1.How do you measure quality of clusters in DBSCAN?

There are two methods to measure quality of clusters in DBSCAN which are Silhouette Method and Visual Cluster Interpretation.

**Silhouette Method:** This method determines the separability of clusters. To begin, an average distance between each point and all other points in a cluster is calculated. The distance between each point and each point in other clusters is then calculated. We divide by whichever average is bigger after subtracting the two averages.

In the end, we desire a high (i.e., near to 1) score, which indicates a short intra-cluster average distance (tight clusters) and a large inter-cluster average distance (clusters well separated).

**Visual Cluster Interpretation:** It's important to understand each cluster when you've gotten your clusters. This is usually accomplished by merging the original dataset with the clusters and viewing each cluster separately. The more prominent and obvious each cluster is, the better.

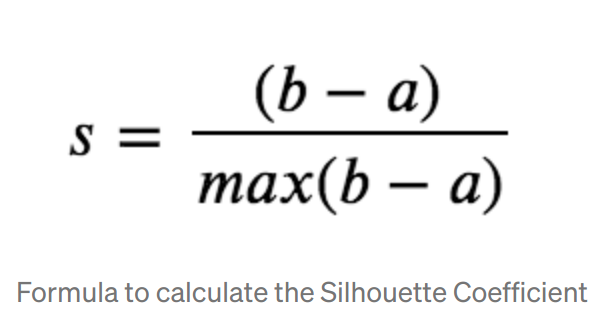
2. How do you evaluate DBSCAN algorithm?

We can evaluate DBSAN algorithm with Mean Silhouette Coefficient.

The Silhouette Coefficient has bounded a range of 1 to -1. 1 is the best value, and -1 is the worst. A higher score suggests that the model's clusters are more defined and denser. Negative values typically indicate that data points have been assigned to the wrong clusters. Values close to 0 imply overlapping clusters, whereas negative values usually indicate that data points have been assigned to the wrong clusters.

The silhouette coefficient is calculated using two scores:

* **a**: The average distance between one data point and the rest of the data points in the same cluster.
* **b**: The average distance between one data point and the next closest cluster's other points.



3. What do you understand by market basket analysis?

Market basket analysis is a data mining approach used by merchants to better understand customer purchase patterns and thereby enhance revenue. It entails evaluating huge data sets, such as purchase histories, to identify product groups and products that are likely to be bought together.

The introduction of electronic point-of-sale (POS) systems boosted the implementation of market basket analysis. The digital records generated by POS systems made it easier for apps to process and analyse massive volumes of purchase data when compared to handwritten records held by store owners.

4. Explain centroid formation technique in K Means algorithm.

A centroid is an imaginary or real location that represents the cluster's centre. By lowering the in-cluster sum of squares, each data point is assigned to one of the clusters.

To put it another way, the K-means algorithm finds k centroids and then assigns each data point to the closest cluster while keeping the centroids as small as possible.The average of the data, or determining the centroid, is what the ‘means' in K-means refers to.

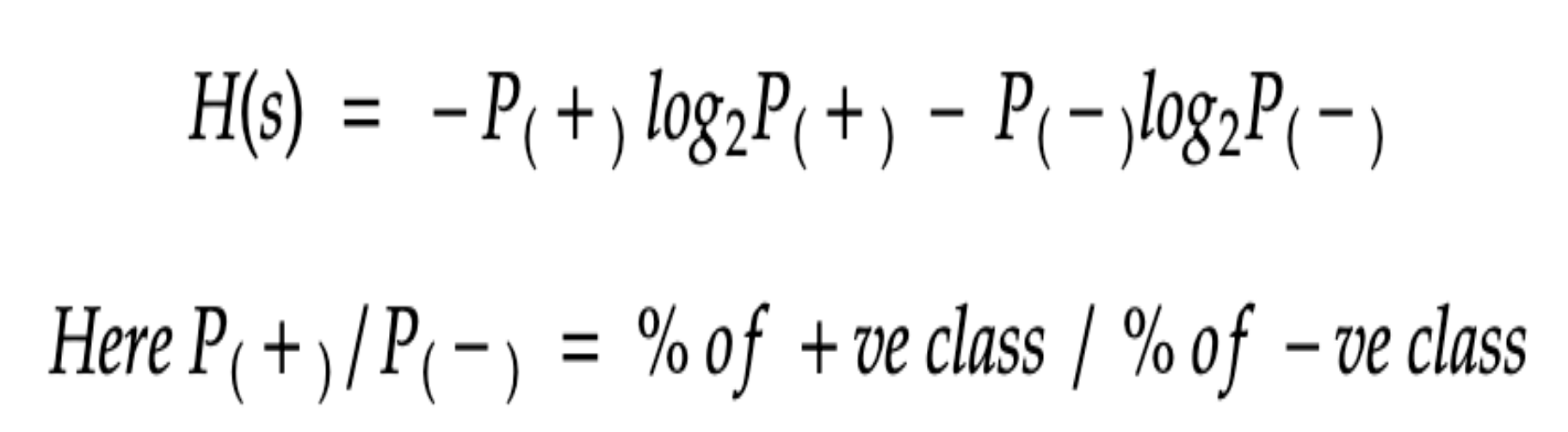
5. Have you ever used SVM regression in any of your project, If yes, why?

Yes, I had used in one of my projects because Support Vector Regression (SVR) recognises the existence of non-linearity in the data and offers a reliable prediction model.

6. Explain the concept of GINI Impurity.

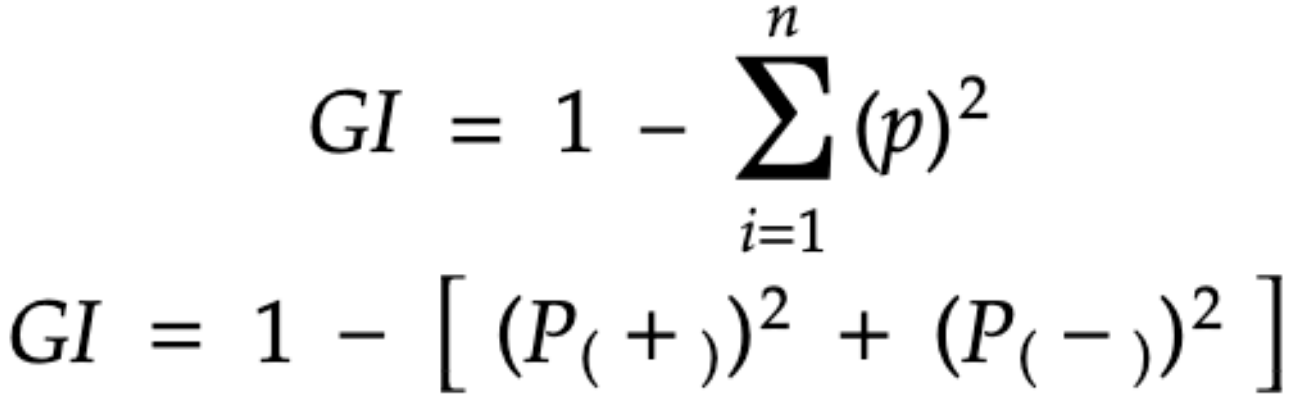
We need to understand Entropy first for better understanding of Gini Impurity.

**Entropy:** Entropy helps us to build an appropriate decision tree for selecting the best splitter. The entropy of a sub split can be defined as a measure of its purity. Entropy is always between 0 and 1. This formula can be used to compute the entropy of any split.



Gini Impurity: Gini Impurity is a measure of the likelihood of a new instance of a random variable being incorrectly categorised if it were randomly classified using the distribution of class labels from the data set.

If the data set comprises only one class, the Gini impurity is lower bounded by 0.



7. Let’s suppose I have given you dataset with 100 columns how you will be able to control growth of decision tree?

First, I'll apply a dimensionality reduction on the dataset by using PCA, Intractive binning (IB). Methods. Then applying Decision tree on it and check via tree chart.

In the decision tree chart, each internal node has a decision rule that splits the data. Gini referred to as the Gini ratio, which measures the impurity of the node. You can say a node is pure when all of its records belong to the same class, such nodes known as the leaf node.

8. If you are using Ada-boost algorithm & if it is giving you underfitted result What is the hyperparameter tuning you will do?

Will use different parameters in hyperparameter tuning which are base\_estimator, estimators, learning\_rate and random\_state. The variables we supply to a model before we start the modelling process are known as hyper-parameters. Let's have a look at them all.

**base estimator:** The ensemble's model; by default, a decision tree is used.

**n estimators:** The number of models that will be created.

**learning rate:** reduces each classifier's contribution by this amount.

**random state:** The seed for the random number generator, which ensures that the same random numbers are created each time.

Will do some twitching in these parameters and getting the good result.

9. Explain gradient boosting algorithm.

Gradient boosting is a sort of boosting used in machine learning. It is based on the assumption that when the best potential next model is coupled with prior models, the overall prediction error is minimised. To decrease error, the fundamental notion is to specify the target outcomes for the next model. How are the goals determined? The goal outcome for each case in the data is determined by how much modifying the prediction for that case affects the overall prediction error:

* If a slight change in a case's prediction results in a big reduction in error, the case's next target outcome is a high value. The error will be reduced if the new model's predictions are near to its targets.
* If a slight modification in a case's prediction generates no change in error, the case's next target result is zero. Changes to this prediction have no effect on the error.

10. Can we use PCA to reduce dimensionality of highly non-linear data.

No, PCA cannot handle non-linear data.

11. How do you evaluate performance of PCA.

Reconstruction error is one way to measure performance. Indeed, one way to think of PCA is that it reduces the amount of data on the training set. To project the points into a low-dimensional space, use PCA. Then, by projecting the low-dimensional representations back into the original, high-dimensional space, recreate the original points. The distance between the original points and their reconstructions is inversely proportional to the model's ability to capture the data's structure. This is related to PCA's reputation as a lossy data compression method. The original points can be recreated more precisely when the low-dimensional representation contains more information. The commonly used performance measure R2 can also be calculated using reconstruction error (fraction of variance accounted for).

12. Have you ever used multiple dimensionality techniques in any project? if yes, give reason. If no, where can we use it?

In One of my projects was to estimate churn using the enormous data set. The enormous dimensionality of this data set, with 15K data columns, is its unique feature. Most data mining techniques are implemented column-by-column, which makes them slower and slower as the number of data columns grows. The project's first milestone was to lower the number of columns in the data collection while sacrificing the least amount of information possible. So, I used PCA (Principal Component analysis) for dimensionality reduction.

13. What do you understand by curse of dimensionality explain.

When working with high-dimensional data, the "Curse of Dimensionality" refers to a set of issues. The number of attributes/features in a dataset corresponds to the dataset's dimension. High dimensional data is a dataset containing a large number of attributes, usually on the order of a hundred or more. Some of the challenges that come with high-dimensional data show up while analysing or displaying the data to look for trends, and others show up when training machine learning models. The ‘Curse of Dimensionality' refers to the difficulty in training machine learning models due to high dimensional data. ‘Data sparsity' and ‘distance concentration' are two well-known characteristics of the curse of dimensionality.

14. What is the difference between anomaly detection and novelty detection?

Anomaly detection (also known as outlier analysis) is a data mining step that detects data points, events, and/or observations that differ from the expected behaviour of a dataset. Atypical data might reveal significant situations, such as a technical fault, or prospective possibilities, such as a shift in consumer behaviour. Anomaly detection is increasingly automated thanks to machine learning.

Novelty detection, as the name implies, is the process of identifying new or uncommon data inside a dataset. Outliers, also known as anomalies, are frequently detected as a result of their variances from the rest of the data. However, novelty detection algorithms may need to be tweaked to look for groups or bursts of uncommon data rather than single incidences of odd data. Cluster analysis is an approach that is commonly used in bank fraud algorithms to monitor suspicious activity patterns.

15. Explain gaussian mixture model.

A Gaussian mixture model (GMM) is a type of probabilistic model in which all data points are generated from a mixture of finite Gaussian distributions with unknown parameters. The parameters for Gaussian mixture models are produced from a well-trained prior model using either maximum a posteriori estimation or an iterative expectation-maximization approach. When it comes to modelling data, especially data from multiple groups, Gaussian mixture models are quite effective.

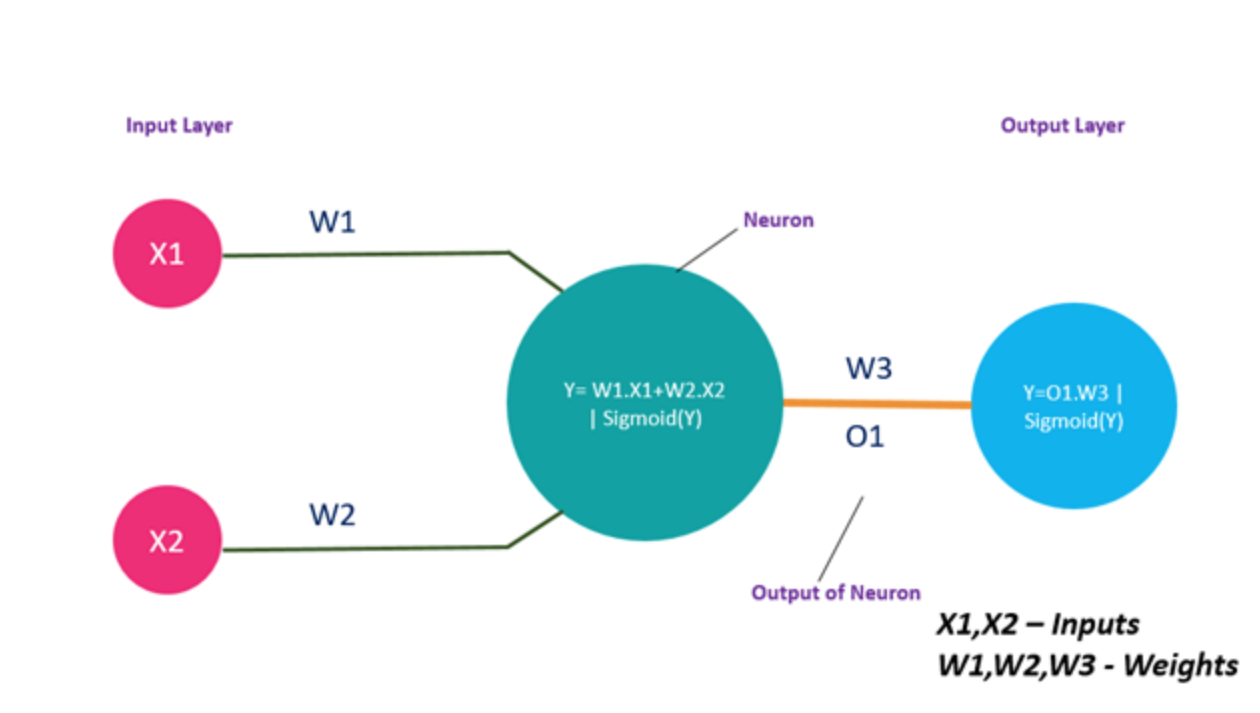
16. Explain neural network in terms of mathematical function.

Two principles govern the operation of a neural network.

* Forward Propagation
* Backward Propagation

Let's use an example to better comprehend these building blocks. To make the understanding obvious, I am considering a single input layer, hidden layer, and output layer.

Forward Propagation



1. Given that we have data, we'd like to use binary classification to obtain the desired result.
2. Consider a sample with features such as X1 and X2, which will be used to forecast the outcome using a series of processes.
3. Each feature is assigned a weight, with X1, X2 representing features and W1, W2 representing weights. These are fed into a neuron as input.
4. Both functions are carried out by a neuron. a) Activation b) Summation
5. All features are multiplied by their weights in the summing, and bias is totalled. (Y=W1X1+W2X2+b).
6. This summing function is used in conjunction with an Activation function. The output of this neuron is multiplied by the weight W3 and fed to the output layer as input.
7. Each neuron goes through the same procedure, although the activation functions in hidden layer neurons differ from those in the output layer.

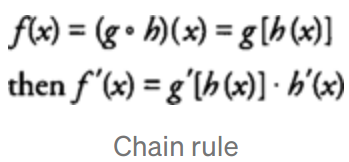
We just initialised the weights at random and carried on with the process. Initializing the weights can be done in a variety of ways. But you might be wondering how these weights are updated, right??? Back propagation will be used to answer this question.

Backward Propagation

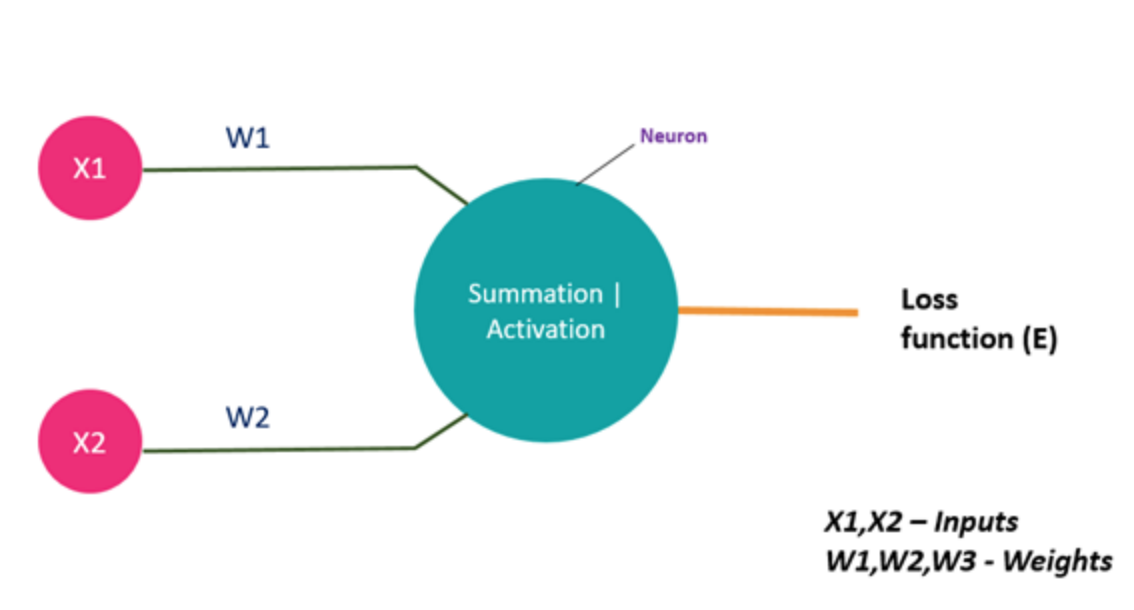
Let us return to our calculus basics, and we will update the weights using the **chain rule** that we learnt in school.

Chain Rule

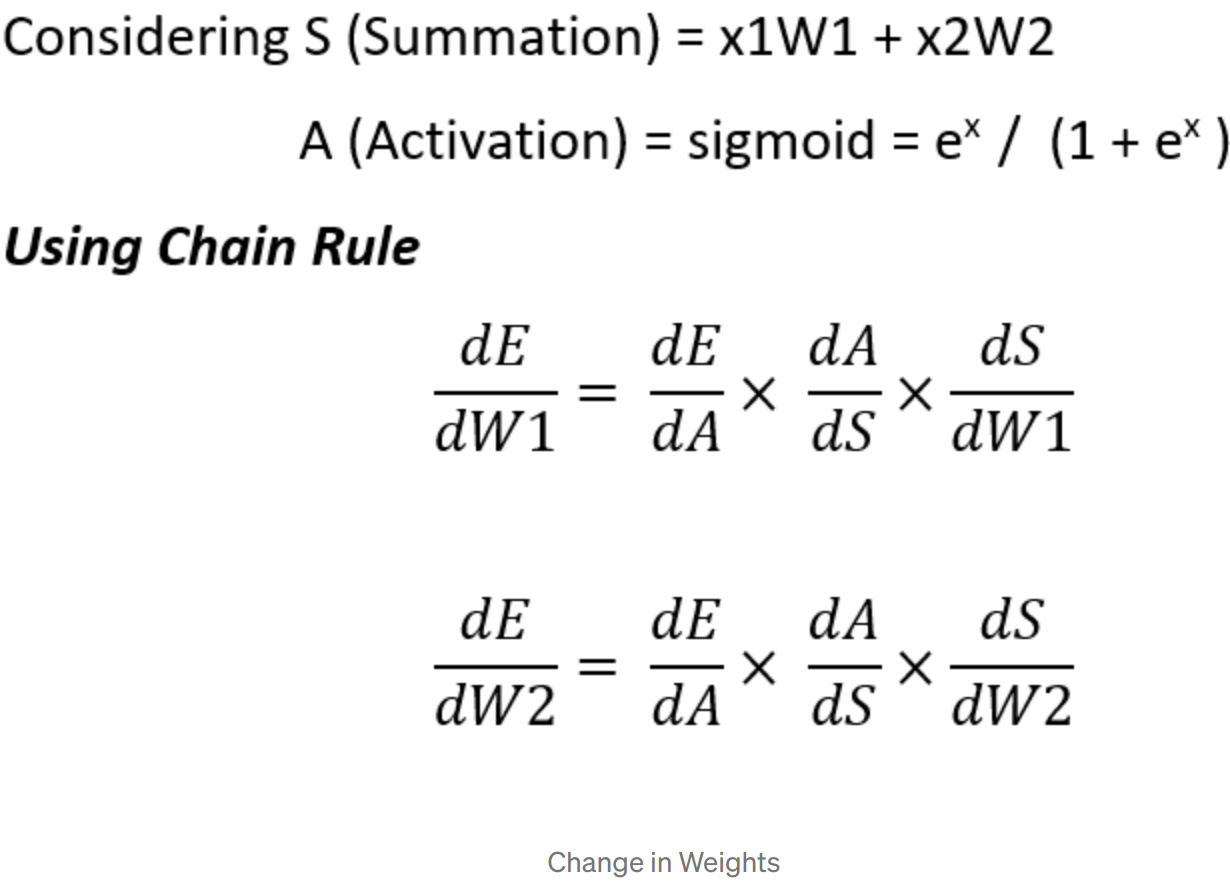
The chain rule is a method for computing the derivative of composite functions, with the number of functions in the composition influencing the number of differentiation steps required. If a composite function f(x) is defined as, for example,

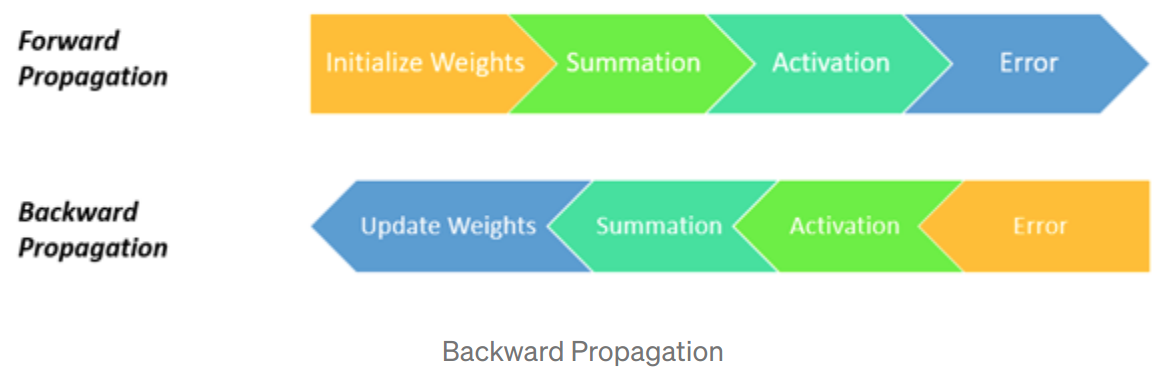


Let's take a single neuron and apply the chain rule to it.

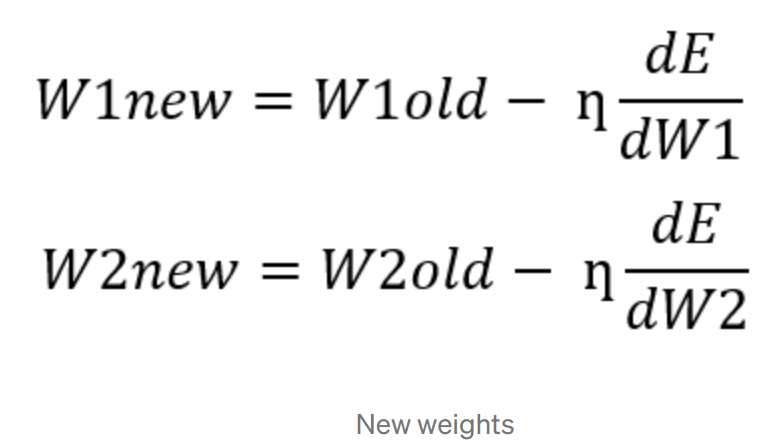


Our primary goal in neural networks will be to reduce error, which will necessitate updating all weights via backpropagation. We need to determine a change in weights that will result in the least amount of inaccuracy. To do so, we use the dE/dW1 and dE/dW2 formulas.





After you've calculated the changes in weights concern mistake, we'll use the gradient descent process to update the weights.

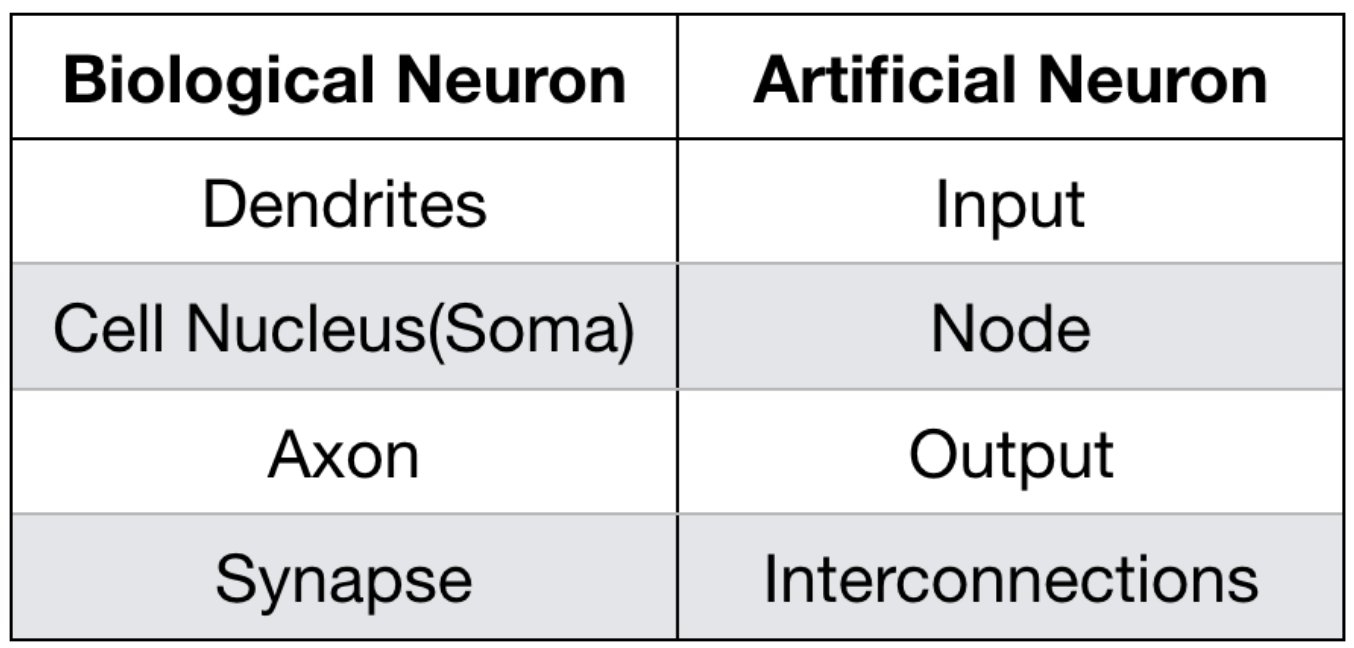


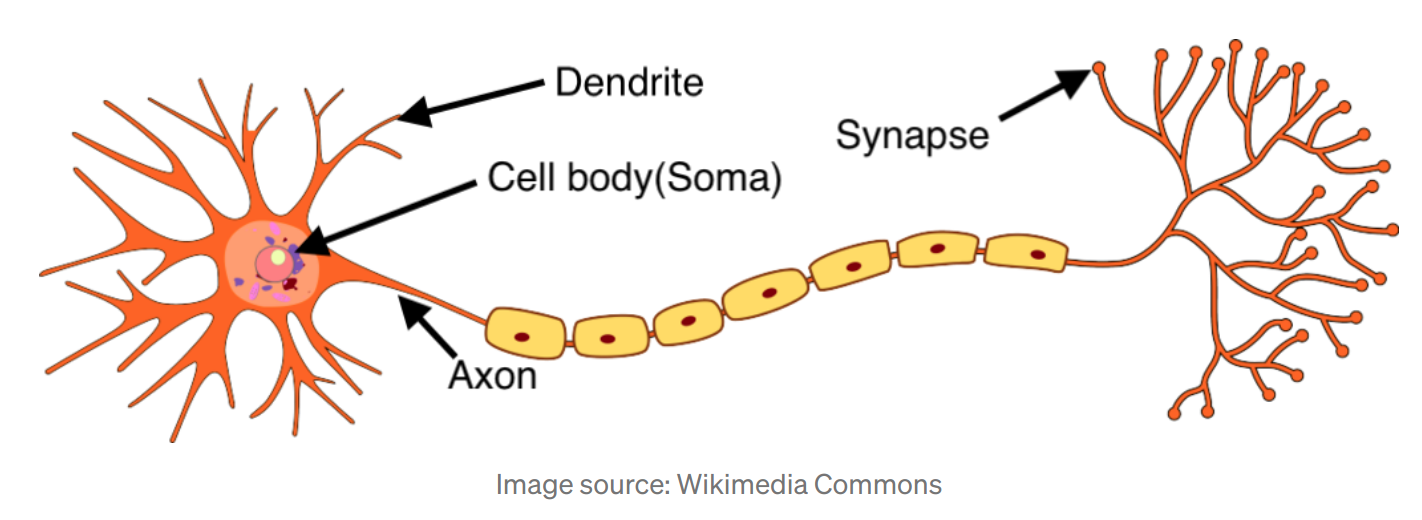
For all samples, forward and backward propagation will continue until the error hits a minimum value.

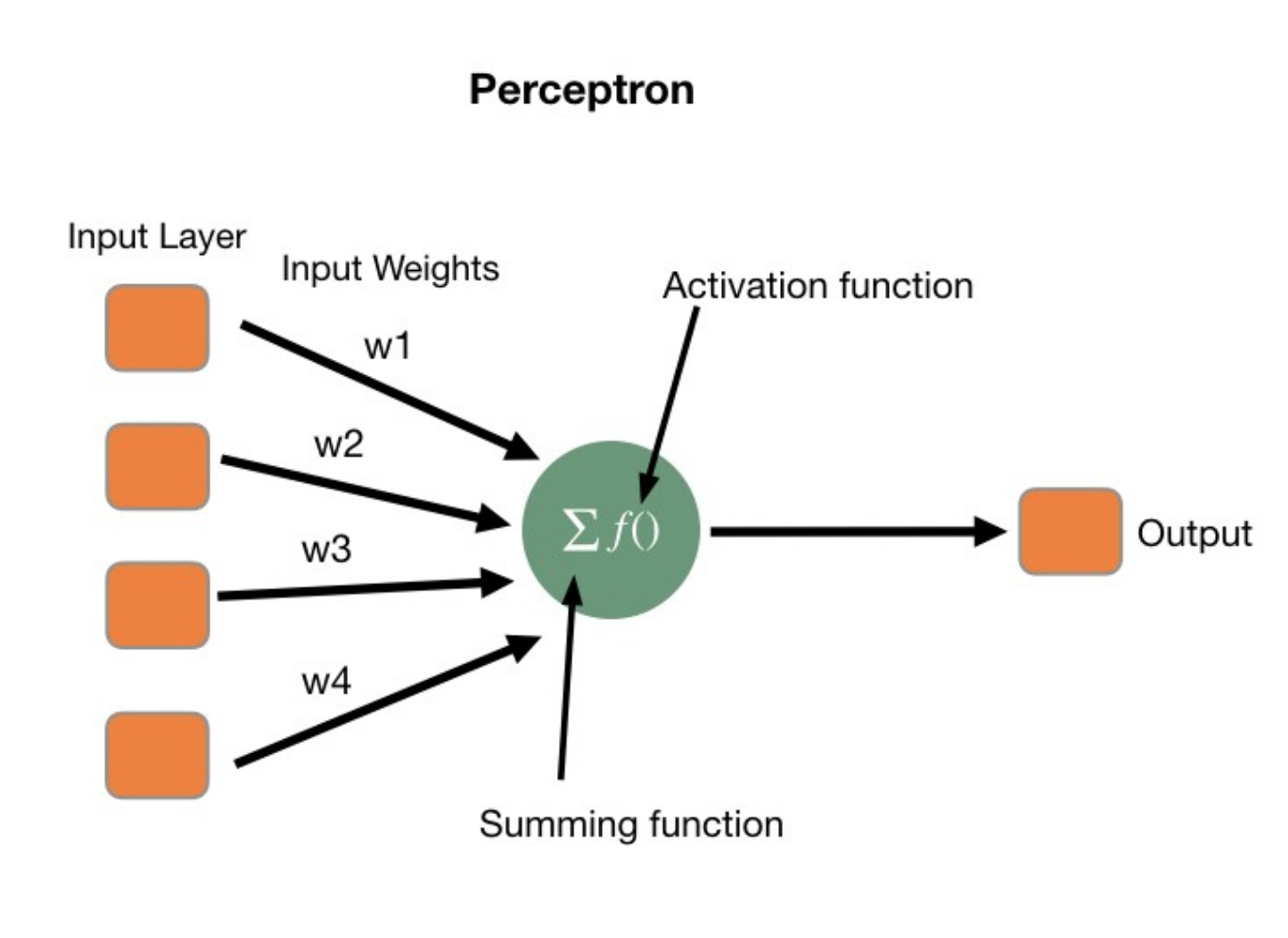
17. Can you please corelate a biological neuron and artificial neuron?

The complexity of biological neural networks far exceeds that of DNNs, making understanding the representations they learn even more difficult. As a result, both machine learning and computational neuroscience face a similar problem: how can we evaluate their representations to learn how they handle complex problems? We look at how computational neuroscientists' data-analysis concepts and methodologies may be applied to analysing representations in DNNs, and how recently discovered DNN-analysis approaches can be applied to understanding representations in biological neural networks.

By figure you can analyse the difference:





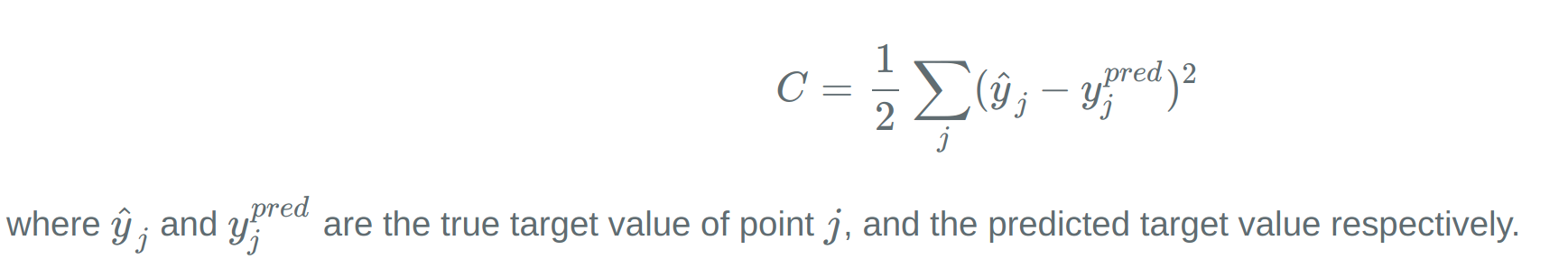


18. Give list of cost functions you heard of.

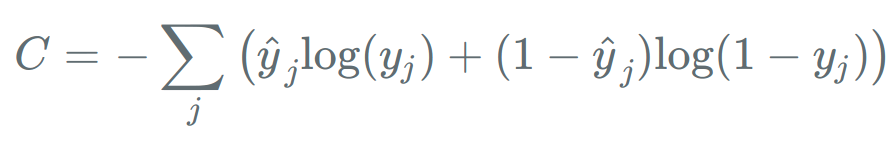
List of cost functions used in Neural Networks.

A cost function is a quantitative assessment of a model's fit quality: how well it reproduces the data. A cost function is a single number that represents the sum of the model's divergence from the true value for all points in the dataset.

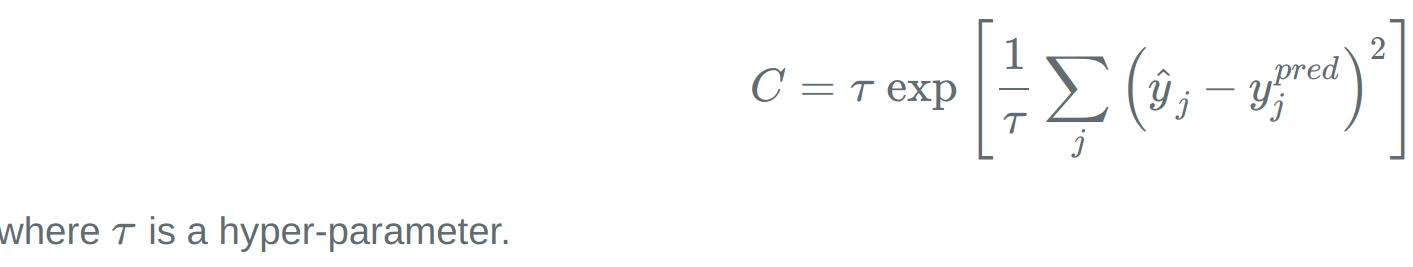
1. Quadratic Cost function: regression



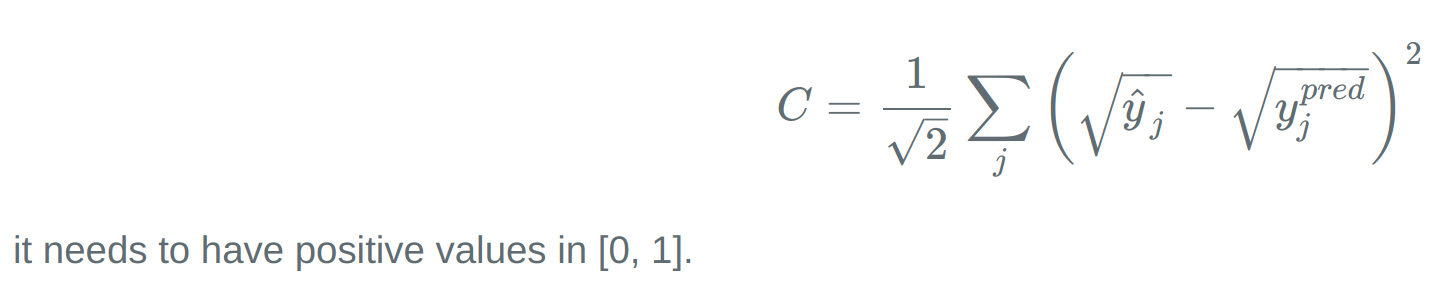
2. Cross Entropy Cost: Classification



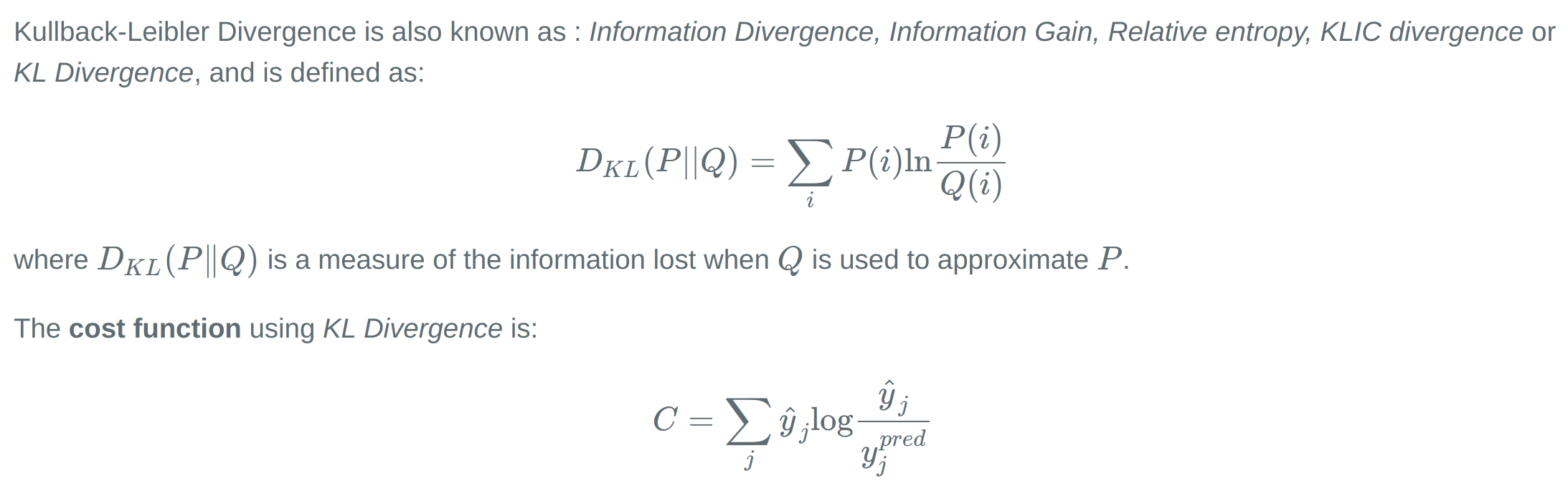
### 3. Exponential Cost



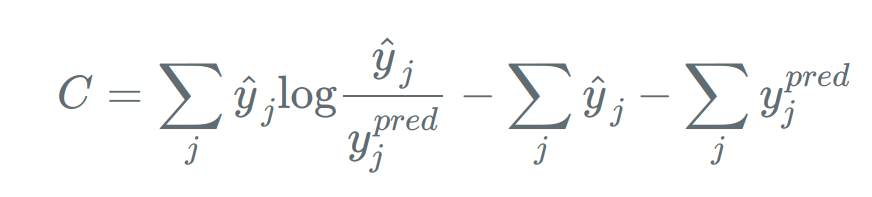
### 4. Hellinger Distance



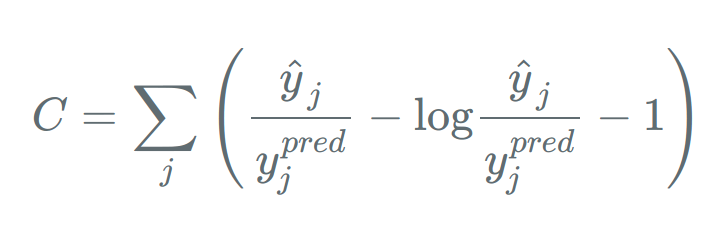
5. Kullback-Leibler Divergence



6. Generalized Kullback-Leibler Divergence



7. Itakura-Saito Distance



19. Can I solve problem of classification with tabular data in neural network?

Yes, OfCourse We can solve tabular data classification problem in neural network.

20. What do you understand by backword propagation in neural network?

The central technique by which artificial neural networks learn is backpropagation. It is doing the messenger's job to inform the neural network whether it made a mistake when making a forecast.

To propagate something (light, sound, motion, or information) means to send it in a certain direction or through a specific medium. Backpropagate is to communicate anything in response, or to send information back upstream - in this case, to fix an error. When we talk about backpropagation in deep learning, we're talking about information transfer, and that information is related to the error that the neural network produces when it makes a guess about data. Correction is synonymous with backpropagation.