained by the regression model.

The equation relating these three metrics is: TSS = ESS + RSS

3. What is the need of regularization in machine learning?

NEED OF REGULARIZATION:

1. Regularization is a technique used to avoid overfitting in machine learning models.
2. It works by introducing additional information in order to prevent the model from learning too much from the training data.
3. Regularization can help prevent a model from overfitting by adding a penalty to the loss function that penalizes complex models and encourages simpler models.
4. Regularization also helps improve the generalization ability of the model, which is important for making accurate predictions on new unseen data.

4. What is Gini–impurity index?

Gini-impurity index is a measure of the impurity of a node in a decision tree. It is calculated by subtracting the sum of the squared probability of each class from one. The resulting value is often used as a measure of how “pure” a node is. A lower Gini-impurity index means that a node is more likely to contain instances of a single class, and therefore is more “pure”.

Gini Impurity

Used by the CART (classification and regression tree) algorithm for classification trees, Gini impurity is a measure of how often a randomly chosen element from the set would be incorrectly labeled if it was randomly labeled according to the distribution of labels in the subset.

Mathematically, we can write Gini Impurity as following

$$I_{Gini} = 1 - \sum_{i=1}^{j} p_i^2 \qquad (3)$$

where j is the number of classes present in the node and p is the distribution of the class in the node.

5. Are unregularized decision-trees prone to overfitting? If yes, why?

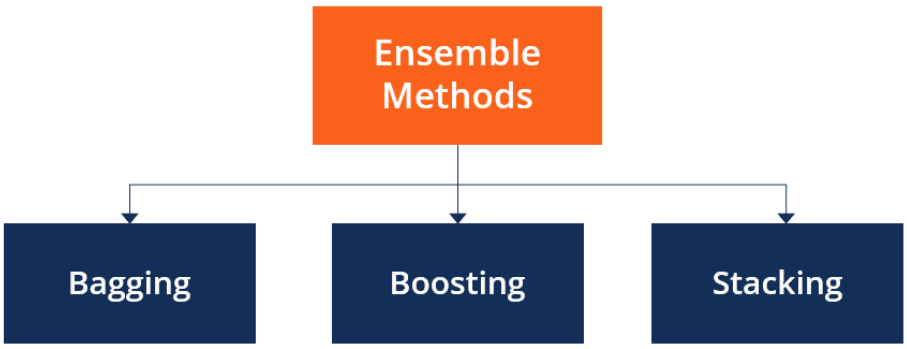
Decision trees are prone to overfitting, especially when a tree is particularly deep. This is due to the amount of specificity we look at leading to smaller sample of events that meet the previous assumptions. This small sample could lead to unsound conclusions. An example of this could be predicting if the Boston Celtics will beat the Miami Heat in tonight's basketball game. The first level of the tree could ask if the Celtics are playing home or away. The second level might ask if the Celtics have a higher win percentage than their opponent, in this case the Heat. The third level asks if the Celtic's leading scorer is playing? The fourth level asks if the Celtic's second leading scorer is playing. The fifth level asks if the Celtics are traveling back to the east coast from 3 or more consecutive road games on the west coast. While all of these questions may be relevant, there may only be two previous games where the conditions of tonights game were met. Using only two games as the basis for our classification would not be adequate for an informed decision. One way to combat this issue is by setting a max depth. This will limit our risk of overfitting; but as always, this will be at the expense of error due to bias. Thus if we set a max depth of three, we would only ask if the game is home or away, do the Celtics have a higher winning percentage than their opponent, and is their leading scorer playing. This is a simpler model with less variance sample to sample but ultimately will not be a strong predictive model.

Ideally, we would like to minimize both error due to bias and error due to variance. Enter random forests. Random forests mitigate this problem well. A random forest is simply a collection of decision trees whose results are aggregated into one final result. Their ability to limit overfitting without substantially increasing error due to bias is why they are such powerful models.

One way Random Forests reduce variance is by training on different samples of the data. A second way is by using a random subset of features. This means if we have 30 features, random forests will only use a certain number of those features in each model, say five. Unfortunately, we have omitted 25 features that could be useful. But as stated, a random forest is a collection of decision trees. Thus, in each tree we can utilize five random features. If we use many trees in our forest, eventually many or all of our features will have been included. This inclusion of many features will help limit our error due to bias and error due to variance. If features weren't chosen randomly, base trees in our forest could become highly correlated. This is because a few features could be particularly predictive and thus, the same features would be chosen in many of the base trees. If many of these trees included the same features we would not be combating error due to variance.

6. What is an ensemble technique in machine learning?

Ensemble methods are techniques that aim at improving the accuracy of results in models by combining multiple models instead of using a single model. The combined models increase the accuracy of the results significantly. This has boosted the popularity of ensemble methods in machine learning.



Main Types of Ensemble Methods

1. Bagging

Bagging, the short form for bootstrap aggregating, is mainly applied in classification and regression. It increases the accuracy of models through decision trees, which reduces variance to a large extent. The reduction of variance increases accuracy, eliminating overfitting, which is a challenge to many predictive models.

Bagging is classified into two types, i.e., bootstrapping and aggregation. **Bootstrapping** is a sampling technique where samples are derived from the whole population (set) using the replacement procedure. The sampling with replacement method helps make the selection procedure randomized. The base learning algorithm is run on the samples to complete the procedure.

Aggregation in bagging is done to incorporate all possible outcomes of the prediction and randomize the outcome. Without aggregation, predictions will not be accurate because all

outcomes are not put into consideration. Therefore, the aggregation is based on the probability bootstrapping procedures or on the basis of all outcomes of the predictive models.

Bagging is advantageous since weak base learners are combined to form a single strong learner that is more stable than single learners. It also eliminates any variance, thereby reducing the overfitting of models. One limitation of bagging is that it is computationally expensive. Thus, it can lead to more bias in models when the proper procedure of bagging is ignored.

2. Boosting

Boosting is an ensemble technique that learns from previous predictor mistakes to make better predictions in the future. The technique combines several weak base learners to form one strong learner, thus significantly improving the predictability of models. Boosting works by arranging weak learners in a sequence, such that weak learners learn from the next learner in the sequence to create better predictive models.

Boosting takes many forms, including gradient boosting, Adaptive Boosting (AdaBoost), and XGBoost (Extreme Gradient Boosting). AdaBoost uses weak learners in the form of decision trees, which mostly include one split that is popularly known as decision stumps. AdaBoost's main decision stump comprises observations carrying similar weights.

Gradient boosting adds predictors sequentially to the ensemble, where preceding predictors correct their successors, thereby increasing the model's accuracy. New predictors are fit to counter the effects of errors in the previous predictors. The gradient of descent helps the gradient booster identify problems in learners' predictions and counter them accordingly.

XGBoost makes use of decision trees with boosted gradient, providing improved speed and performance. It relies heavily on the computational speed and the performance of the target model. Model training should follow a sequence, thus making the implementation of gradient boosted machines slow.

3. Stacking

Stacking, another ensemble method, is often referred to as stacked generalization. This technique works by allowing a training algorithm to ensemble several other similar learning algorithm predictions. Stacking has been successfully implemented in regression, density estimations, distance learning, and classifications. It can also be used to measure the error rate involved during bagging.

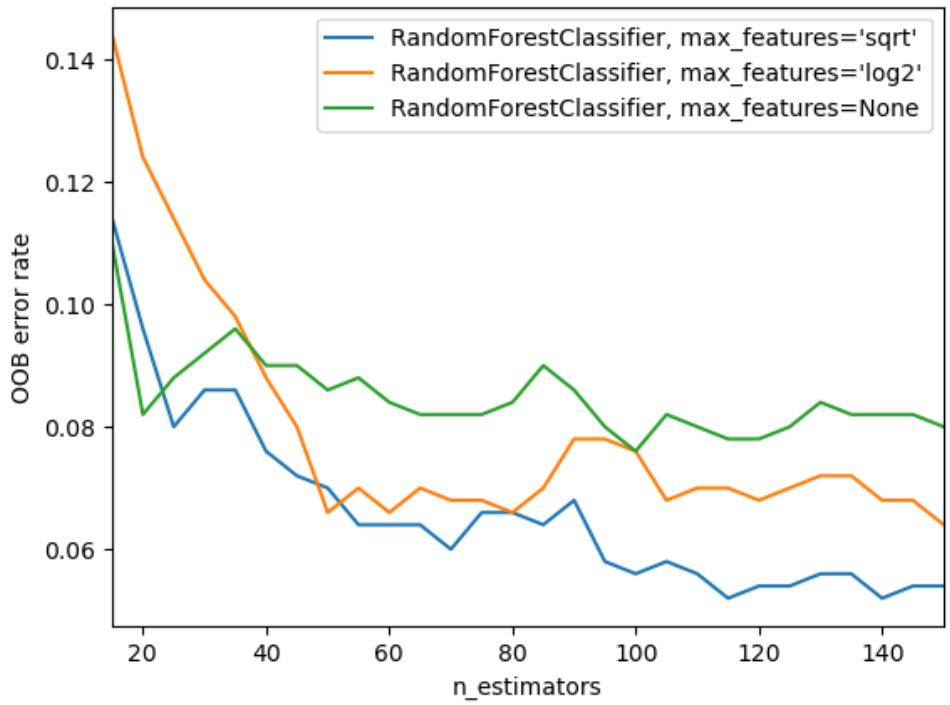
7. What is the difference between Bagging and Boosting techniques?

S.NO	Bagging	Boosting
1.	The simplest way of combining predictions that belong to the same type.	A way of combining predictions that belong to the different types.
2.	Aim to decrease variance, not bias.	Aim to decrease bias, not variance.
3.	Each model receives equal weight.	Models are weighted according to their performance.
4.	Each model is built independently.	New models are influenced by the performance of previously built models.
5.	Different training data subsets are selected using row sampling with replacement and random sampling methods from the entire training dataset.	Every new subset contains the elements that were misclassified by previous models.
6.	Bagging tries to solve the over-fitting problem.	Boosting tries to reduce bias.
7.	If the classifier is unstable (high variance), then apply bagging.	If the classifier is stable and simple (high bias) the apply boosting.
8.	In this base classifiers are trained parallelly.	In this base classifiers are trained sequentially.
9	Example: The Random forest model uses Bagging.	Example: The AdaBoost uses Boosting techniques

8. What is out-of-bag error in random forests?

The RandomForestClassifier is trained using *bootstrap aggregation*, where each new tree is fit from a bootstrap sample of the training observations $z_i=(x_i, y_i)$ The *out-of-bag* (OOB) error is the average error for each z_i calculated using predictions from the trees that do not contain z_i in their respective bootstrap sample. This allows the RandomForestClassifier to be fit and validated whilst being trained .

The example below demonstrates how the OOB error can be measured at the addition of each new tree during training. The resulting plot allows a practitioner to approximate a suitable value of n_estimators at which the error stabilizes.



9. What is K-fold cross-validation?

K-fold cross-validation

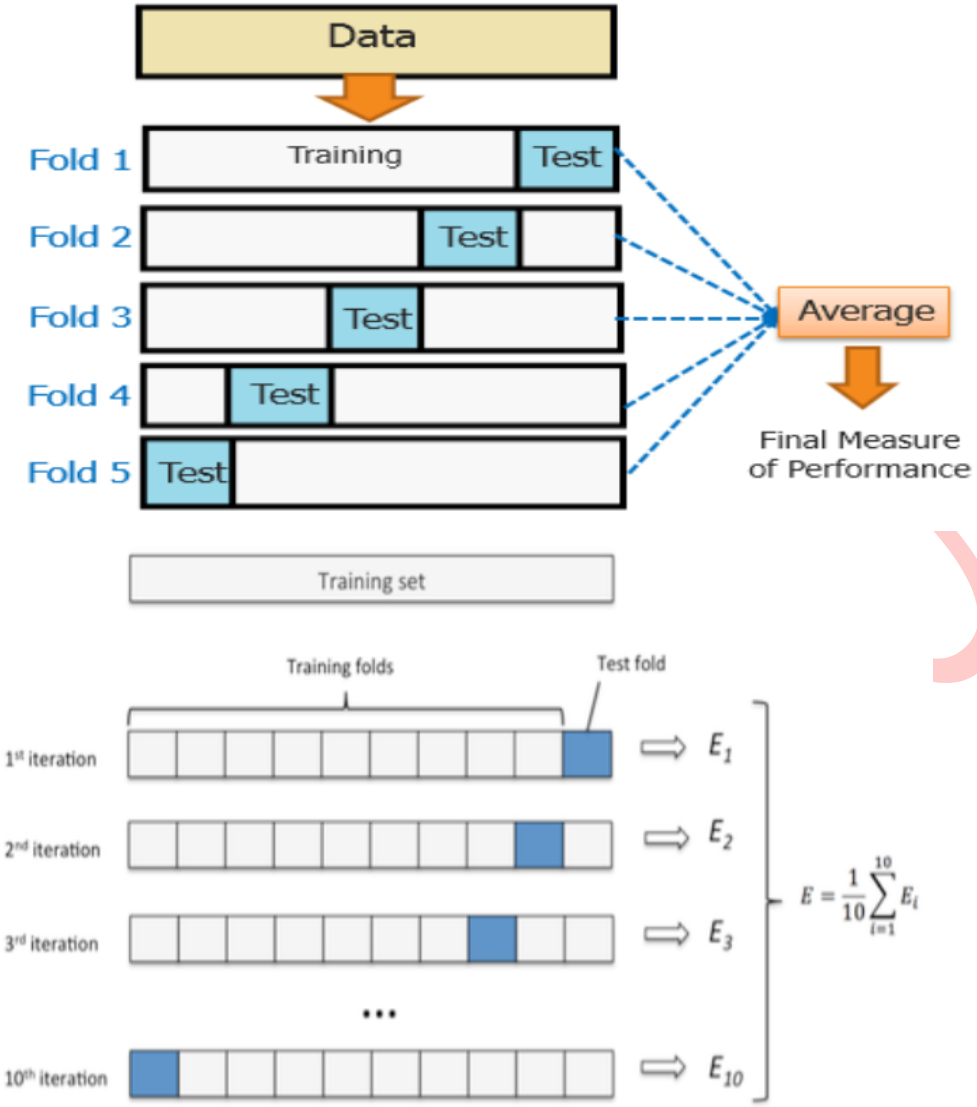
K-fold cross-validation randomly splits the training data into k subset called fold. For validation process we have to take care of following

1. We should train the model on large portion of the dataset. Otherwise we'll fail to read and recognised the underlying trend in the data. This will eventually result in higher bias.
2. We also need a good ration of testing data points. As we have seen above, less amount of the data points can lead a variance error while testing the effectiveness of the model.
3. We should iterate on the training and testing process multiple times. We should change the train and test dataset distribution. This help in validating the model effectiveness properly.

After taking care of this requirement, we have to follow following steps for k=fold validation

1. Randomly split your entire dataset into k folds.
2. For each k fold in dataset, build a model on k-1 folds of dataset. Then, test the model to check effectiveness of kth fold.
3. Record the error see on each prediction.
4. Repeat this until each of the k folds has searved as a test set
5. The average of your k recorded error is called cross validation error and it will seaved as performance metric for the model.

Visualization of k fold validation when k=5



10. What is hyper parameter tuning in machine learning and why it is done?

Hyperparameter tuning consists of finding a set of optimal hyperparameter values for a learning algorithm while applying this optimized algorithm to any data set. That combination of hyperparameters maximizes the model's performance, minimizing a predefined loss function to produce better results with fewer errors.

11. What issues can occur if we have a large learning rate in Gradient Descent?

1. Overshooting: A large learning rate can cause the model to overshoot the minimum and result in a poor convergence. 2. Oscillation: If the learning rate is too high, the algorithm may oscillate around the minimum instead of converging towards it. 3. Loss of Precision: A large learning rate can lead to a loss of precision, as the algorithm will take larger steps and miss out on important details. 4. Divergence: In extreme cases, a large learning rate can lead to divergence, where the algorithm never converges to a solution.

12. Can we use Logistic Regression for classification of Non-Linear Data? If not, why?

No, we cannot use logistic regression for classification of non-linear data because logistic regression is a linear classifier, meaning that it seeks to classify data points based on their

linear relationships. Non-linear data does not have linear relationships, so logistic regression is not an appropriate method for this type of data.

13. Differentiate between Adaboost and Gradient Boosting.

Features	Gradient boosting	Adaboost
Model	It identifies complex observations by huge residuals calculated in prior iterations	The shift is made by up-weighting the observations that are miscalculated prior
Trees	The trees with week learners are constructed using a greedy algorithm based on split points and purity scores. The trees are grown deeper with eight to thirty-two terminal nodes. The week learners should stay a week in terms of nodes, layers, leaf nodes, and splits	The trees are called decision stumps.
Classifier	The classifiers are weighted precisely and their prediction capacity is constrained to learning rate and increasing accuracy	Every classifier has different weight assumptions to its final prediction that depend on the performance.
Prediction	It develops a tree with help of previous classifier residuals by capturing variances in data. The final prediction depends on the maximum vote of the week learners and is weighted by its accuracy.	It gives values to classifiers by observing determined variance with data. Here all the week learners possess equal weight and it is usually fixed as the rate for learning which is too minimum in magnitude.
Short-comings	Here, the gradients themselves identify the shortcomings.	Maximum weighted data points are used to identify the shortcomings.
Loss value	Gradient boosting cut down the error components to provide clear explanations and its concepts are easier to adapt and understand	The exponential loss provides maximum weights for the samples which are fitted in worse conditions.
Applications	This method trains the learners and depends on reducing the loss functions of that week learner by training the residues of the model	Its focus on training the prior miscalculated observations and it alters the distribution of the dataset to enhance the weight on sample values which are hard for classification

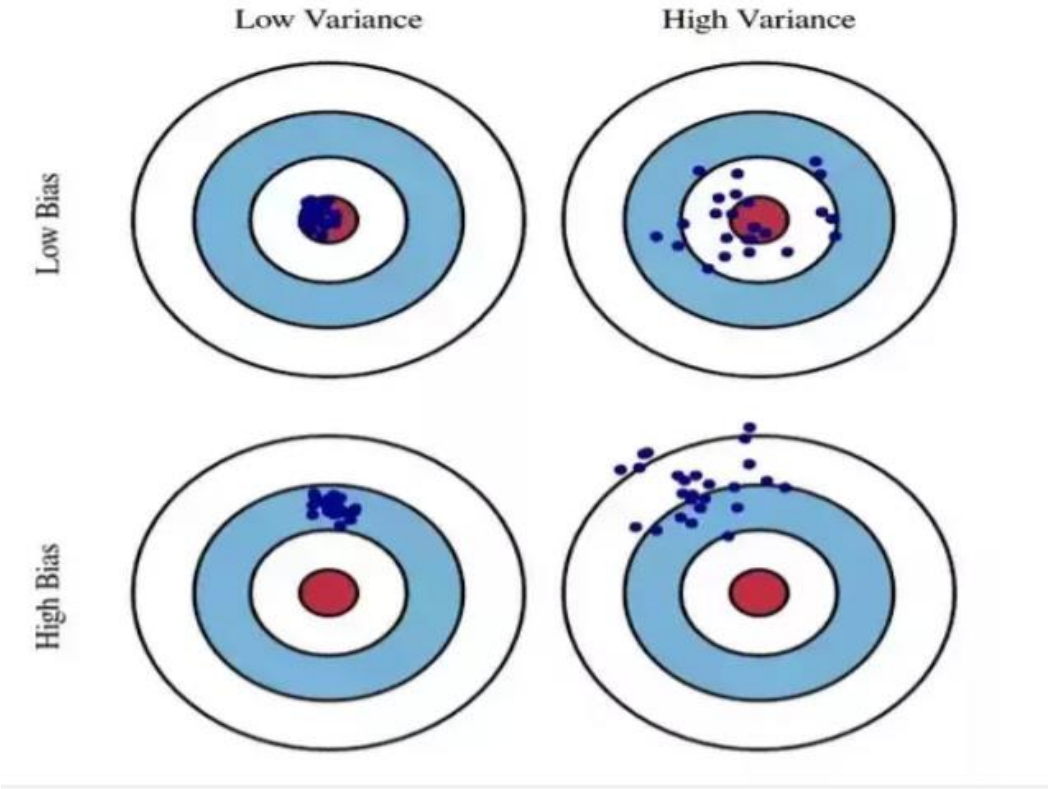
14. What is bias-variance trade off in machine learning?

BIAS /Variance Trade off: The bias is error from erroneous assumptions in the learning algorithm. High bias can cause an algorithm to miss the relevant relations between features and target outputs (underfitting).
The variance is error from sensitivity to small fluctuations in the training set. High variance can cause overfitting: modeling the random noise in the training data, rather than the intended outputs.

High bias is equivalent to aiming in the wrong place. High variance is equivalent to having an unsteady aim.
This can lead to the following scenarios:Low bias, low variance: Aiming at the target and hitting it with good precision.

- Low bias, high variance: Aiming at the target, but not hitting it consistently.
- High bias, low variance: Aiming off the target, but being consistent.
- High bias, high variance: Aiming off the target and being inconsistent.

This is an often used illustration:



15. Give short description each of Linear, RBF, Polynomial kernels used in SVM

Kernel: The main function of the kernel is to transform the given dataset input data into the required form. There are various types of functions such as linear, polynomial, and radial basis function (RBF). Polynomial and RBF are useful for non-linear hyperplane. Polynomial and RBF kernels compute the separation line in the higher dimension. In some of the applications, it is suggested to use a more complex kernel to separate the classes that are curved or nonlinear. This transformation can lead to more accurate classifiers.

SVM Kernels

The SVM algorithm is implemented in practice using a kernel. A kernel transforms an input data space into the required form. SVM uses a technique called the kernel trick. Here, the kernel takes a low-dimensional input space and transforms it into a higher dimensional space. In other words, you can say that it converts nonseparable problem to separable problems by adding more dimension to it. It is most useful in non-linear separation problem. Kernel trick helps you to build a more accurate classifier.

- **Linear Kernel** A linear kernel can be used as normal dot product any two given observations. The product between two vectors is the sum of the multiplication of each pair of input values.

```
K(x, xi) = sum(x * xi)
```

- **Polynomial Kernel** A polynomial kernel is a more generalized form of the linear kernel. The polynomial kernel can distinguish curved or nonlinear input space.

```
K(x,xi) = 1 + sum(x * xi)^d
```

Where d is the degree of the polynomial. d=1 is similar to the linear transformation. The degree needs to be manually specified in the learning algorithm.

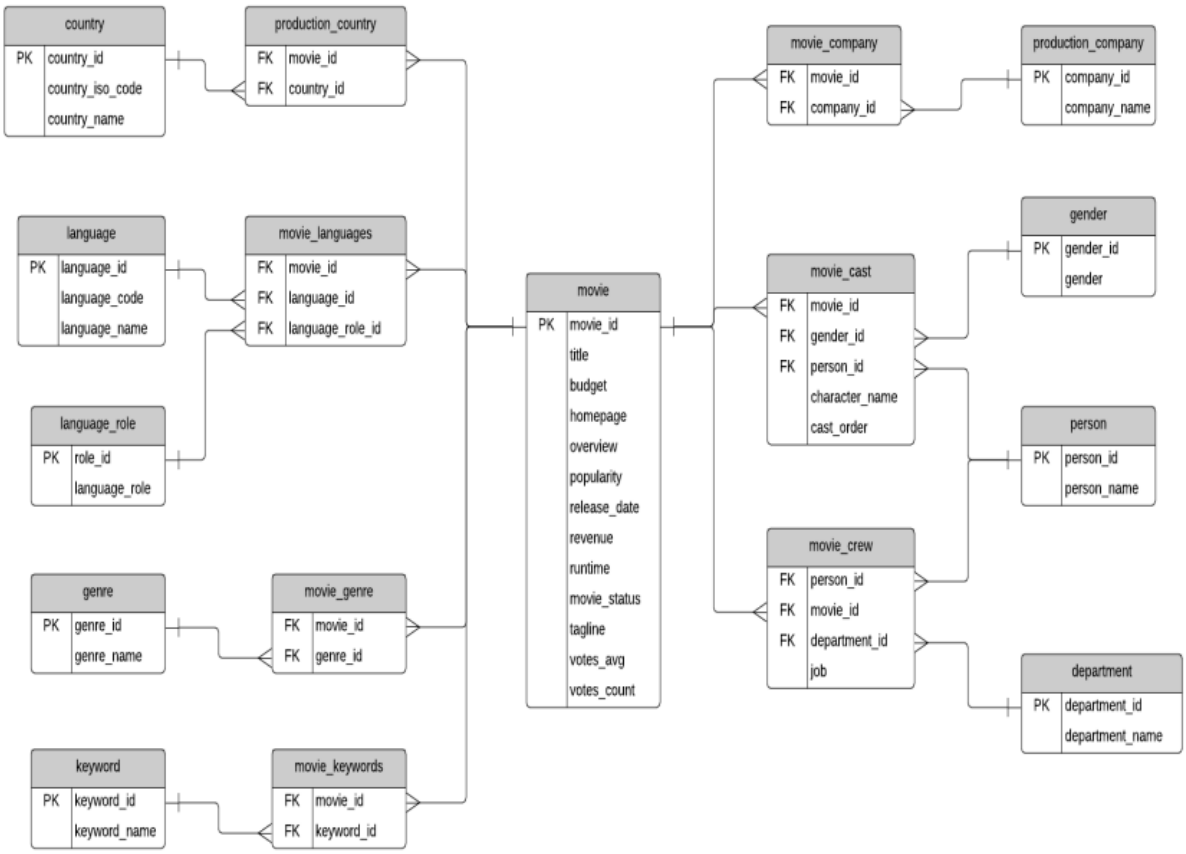
- **Radial Basis Function Kernel** The Radial basis function kernel is a popular kernel function commonly used in support vector machine classification. RBF can map an input space in infinite dimensional space.

K(x,xi) = exp(-gamma * sum((x – xi^2))

Here gamma is a parameter, which ranges from 0 to 1. A higher value of gamma will perfectly fit the training dataset, which causes over-fitting. Gamma=0.1 is considered to be a good default value. The value of gamma needs to be manually specified in the learning algorithm.

WORKSHEET 5 SQL

Refer the following ERD and answer all the questions in this worksheet. You have to write the queries using MySQL for the required Operation.



CREATE TABLE country (country_id INT PRIMARY KEY,
country_iso_code CHAR(2) NOT NULL,
country_name VARCHAR(255) NOT NULL);

CREATE TABLE production_country (movie_id INT NOT NULL,
country_id INT NOT NULL,
FOREIGN KEY (movie_id) REFERENCES movie(movie_id),
FOREIGN KEY (country_id) REFERENCES country(country_id));

CREATE TABLE language (language_id INT PRIMARY KEY,
language_name VARCHAR(255) NOT NULL,
language_code CHAR(2) NOT NULL);

CREATE TABLE movie_languages (movie_id INT NOT NULL,
language_id INT NOT NULL,
language_role_id INT NOT NULL,
FOREIGN KEY (movie_id) REFERENCES movie(movie_id),
FOREIGN KEY (language_id) REFERENCES language(language_id));

```
CREATE TABLE language_role (role_id INT PRIMARY KEY,
language_role VARCHAR(255) NOT NULL);

CREATE TABLE genre (genre_id INT PRIMARY KEY,
genre_name VARCHAR(255) NOT NULL);
CREATE TABLE movie_genre (movie_id INT NOT NULL,
genre_id INT NOT NULL,
FOREIGN KEY (movie_id) REFERENCES movie(movie_id),
FOREIGN KEY (genre_id) REFERENCES genre(genre_id));

CREATE TABLE keyword (keyword_id INT PRIMARY KEY,
keyword_name VARCHAR(255) NOT NULL);

CREATE TABLE movie_keywords (movie_id INT NOT NULL,
keyword_id INT NOT NULL,
FOREIGN KEY (movie_id) REFERENCES movie(movie_id),
FOREIGN KEY (keyword_id) REFERENCES keyword(keyword_id));

CREATE TABLE movie (movie_id INT PRIMARY KEY,
title VARCHAR(255) NOT NULL,
budget BIGINT,
homepage VARCHAR(255),
overview TEXT,
popularity FLOAT,
release_date DATE,
revenue BIGINT,
runtime INTEGER,
movie_status VARCHAR(255),
tagline VARCHAR(255),
votes_avg FLOAT,
votes_count INTEGER);

CREATE TABLE movie_company (movie_id INT NOT NULL,
company_id INT NOT NULL,
FOREIGN KEY (movie_id) REFERENCES movie(movie_id),
FOREIGN KEY (company_id) REFERENCES production_company(company_id));

CREATE TABLE production_company (company_id INT PRIMARY KEY,
company_name VARCHAR(255) NOT NULL);

CREATE TABLE gender (gender_id INT PRIMARY KEY,
gender VARCHAR(10) NOT NULL);

CREATE TABLE movie_cast (movie_id INT NOT NULL,
gender_id INT NOT NULL,
person_id INT NOT NULL,
character_name VARCHAR(255),
```

```
cast_order INTEGER,
FOREIGN KEY (movie_id) REFERENCES movie(movie_id),
FOREIGN KEY (gender_id) REFERENCES gender(gender_id),
FOREIGN KEY (person_id) REFERENCES person(person_id));

CREATE TABLE person (person_id INT PRIMARY KEY,
person_name VARCHAR(255) NOT NULL);

CREATE TABLE movie_crew (person_id INT NOT NULL,
movie_id INT NOT NULL,
department_id INT NOT NULL,
job VARCHAR(255),
FOREIGN KEY (person_id) REFERENCES person(person_id),
FOREIGN KEY (movie_id) REFERENCES movie(movie_id),
FOREIGN KEY (department_id) REFERENCES department(department_id));

CREATE TABLE department (department_id INT PRIMARY KEY,
department_name VARCHAR(255) NOT NULL);
```

1. **Write SQL query to show all the data in the Movie table**.
Select * from Movie

2. Write SQL query to show the title of the longest runtime movie.
SELECT title, runtime FROM Movie WHERE runtime = (SELECT MAX(runtime) FROM Movie);

3.Write SQL query to show the highest revenue generating movie title.
SELECT title, revenue FROM Movie WHERE revenue = (SELECT MAX(revenue) FROM Movie);

4.Write SQL query to show the movie title with maximum value of revenue/budget.
SELECT title, budget FROM Movie WHERE budget = (SELECT MAX(runtime) FROM Movie);

5. Write a SQL query to show the movie title and its cast details like name of the person, gender, character name, cast order.
Select title from movie join Movie_cast ;

6. Write a SQL query to show the country name where maximum number of movies has been produced, along with the number of movies produced.
7. Write a SQL query to show all the genre_id in one column and genre_name in second column.
Select genre_id, genre_name from genre;

8. Write a SQL query to show name of all the languages in one column and number of movies in that particular column in another column.
Select language_name from Language join Movie_Language on language_name.language_id=movie_id.language_id ;

9. Write a SQL query to show movie name in first column, no. of crew members in second column and number of cast members in third column.
Select movie_title, movie_crew , movie_cast from movie;
10. Write a SQL query to list top 10 movies title according to popularity column in decreasing order.
SELECT title FROM Movie ORDER BY popularity DESC limit10;

11. Write a SQL query to show the name of the 3rd most revenue generating movie and its revenue.
SELECT title FROM Movie ORDER BY revenue DESC limit1 OFFSET 2;
12. Write a SQL query to show the names of all the movies which have "rumoured" movie status.
SELECT title FROM Movie WHERE movie_status= "rumoured" ;

13. Write a SQL query to show the name of the "United States of America" produced movie which generated maximum revenue.
SELECT country_name, country_id FROM country JOIN production_country on movie_id = movie_id.country_id JOIN movie on movie_id WHERE revenue = (SELECT MAX(revenue) FROM Movie AND country_name='United States of America' FROM country);

14. Write a SQL query to print the movie_id in one column and name of the production company in the second column for all the movies.
Select movie_id from Movie_Componey join Production_Componey on movie_id.componey_id=componey_name.componey_id ;

15. Write a SQL query to show the title of top 20 movies arranged in decreasing order of their budget.
SELECT title FROM Movie ORDER BY budget DESC limit20