Machine Learning Introduction

Definition

Machine learning is a sub-field of artificial intelligence (AI) that provides **systems** the ability to **automatically learn** and **improve** from **experience without** being **explicitly** programmed.

Machine Learning algorithms are classified as either supervised or unsupervised

Supervised
In supervised learning, we use known or labeled data for the training data.
For the training procedure, the input is a known training data set with its **corresponding labels**, and the learning algorithm produces an inferred function to finally make predictions about some new unseen observations that one can give to the model.

Supervised models can be further grouped into regression and classification cases:

- **Classification**: A classification problem is when the output variable is a category e.g. “disease” / “no disease”.
  Ex: NaiveBayes, Logistic regression, decision tree, random forest, KNN etc
- **Regression**: A regression problem is when the output variable is a real continuous value e.g. stock price prediction
  Ex: Linear regression, Polynomial regression etc.

Unsupervised
In unsupervised learning, the training data is unknown and unlabeled - meaning that no one has looked at the data before
This data is fed to the Machine Learning algorithm and is used to train the model. The trained model tries to search for a pattern and give the desired response.
The system doesn’t predict the right output, but instead, it explores the data and can draw inferences from datasets to describe hidden structures from unlabeled data.

Unsupervised models can be further grouped into clustering and association cases.

- **Clustering**: A clustering problem is where you want to unveil the inherent groupings in the data, such as grouping animals based on some characteristics/features e.g. number of legs.
  Ex: K-Means
- **Association:** An association rule learning is where you want to discover association rules such as people that buy X also tend to buy Y.
  Ex: Apriori

Reinforcement Learning

Reinforcement Learning is a feedback-based Machine learning technique in which an agent learns to behave in an environment by performing the actions and seeing the results of actions. For each good action, the agent gets positive feedback, and for each bad action, the agent gets negative feedback or penalty.
Ex: Q-Learning

Programs

20 June 2021 08:18 PM

1. Naïve Bayes - playtennis.csv
2. Naïve Bayes - gender.csv
3. DecisionTree - titanic.csv
4. SimpleLinearRegression - weather.csv
5. KNN - (Default) Breastcancer
6. Mul LinearRegression - wine-quality.csv
7. Single Layer Perceptron - (default) iris
8. Single Layer Perceptron - (default) MNIST digit
9. Q -Learning Tic Tac Toe - policy1,policy2
10. K means - iris.csv
11. SVM - creditcard.csv
12. Ensemble model - data.csv

Programming Steps

Programming Steps:

1. **Import dataset**

```
import pandas as pd
df = pd.read_csv("…..")
```

2. **Preprocess data**
   a. **Check for null values (if any)**

```
df.isnull().sum()
```

   b. **Apply encoding** using LabelEncoder or **Scale the data**

```
from sklearn.preprocessing import StandardScaler
from sklearn.preprocessing import LabelEncoder
```

3. **Find the features and target**

```
x = df.iloc[:,:-1]
y = df.iloc[:,-1]
```

4. **Split data into train and test data**

```
from sklearn.model_selection import train_test_split
x_train , x_test , y_train , y_test = train_test_split(x,y,
                        test_size= 0.25,random_state= 25)
```

5. **Apply ML algorithm**
   a. **Import the algorithm**

```
from sklearn.naive_bayes import GaussianNB
from sklearn.neighbors import KNeighborsClassifier
from sklearn.tree import DecisionTreeClassifier
from sklearn.linear_model import LinearRegression
from sklearn.cluster import Kmeans
from sklearn.svm import SVC
from sklearn.linear_model import LogisticRegression
From sklearn.cluster import KMeans
.
Etc…
```

   b. **Train the model**

```
model.fit(x_train,y_train)
```

7. **Find the accuracy**

```
from sklearn.metrics import accuracy_score
y_pred = model.predict(X_test)
acc = accuracy_score(y_test, y_pred)
```

Naïve Bayes Algorithm

Naive Bayes is a supervised Machine Learning algorithm inspired by the Bayes theorem. It works on the principles of conditional probability. Naive Bayes is a classification algorithm for binary and multi-class classification. The Naive Bayes algorithm uses the probabilities of each attribute belonging to each class to make a prediction.

Example

What is the probability of playing tennis when it is sunny, hot, highly humid and windy?

$$P(A|B) = \frac{P(B|A) * (P(A)}{P(B)}$$

- P(A|B) - **Posterior Probability**
  The conditional probability of the response variable (target variable) given the training data inputs.
- P(A) - **Prior Probability**
  The probability of the response variable (target variable).
- P(B) - **Evidence**
  The probability of the training data.
- P(B|A) - **Likelihood**
  The conditional probability of the training data given the response variable.

Type of Naive Bayes Algorithm

Python's Scikitlearn gives the user access to the following 3 Naive Bayes models.

1. Gaussian
   The gaussian NB Alogorithm assumes all contnuous features (predictors) and all follow a Gaussian (Normal Distribution).
2. Multinomial
   Multinomial NB is suited for discrete data that have frequencies and counts. Spam Filtering and Text/Document Classification are two very well-known use cases.
3. Bernoulli
   Bernoulli is similar to Multinomial except it is for boolean/binary features. Like the multinomial method it can be used for spam filtering and document classification in which binary terms (i.e. word occurrence in a document represented with True or False).

Decision Tree

Decision Tree is a **Supervised learning technique** which is mostly used to solve classification problems
It is a tree-structured classifier, where **internal nodes represent the features of a dataset, branches represent the decision rules** and **each leaf node represents the outcome.**

Gini Index

- Measure of impurity
- Gini Index is a metric to measure how often a randomly chosen element would be incorrectly identified.
- **Gini index varies between values 0 and 1**, where 0 expresses the purity of classification, i.e. All the elements belong to a specified class or only one class exists there. And 1 indicates the random distribution of elements across various classes. The value of 0.5 of the Gini Index shows an equal distribution of elements over some classes.

$$\text{Gini Index} = 1 - \sum_{i=1}^{n} (P_i)^2$$

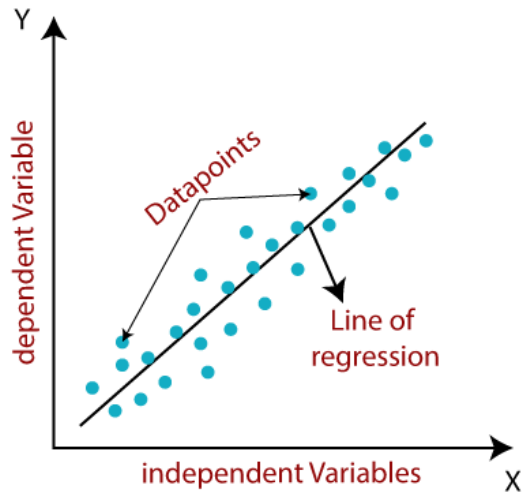
For this leaf, the Gini impurity = 1 - (the probability of "yes")^2 - (the probability of "no")^2

$$= 1 - \left(\frac{105}{105 + 39}\right)^2 - \left(\frac{39}{105 + 39}\right)^2$$

$$= 0.395$$

Linear Regression

Linear regression algorithm shows a linear relationship between a dependent (y) and one or more independent (x) variables.
Since linear regression shows the linear relationship, which means it finds how the value of the dependent variable is changing according to the value of the independent variable.
When working with linear regression, our main goal is to find the best fit line that means the error between predicted values and actual values should be minimized. The best fit line will have the least error.



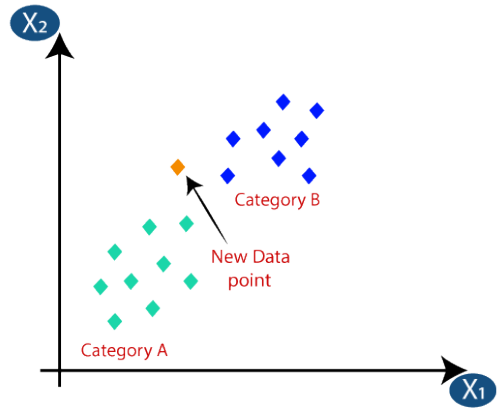
Linear regression can be further divided into two types of the algorithm:

- **Simple Linear Regression:**
  If a **single independent variable** is used to predict the value of a numerical dependent variable, then such a Linear Regression algorithm is called Simple Linear Regression.
- **Multiple Linear regression:**
  If **more than one independent variable** is used to predict the value of a numerical dependent variable, then such a Linear Regression algorithm is called Multiple Linear Regression.

K Nearest Neighbours

- KNN is a supervised machine learning algorithm
- K-NN algorithm assumes the similarity between the new case/data and available cases and put the new case into the category that is most similar to the available categories.
- K-NN algorithm stores all the available data and classifies a new data point based on the similarity. This means when new data appears then it can be easily classified into a well suite category by using K- NN algorithm.

Assume k = 5



Performance Metrics

The cost function is the technique of evaluating “the performance of our algorithm/model”

It takes both predicted outputs by the model and actual outputs and calculates how much wrong the model was in its prediction.

Regression cost Function

Error = y-y’

Where,

Y – Actual Input
Y’ – Predicted output

Mean Squared Error

$$MSE = \frac{\sum_{i=0}^{n}(y-y')^2}{n}$$

Mean Absolute Error

$$MAE = \frac{\sum_{i=0}^{n}|y-y'|}{n}$$

	Predicted Positives	Predicted Negatives
Positives	True Positives	False Negatives
Negatives	False Positives	True Negatives

- **Precision :** It is the number of correct positive results divided by the number of positive results predicted by the classifier.

$$Precision = \frac{TruePositives}{TruePositives + FalsePositives}$$

- **Recall :** It is the number of correct positive results divided by the number of ***all*** relevant samples (all samples that should have been identified as positive).

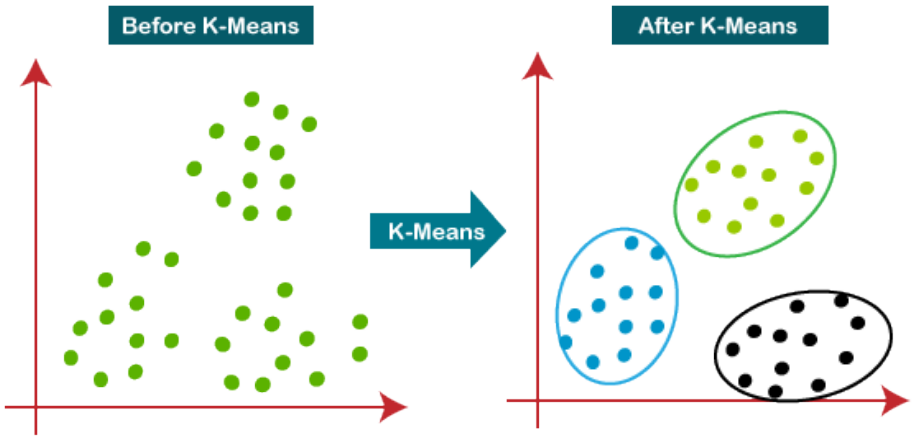
$$Precision = \frac{TruePositives}{TruePositives + FalseNegatives}$$

- **F1 Score**: is used to measure a test's accuracy
  It tells you how precise your classifier is

$$F1 = 2 * \frac{1}{\frac{1}{precision} + \frac{1}{recall}}$$

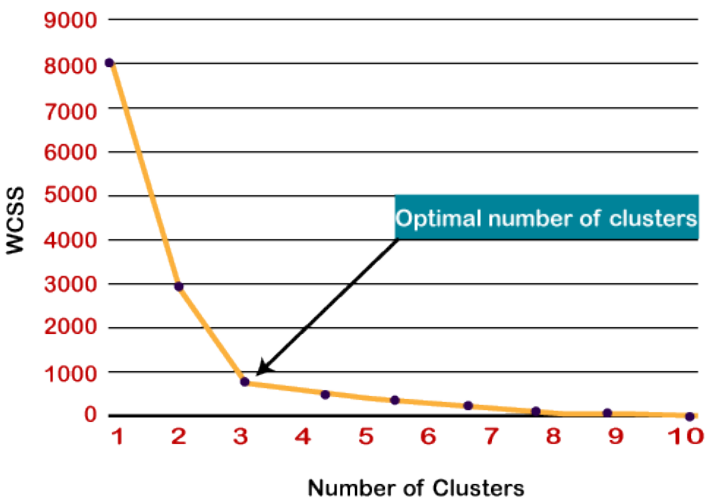
K Means Clustering

K-Means Clustering is an Unsupervised Learning Algorithm which groups the unlabeled dataset into different clusters. ***Kmeans Algorithm is an Iterative algorithm that divides a group of n datasets into k subgroups /clusters based on the similarity and their mean distance from the centroid of that particular subgroup/ formed***.



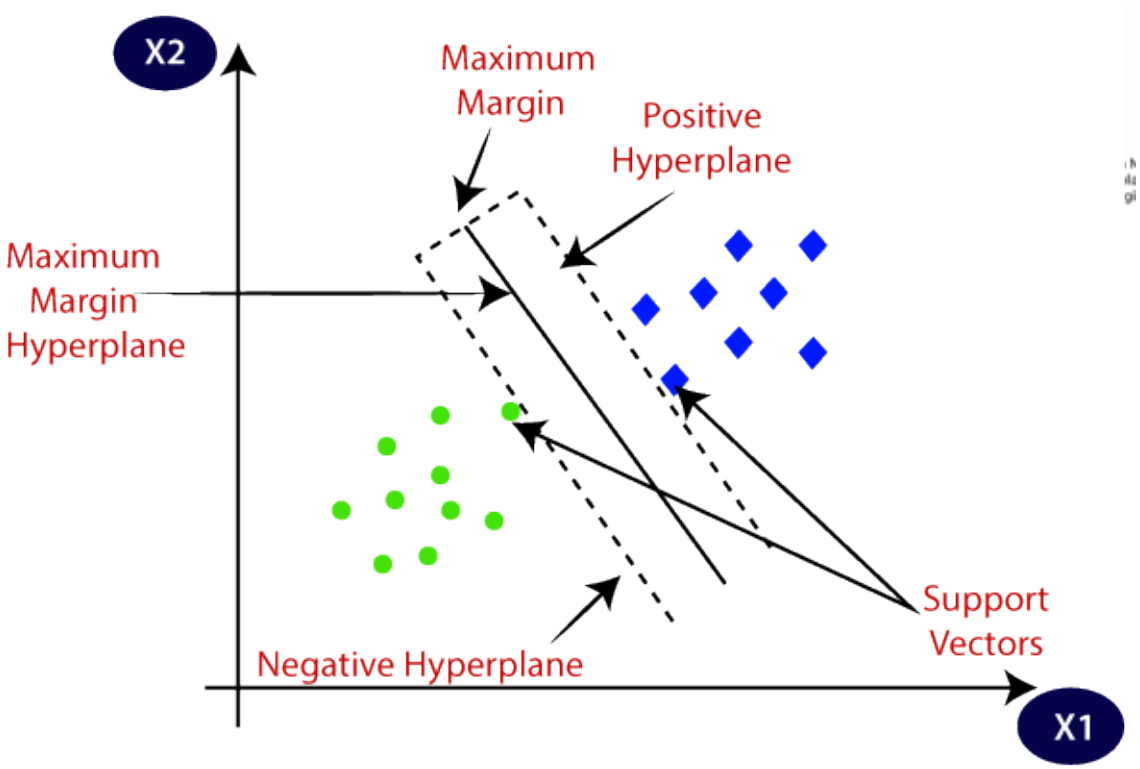
Elbow Method

The Elbow method is one of the most popular ways to find the optimal number of clusters. This method uses the concept of WCSS value. **WCSS** stands for **Within Cluster Sum of Squares**



SVM

- SVM is a supervised learning algorithm which is mainly used for classification
- The goal of the SVM algorithm is to create the best line or decision boundary that can segregate n-dimensional space into classes so that we can easily put the new data point in the correct category in the future.
- This best decision boundary is called a hyperplane.
- The data points or vectors that are the closest to the hyperplane and which affect the position of the hyperplane are termed as Support Vector. Since these vectors support the hyperplane, hence called a Support vector.



Voting Ensemble Model

A voting ensemble (or a "*majority voting ensemble*") is an ensemble machine learning model that combines the predictions from multiple other models.c

- **Classification Voting Ensemble**: Predictions are the majority vote of contributing models.
    - **Hard Voting**. Predict the class with the largest sum of votes from models
    - **Soft Voting**. Predict the class with the largest summed probability from models.

Perceptron

20 June 2021 06:46 PM

The perceptron is a single processing unit of any neural network
It consists of four parts:

- Inputs
- Weights
- Net sum
- Activation function