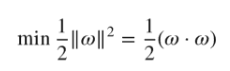
1. What is the definition of a target function? In the sense of a real-life example, express the target function. How is a target function's fitness assessed?

Ans: In machine learning, the target function is typically a function that maps input data to desired output labels. E.g example, in a classification task, the target function would map input data points (e.g., images of handwritten digits) to output class labels (e.g., 0, 1, 2, 3, 4, 5, 6, 7, 8, or 9). The function as defined by*f* is applied to the input (*I*) to produce the output (*I*), Therefore *O= f(I).*

For example, an SVM’s learning process takes a set of input data(x1, y1), (x2, y2)…(xk, yk)  
and finds a target function with a general form f(x) = (ω… x) + b.

Constraints are used during the SVM process to ensure the results are reasonable. For example, if you assume that φ(x) is a conversion function that coverts the sample data into a high-dimensional feature space, then the optimization for the algorithm can be represented by  


Target function's fitness is assessed :

* Target function is differentiable. This means that it should be smooth and not have any sharp discontinuities.
* Target function must be as simple as possible. This will make it easier for the model to optimize it and will also reduce the risk of overfitting.
* Target function is convex. This means that it should have a single global minimum or maximum value; if there are multiple local minima or maxima, it will be difficult for the model to find the global optimum.

1. What are predictive models, and how do they work? What are descriptive types, and how do you use them? Examples of both types of models should be provided. Distinguish between these two forms of models.

Ans: Predictive modelling is a statistical technique using machine learning to predict and forecast likely future outcomes with the aid of historical and existing data. It works by analysing current and historical data and projecting what it learns on a model generated to forecast likely outcomes. it is validated or revised regularly to incorporate changes in the underlying data.e.g

* Ecommerce businesses that use a customer’s browsing and purchasing history to make product recommendations.
* Financial organizations that need help determining whether a customer is likely to pay their credit card bill on time.
* Marketers who analyse data to determine the likelihood that new customers will respond favourably to a given campaign or product offering.
* Descriptive types: deals with organizing and summarizing data with the help of various charts/plots.e.g Annual revenue reports,Survey response summaries,Year-over-year sales reports

1. Describe the method of assessing a classification model's efficiency in detail. Describe the various measurement parameters.

Ans:

4 metrics **accuracy**, **precision**, **recall**, and the **F1 score are**  commonly used to evaluate classification models

* True Positives (TP) are instances where the classifier predicts a 1, and the true value is 1
* False Positives (FP) are instances where the classifier predicts a 1, and the true value is 0
* False Negatives (FN) are instances where the classifier predicts a 0, and the true value is 1
* True Negatives (TN) are instances and the classifier predicts a 0, where the true value is 0
* Accuracy measures the fraction of correctly classified samples. The following equation defines this value:

Accuracy=TP+FP/ TP+FP+ TN+ FN​    (1)

* Precision measures the ability of the classifier to correctly label positive values. The following equation defines this value:

Precision=TP/FP+TP​    (2)

* Recall measures the ability of the model to find all positive values. The following equation defines this value:

Recall=TP/FN+TP​

* The F1 score is a weighted average of the precision and recall metrics. The following equation defines this value:

F1= 2×Precision×Recall​/Precision+Recall    (4)

4.

i. In the sense of machine learning models, what is underfitting? What is the most common reason for underfitting?

Ans: A statistical model or a machine learning algorithm is said to have underfitting when it cannot capture the underlying trend of the data, i.e., it only performs well on training data but performs poorly on testing data. Underfitting destroys the accuracy of our machine learning model.

 Reasons for Underfitting:

1. High bias and low variance
2. The size of the training dataset used is not enough.
3. The model is too simple.
4. Training data is not cleaned and also contains noise in it.

ii. What does it mean to overfit? When is it going to happen?

Ans: A statistical model is said to be overfitted when the model does not make accurate predictions on testing data. When a model gets trained with so much data, it starts learning from the noise and inaccurate data entries in our data set. And when testing with test data results in High variance. Then the model does not categorize the data correctly, because of too many details and noise.

Reasons for Overfitting are as follows:

1. High variance and low bias
2. The model is too complex
3. The size of the training data

iii. In the sense of model fitting, explain the bias-variance trade-off.

Ans: If the algorithm is too simple (hypothesis with linear eq.) then it may be on high bias and low variance condition and thus is error-prone. If algorithms fit too complex ( hypothesis with high degree eq.) then it may be on high variance and low bias. In the latter condition, the new entries will not perform well,there is something between both of these conditions, known as Trade-off or Bias Variance Trade-off.

An algorithm can’t be more complex and less complex at the same time. For the graph, the perfect tradeoff will be like.

Chart, scatter chart

Description automatically generated  
The best fit will be given by hypothesis on the tradeoff point.

The error to complexity graph to show trade-off is given as –  
A picture containing diagram

Description automatically generated  
This is referred to as the best point chosen for the training of the algorithm which gives low error in training as well as testing data.

5.Is it possible to boost the efficiency of a learning model? If so, please clarify how.

#### Ans:yes,

#### Enrich training data with public datasets: generating new features from existing datasets, another idea is to get features from public datasets,

#### Feature selection and Use dimensionality reduction techniques: Remove irrelevant and noisy features could help reduce model training time and improve model performance.it prevents Overfitting,improves simplicity of ML and more computational Efficiency

#### Remove data leakage :

## Feature Scaling: will help improve the quality and predictive power of your model

## Tune hyperparameters: train your model against all hyperparameters in order to determine the model with optimal performance.

## Quantify random error:quantify how random error affects the predictive power of your model. This would help improve the reliability and quality of your model.

## Compare different algorithms: It is important to compare the predictive power of several different algorithms before selecting your final model.

6.How would you rate an unsupervised learning model's success? What are the most common success indicators for an unsupervised learning model?

Ans:Unsupervised learning models’s success is evaluated /indicated based on some similarity or dissimilarity measure such as the distance between cluster points. Commonly used evaluation metrics are:

* **Silhouette Coefficient:** The Silhouette Coefficient is defined for each sample and is composed of two scores:  
  a: The mean distance between a sample and all other points in the same cluster.  
  b: The mean distance between a sample and all other points in the next nearest cluster

Silhouette Coefficient

.

The Silhouette Coefficient for a set of samples is given as the mean of the Silhouette Coefficient for each sample. The score is bounded between -1 for incorrect clustering and +1 for highly dense clustering. Scores around zero indicate overlapping clusters. The score is higher when clusters are dense and well separated, which relates to a standard concept of a cluster.

* **Dunn’s Index:**Dunn’s Index (DI) is another metric for evaluating a clustering algorithm. Dunn’s Index is equal to the minimum inter-cluster distance divided by the maximum cluster size. Large inter-cluster distances (better separation) and smaller cluster sizes (more compact clusters) lead to a higher DI value. A higher DI implies better clustering. It assumes that better clustering means that clusters are compact and well-separated from other clusters.

7.Is it possible to use a classification model for numerical data or a regression model for categorical data with a classification model? Explain your answer.

Ans: yes, it is possible to use a classification model for numerical data or a regression model for categorical data with a classification model.

There are some ML models which use both categorical and numerical data

* Decision trees(with bagging),
* Random forest(with bagging & random subspace)
* Naive Bayes(numeric by Gaussian distribution or kernel density estimation)
* KNN based approach
* Ensemble Techniques
* linear regression

Logistic regression is used only for binary classification.Any supervised classification algorithm can be used when having categorical features by applying some encoding technique. That is a regression problem can be converted into a classification problem by simply **discretizing** the response variable into buckets.

8.Describe the predictive modeling method for numerical values. What distinguishes it from categorical predictive modeling?

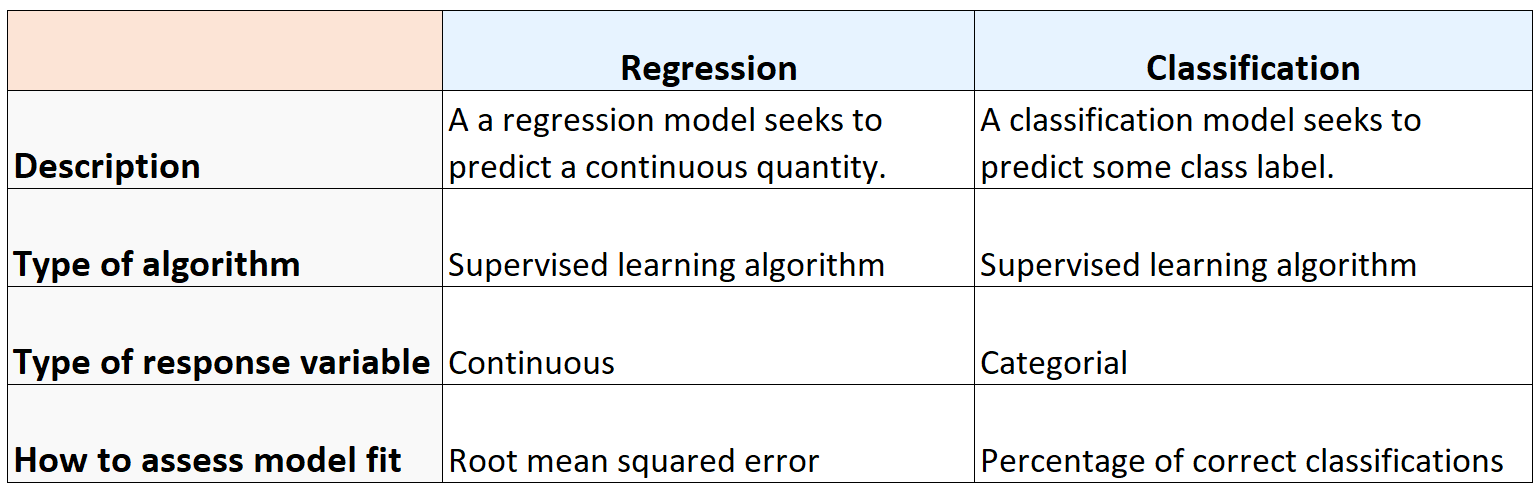
Ans: Predictive modeling is a mathematical and statistical method to predict future behavior or trends by analyzing the patterns of input data. Most commonly used predictive models for numerical values are Simple/Multiple linear regression and polynomial regression, support vector regressor, decision tree regressor.

For e.g if the relation ship between two or more variables is linear which is followed the equation of straight line then we use Simple/Multiple linear regression else for non linear relationship polynomial regression.

y=β₀+β₁x₁+…+βᵢxᵢ, β0=intercept, β1, β2=slopes

 β₀ is the y-intercept, the y-value when all explanatory variables are set to zero. β₁ to βᵢ are the coefficients for variables x₁ to xᵢ, the amount y increases or decreases with a one unit change in that variable

The classification model classifies the data sample into different categories or classes specified. Spam detection and fraud transaction detection are good examples of this category.



9. The following data were collected when using a classification model to predict the malignancy of a group of patients' tumors:

i. Accurate estimates – 15 cancerous, 75 benign

ii. Wrong predictions – 3 cancerous, 7 benign

Determine the model's error rate, Kappa value, sensitivity, precision, and F-measure.

Ans:

|  |  |  |
| --- | --- | --- |
| Actual  Predicted | (1) cancer | (0) bengin |
| (1) | TP 12 | FP 7 |
| (0) | FN 3 | TN 68 |

model's error rate = FP+FN/(TP+FN+FP+TP)=7+3/(15+75)=10/90=0.111

Kappa value=2[(TP\*TN)-(FN\*FP)]/{[(TP+FP)\*(FP+TN)]+(TP+FN)\*(FN+TN)

2(12\*68-3\*7)/[(12+7)\*(7+68)+(12+3)( 3+68)]=0.319277

Sensitivity /Recall(R)=TP/TP+FN=12/15 =0.8

Precision(P):TP/TP+FP=12/17=0.70588

F-measure=F1 score=2[(P\*R)/P+R] =1.129/1.5=0.7529

Kappa value=2(12\*68-3\*7)/[(12+7)\*(7+68)+(12+3)( 3+68)]=0.319277

10. Make quick notes on:

1. The process of holding out: The **hold-out method** for training the machine learning models is a technique that involves splitting the data into different sets: one set for training, and other sets for validation and testing. The hold-out method is used to check how well a machine learning model will perform on the new data

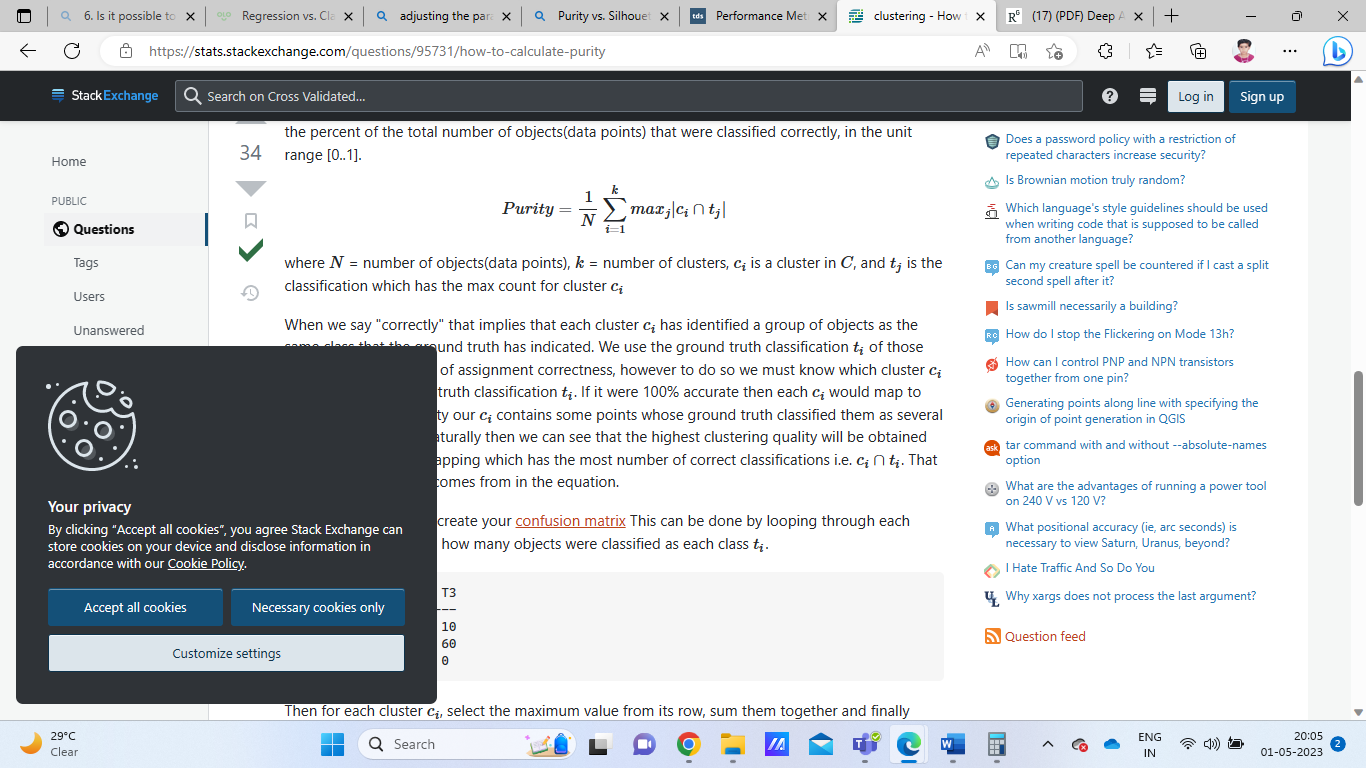
2. Cross-validation by tenfold: in Cross-validation by tenfold, the data-set is spitted into 10 number of subsets(known as folds) then training is perform on the all the subsets leaving one subset for the evaluation of the trained model. In this method, we iterate 10 times with a different subset reserved for testing purpose each time.

3. Adjusting the parameters: To improve the performance of the ML algorithm, adjusting the parameters or hyperparameter tuning play a vital role as their values control the learning process and determine the values of model parameters that a learning algorithm ends up learning.

11. Define the following terms:

1. Purity vs. Silhouette width:

Ans: Purity :It is the percent of the total number of objects(data points) that were classified correctly, in the unit range [0..1].



. where N = number of objects(data points), k = number of clusters, ci is a cluster in C, and tj is the classification which has the max count for cluster ci.

Purity is calculated first by creating  [confusion matrix](http://en.wikipedia.org/wiki/Confusion_matrix) This can be done by looping through each cluster ci and counting how many objects were classified as each class ti.

| T1 | T2 | T3

---------------------

C1 | 0 | 53 | 10

C2 | 0 | 1 | 60

C3 | 0 | 16 | 0

Then for each cluster ci, select the maximum value from its row, sum them together and finally divide by the total number of data points.

Purity = (53 + 60 + 16) / 140 = 0.92142

While The Silhouette Score is calculated using the mean intra-cluster distance (i) and the mean nearest-cluster distance (n) for each sample. The Silhouette Coefficient for a sample is (n - i) / max(i, n). n is the distance between each sample and the nearest cluster that the sample is not a part of while i is the mean distance within each cluster. The higher the Silhouette Coefficients (the closer to +1), the further away the cluster’s samples are from the neighbouring clusters samples. A value of 0 indicates that the sample is on or very close to the decision boundary between two neighbouring clusters. Negative values, instead, indicate that those samples might have been assigned to the wrong cluster.

2. Boosting vs. Bagging

### Differences Between Bagging and Boosting

| 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 |

1. The eager learner vs. the lazy learner

Ans:

* **Lazy learning adapts to new data quickly** as it’s during the query phase when it determines which objects to use for prediction. So, all that we need to do to use new data is to store them. In contrast, eager learning generalizes during the training phase. Incorporating the information from new data afterward requires retraining.
* **Lazy learning often results in simpler model**s since it inspects only a neighborhood of the query point, whereas eager learning may generate more complex models as it aims to find global patterns in the data.
* Lazy learning requires less training time as the algorithms focus on storing data so that they can be easily retrieved later. In contrast, **eager learning has a training phase in which it uses the entire training dataset**to construct a comprehensive model for prediction. **Lazy learning is slower at making predictions** since it searches through data to find the query object’s neighborhood, while eager learning has a faster query time as completed models are ready to use and usually produce predictions quickly.
* **Memory usage is higher in lazy learning** as it stores all data instances, whereas eager learning has lower memory usage since only the model is stored.
* **Lazy learning is well-suited for online learning** as it can easily update the stored data, while eager learning requires retraining the model, which can be time-consuming.
* **Lazy learning is less robust due to its sensitivity to noise** and reliance on local patterns, whereas eager learning is generally more robust as it finds global patterns and is less sensitive to noise.
* E.g of eager learning algorithm:[decision trees](https://www.baeldung.com/cs/decision-trees-vs-random-forests) ,.[Support vector machines (SVM)](https://www.baeldung.com/cs/ml-support-vector-machines)  [Naive Bayes](https://www.baeldung.com/cs/decision-tree-vs-naive-bayes) and [artificial neural networks (ANN)](https://www.baeldung.com/cs/neural-networks-neurons).
* E.g of Lazy learning algorithms: [k-nearest neighbors](https://www.baeldung.com/cs/k-nearest-neighbors) algorithm (kNN) ,[Radial Basis Function](https://www.baeldung.com/cs/ml-parametric-vs-non-parametric-models) (RBF)