# 1. Define the term “KNN” classification. Write two limitations of this classification.

**Ans**: KNN is a non-parametric, lazy learning algorithm. Its purpose is to use a database in which the data points are **s**eparated into several classes to predict the classification of a new sample point. Just for reference, this is “where” KNN is positioned in the algorithm list of scikit learn.

KNN classifier-

* Does not build models explicitly
* Unlike eager learners such as decision tree induction and rule-based systems
* Classifying unknown records are relatively expensive

**Two limitations of KNN classification:**

1. The main disadvantage of the KNN algorithm is that it is a *lazy learner*, i.e. it does not learn anything from the training data and simply uses the training data itself for classification, which can result in the algorithm not generalizing well and also not being robust to noisy data.
2. To predict the label of a new instance the KNN algorithm will find the *K* closest neighbors to the new instance from the training data, the predicted class label will then be set as the most common label among the *K* closest neighboring points. The main disadvantage of this approach is that the algorithm must compute the distance and sort all the training data at each prediction, which can be slow if there are a large number of training examples. Further, changing *K* can change the resulting predicted class label.

**2. Define the term True Positive and False Negative.**

**Ans:**

**True Positive**: A **true positive** test result is one that detects the condition when the condition is present.

**False Negative:** A **true negative** test result is one that does not detect the condition when the condition is absent.

**3. Define the terms: accuracy, recall, F-measures and precision.**

**Ans:**

[**Accuracy**](https://blog.exsilio.com/all/accuracy-precision-recall-f1-score-interpretation-of-performance-measures/) : [Accuracy](https://blog.exsilio.com/all/accuracy-precision-recall-f1-score-interpretation-of-performance-measures/) is the most intuitive performance measure and it is simply a ratio of correctly predicted observation to the total observations. One may think that, if we have high [accuracy](https://blog.exsilio.com/all/accuracy-precision-recall-f1-score-interpretation-of-performance-measures/) then our model is best. Yes, [accuracy](https://blog.exsilio.com/all/accuracy-precision-recall-f1-score-interpretation-of-performance-measures/) is a great measure but only when you have symmetric datasets where values of false positive and false negatives are almost same. Therefore, you have to look at other parameters to evaluate the performance of your model. For our model, we have got 0.803 which means our model is approx. 80% accurate.

[Accuracy](https://blog.exsilio.com/all/accuracy-precision-recall-f1-score-interpretation-of-performance-measures/) = TP+TN/TP+FP+FN+TN

[**Precision**](https://blog.exsilio.com/all/accuracy-precision-recall-f1-score-interpretation-of-performance-measures/) - [Precision](https://blog.exsilio.com/all/accuracy-precision-recall-f1-score-interpretation-of-performance-measures/) is the ratio of correctly predicted positive observations to the total predicted positive observations. The question that this metric answer is of all passengers that labeled as survived, how many actually survived? High [precision](https://blog.exsilio.com/all/accuracy-precision-recall-f1-score-interpretation-of-performance-measures/) relates to the low false positive rate. We have got 0.788 [precision](https://blog.exsilio.com/all/accuracy-precision-recall-f1-score-interpretation-of-performance-measures/) which is pretty good.

[Precision](https://blog.exsilio.com/all/accuracy-precision-recall-f1-score-interpretation-of-performance-measures/) = TP/TP+FP

**Recall** (Sensitivity) - Recall is the ratio of correctly predicted positive observations to the all observations in actual class - yes. The question recall answers is: Of all the passengers that truly survived, how many did we label? We have got recall of 0.631 which is good for this model as it’s above 0.5. Recall = TP/TP+FN

**F1 score** - F1 Score is the weighted average of [Precision](https://blog.exsilio.com/all/accuracy-precision-recall-f1-score-interpretation-of-performance-measures/) and Recall. Therefore, this score takes both false positives and false negatives into account. Intuitively it is not as easy to understand as [accuracy,](https://blog.exsilio.com/all/accuracy-precision-recall-f1-score-interpretation-of-performance-measures/) but F1 is usually more useful than [accuracy,](https://blog.exsilio.com/all/accuracy-precision-recall-f1-score-interpretation-of-performance-measures/) especially if you have an uneven class distribution. [Accuracy](https://blog.exsilio.com/all/accuracy-precision-recall-f1-score-interpretation-of-performance-measures/) works best if false positives and false negatives have similar cost. If the cost of false positives and false negatives are very different, it’s better to look at both [Precision](https://blog.exsilio.com/all/accuracy-precision-recall-f1-score-interpretation-of-performance-measures/) and Recall. In our case, F1 score is 0.701.

F1 Score = 2\*(Recall \* [Precision)](https://blog.exsilio.com/all/accuracy-precision-recall-f1-score-interpretation-of-performance-measures/) / (Recall + [Precision)](https://blog.exsilio.com/all/accuracy-precision-recall-f1-score-interpretation-of-performance-measures/)

**4. Define tree based and rule based classification.**

**Tree based learning:** Decision tree learning is a method commonly used in data mining. The goal is to create a model that predicts the value of a target variable based on several input variables. An example is shown in the diagram at right. Each [interior node](https://en.wikipedia.org/wiki/Interior_node) corresponds to one of the input variables; there are edges to children for each of the possible values of that input variable. Each leaf represents a value of the target variable given the values of the input variables represented by the path from the root to the leaf.

**Rule based classification:** Rule-based classifier makes use of a set of IF-THEN rules for classification. We can express a rule in the following from −

IF condition THEN conclusion

**Points to remember −**

* + The IF part of the rule is called **rule antecedent** or **precondition**.
  + The THEN part of the rule is called **rule consequent**.
  + The antecedent part the condition consist of one or more attribute tests and these tests are logically ANDed.
  + The consequent part consists of class prediction.

**5. Write the process of classification of any one.**

**Ans:** The process of decision tree based classification is given below:

There are several steps involved in the building of a decision tree.

* **Splitting:** The process of partitioning the data set into subsets. Splits are formed on a particular variable
* **Pruning:** The shortening of branches of the tree. Pruning is the process of reducing the size of the tree by turning some branch nodes into leaf nodes, and removing the leaf nodes under the original branch. Pruning is useful because classification trees may fit the training data well, but may do a poor job of classifying new values. A simpler tree often avoids over-fitting.A pruned tree has less nodes and has less sparsity than a unpruned decision tree.
* **Tree Selection:** The process of finding the smallest tree that fits the data. Usually this is the tree that yields the lowest cross-validated error.

**Key Factors:**

**1. Entropy:** A decision tree is built top-down from a root node and involves partitioning the data into subsets that contain instances with similar values (homogeneous). ID 3 algorithm uses entropy to calculate the homogeneity of a sample. If the sample is completely homogeneous the entropy is zero and if the sample is an equally divided it has entropy of one.

**2. Information Gain:** The information gain is based on the decrease in entropy after a dataset is split on an attribute. Constructing a decision tree is all about finding attribute that returns the highest information gain (i.e., the most homogeneous branches).

**3.** **Steps Involved**

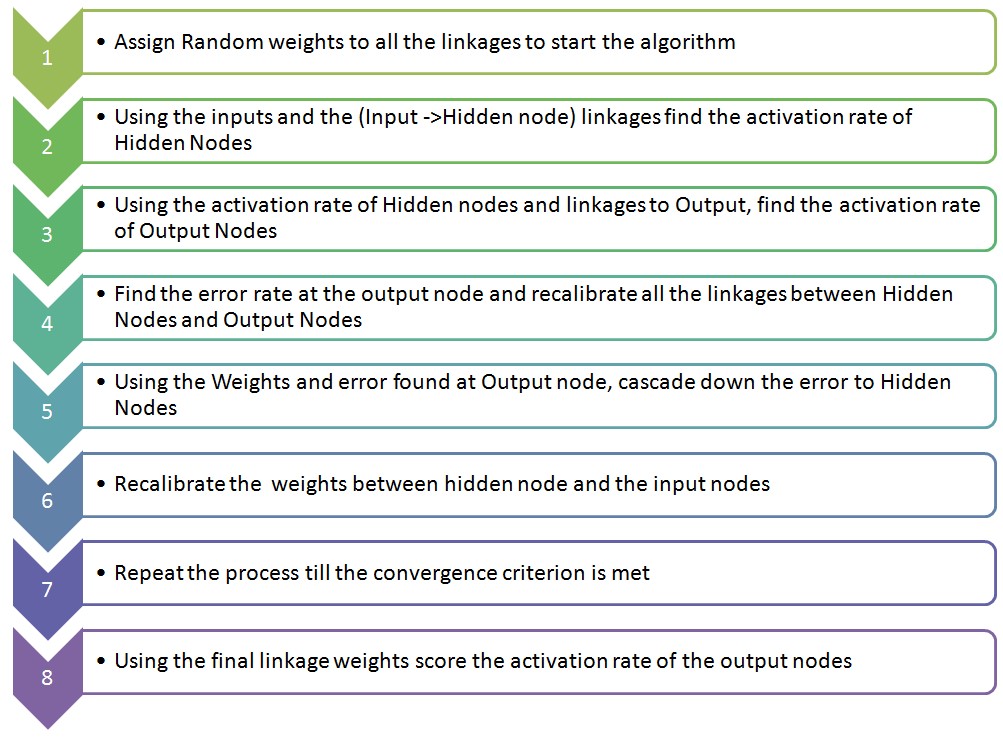
**Step 1:** Calculate entropy of the target.

**Step 2:** The dataset is then split on the different attributes. The entropy for each branch is calculated. Then it is added proportionally, to get total entropy for the split. The resulting entropy is subtracted from the entropy before the split. The result is the Information Gain, or decrease in entropy.

**Step 3:** Choose attribute with the largest information gain as the decision node, divide the dataset by its branches and repeat the same process on every branch.

**6. How ANN classifier works?**

Following is the framework in which artificial neural networks (ANN) work:



**4.How Bayes classifier works?**

**Ans:** Bayes theorem named after Rev. Thomas Bayes. It works on conditional probability. Conditional probability is the probability that something will happen, given that something else has already occurred. Using the conditional probability, we can calculate the probability of an event using its prior knowledge.

Below is the formula for calculating the conditional probability.



* + Consider each attribute and class label as random variables
  + Given a record with attributes (A1, A2,…,An)
    - Goal is to predict class C
    - Specifically, we want to find the value of C that maximizes P(C| A1, A2,…,An )  Can we estimate P(C| A1, A2,…,An ) directly from data?
  + Approach:
    - compute the posterior probability P(C | A1, A2, …, An) for all values of C using the Bayes theorem
    - Choose value of C that maximizes

P(C | A1, A2, …, An)

* + - Equivalent to choosing value of C that maximizes

P(A1, A2, …, An|C) P(C)

**8.** **How do you perform KNN?**

**Ans:** The steps are given below:

1. Determine K = number of nearest neighbors
2. Calculate the distance between the query-instance and all the training samples
3. Sort the distance and determine nearest neighbors based on the k-th minimum distance
4. Gather the category of the nearest neighbors
5. Use sample majority of the category of nearest neighbors as the prediction value of the query instance

**9. How do you validate a classification model?**

**Ans:** The steps are given below:

1. Choose several appropriate models/algorithms.
2. Split the training set into a smaller training set and validation set. The split depends on the problem at hand and the nature of the data.
3. Tune the parameters of each model by cross-validation on the smaller training set.
4. Choose the best parameter set for each model based on the the point above.
5. Test each model separately on the validation set.
6. Choose the best model according to your metric (accuracy is not the best in most cases, although it depends).

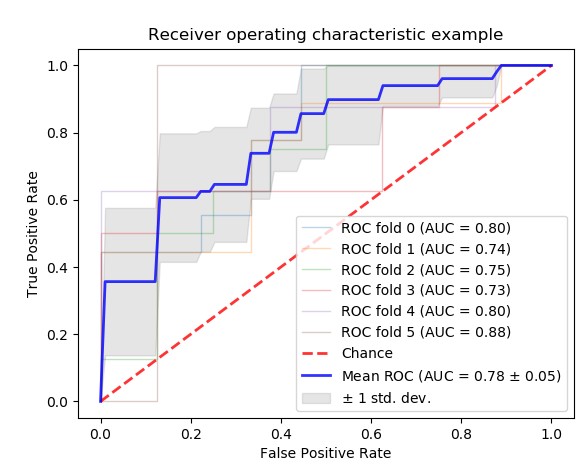
**10. What are the functions of ROC curve in validation?**

**Ans:** Example of Receiver Operating Characteristic (ROC) metric to evaluate classifier output quality using cross-validation.

ROC curves typically feature true positive rate on the Y axis, and false positive rate on the X axis. This means that the top left corner of the plot is the “ideal” point - a false positive rate of zero, and a true positive rate of one. This is not very realistic, but it does mean that a larger area under the curve (AUC) is usually better.

The “steepness” of ROC curves is also important, since it is ideal to maximize the true positive rate while minimizing the false positive rate.

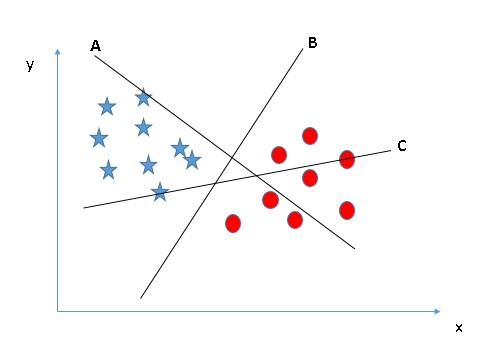
This example shows the ROC response of different datasets, created from K-fold cross-validation. Taking all of these curves, it is possible to calculate the mean area under curve, and see the variance of the curve when the training set is split into different subsets. This roughly shows how the classifier output is affected by changes in the training data, and how different the splits generated by K-fold cross-validation are from one another.



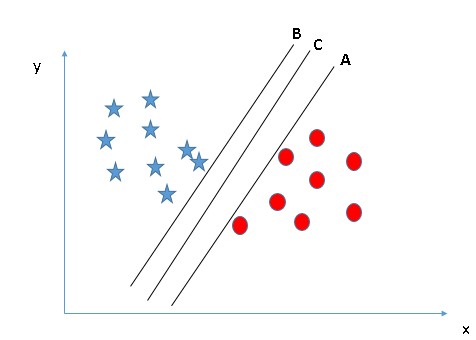
**11. How SVM classifier works?**

**Ans:**

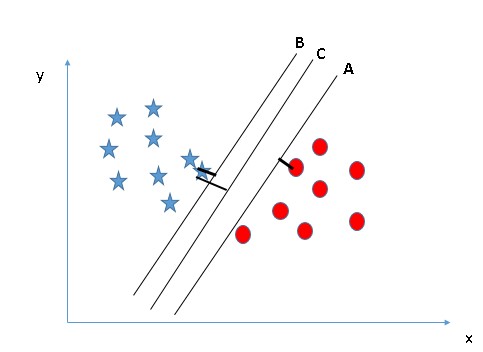
* **Identify the right hyper-plane (Scenario-1):** Here, we have three hyper-planes (A, B and C). Now, identify the right hyper-plane to classify star and circle.



* You need to remember a thumb rule to identify the right hyper-plane: “Select the hyperplane which segregates the two classes better”. In this scenario, hyper-plane “B” has excellently performed this job.
* **Identify the right hyper-plane (Scenario-2):** Here, we have three hyper-planes (A, B and C) and all are segregating the classes well.



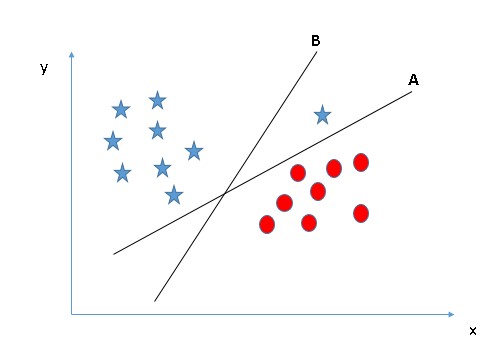
Here, maximizing the distances between nearest data point (either class) and hyper-plane will help us to decide the right hyper-plane. This distance is called as **Margin**. Let’s look



Above, you can see that the margin for hyper-plane C is high as compared to both A and

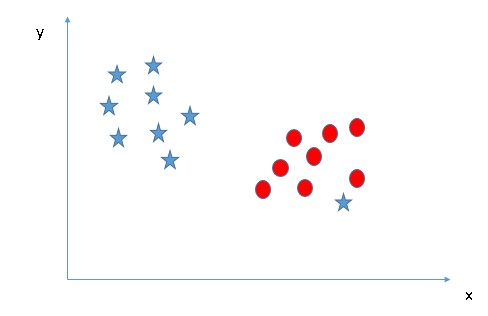
B. Hence, we name the right hyper-plane as C. Another lightning reason for selecting the hyper-plane with higher margin is robustness. If we select a hyper-plane having low margin then there is high chance of miss-classification.

* **Identify the right hyper-plane (Scenario-3):** Hint:Use the rules as discussed in previous section to identify the right hyper-plane

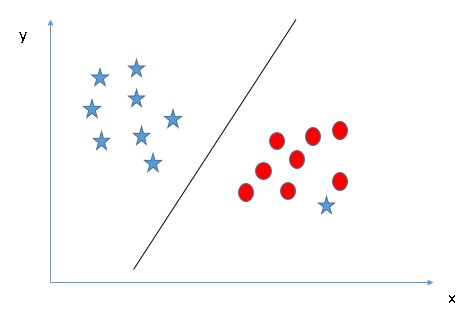


Some of you may have selected the hyper-plane **B** as it has higher margin compared to **A.** But, here is the catch, SVM selects the hyper-plane which classifies the classes accurately prior to maximizing margin. Here, hyper-plane B has a classification error and A has classified all correctly. Therefore, the right hyper-plane is **A.**

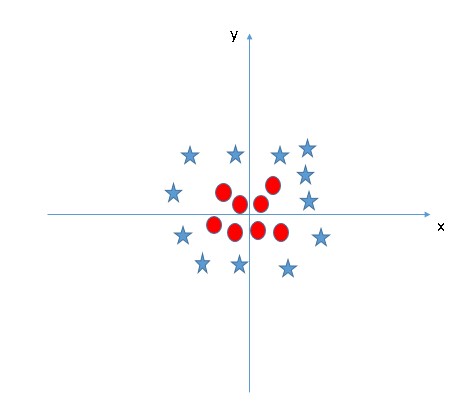
* **Can we classify two classes (Scenario-4)?:** Below, I am unable to segregate the two classes using a straight line, as one of star lies in the territory of other(circle) class as an outlier.



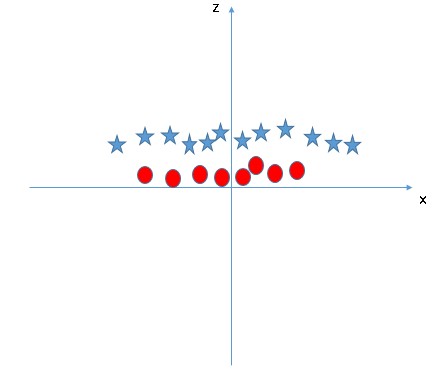
* As I have already mentioned, one star at other end is like an outlier for star class. SVM has a feature to ignore outliers and find the hyper-plane that has maximum margin. Hence, we can say, SVM is robust to outliers.



* **Find the hyper-plane to segregate to classes (Scenario-5):** In the scenario below, we can’t have linear hyper-plane between the two classes, so how does SVM classify these two classes? Till now, we have only looked at the linear hyper-plane.



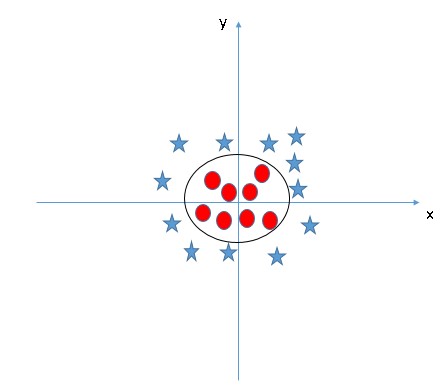
* SVM can solve this problem. Easily! It solves this problem by introducing additional feature. Here, we will add a new feature z=x^2+y^2. Now, let’s plot the data points on axis x and z:



In above plot, points to consider are:

* + - All values for z would be positive always because z is the squared sum of both x and y
    - In the original plot, red circles appear close to the origin of x and y axes, leading to lower value of z and star relatively away from the origin result to higher value of z.

When we look at the hyper-plane in original input space it looks like a circle:



**12. Math on Draw dendrogram for hierarchical clustering.**

**Ans:** Dendrogram for hierarchical clustering



**1**



**2**



**3**



**4**



**5**



**6**



**1**



**2**



**3**



**4**



**5**

3

6

2

5

4

1

0

0.05

0.1

0.15

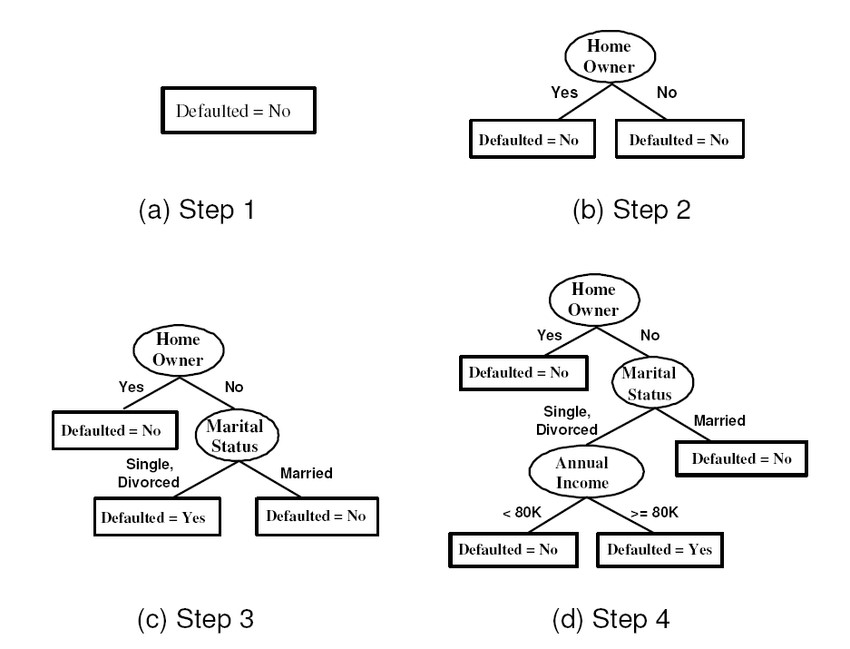
0.2

13. **If you have a data set with class attribute and a new data without class attribute. You want to predict the value of class attribute of new data. Write the process of classifying this new data using decision tree classification (Hunt’s Algorithm).**

**Ans:** Let D be the set of training records that reach a node t t

General Procedure:

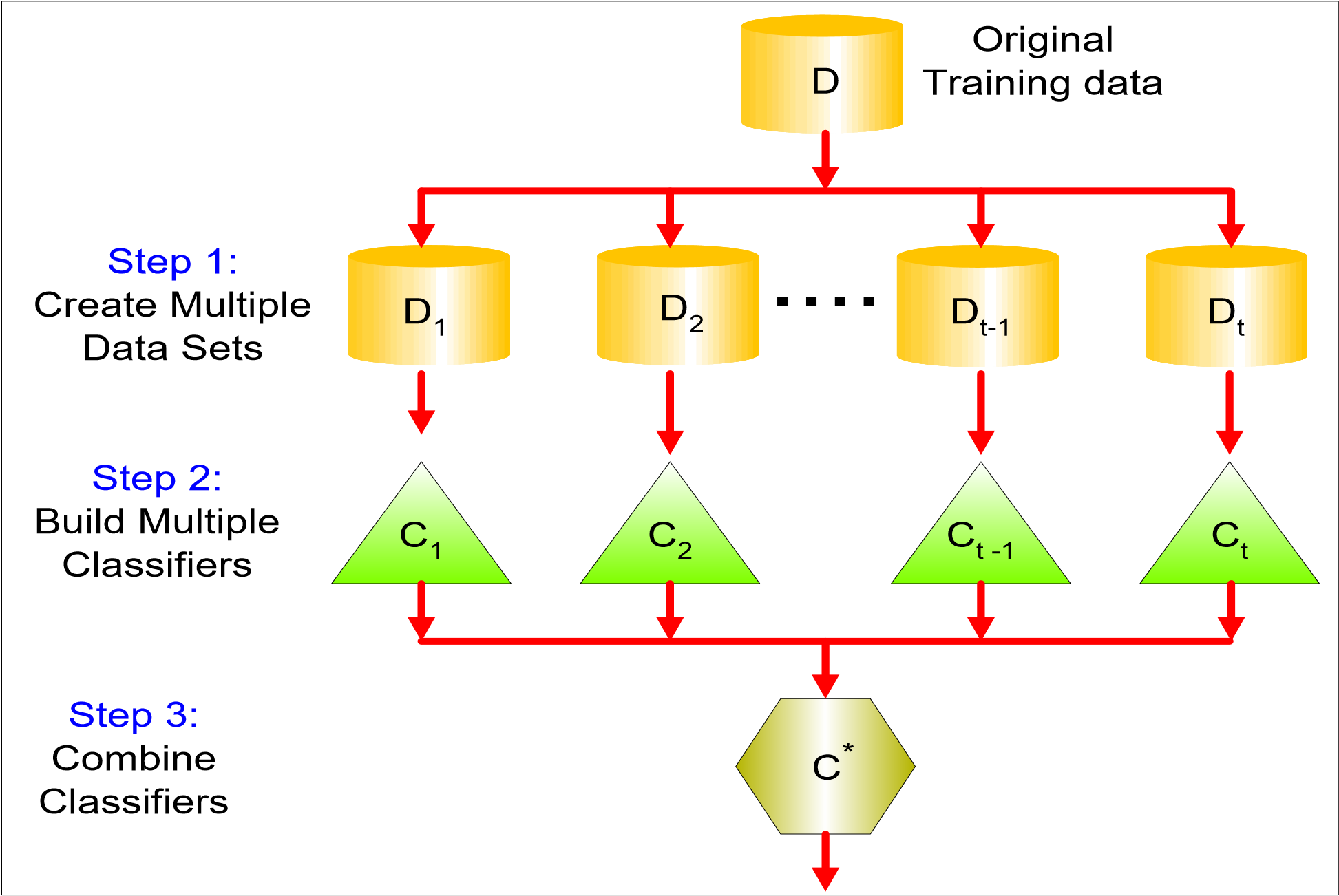
* + If D contains records that belong the same class y , then t is a leaf node labeled as tt y t
  + If D is an empty set, then t is a leaf node labeled by the default class, y t d
  + If D contains records that belong to more than one class, use an attribute test to t split the data into smaller subsets. Recursively apply the procedure to each subset.



**14.** **If you have a data set with class attribute and a new data without class attribute. Write the process of classifying this new data using ensemble method.**

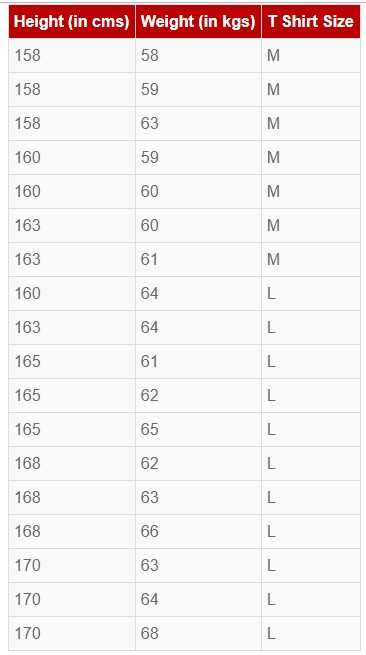
**Ans:**

* Construct a classifier of training data
* Predict class label of previously unseen records by aggregating predictions made by multiple classifiers



**15**. **Write the process of classifying this new data using KNN.**

Ans: Suppose we have height, weight and T-shirt size of some customers and we need to predict the T-shirt size of a new customer given only height and weight information we have. Data including height, weight and T-shirt size information is shown below



In the classification setting, the K-nearest neighbor algorithm essentially boils down to forming a majority vote between the K most similar instances to a given “unseen” observation. Similarity is defined according to a distance metric between two data points. A popular choice is the Euclidean distance given by

*d*(*x*,*x*′)=√( (*x*1−*x*′1)2+(*x*2−*x*′2)2+…+(*xn*−*x*′*n*)2 )

but other measures can be more suitable for a given setting and include the Manhattan, Chebyshev and Hamming distance.

More formally, given a positive integer K, an unseen observation *x* and a similarity metric *d*, KNN classifier performs the following two steps:

 It runs through the whole dataset computing *d* between *x* and each training observation. We’ll call the K points in the training data that are closest to *x* the set A. Note that K is usually odd to prevent tie situations.

It then estimates the conditional probability for each class, that is, the fraction of points in A with that given class label. (Note *I*(*x*) is the indicator function which evaluates to 1 when the argument *x* is true and 0 otherwise)

*P*(*y*=*j*|*X*=*x*)=1/*K*

∑*i*A *I*(*y*(*i*)=*j*)

Finally, our input *x* gets assigned to the class with the largest probability. An alternate way of understanding KNN is by thinking about it as calculating a decision boundary (i.e. boundaries for more than 2 classes) which is then used to classify new points.

**16. What is Validation?**

**Ans: Validation:** In this approach, instead of using the training set to estimate the generalization error, the original training data is divided into two smaller subsets. One of the subsets is used for training, while the other, known as the validation set, is used for estimating the generalization error.

Validation set: Pick algorithm + knob settings

* Pick best-performing algorithm (NB vs. DT vs…)
* Fine-tune knobs (tree depth, k in KNN, c in SVM)

**17. In constructing a decision tree, how do we select an attribute and when do we stop the further expansion of the tree?**

**Ans:** determine the attribute that best classifies the training data; use this attribute at the root of the tree. Repeat this process at for each branch. This means we are performing top-down, greedy search through the space of possible decision trees.

use the attribute with the highest **information gain**in **ID3**. In order to define information gain precisely, we begin by defining a measure commonly used in information theory, called **entropy** that characterizes the (im) purity of an arbitrary collection of examples.

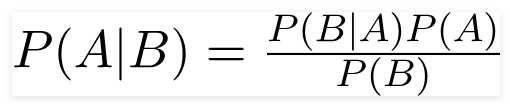
**When to stop:**

* Stop expanding a node when all the records belong to the same class
* Stop expanding a node when all the records have similar attribute value.
* Early termination.

**17. On what principal Bayesian classifier has been built?**

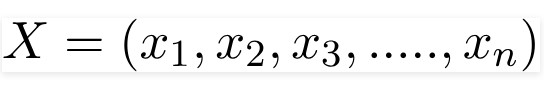
**Ans:** [Naive Bayes classifiers](https://www.geeksforgeeks.org/naive-bayes-classifiers/) are a collection of classification algorithms based on **Bayes’ Theorem**. It is not a single algorithm but a family of algorithms where all of them share a common principle, i.e. every pair of features being classified is independent of each other.

Bayes’ Theorem finds the probability of an event occurring given the probability of another event that has already occurred. Bayes’ theorem is stated mathematically as the following equation:



Now, with regards to our dataset, we can apply Bayes’ theorem in following way:

where, y is class variable and X is a dependent feature vector (of size *n*) where:

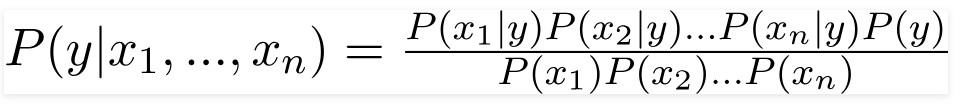


**Naive assumption**

Now, its time to put a naive assumption to the Bayes’ theorem, which is, **independence**among the features. So now, we split **evidence** into the independent parts. Now, if any two events A and B are independent, then

P(A,B)= p(A)p(B)

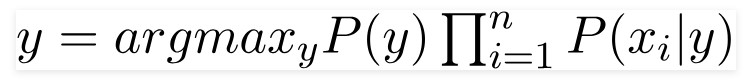
Hence, we reach to the result:



Now, as the denominator remains constant for a given input, we can remove that term:



Now, we need to create a classifier model. For this, we find the probability of given set of inputs for all possible values of the class variable *y* and pick up the output with maximum probability. This can be expressed mathematically as:



So, finally, we are left with the task of calculating P(y) and P(xi | y).

**19.Math on Gini Index**

**Ans:**

**Gini Index:** Gini index is the most commonly used measure of inequality. Also referred as Gini ratio or Gini coefficient.

Gini Index for a given node t:

Gini (t) = 1-∑[𝑝(𝑗|𝑡)]^2

p( j | t) is the relative frequency of class j at node t.

* Maximum (1 - 1/nc) when records are equally distributed among all classes, implying least interesting information.
* Minimum (0.0) when all records belong to one class, implying most interesting information.

**Example of computing Gini:**

Gini (t) = 1-∑[𝑝(𝑗|𝑡)]^2

|  |  |
| --- | --- |
| C1 | 0 |
|  |  |
| C2 | 6 |

P(C1)=0/6 = 0 P(C2) = 6/6 = 1

Gini = 1 - P(C1)2 - P(C2)2 = 1-0-1 =0

|  |  |
| --- | --- |
| C1 | 1 |
| C2 | 5 |

P(C1)=1/6 P(C2) = 5/6

Gini = 1 - P(C1)2 - P(C2)2 = 1 - (1/6)2 – (5/6)2 = .278

|  |  |
| --- | --- |
| C1 | 2 |
| C2 | 4 |

P(C1)=2/6 P(C2) = 4/6

Gini = 1 - (2/6)2 - (4/6)2 = .444

**20. What are the advantage of tree based classification?**

**Ans: Advantage of tree based classification:**

* Decision trees are easy to interpret and visualize.
* It can easily capture Non-linear patterns.
* It requires fewer data preprocessing from the user, for example, there is no need to normalize columns.
* It can be used for feature engineering such as predicting missing values, suitable for variable selection.
* The decision tree has no assumptions about distribution because of the nonparametric nature of the algorithm.
* It does not require any domain knowledge.
* It is easy to comprehend.
* The learning and classification steps of a decision tree are simple and fast.

**21. What are the main principal of Bayesian Classification?**

**Ans:** The Naive Bayes Classifier technique is based on the so-called Bayesian theorem and is particularly suited when the dimensionality of the inputs is high. Despite its simplicity, Naive Bayes can often perform better in many complex real world situations. Requires a small amount of training data to estimate the parameters.

P(A/B) = P(B/A) P(A) / P(B)

P(A) : Prior probability of hypothesis A

P(B) : Prior probability of training data B

P(A/B) : Probability of A given B

P(B/A) : Probability of B given A

* Naive assumption: attribute independence P(x1,…,xk|C) = P(x1|C)·…·P(xk|C)
* If ith attribute is categorical: P(xi|C) is estimated as the relative frequency of samples having value xi as ith attribute in class C.
* If ith attribute is continuous: P(xi|C) is estimated through a Gaussian density function.

**22. What are the sequential steps in doing classification of a set of data?**

**Ans**: **Four steps to data classification**

Step 1: Choose your target. Define your goal.

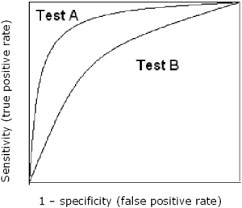
Step 2: Map an approach and appropriate toolset. Determine the metrics you'll collect. ...

Step 3: Gather your data and validate it. ...

Step 4: Organize and communicate the data in a form that will lead to positive change/action.

**23. What are the use of ROC curve?**

**Ans:** The best cut-off has the highest true positive rate together with the lowest false positive rate. As the area under a ROC curve is a measure of the usefulness of a test in general, where a greater area means a more useful test, the areas under ROC curves are used to compare the usefulness of tests.



**24. What are the uses of support vector machine (SVM)?**

**Ans: Support Vector Machine (SVM):** “Support Vector Machine” (SVM) is a supervised machine learning algorithm which can be used for either classification or regression challenges. However, it is mostly used in classification problems. It uses a technique called the kernel trick to transform user data. SVM is capable of doing both classification and regression. SVM also works very well with high-dimensional data and avoids the curse of dimensionality problem.

**Uses of SVM:**

* It works really well with clear margin of separation  It is effective in high dimensional spaces.
* It is effective in cases where number of dimensions is greater than the number of samples.
* It uses a subset of training points in the decision function (called support vectors), so it is also memory efficient.

**25. What do you mean by the term precision and recall? When do we use these?**

**Ans: Precision:** In the field of [information retrieval,](https://en.wikipedia.org/wiki/Information_retrieval) precision is the fraction of retrieved documents that are [relevant](https://en.wikipedia.org/wiki/Relevance_(information_retrieval)) to the query:

For example, for a text search on a set of documents, precision is the number of correct results divided by the number of all returned results. Precision takes all retrieved documents into account, but it can also be evaluated at a given cutoff rank, considering only the topmost results returned by the system. This measure is called precision at n or P@n. Precision is used with [recall,](https://en.wikipedia.org/wiki/Recall_(information_retrieval)) the percent of all relevant documents that is returned by the search. The two measures are sometimes used together in th[e F1 Score](https://en.wikipedia.org/wiki/F1_Score) (or f-measure) to provide a single measurement for a system.

Note that the meaning and usage of "precision" in the field of information retrieval differs from the definition of [accuracy and precision](https://en.wikipedia.org/wiki/Accuracy_and_precision) within other branches of science and technology.

**Recall:** In information retrieval, recall is the fraction of the relevant documents that are successfully retrieved.

For example, for a text search on a set of documents, recall is the number of correct results divided by the number of results that should have been returned.

In binary classification, recall is called [sensitivity.](https://en.wikipedia.org/wiki/Sensitivity_and_specificity#Sensitivity) It can be viewed as the probability that a relevant document is retrieved by the query.

It is trivial to achieve recall of 100% by returning all documents in response to any query. Therefore, recall alone is not enough but one needs to measure the number of non-relevant documents also, for example by also computing the precision.

**When do we use:**

It describes how good a model is at predicting the positive class. Precision is referred to as the positive predictive value. Recall is calculated as the ratio of the number of true positives divided by the sum of the true positives and the false negatives. Recall is the same as sensitivity. Precision Recall curves should be used when there is a moderate to large class imbalance.

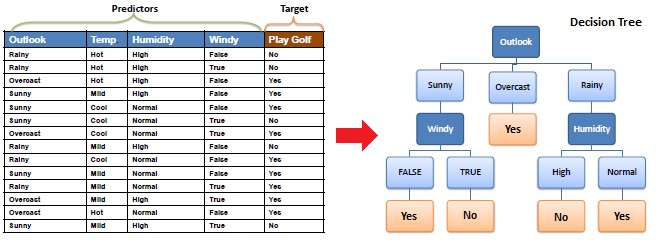
**26. What do you mean by rule base classification?**

**Ans:** The term rule[-based classification](https://link.springer.com/referenceworkentry/10.1007%2F978-0-387-39940-9_559) can be used to refer to any classification scheme that make use of IF-THEN rules for class prediction. Rule[-based classification](https://link.springer.com/referenceworkentry/10.1007%2F978-0-387-39940-9_559) schemes typically consist of the following components:

* Rule Induction Algorithm This refers to the process of extracting relevant IF-THEN rules from the data which can be done directly using sequential covering algorithms or indirectly from other data mining methods like decision tree building or association rule mining.
* Rule Ranking Measures This refers to some values that are used to measure the usefulness of a rule in providing accurate prediction. Rule ranking measures are often used in the rule induction algorithm to prune off unnecessary rules and improve efficiency. They are also used in the class prediction algorithm to give a ranking to the rules which will be then be utilized to predict the class of new cases. Class Prediction *Algorithm*Given a new record.

**27. What is decision tree classification?**

**Ans:** Decision tree builds classification or regression models in the form of a tree structure. It breaks down a dataset into smaller and smaller subsets while at the same time an associated decision tree is incrementally developed. The final result is a tree with **decision nodes** and **leaf nodes**. A decision node (e.g., Outlook) has two or more branches (e.g., Sunny, Overcast and Rainy). Leaf node (e.g., Play) represents a classification or decision. The topmost decision node in a tree which corresponds to the best predictor called **root node**. Decision trees can handle both categorical and numerical data.



**28. What is ensemble method of classification? Explain with pictorial example.**

**Ans:** *Ensemble models in machine learning combine the decisions from multiple models to improve the overall performance.* They operate on the similar idea as employed while buying headphones. The main causes of error in learning models are due to **noise, bias and variance**.

**Ensemble methods help to minimize these factors**. These methods are designed to improve the stability and the accuracy of Machine Learning algorithms. **Simple Ensemble techniques**

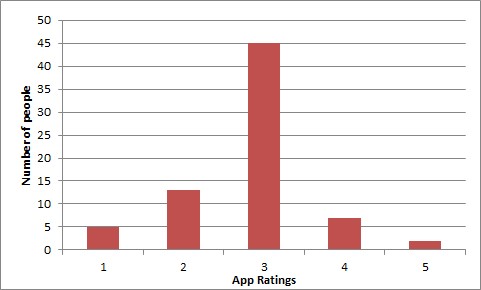
1. **Taking the mode of the results**

MODE: The mode is a statistical term that refers to the most frequently occurring number found in a set of numbers.

In this technique, multiple models are used to make predictions for each data point. The predictions by each model are considered as a separate vote. The prediction which we get from the majority of the models is used as the final prediction.

For instance: We can understand this by referring back to Scenario 2 above. I have inserted a chart below to demonstrate the ratings that the beta version of our health and fitness app got from the user community. (*Consider each person as a different model*)

Output= MODE=3, as majority people voted this



1. **Taking the average of the results**

In this technique, we take an average of predictions from all the models and use it to make the final prediction.

AVERAGE= sum(Rating\*Number of people)/Total number of people= (1\*5)+(2\*13)+(3\*45)+(4\*7)+(5\*2)/72

= 2.833

=Rounded to nearest integer would be 3

1. **Taking weighted average of the results**

This is an extension of the averaging method. All models are assigned different weights defining the importance of each model for prediction. For instance, if about 25 of your responders are professional app developers, while others have no prior experience in this field, then the answers by these 25 people are given more importance as compared to the other people.

For example: For posterity, I am trimming down the scale of the example to 5 people

WEIGHTED AVERAGE= (0.3\*3)+(0.3\*2)+(0.3\*2)+(0.15\*4)+(0.15\*3) =3.15 = rounded to nearest integer would give us 3



**29. What is the main principal of Gini index? – explain.**

**Ans:** The Gini coefficient measures the inequality among values of a frequency distribution (for example levels of income). A Gini coefficient of zero expresses perfect equality where all values are the same (for example, where everyone has an exactly equal income)

**Main principal of Gini Index-**

1. *Anonymity:*

The coefficient does not disclose the identities of high-income and low-income individuals in a population.

1. *Scale of independence*

The calculation of the Gini coefficient does not depend on how large the economy is, how it is measured, or how wealthy a country is. For example, both rich and poor countries may show the same coefficient due to similar income distribution.

1. *Population independence*

The coefficient does not depend on the size of the population.

1. *Transfer principle*

The coefficient reflects situations when income is transferred from a rich to a poor individual.

**30. When we have to use Gini index in splitting?**

**Ans:** We use the Gini Index as our cost function used to evaluate splits in the dataset. our target variable is Binary variable which means it take two values (Yes and No). There can be 4 combinations. A Gini score gives an idea of how good a split is by how mixed the classes are in the two groups created by the split.

**31. Which classification technique will you use?**

**Ans:** I will prefer decision tree-based classification, as it is-

* Inexpensive to construct.
* Extremely fast at classifying unknown records.
* Easy to interpret for small-sized trees.
* Accuracy is comparable to other classification techniques for many simple data sets.

**32. Why and when do researchers like to use SVM classifier?**

**Ans:** Why to use:

1. It uses **Kernel** trick
2. It is Optimal margin based classification technique in Machine Learning.
3. Good number of algorithms are proposed which utilizes **problem structures** and other smallersmaller things like **problem shrinking** during optimization etc.

When to use:

1. When number of features (variables) and number of training data is very large (say millions of features and millions of instances (data)).
2. When sparsity in the problem is very high, i.e., most of the features have zero value.
3. It is the best for document classification problems where sparsity is high and features/instances are also very high.
4. It also performs very well for problems like image classification, genes classification, drug disambiguation etc. where number of features are high.

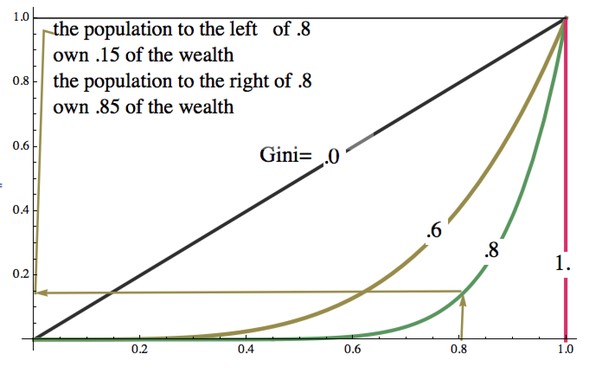
**33. Why and when do we use Gini coefficient or entropy?**

**Ans:**

**Why:** The Gini index measures the **distribution** of wealth, income, or anything else for that matter.

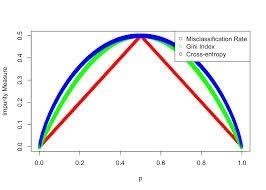
Since wealth distribution is vital to the stability of a society and the well being of its people, lets use it for an explanation of its use.

First, for those who aren’t familiar with Dr. Gini’s index, in the plot below,

 the y axis is the fraction of total wealth owned by the fraction on the x axis.The x axis is the fraction of the population who own that wealth. For the US, the wealth Gini index can be calculated for the populations net wealth with home equity included, around .6 fraction of households, for those who have substantial other assets, those between .9 and .99. And those who have extreme wealth, between .99 and 1.0.

**34. Why entropy is used instead of Gini index?**

**Ans:** From what I can tell, both are used largely for the same purpose. If we visualize these two metrics (and throw in the miss-classification error) for a binary classification, we'd see something like this:



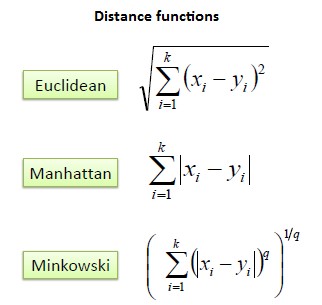
We notice that gini and cross-entropy look incredibly similar. Furthermore, while we often make the distinction between using miss-classification error versus gini or cross-entropy when growing trees, we don't often hear good reasons to use gini over cross-entropy, or vice versa (at least I haven't). I've come across people who like the interpretation of one over the other, but in practice it seems like we often try both (or just use one) and see which one gives us better results.

Why the two formulas, then? It appears that they arose from the same motivations, but from two different fields (Gini from statistics and cross-entropy from computer science/information theory). This paper discusses a little of how the two arose, but probably more importantly gives some empirical evidence to support the idea that one isn't really better than the other.

**35. Write the algorithm of K-nearest neighbor classification.**

**Ans:** The algorithm of K-nearest neighbor:

A case is classified by a majority vote of its neighbors, with the case being assigned to the class most common amongst its K nearest neighbors measured by a distance function. If K = 1, then the case is simply assigned to the class of its nearest neighbor.



It should also be noted that all three distance measures are only valid for continuous variables. In the instance of categorical variables the Hamming distance must be used. It also brings up the issue of standardization of the numerical variables between 0 and 1 when there is a mixture of numerical and categorical variables in the dataset.

1. **Write the limitations of KNN.**

**Ans*:***

**Limitations:**

* k-NN classifiers are lazy learners
* It does not build models explicitly
* Unlike eager learners such as decision tree induction and rule-based systems
* Classifying unknown records are relatively expensive

**37. What is K-nearest neighbor classification and k-means clustering?**

**Ans: KNN neighbor:** In [pattern recognition,](https://en.wikipedia.org/wiki/Pattern_recognition) the ***k*-nearest neighbors algorithm** (***k*-NN**) is a [nonparametric](https://en.wikipedia.org/wiki/Non-parametric_statistics) method used for [classification](https://en.wikipedia.org/wiki/Statistical_classification) and [regression.](https://en.wikipedia.org/wiki/Regression_analysis) In both cases, the input consists of the *k* closest training examples in the [feature space.](https://en.wikipedia.org/wiki/Feature_space) The output depends on whether *k*-NN is used for classification or regression:

In *k-NN classification*, the output is a class membership. An object is classified by a plurality vote of its neighbors, with the object being assigned to the class most common among its *k* nearest neighbors (*k* is a positive [integer,](https://en.wikipedia.org/wiki/Integer) typically small). If *k* = 1, then the object is simply assigned to the class of that single nearest neighbor.

In *k-NN regression*, the output is the property value for the object. This value is the average of the values of *k' nearest neighbors.*

*k*-NN is a type of [instance-based learning,](https://en.wikipedia.org/wiki/Instance-based_learning) or [lazy learning,](https://en.wikipedia.org/wiki/Lazy_learning) where the function is only approximated locally and all computation is deferred until classification. The *k*-NN algorithm is among the simplest of all [machine learning](https://en.wikipedia.org/wiki/Machine_learning) algorithms.

**K-means clustering:** K-means clustering is a method used for clustering analysis, especially in data mining and statistics. It aims to partition a set of observations into a number of clusters (k), resulting in the partitioning of the data into Voronoi cells. It can be considered a method of finding out which group a certain object really belongs to.

It is used mainly in statistics and can be applied to almost any branch of study. For example, in marketing, it can be used to group different demographics of people into simple groups that make it easier for marketers to target. Astronomers use it to sift through huge amounts of astronomical data; since they cannot analyze each object one by one, they need a way to statistically find points of interest for observation and investigation.

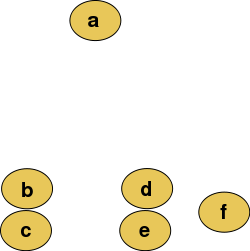
The algorithm:

* + 1. K points are placed into the object data space representing the initial group of centroids.
    2. Each object or data point is assigned into the closest k.
    3. After all objects are assigned, the positions of the k centroids are recalculated.
    4. Steps 2 and 3 are repeated until the positions of the centroids no longer move.

**38. Define hierarchical clustering with example.**

**Ans:** Hierarchical clustering is where you build a cluster tree (a dendrogram) to represent data, where each group (or “node”) links to two or more successor groups. The groups are [nested a](https://www.statisticshowto.datasciencecentral.com/nested-model-anova-factors/#model)nd organized as a tree, which ideally ends up as a meaningful classification scheme.

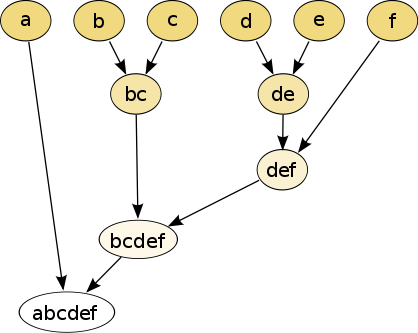
Each node in the cluster tree contains a group of similar data; Nodes group on the graph next to other, similar nodes. Clusters at one level join with clusters in the next level up, using a degree of similarity; The process carries on until all nodes are in the tree, which gives a visual snapshot of the data contained in the whole set. The total number of clusters is *not* predetermined before you start the tree creation.



Raw data

For example, suppose this data is to be clustered, and the [Euclidean distance](https://en.wikipedia.org/wiki/Euclidean_distance) is the [distance metric.](https://en.wikipedia.org/wiki/Metric_(mathematics))

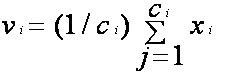
The hierarchical clustering [dendrogram](https://en.wikipedia.org/wiki/Dendrogram) would be as such:



**39. Describe the steps of K-means clustering.**

**Ans:** Let X = {x1,x2,x3,……..,xn} be the set of data points and V = {v1,v2,…….,vc} be the set of centers.

1. Randomly select *‘c’* cluster centers.
2. Calculate the distance between each data point and cluster centers.
3. Assign the data point to the cluster center whose distance from the cluster center is minimum of all the cluster centers..
4. Recalculate the new cluster center using:



where, *‘ci’* represents the number of data points in *ith* cluster.

1. Recalculate the distance between each data point and new obtained cluster centers.
2. If no data point was reassigned then stop, otherwise repeat from step 3.

**40. What are the different types of clustering?**

**Ans:**

**1. Partitional Clustering:**

– A division data objects into non-overlapping subsets (clusters) such that each data object is in exactly one subset.

**2. Hierarchical clustering:**

* + - A set of nested clusters organized as a hierarchical tree.

**3. Well-Separated Clusters:**

* + - A cluster is a set of points such that any point in a cluster is closer (or more similar) to every other point in the cluster than to any point not in the cluster.

**4. Center-based:**

* + - A cluster is a set of objects such that an object in a cluster is closer (more similar) to the “center” of a cluster, than to the center of any other cluster.
    - The center of a cluster is often a centroid, the average of all the points in the cluster, or a medoid, the most “representative” point of a cluster.

**5. Contiguous Cluster (Nearest neighbor or Transitive):**

* + - A cluster is a set of points such that a point in a cluster is closer (or more similar) to one or more other points in the cluster than to any point not in the cluster.

**6. Density-based:**

* + - A cluster is a dense region of points, which is separated by low-density regions, from other regions of high density.
    - Used when the clusters are irregular or intertwined, and when noise and outliers are present.

**7. Shared Property or Conceptual Clusters:**

* + - Finds clusters that share some common property or represent a particular concept.