**MACHINE LEARNING**

CREATED BY VISHNU MALLELA

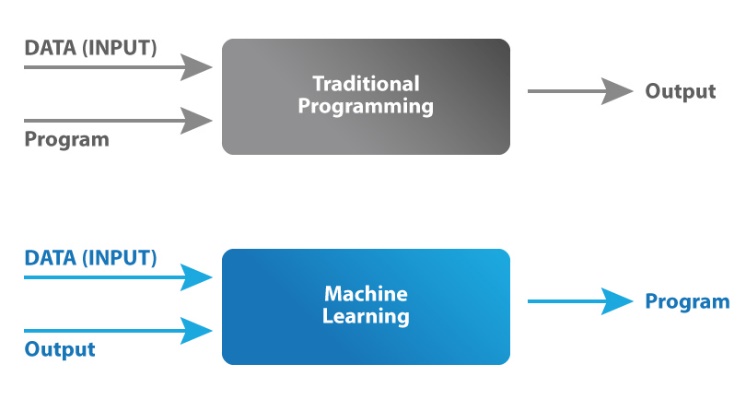
**MODULE-1:**

**(INTRODUCTION TO ML)**

**ML AND APPLICATION, LEARNING PARADIGMS, PERSPECTIVES AND ISSUES, FINITE AND INFINITE HYPOTHESIS, PAC LEARNING, FIND S AND CANDIDATE ELIMINATION**

**Machine Learning:**

Machine learning is defined as use of algorithms and computational statistics to learn from data without being explicitly programmed.



**Applications:**

* Image Recognition

Image Processing, or more specifically, Digital Image Processing, is the process of processing a digital image using a series of algorithms. Digital Image Processing Using Neural Networks has grown in popularity in recent years.

* Speech Recognition

speech recognition is the process of turning spoken commands to text and subsequently classifying, segmenting, and so on. This method is used by a variety of virtual assistants, including Google Assistant, Siri, Cortana, and others.

* Self-driving Automobiles

Machine learning is crucial in self-driving cars. Tesla, the most well-known vehicle manufacturer, is developing a self-driving car. It trains automobile models to recognize people and objects using unsupervised learning and reinforcement learning methods.

* Automatic Language Translation

Machine translation is a task that generally uses machine learning models generated using extremely complex linguistic knowledge and other related data to produce accurate text translation from one language to another. Machine Translations have become an essential aspect of corporate transactions when combined with Natural Language Understanding, which also employs Supervised Learning.

* Online Fraud Detection

Machine learning is proving its potential to make cyberspace a secure place and tracking monetary frauds online is one of its examples. For example: PayPal is using ML for protection against money laundering.

* Product suggestions

Various retail, eCommerce, and entertainment firms can produce suggestions for their users based on various levels of associativity by utilizing various association rule engines. For example, Amazon product recommendations, Netflix, and so on.

* Transportation and Commuting

If you have used an app to book a cab, you are already using Machine Learning to an extent. It provides a personalized application that is unique to you. Automatically detects your location and provides options to either go home or office or any other frequent place based on your History and Patterns.

**Advantages and Challenges:**

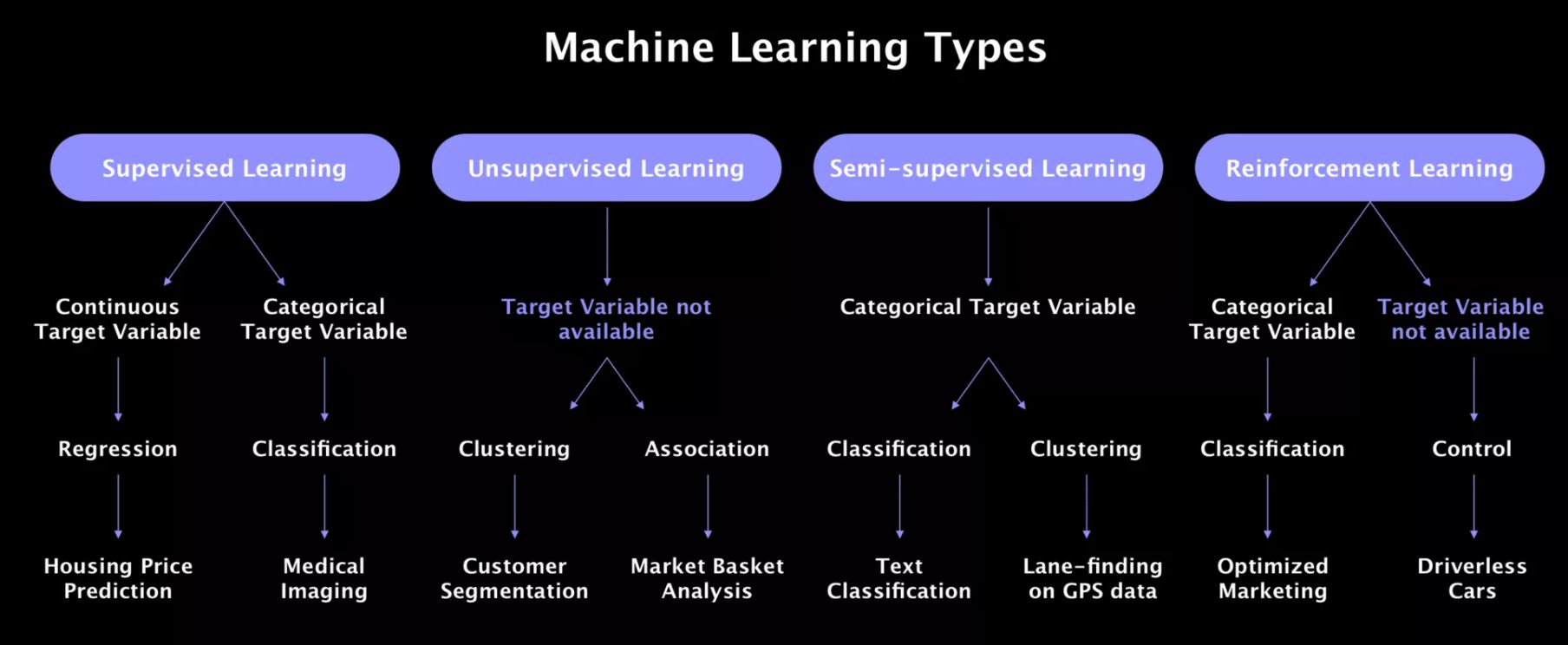
**Advantages:**

1. Easily identifies trends and patterns
2. No human intervention needed (automation)
3. Continuous Improvement
4. Handling multi-dimensional and multi-variety data
5. Wide Applications
6. Solve unprogrammable tasks

**Disadvantages:**

1. Requires Big Data
2. Requires Supervised Data
3. Time, Space and Resources
4. Data Acquisition
5. High error-susceptibility
6. Algorithm Selection

**ML Paradigms:**

****

* **Supervised Learning**

Supervised learning means that the training data that we feed our algorithm with has labels on it. As a result, it maps the input (training data) to output and labels the data accordingly.

Supervised learning is a machine learning task in which a function maps the input to output data using the provided input-output pairs.

1. **Classification:**

The model assigns a category to the target variable. The target variable is the category you want your algorithm to find.

1. **Regression:**

The model assigns a continuous variable to the target variable.

* **Unsupervised Learning**

In unsupervised learning, we don’t have a column for the target variable — we actually don’t really know what we are looking for.

In this type of learning paradigm, the computer is provided with just the input to develop a learning pattern. It is basically Learning from no results!!

1. **Clustering:**

identifies the consumption behavior of each category and identifies how many groups we should have and who should be placed into each group. In summary, our models divide the dataset according to its similarities. This is called Clustering, a subcategory of Unsupervised Learning.

1. **Association:**

An association rule learning problem is where you want to discover rules that describe large portions of your data

* **Reinforcement Learning**

Reinforcement Learning is a type of Machine Learning, and thereby also a branch of Artificial Intelligence. It allows machines and software agents to automatically determine the ideal behavior within a specific context, in order to maximize its performance.

**CONCEPT LEARNING**

‘’ Problem of searching through a predefined space of potential hypotheses for the hypothesis that best fits the training examples”. Concept Learning is a type of Supervised Learning.

Teaching a machine to distinguish between examples and non-examples of ideas such as symphony, anger, beauty, dog, cat etc. is called concept learning. In concept learning, we aim to use data to teach a machine to solve a binary classification problem. That is, to classify a data-point as either belonging to or not belonging to a particular concept or idea.

Feature Space:

a feature space is just the set of all possible values for a chosen set of features from that data. It is always possible to represent feature values and thus a feature space using only numbers, and further to do so in such a way that the feature space can be interpreted as a real space.

**FIND -S ALOGORITHM**

(CONSIDER ONLY +VE EXAMPLES)

The find-S algorithm starts with the most specific hypothesis and generalizes this hypothesis each time it fails to classify an observed positive training data. Hence, the Find-S algorithm moves from the most specific hypothesis to the most general hypothesis.

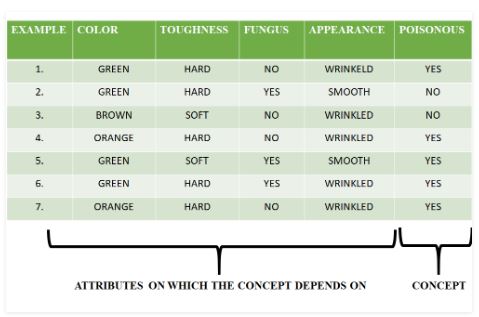
**Most general hypothesis: {?, ?, ?, ?, ?, ?}**

**Most specific hypothesis:** **{ϕ, ϕ, ϕ, ϕ, ϕ, ϕ}**

Steps Involved in Find-S:

1. Start with the most specific hypothesis.   
   **h = {ϕ, ϕ, ϕ, ϕ, ϕ, ϕ}**
2. Take the next example and if it is negative, then no changes occur to the hypothesis.
3. If the example is positive and we find that our initial hypothesis is too specific then we update our current hypothesis to a general condition.
4. Keep repeating the above steps till all the training examples are complete.
5. After we have completed all the training examples, we will have the final hypothesis when can use to classify the new examples.

EXAMPLE:



First, we consider the hypothesis to be a more specific hypothesis. Hence, our hypothesis would be:

h = {ϕ, ϕ, ϕ, ϕ, ϕ, ϕ}

Consider example 1:

The data in example 1 is {GREEN, HARD, NO, WRINKLED}. We see that our initial hypothesis is more specific and we have to generalize it for this example. Hence, the hypothesis becomes:

h = {GREEN, HARD, NO, WRINKLED}

Consider example 2:

Here we see that this example has a negative outcome. Hence, we neglect this example and our hypothesis remains the same.

h = {GREEN, HARD, NO, WRINKLED}

Consider example 3:

Here we see that this example has a negative outcome. Hence, we neglect this example and our hypothesis remains the same.

h = {GREEN, HARD, NO, WRINKLED}

Consider example 4:

The data present in example 4 is {ORANGE, HARD, NO, WRINKLED}. We compare every single attribute with the initial data and if any mismatch is found we replace that particular attribute with a general case (”?”). After doing the process the hypothesis becomes:

h = {?, HARD, NO, WRINKLED }

Consider example 5:

The data present in example 5 is {GREEN, SOFT, YES, SMOOTH}. We compare every single attribute with the initial data and if any mismatch is found we replace that particular attribute with a general case (”?”). After doing the process the hypothesis becomes:

h = { ?, ?, ?, ? }

Since we have reached a point where all the attributes in our hypothesis have the general condition, example 6 and example 7 would result in the same hypothesizes with all general attributes.

h = { ?, ?, ?, ? }

Hence, for the given data the final hypothesis would be:

Final Hypothesis: h = { ?, ?, ?, ? }

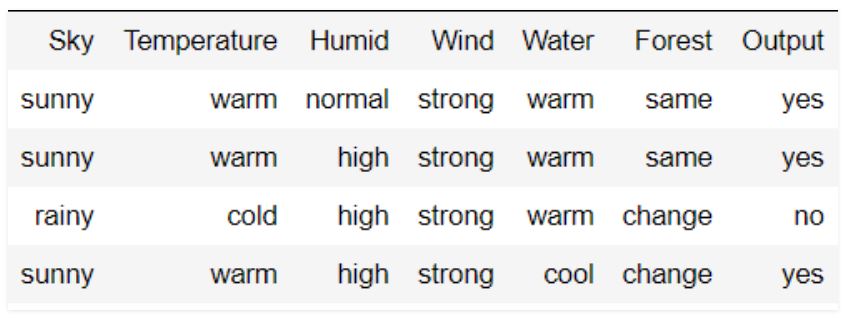
**CANDIDATE ELIMINATION ALGORITHM:**

(CONSIDER BOTH +VE AND -VE EXAMPLES)

The candidate elimination algorithm incrementally builds the version space given a hypothesis space H and a set E of examples. The examples are added one by one; each example possibly shrinks the version space by removing the hypotheses that are inconsistent with the example. The candidate elimination algorithm does this by updating the general and specific boundary for each new example.

* You can consider this as an extended form of Find-S algorithm.
* Consider both positive and negative examples.
* Actually, positive examples are used here as Find-S algorithm (Basically they are generalizing from the specification).
* While the negative example is specified from generalize form**.**

EXAMPLE:



**Initially:** G =[[?, ?, ?, ?, ?, ?], [?, ?, ?, ?, ?, ?], [?, ?, ?, ?, ?, ?],

[?, ?, ?, ?, ?, ?], [?, ?, ?, ?, ?, ?], [?, ?, ?, ?, ?, ?]]

S = [Null, Null, Null, Null, Null, Null]

**For instance 1 :** <'sunny','warm','normal','strong','warm ','same'> and positive output.

G1 = G

S1 = ['sunny','warm','normal','strong','warm ','same']

**For instance 2 :** <'sunny','warm','high','strong','warm ','same'> and positive output.

G2 = G

S2 = ['sunny','warm',?,'strong','warm ','same']

**For instance 3 :** <'rainy','cold','high','strong','warm ','change'> and negative output.

G3 = [['sunny', ?, ?, ?, ?, ?], [?, 'warm', ?, ?, ?, ?], [?, ?, ?, ?, ?, ?],

[?, ?, ?, ?, ?, ?], [?, ?, ?, ?, ?, ?], [?, ?, ?, ?, ?, 'same']]

S3 = S2

**For instance 4 :** <'sunny','warm','high','strong','cool','change'> and positive output.

G4 = G3

S4 = ['sunny','warm',?,'strong', ?, ?]

At last, by synchronizing the G4 and S4 algorithm produce the output.

**MODULE -2**

**(SUPERVISED LEARNING - 1)**

**Learning a Class from Examples, Linear, Non-linear, Multi-class and Multi-label classification, Generalization error bounds: VC Dimension, Decision Trees: ID3, Classification and Regression Trees, Regression: Linear Regression, Multiple Linear Regression, Logistic Regression.**

**REGRESSION:**

Regression is a technique for investigating the relationship between independent variables or features and a dependent variable or outcome. It’s used as a method for predictive modelling in machine learning, in which an algorithm is used to predict continuous outcomes.

**Types of Regression:**

1. Simple Linear Regression:

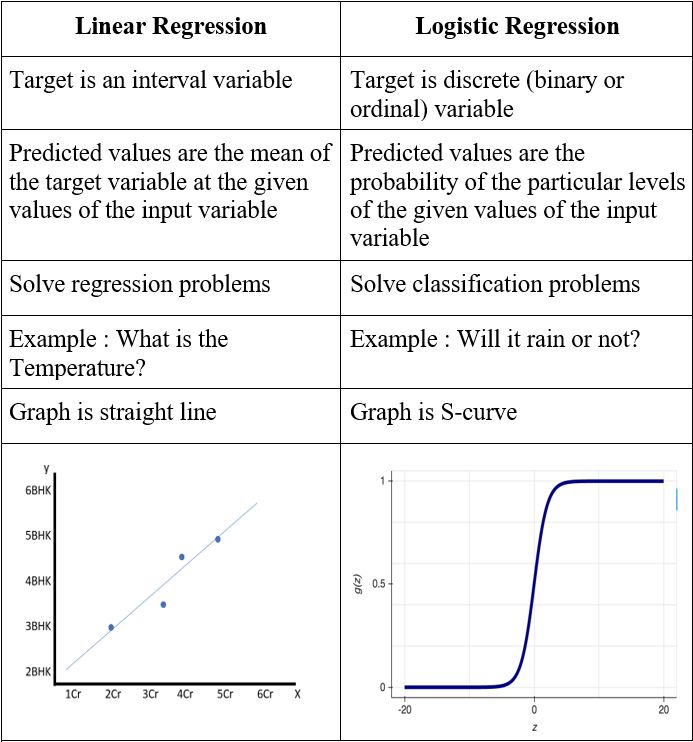
Simple Linear regression is a linear regression technique which plots a straight line within data points to minimise error between the line and the data points.

1. Multiple linear regression:

Multiple linear regression is a technique used when more than one independent variable is used. Polynomial regression is an example of a multiple linear regression technique. It is a type of multiple linear regression, used when there is more than one independent variable. It achieves a better fit in the comparison to simple linear regression when multiple independent variables are involved. The result when plotted on two dimensions would be a curved line fitted to the data points.

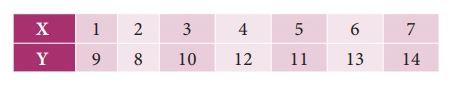
1. Logistic regression:

Logistic regression is used when the dependent variable can have one of two values, such as true or false, or success or failure. Logistic regression models can be used to predict the probability of a dependent variable occurring. Generally, the output values must be binary. A sigmoid curve can be used to map the relationship between the dependent variable and independent variables.

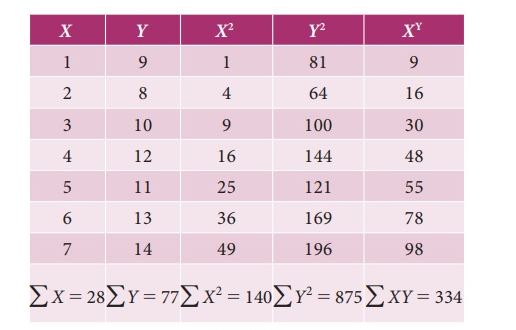


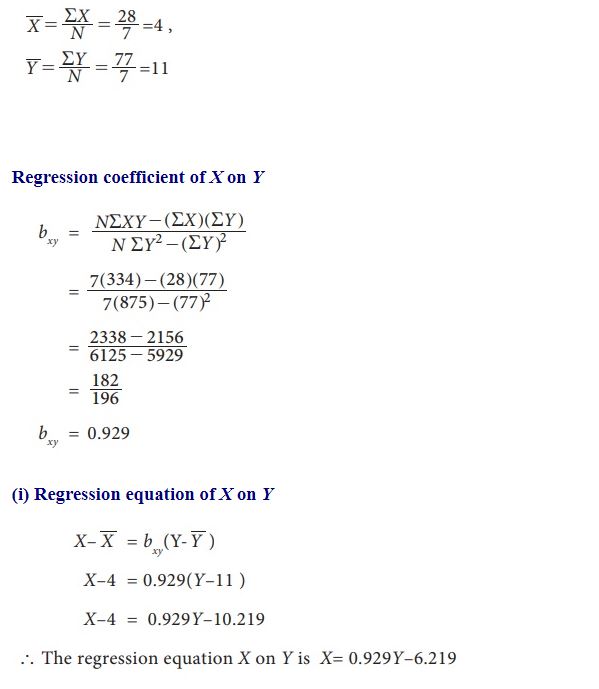
**PROBLEMS:**

1. Calculate the regression coefficient and obtain the lines of regression for the following data (SIMPLE LINEAR REGRESSION)



SOLUTION:





**CLASSIFICATION:**

Classification in machine learning and statistics is a supervised learning approach in which the computer program learns from the data given to it and make new observations or classifications.

**Types of Classification:**

1. **Binary Classification:**

It is a type of classification with two outcomes, for e.g. – either true or false, spam or not.

Algorithms:

1. Logistic Regression
2. k-Nearest Neighbors
3. Decision Trees
4. Support Vector Machine
5. Naive Bayes
6. **Multi-Class Classification:**

The classification with more than two classes, in multi-class classification each sample is assigned to one and only one label or target.

Algorithms:

1. k-Nearest Neighbors.
2. Decision Trees.
3. Naive Bayes.
4. Random Forest.
5. Gradient Boosting.
6. **Multi-label Classification**

This is a type of classification where each sample is assigned to a set of labels or targets.

This is unlike binary classification and multi-class classification, where a single class label is predicted for each example.

Algorithms:

1. Multi-label Decision Trees
2. Multi-label Random Forests
3. Multi-label Gradient Boosting

**DECISION TREES**

A decision tree is a type of supervised machine learning used to categorize or make predictions based on how a previous set of questions were answered. The model is a form of supervised learning, meaning that the model is trained and tested on a set of data that contains the desired categorization.

A decision tree is a graphical representation of all possible solutions to a decision based on certain conditions.

Decision Tree models are created using 2 steps: Induction and Pruning.

1. **CLASSIFICATION TREES (Discrete/categorical):**

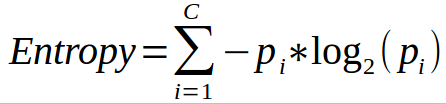
where the outcome was a variable like ‘fit’ or ‘unfit’. Here the decision variable is Categorical

1. **REGRESSION TREES (Continuous):**

Here the decision or the outcome variable is Continuous, there are many algorithms out there which construct Decision Trees, but one of the best is called as ID3 Algorithm. ID3 Stands for Iterative Dichotomiser 3(produces more than two children at node in tree).

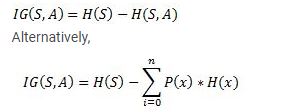
**ENTROPY (Predictability of outcome /Randomness):**

Entropy, also called as Shannon Entropy is denoted by H(S) for a finite set S, is the measure of the amount of uncertainty or randomness in data



**INFORMATION GAIN:**

Information gain is also called as Kullback-Leibler divergence denoted by IG(S,A) for a set S is the effective change in entropy after deciding on a particular attribute A. It measures the relative change in entropy with respect to the independent variables



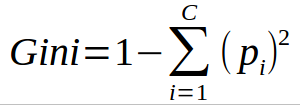
Information gain is calculated after node splitting.

**Information gain = Entropy(s) – [ (average weight) \* Entropy (each feature)]**

Information Gain = Entropy before splitting - Entropy after splitting

**Gini Index:**

Gini Index is the measure of impurity or the purity that is used in building a decision tree in the CART Algorithm.



**CART ALGORITHM:**

**ONLY BINARY TREES**

**USE “GINI INDEX” FOR BEST ATTRIBUTE MEASUREMENT**

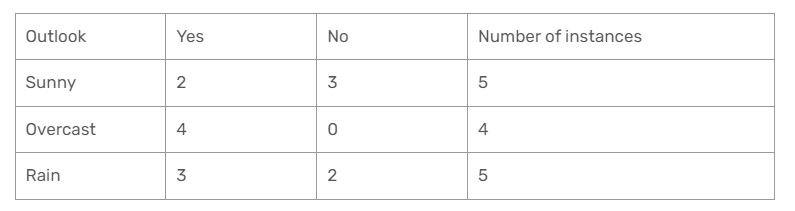
The CART stands for Classification and Regression Trees is a greedy algorithm that greedily searches for an optimum split at the top level, then repeats the same process at each of the subsequent levels.

PROBLEM:



Gini index is a metric for classification tasks in CART.

1. **OUTLOOK**

****

**Gini(Temp=Hot) = 1 – (2/4)2 – (2/4)2 = 0.5**

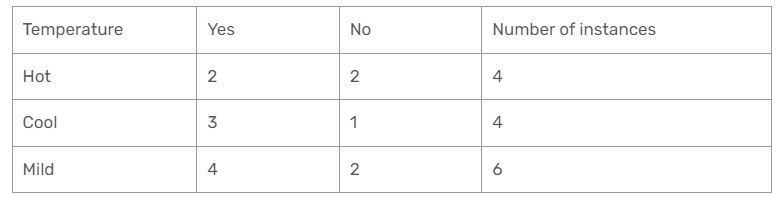
**Gini(Temp=Cool) = 1 – (3/4)2 – (1/4)2 = 1 – 0.5625 – 0.0625 = 0.375**

**Gini(Temp=Mild) = 1 – (4/6)2 – (2/6)2 = 1 – 0.444 – 0.111 = 0.445**

**We’ll calculate weighted sum of gini index for temperature feature**

**Gini(Temp) = (4/14) x 0.5 + (4/14) x 0.375 + (6/14) x 0.445 = 0.142 + 0.107 + 0.190 = 0.439**

1. **Temperature**

****

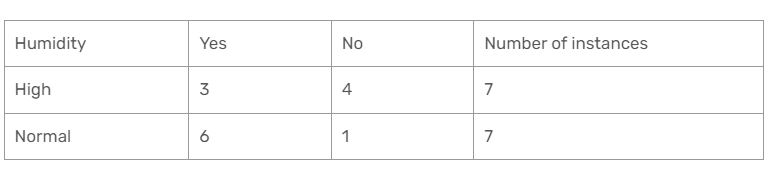
**Gini(Humidity=High) = 1 – (3/7)2 – (4/7)2 = 1 – 0.183 – 0.326 = 0.489**

**Gini(Humidity=Normal) = 1 – (6/7)2 – (1/7)2 = 1 – 0.734 – 0.02 = 0.244**

**Weighted sum for humidity feature will be calculated next**

**Gini(Humidity) = (7/14) x 0.489 + (7/14) x 0.244 = 0.367**

1. **Humidity**

****

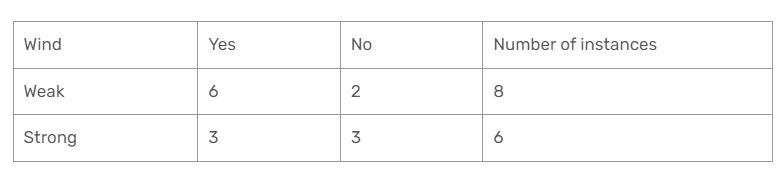
**Gini(Humidity=High) = 1 – (3/7)2 – (4/7)2 = 1 – 0.183 – 0.326 = 0.489**

**Gini(Humidity=Normal) = 1 – (6/7)2 – (1/7)2 = 1 – 0.734 – 0.02 = 0.244**

**Weighted sum for humidity feature will be calculated next**

**Gini(Humidity) = (7/14) x 0.489 + (7/14) x 0.244 = 0.367**

1. **WIND**

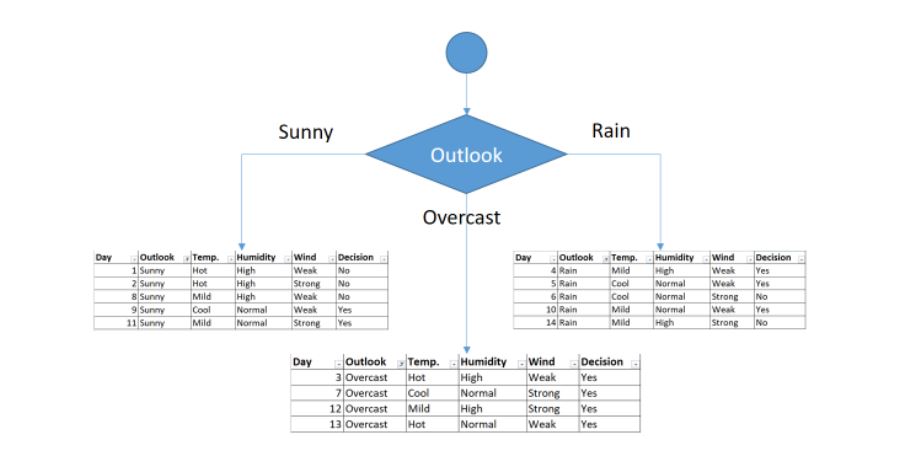
****

**Gini(Wind=Weak) = 1 – (6/8)2 – (2/8)2 = 1 – 0.5625 – 0.062 = 0.375**

**Gini(Wind=Strong) = 1 – (3/6)2 – (3/6)2 = 1 – 0.25 – 0.25 = 0.5**

**Gini(Wind) = (8/14) x 0.375 + (6/14) x 0.5 = 0.428**

**Now, Lowest Gini attribute will highest priority.**

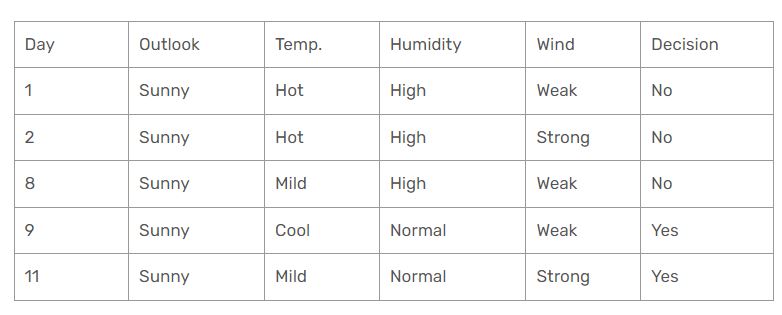
****

You might realize that sub dataset in the overcast leaf has only yes decisions. This means that overcast leaf is over.

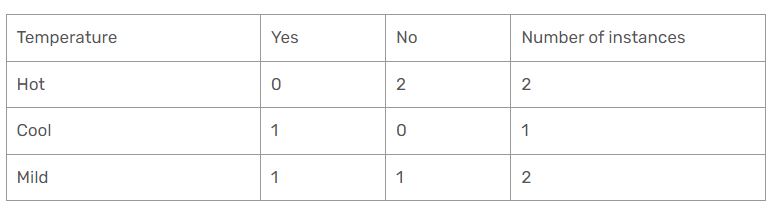
**We will apply same principles to those sub datasets in the following steps.**

**Focus on the sub dataset for sunny outlook. We need to find the gini index scores for temperature, humidity and wind features respectively.**

1. **SUNNY**

****

**Gini of temperature for sunny outlook:**

****

**Gini(Outlook=Sunny and Temp.=Hot) = 1 – (0/2)2 – (2/2)2 = 0**

**Gini(Outlook=Sunny and Temp.=Cool) = 1 – (1/1)2 – (0/1)2 = 0**

**Gini(Outlook=Sunny and Temp.=Mild) = 1 – (1/2)2 – (1/2)2 = 1 – 0.25 – 0.25 = 0.5**

**Gini(Outlook=Sunny and Temp.) = (2/5)x0 + (1/5)x0 + (2/5)x0.5 = 0.2**

**Gini of humidity for sunny outlook**

**Gini(Outlook=Sunny and Humidity=High) = 1 – (0/3)2 – (3/3)2 = 0**

**Gini(Outlook=Sunny and Humidity=Normal) = 1 – (2/2)2 – (0/2)2 = 0**

**Gini(Outlook=Sunny and Humidity) = (3/5)x0 + (2/5)x0 = 0**

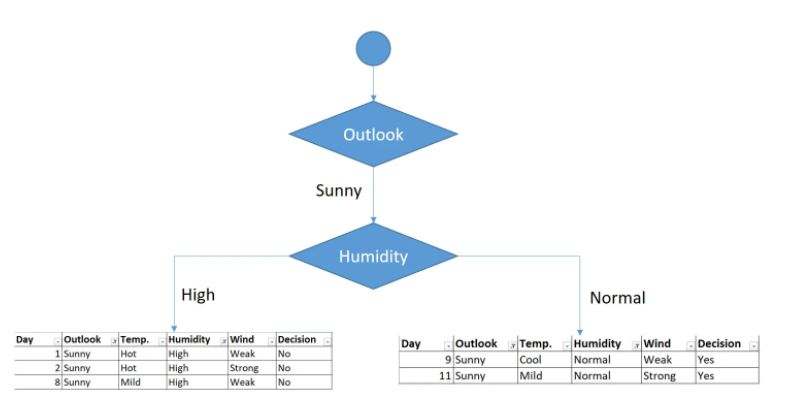
**Gini of wind for sunny outlook**

**Gini(Outlook=Sunny and Wind=Weak) = 1 – (1/3)2 – (2/3)2 = 0.266**

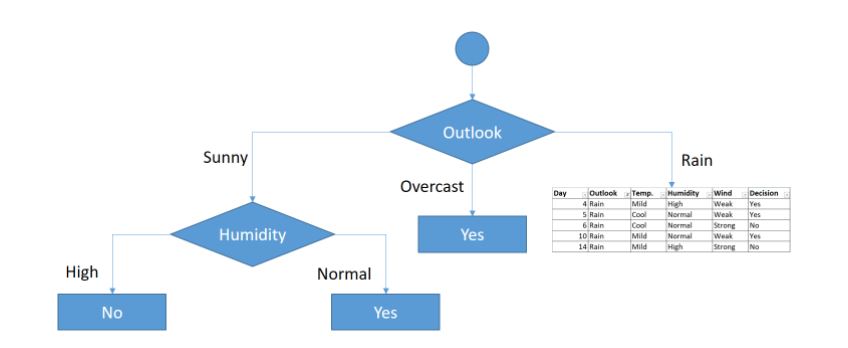
**Gini(Outlook=Sunny and Wind=Strong) = 1- (1/2)2 – (1/2)2 = 0.2**

**Gini(Outlook=Sunny and Wind) = (3/5)x0.266 + (2/5)x0.2 = 0.46**

**We’ll put humidity check at the extension of sunny outlook.**

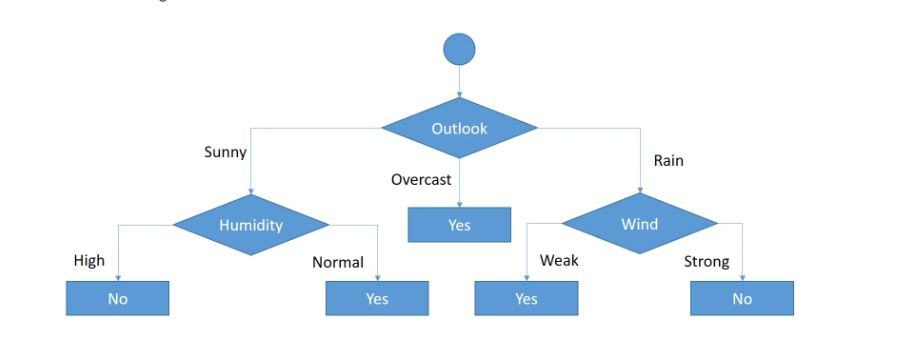
****

As seen, decision is always no for high humidity and sunny outlook. On the other hand, decision will always be yes for normal humidity and sunny outlook. This branch is over.



Now , repeating the steps for Rain.

Finally we get,



**ID-3 ALGORITHM**