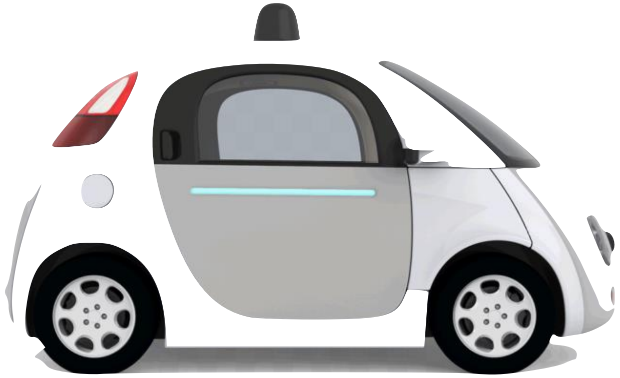
PRIME INTUIT
Finishing School

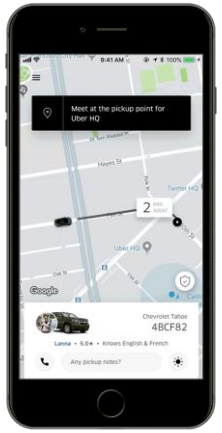
Getting started with Machine Learning

PRIME INTUIT
Finishing School

What is Artificial Intelligence



Google AI

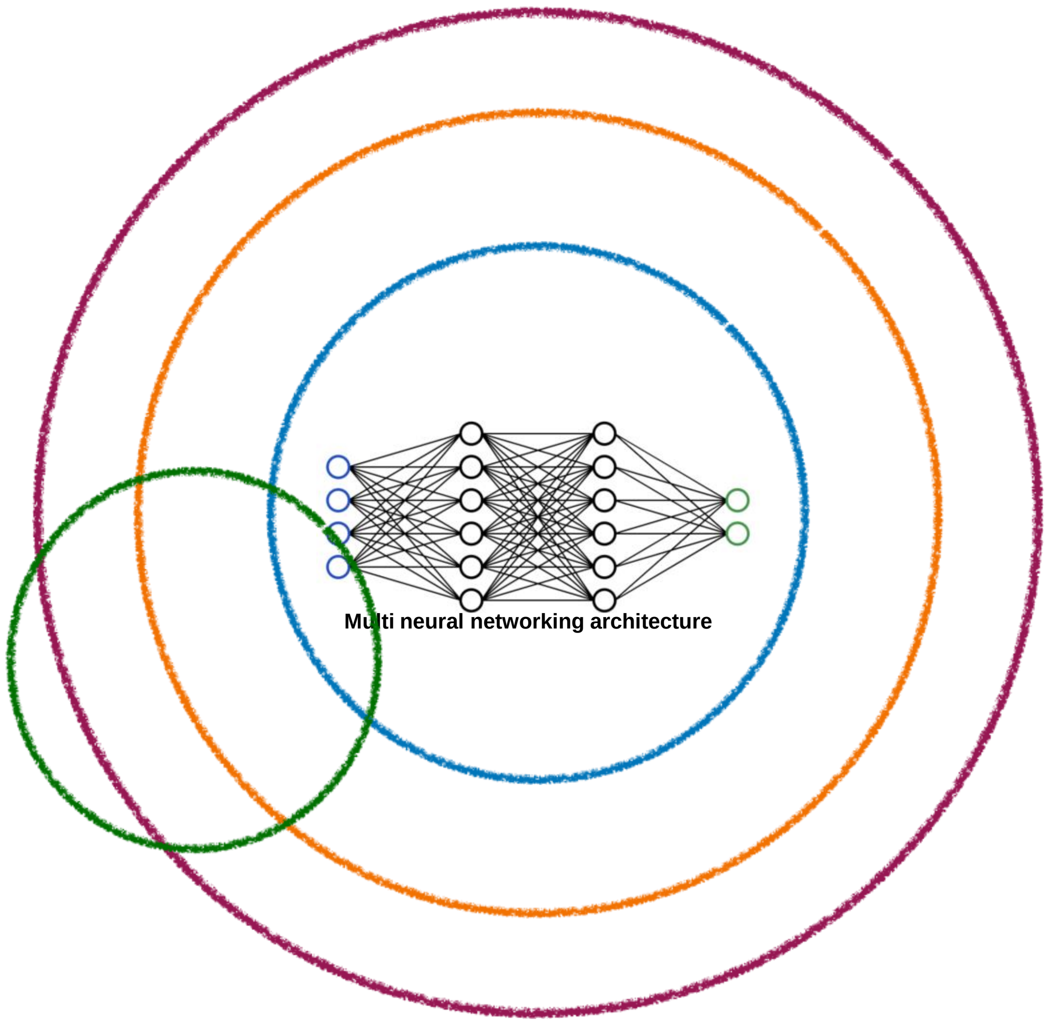


Artificially Providing (Applications) ability for machines to learn, think and make decisions on there own without human intervention

Ajathasatru’s Bhuta Vahana Yantra

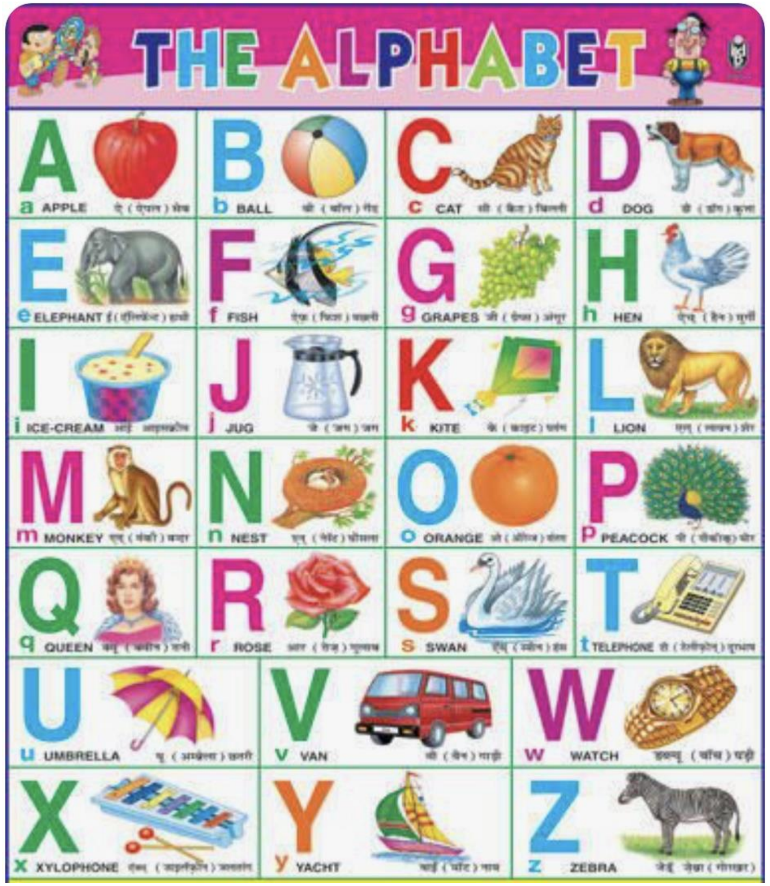
Talos the Greek Automaton

Different Terminologies



Artificial Intelligence	Applications that enables the machines to read, understand, think and learn from data	
Machine Learning	Statistical tools which can analyse, explore & understand data	• Supervised • Unsupervised • Semi Supervised • Reinforcement learning
Deep Learning	Architecture that helps machines to learn by mimicking human brain	• ANN • CNN • RNN • DNN
Data Science	Science of collecting, storing, processing and describing data using ML,DL and mathematical tools to predict or draw inferences from data	• Statistics • Probability • Learner algebra

Supervised learning



Input to Output Pair

Past Labeled Data / Training Data

Learning a function appropriate to the problem Y = f(x)

It infers a function from *labelled training data* consisting of a set of *training examples*. In supervised learning, each example is a *pair* consisting of an input object (typically a vector) and a desired output value (also called the *supervisory signal*).

Classification / regression

Common Use cases:

Bioinformatics
Spam detection
Speech recognition

Unsupervised learning

Data is not labeled classified or categorized

Mathematical model tries to identify similarities in dataset and based on that it tries to deduce a structure present in the input data.

Clustering / association problems

Common Use cases:

Bioinformatics
Spam detection
Speech recognition

PRIME INTUIT
Finishing School

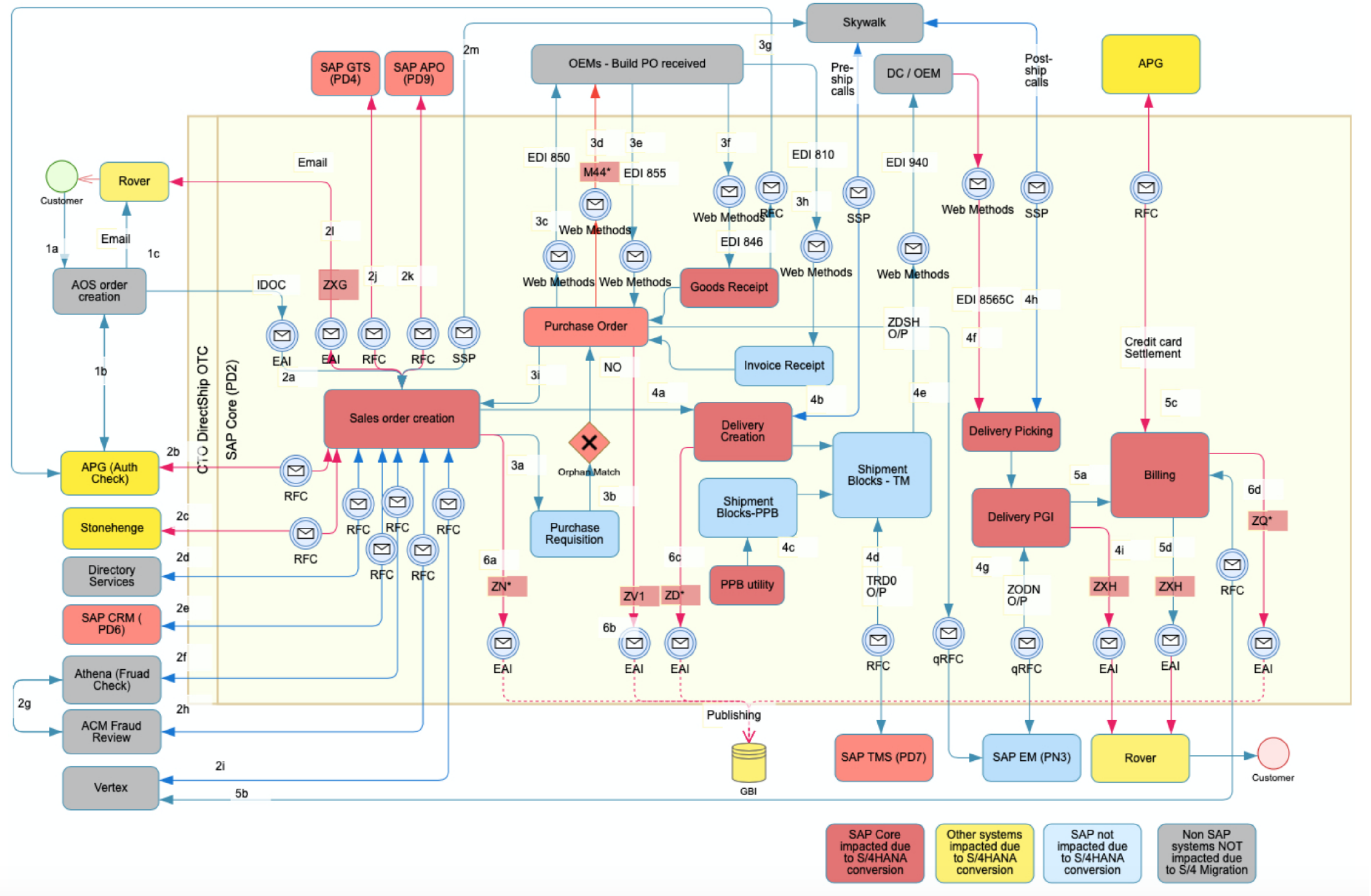
Semi-supervised learning

Dataset is a mixture of both labeled and unlabeled data. is not labeled classified or categorized

Mathematical model uses labeled data to learn the structure of unlabeled data and tries to make predictions.

Common Use cases:

Fraud Detection
Speech Analysis
Internet content - classification



Skywalk
SAP GTS (PD4)
SAP APO (PD9)
OEMs - Build PO received
DC / OEM
APG
2m
3g
Pre-ship calls
Post-ship calls
Customer
Rover
Email
1a
1c
AOS order creation
IDOC
ZXG
2l
2j
2k
EDI 850
3d
3e
M44*
EDI 855
3c
Web Methods
3f
Web Methods
RFC
EDI 846
Goods Receipt
EDI 810
3h
SSP
Web Methods
EDI 940
Web Methods
EDI 8565C
4h
SSP
RFC
Credit card Settlement
Purchase Order
EAI
RFC
SSP
2a
1b
CTO DirectShip OTC
SAP Core (PD2)
Sales order creation
3i
NO
Invoice Receipt
ZDSH O/P
4f
4a
4b
4e
5c
Delivery Creation
Shipment Blocks - TM
Delivery Picking
Billing
APG (Auth Check)
2b
3a
Orphan Match
5a
6d
Stonehenge
2c
3b
Purchase Requisition
Shipment Blocks-PPB
Delivery PGI
ZQ*
Directory Services
2d
6a
6c
4c
4d
4g
4i
5d
ZN*
ZV1
ZD*
PPB utility
TRDO O/P
ZODN O/P
ZXH
SAP CRM (PD6)
2e
6b
qRFC
Athena (Fruad Check)
2f
2g
ACM Fraud Review
2h
Publishing
GBI
SAP TMS (PD7)
SAP EM (PN3)
Vertex
2i
5b
SAP Core impacted due to S/4HANA conversion
Other systems impacted due to S/4HANA conversion
SAP not impacted due to S/4HANA conversion
Non SAP systems NOT impacted due to S/4 Migration

Reinforcement machine learning

Specialized subfield of machine learning

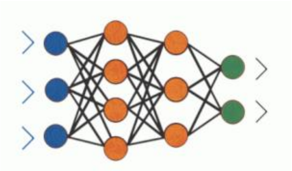
Learning is from action in a contained environment

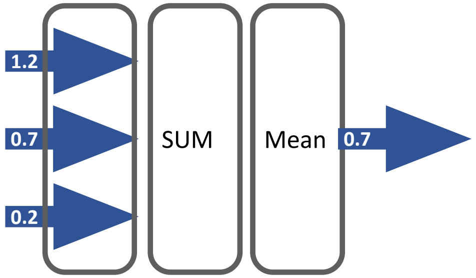
Common Use cases:

Self Driving cars
Robotics
Traffic light controls
Personalised Recommendations
Real time bidding
dynamic treatment regimes(DTRs) in chronic disease

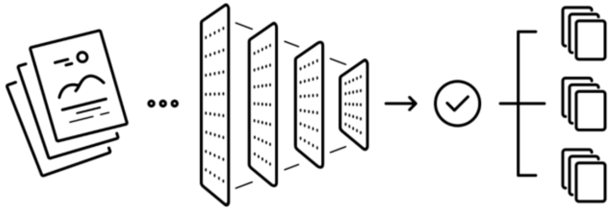
Different Terminologies

ANN

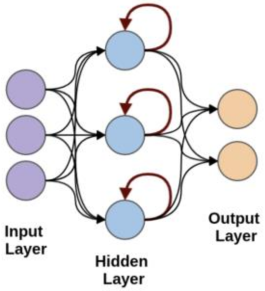




CNN



RNN



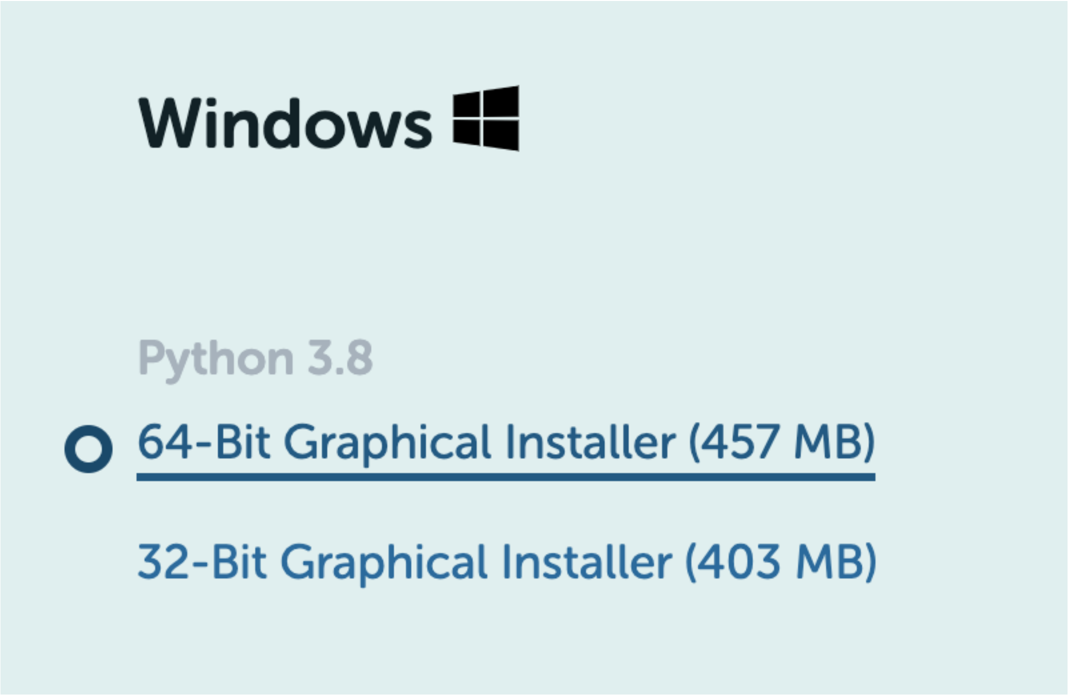
	Data Science	Artificial Intelligence
Scope	Involves various underlaying data operations	Limited to the implementation of ML algorithms
Type of Data	Structured and unstructured	Standardized in the form of embeddings and vectors
Tools	R, Python, SAS, SPSS,Keran, Scikit Learn	Scikit Learn, Kaffe, PyTorch,Shogun
Applications	Advertising, Marketing, internet search engine	Manufacturing, Automation, Robotics, Healthcare

Natural language processing

- What is Natural Language
  - It is the language spoken by humans to express thoughts, actions, feelings, intents.
  - What ever a human speaks, writes, listens to is in the form of Natural Language
- Natural Language Processing or NLP is
  - Machine/Program that interacts with human, understand and process the Natural Language human beings speak
  - Application of computational techniques to the analysis and synthesis of Natural Language and Speech
  - Based on Statistical Language Modeling to “understand” the words

Getting started with Anaconda

https://www.anaconda.com/products/individual



Machine Learning

- ✓ What is Machine Learning
  - ✓ How does it contrast with other means of problem solving
- ✓ When to use Machine Learning
- ✓ How to use Machine Learning (Classification Problems)

PRIME INTUIT
Finishing School

Machine Learning

Working with Real life problem – Spam Detection

- Classify emails as Spam or Ham
- Figure this out before the email is sent to the inbox

Machine Learning

Spam detection – Rule based approach

- Define set of rules to identify spam
- Rules can be intuitive or logical
- Requires domain knowledge
- Needs access to historical data

Machine Learning

Spam detection – Rule based approach (Technique)

- Blacklist
  - Emails from specific IP address
  - Specific words and Pharses
- Whitelist
  - Emails form contacts of contacts of contacts
  - Certain domains

Machine Learning

Spam detection – Rule based approach

From: Spammer@donottrust.com
Subject: You won a lottery !

Nirionline,

Congratulations! You won a 10 million lottery, we need the following details to credit the same to your account.

PRIME INTUIT
Finishing School

Machine Learning

Spam detection – Rule based approach

From: Spammer@donottrust.net
Subject: look who won a lucky draw !

Niranjan,

You have won a lucky draw, we need the following details to credit the same to your account.

PRIME INTUIT
Finishing School

Machine Learning

Spam detection – Rule based approach (limitations)

- Rules get outdated
- Rules are static – Spammers are dynamic
- Hard to maintain many rules
- Rethinking rules is time consuming
- Easier for spammer to get around your rules than the time taken for you to build the rules.

Spam detection – ML based approach

Input

Email

ML Based Spam Classifier

Output

Spam / Ham Output

A larger collection (corpus) of spam and ham emails

Classifiers will look for patterns for spam / ham emails

Machine Learning

Spam detection – ML based approach (Advantages)

- Algorithm can vary approach based on data
- Corpus can be updated based on user feedback
- Can detect patterns not visible to humans
- May be less complex than rule based - approach

Machine Learning

What is machine learning ?

It’s the ability of a program to read, understand and learn from the data and there by draw inferences about the data.

Machine Learning

When to use machine learning

- The alternative is a complex set of hard to maintain rules
- Dynamic environments with changing data
- Rule based solutions can not solve the problem
- To get insights on to your data

Machine Learning

ML – based Vs Rule based

ML Based	Rule Based
Dynamic	Static
Experts optional	Experts required
Need corpus	Corpus required
Training required	No training

Types of Machine Learning

PRIME INTUIT
Finishing School

Machine Learning

Supervised Learning

- Learning under the supervision of a coach / Teacher
- ML algorithm given data along with labels / responses
- Learns from looking at questions and answers

Machine Learning

Supervised Learning

- Classification
  - Spam or Ham
  - Mammal or fish or reptiles or birds or Amphibian
- Regression (predict numeric values)
  - Income of shopper

Machine Learning

Supervised Learning

Cause

Effect

X Cause Y

Machine Learning

X Variables

- Attributes which an ML algorithm focuses on are called features
- Each data point is a list (or vector) of such features
- Input to an ML algorithm is a feature vector
- Feature vectors are called x variables
- Also called independent variables or predictors

Machine Learning

Y Variables

- Attributes which an ML algorithm tries to predict are called labels
- Labels can be:
  - Categorical (classification)
  - Continuous (regression)
- Labels are referred to as y variables
- Also called dependent variable

PRIME INTUIT
Finishing School

Machine Learning

Data Set

Target Variables

Attributes

Features

Sender	Subject	IP	Body	Result
Xyz@bank.com	Loan	77.44.5.89	Dear customer your loan application….	Spam
123@builder.com	quotation	101.22.76.81	Dear Niranjan Your quotation……	Ham
lmn@school.com	reciept	301.99.23.45	Hi sir your Sons school fee………	Ham

Labels

Feature Vector

Machine Learning

Supervised Learning
(Training Stage)

$$y = f(x)$$

Most machine learning algorithms aim to learn the function

f which links the function x to label y

PRIME INTUIT
Finishing School

Machine Learning

Supervised Learning models

- Linear Regression
- Logistic Regression

Machine Learning

Supervised Learning models

- Linear Regression
- Logistic Regression

Machine Learning

Linear Regression

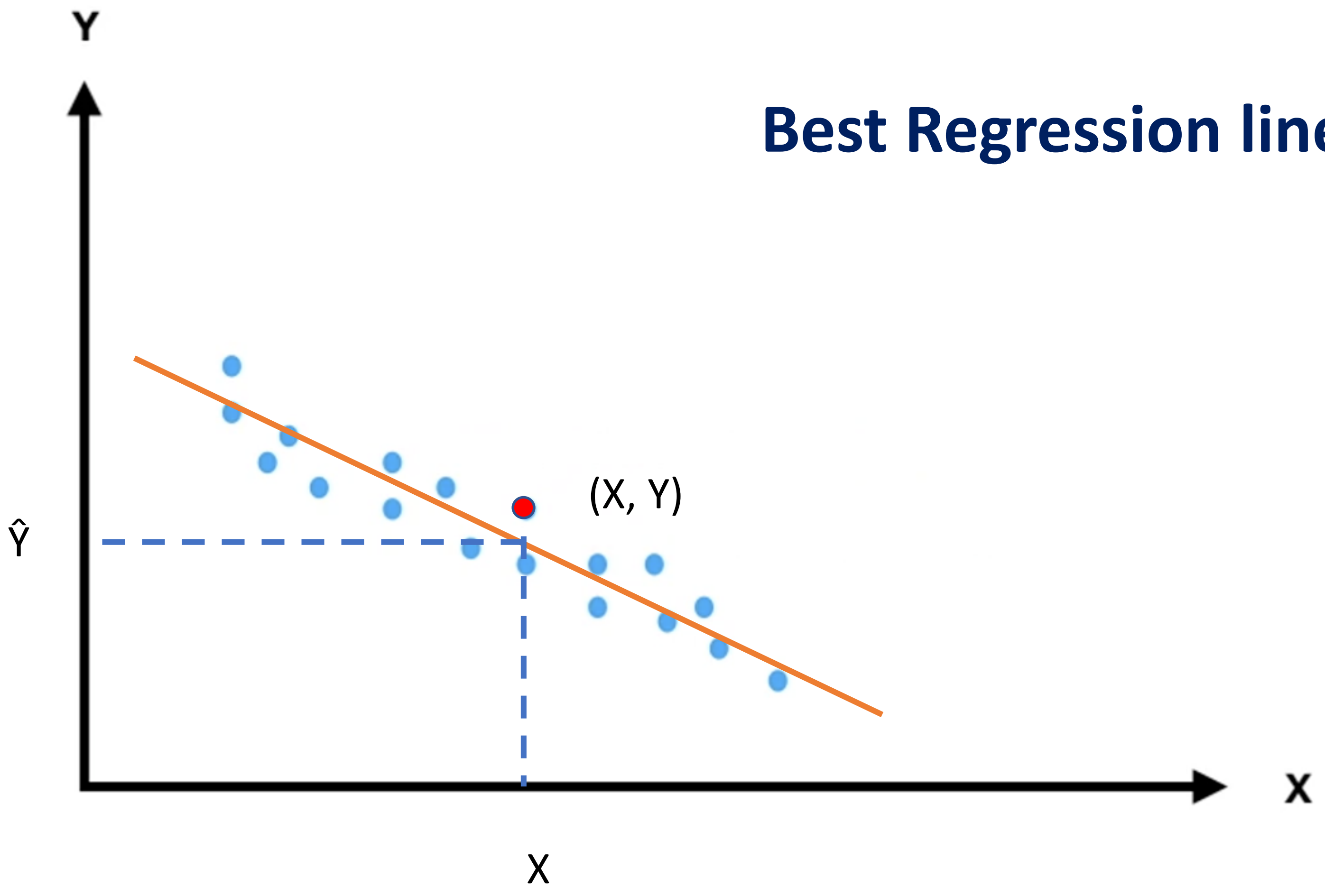
$$Y = m(x) + C$$

$$f(y) = m(x) + C$$

Here Linear regression specifies upfront that the relationship between x & y represented by f is linear

Machine Learning

Linear Regression



PRIME INTUIT
Finishing School

Machine Learning

Best regression line

Residual = e = Y - Ŷ

Data = (x1, y1), (x2, y2), (x3, y3) ---------- (xn, yn)

Residual = (Y1 – Ŷ1), (Y2 – Ŷ2), (Y3 – Ŷ3) ---------- (Yn – Ŷn)

To get the best regression line our aim should be to get the minimum error for training data

Eg: least Square method

Minise: (Y1 – Ŷ1)2 + (Y2 – Ŷ2)2 + (Y3 – Ŷ3)2 ---------- (Yn – Ŷn)2

Machine Learning

Cavoites of regression line

- Can be used only when the errors are normally distributed
- Can also be used with multiple independent variables

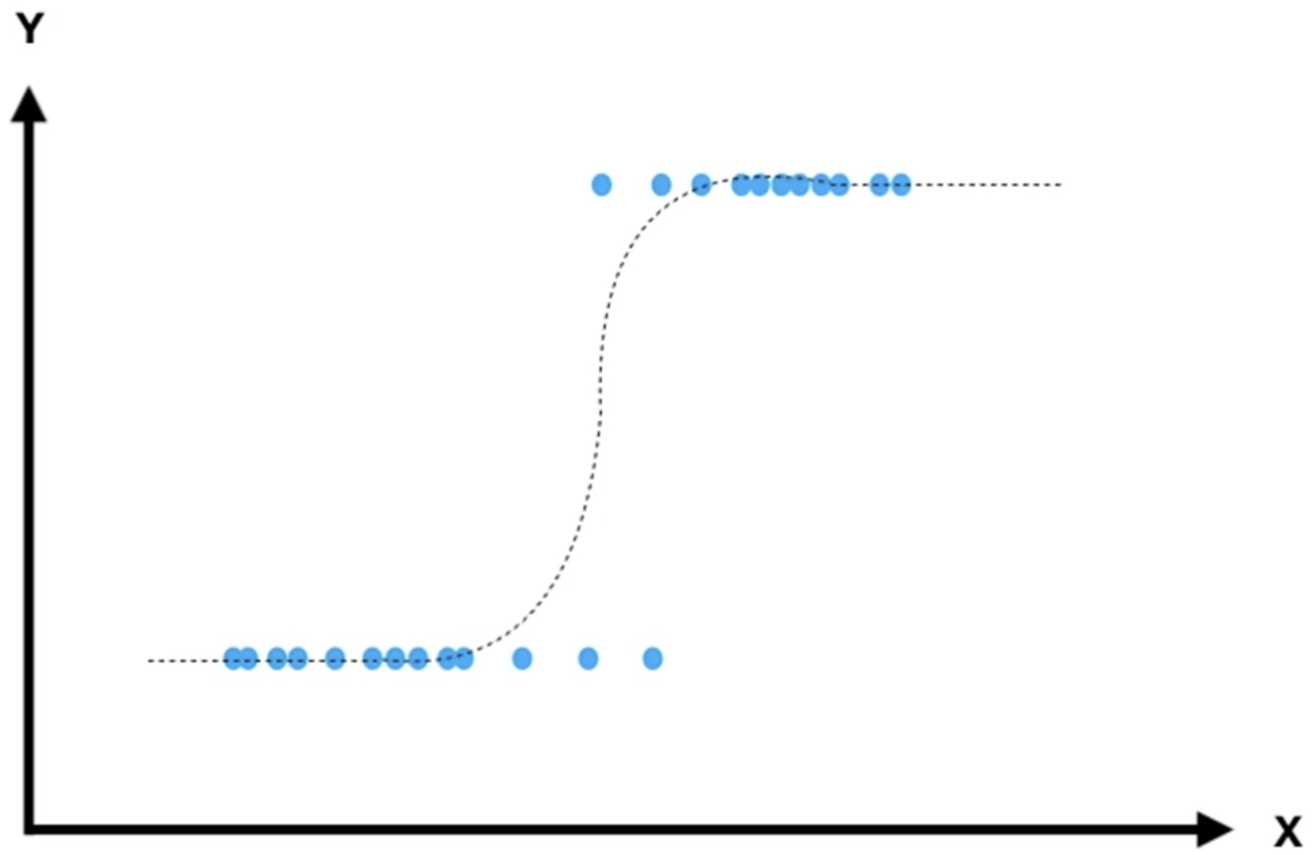
PRIME INTUIT
Finishing School

Logistic Regression

- Used when the dependent variable is categorical (y)
- Mutiple Independent variables can be continuous or categorical (x)
- Predict probability of each outcome – assign result to category with highest probability

Machine Learning

Logistic Regression



Machine Learning

Logistic Regression

Objective function: Minimize cross entropy

Cross-entropy is a measure from the field of information theory, building upon entropy and generally **calculating the difference between two probability distributions.** It is closely related to but is different from KL divergence that calculates the relative entropy between two probability distributions, whereas cross-entropy can be thought to calculate the total entropy between the distributions.

Cross-entropy is also related to and often confused with logistic loss, called log loss. Although the two measures are derived from a different source, when used as loss functions for classification models, both measures calculate the same quantity and can be used interchangeably.

PRIME INTUIT
Finishing School

Machine Learning

Unsupervised Learning

- There is only input data X – no output data
- Model the underlaying structure to learn more about the data
- Algorithms self discover patterns of structure in the data

Machine Learning

Unsupervised Learning - algorithms

- Autoencoding – Identify latent factors that drive data (eg: PCA)
- Clustering – Identify patterns in data items

Looking within

- Be emotionally self sufficient
- Learn what matters (to you)
- Identify others who share them and those who don't
- Eliminate what does not matter
- Train yourself to navigate the outside world

PRIME INTUIT
Finishing School

Machine Learning

Why Look with in

In Life

- Be emotionally self sufficient
- Learn what matters (to you)
- Identify others who share them
- and those who don't
- Eliminate what does not matter
- Train yourself to navigate the outside world

In ML

- Make unlabeled data self sufficient
- Latent factor analysis (fields relevant for a cause)
- Clustering
- Anomaly detection
- Quantisation

Machine Learning

Why Look with in

ML Technique	In Life
Make unlabeled data self sufficient	Identify photo of specific individual
Latent factor analysis (fields relevant for a cause)	Find common drivers for 100 Stocks
Clustering	Find relevant document in a corpus
Anomaly detection	Flag fraudulent credit card transactions
Quantisation	Compress 24 bit true color to 8 bit

Machine Learning

Why Look with in

What

- Make unlabeled data self sufficient
- Latent factor analysis (fields relevant for a cause)
- Clustering
- Anomaly detection
- Quantisation

How

- Autoencoding
- Autoencoding
- Clustering
- Autoencoding
- Clustering

Machine Learning

Support Vector Machines

Machine Learning

What is an SVM

- SVMs are used to build binary classifiers
- Make Classification decision on basis of a “Linear function” of point’s co-ordinates
- Does not require prior knowledge of probability distribution of the points
- Involves an explicit training stage

Machine Learning

What is an SVM

- SVMs are supervised machine learning approach used to build linear, non probabilistic and binary classifiers

Machine Learning

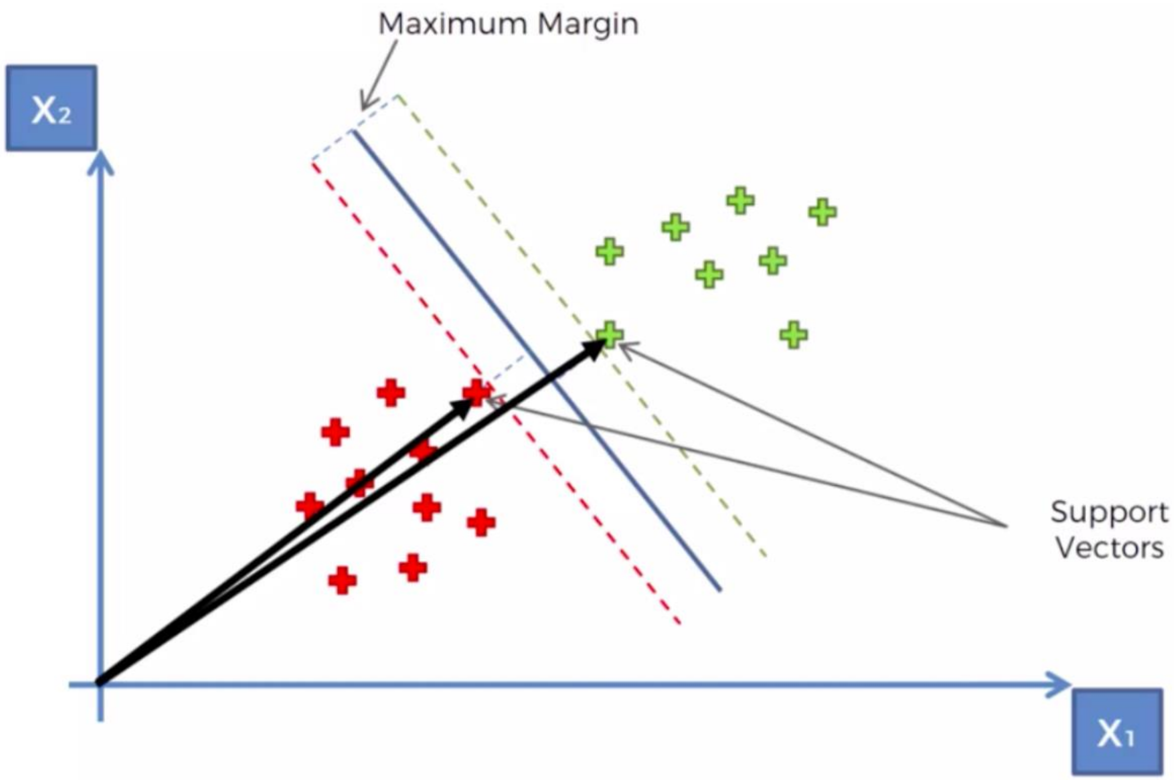
What is an SVM

The linear SVM classifier works by drawing a straight line between two classes. All the data points that fall on one side of the line will be labeled as one class and all the points that fall on the other side will be labeled as the second. Sounds simple enough, but there's an infinite amount of lines to choose from. How do we know which line will do the best job of classifying the data? This is where the LSVM algorithm comes in to play.

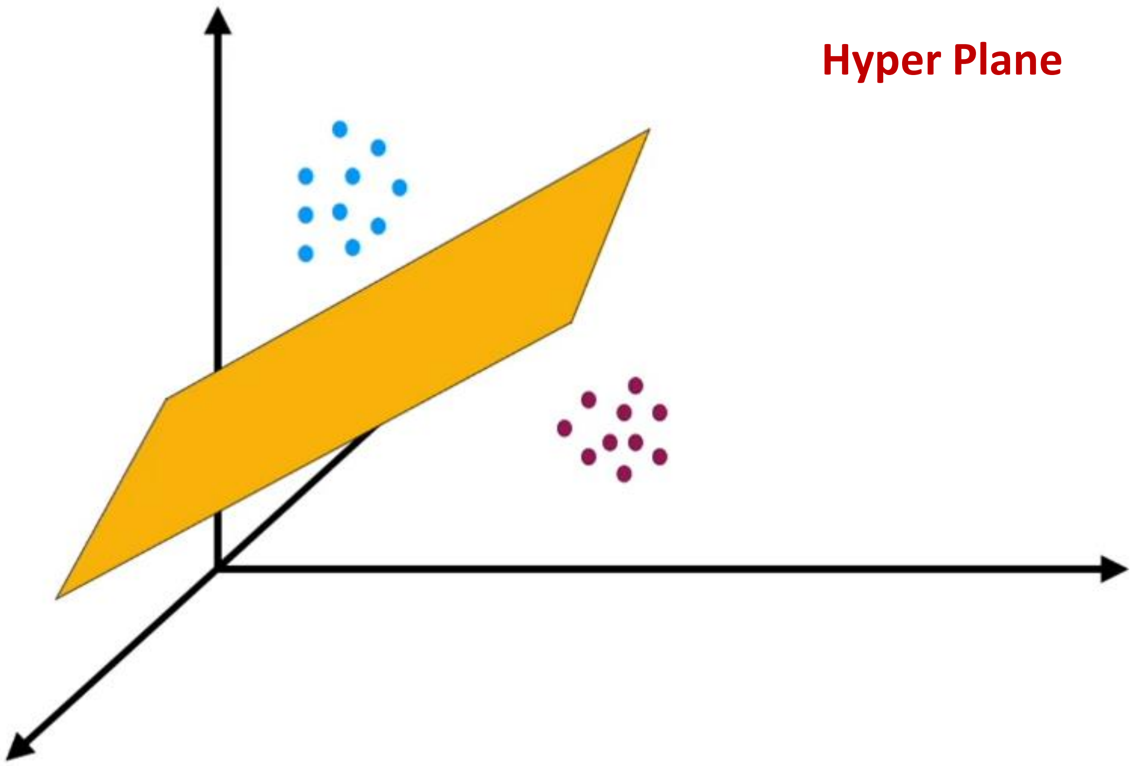
Machine Learning

What is an SVM

The LSVM algorithm will select a line that not only separates the two classes but stays as far away from the closest samples as possible. In fact, the “support vector” in “support vector machine” refers to two position vectors drawn from the origin to the points which dictate the decision boundary.



How SVMs work

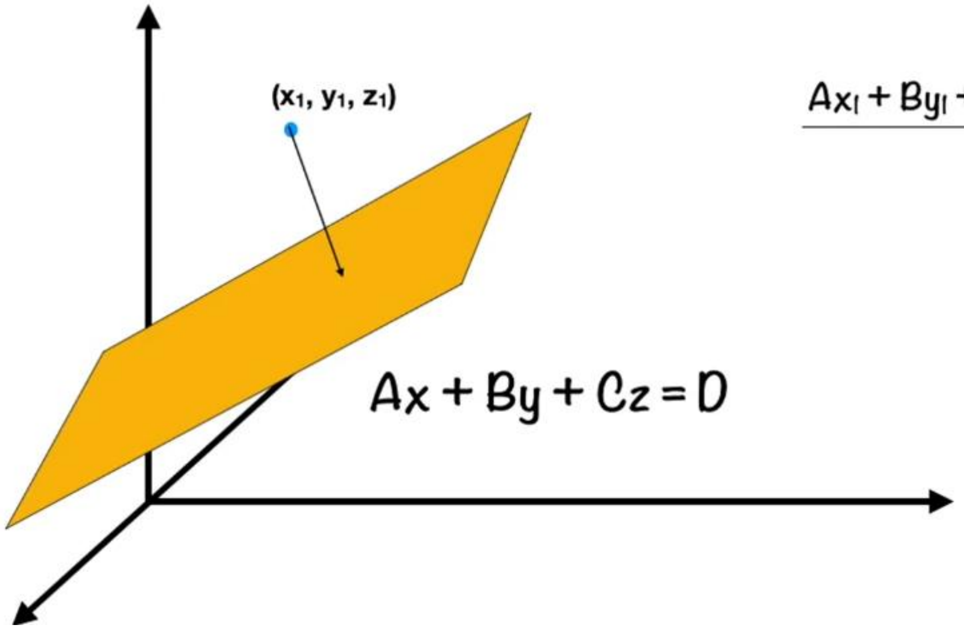


In an n dimensional data/ space SVM finds an n-1 hyperplane to separate points into two categories

Machine Learning

Hyper plane

In a vector space of n dimensions, a hyperplane is a geometrical shape with (n-1) dimensions and zero thickness in one dimension.



$$\frac{Ax_1 + By_1 + Cz_1 - D}{}$$

The best hyper plane is the one which maximizes the sum of distances of the nearest points on either side of the plane

Machine Learning

Hyper plane

In a vector space of n dimensions, a hyperplane is a geometrical shape with (n-1) dimensions and zero thickness in one dimension.

In our example $Ax + By + Cz = D$

Points on one side of the hyperplane :

$Ax + By + Cz > D$

Points on other side of hyperplane:

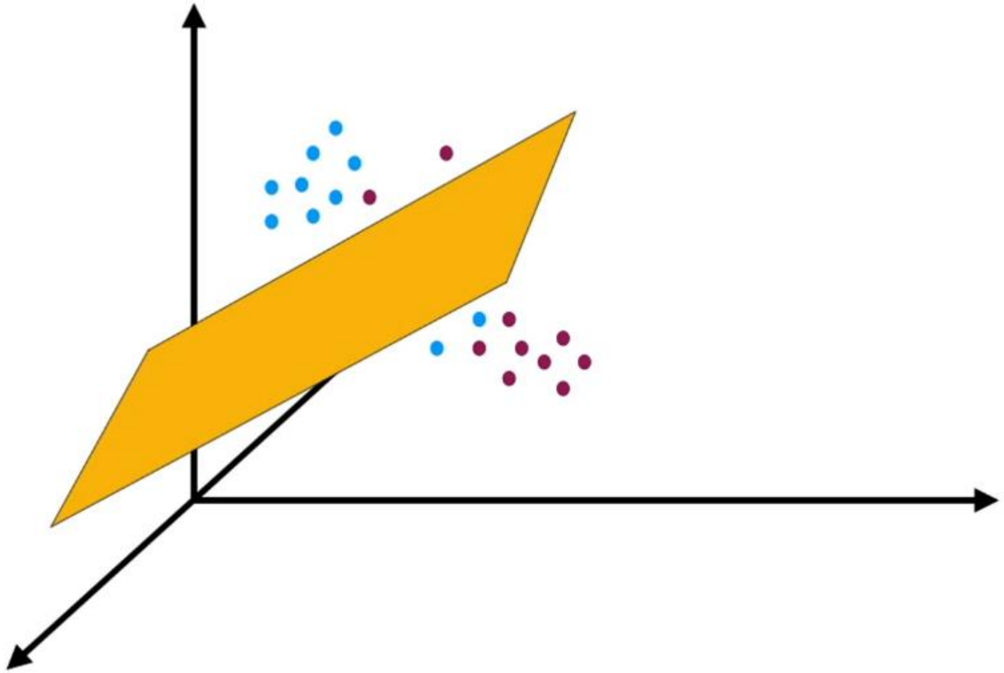
$Ax + By + Cz < D$

SVM tries to find such a linear equation

Machine Learning

Hyper plane

What if the points on the hyperplane are not linearly separable



The soft margin method finds a hyper plane which performs as clean a separation of points as possible

Machine Learning

Non linear separation

SVM is a linear classifier

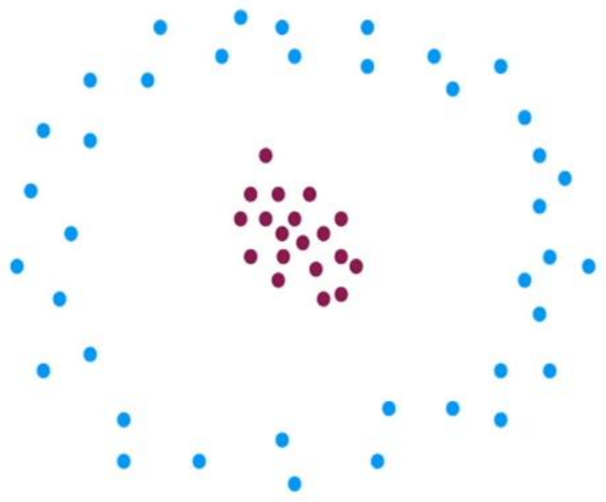
But can be used to perform non linear separation

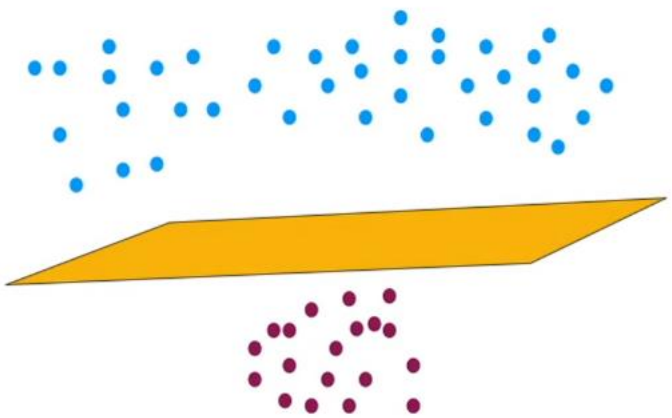
Achieved by using Kernel Trick

PRIME INTUIT Finishing School

Machine Learning

Kernel Trick





Transformed / modified to be separated by a linear plane

Can be separated only by a circle (quadratic)

Machine Learning

Linear Classification does a dot product of vectors with many elements

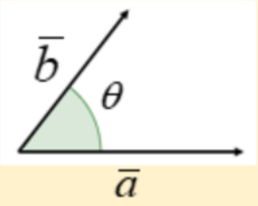
Algebraically, the dot product is **the sum of the products of the corresponding entries of the two sequences of numbers**. Geometrically, it is the product of the Euclidean magnitudes of the two vectors and the cosine of the angle between them. ... In modern geometry, Euclidean spaces are often defined by using vector spaces.

Dot Product

If $\bar{a} = \langle a_1, a_2, a_3 \rangle$ and $\bar{b} = \langle b_1, b_2, b_3 \rangle$

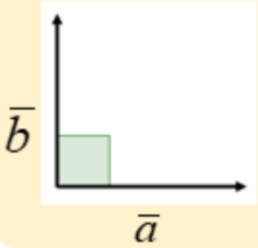
then the dot product is

$$\bar{a} \cdot \bar{b} = a_1 b_1 + a_2 b_2 + a_3 b_3$$



If θ is the angle between $\bar{a}$ and $\bar{b}$ then

$$\bar{a} \bullet \bar{b} = |\bar{a}||\bar{b}|\cos\theta$$



$\bar{a} \bullet \bar{b}$ are orthogonal (perpendicular) if and only if $\bar{a} \bullet \bar{b} = 0$

PRIME INTUIT
Finishing School

Machine Learning

Non linear separation

Kernel function operates in feature space which may have which may have many more dimensions than original feature space

Finds the maximum margin in hyper plane in modified feature space

It’s a linear function in the modified feature space

Kernel trick allows a way to solve problems where the data is not linearly separable by projecting such data into a higher dimensional space

Machine Learning

SVM – Use case

https://archive.ics.uci.edu/ml/machine-learning-databases/adult/adult.data

University of California, Irvine, School of Information and Computer Sciences

PRIME INTUIT
Finishing School

Machine Learning

SVM – Use case

```
import numpy as np

import pandas as pd

original_data = pd.read_csv("adult.csv",

                    names=['Age','Workclass', 'fnlwgt', 'Education', 'Education-Num',
'Marital Status',

                        'Occupation', 'Relationship', 'Race', 'Gender', 'Capital Gain',
'Capital Loss',

                        'Hours per week', 'Country', 'Target'], sep =r'\s*,\s*', engine
='python', na_values = "?")

original_data.head()
```

PRIME INTUIT
Finishing School

Machine Learning

SVM – Use case

```
import matplotlib.pyplot as plt

import math

%matplotlib inline

fig = plt.figure(figsize = (20,20))

cols = 3

rows = math.ceil(float(original_data.shape[1] / cols))
```

Machine Learning

SVM – Use case

```
for i, column in enumerate(['Age', 'Workclass', 'Education', 'Occupation', 'Race',
'Gender']):
    ax = fig.add_subplot(rows, cols, i+1)
    ax.set_title(column)
    if original_data.dtypes[column] ==np.object:
        original_data[column].value_counts().plot(kind='bar', axes = ax)
        plt.xticks(rotation='vertical')
plt.subplots_adjust(hspace=0.7, wspace=0.2)
plt.show()
```

Machine Learning

SVM – Use case

```
# Use label encoder to convert text to numeric
import sklearn.preprocessing as preprocessing


le = preprocessing.LabelEncoder()
original_data['Occupation'] =
le.fit_transform(original_data['Occupation'].astype(str))
original_data.head()
()
```

Machine Learning

SVM – Use case

```
original_data['Target'] =
le.fit_transform(original_data['Target'].astype(str))
original_data.tail()
```

```
original_data.Target.unique()
```

Machine Learning

SVM – Use case

```
original_data.groupby('Education-Num').Target.mean().plot(kind = 'bar')
plt.show()
```

SVM – Use case

```
# machin learning part
# Take only the feature that is req for now
from sklearn.model_selection import train_test_split
x = original_data[['Education-Num', 'Occupation']]
y = original_data['Target']

# split data by 80:20 for training:Test
x_train, x_test, y_train, y_test = train_test_split(x,y,test_size = 0.2,
random_state = 0)
```

Machine Learning

SVM – Use case

```
# import the SVM model, SVC C support vector classification from sklearn.svm
import SVC
#declare the SVC with no tunning
classifier = SVC()
# fitting the data or training the SVM
classifier.fit(x_train, y_train)
# Predecting the result and giving the accuracy
score = classifier.score(x_test, y_test)
print(score)
```

Machine Learning

SVM – Use case

```
# plotting a correlation matrix to understand the data
import seaborn as sns
corrmat = original_data.corr()
f, ax = plt.subplots(figsize = (7,7))
sns.heatmap(corrmat, vmax = .8, square = True)
plt.show()
# only fields with numeric value show up in correlation matrix
```

Machine Learning

SVM – Use case

```
original_data['Race'] =
le.fit_transform(original_data['Race'].astype(str))
original_data['Gender'] =
le.fit_transform(original_data['Gender'].astype(str))
original_data['Marital Status'] =
le.fit_transform(original_data['Marital Status'].astype(str))
original_data['Education'] =
le.fit_transform(original_data['Education'].astype(str))
```

PRIME INTUIT Finishing School

Machine Learning

SVM – Use case

```
# Recreate the correlation plot
import seaborn as sns
corrmat = original_data.corr()
f, ax = plt.subplots(figsize = (7,7))
sns.heatmap(corrmat, vmax = .8, square = True)
plt.show()
```

PRIME INTUIT
Finishing School

Machine Learning

SVM – Use case

```
# replot with annotation
corrmat = original_data.corr()
f, ax = plt.subplots(figsize = (7,7))
sns.heatmap(corrmat, vmax = .8, square = True, annot = True, fmt
= '.2f')
plt.show()
```

PRIME INTUIT
Finishing School

Machine Learning

SVM – Use case

```
# resetting features
x = original_data[['Education-Num', 'Occupation', 'Age', 'Gender']]
y = original_data['Target']
x_train, x_test, y_train, y_test = train_test_split(x,y,test_size = 0.2, random_state = 0)
classifier.fit(x_train, y_train)
# Predecting the result and giving the accuracy
score = classifier.score(x_test, y_test)
print(score)
```

PRIME INTUIT
Finishing School

Machine Learning

SVM – Use case

```
# setting our kernel to Radial Basis Function with penalty parameter c =1.0
classifier = SVC(kernel = 'rbf', C=1.0)
classifier.fit(x_train, y_train)
score = classifier.score(x_test, y_test)
print(score)
```

Machine Learning

SVM – Use case

setting our kernel to Radial Basis Function with penalty parameter c =10,

Process called Hyper parameter tuning

```
classifier = SVC(kernel = 'rbf', C=10)
classifier.fit(x_train, y_train)
score = classifier.score(x_test, y_test)
print(score)
```

Machine Learning

SVM – Use case

```
# using a linear kernel instead of rbf
classifier = SVC(kernel = 'linear', C=10)
classifier.fit(x_train, y_train)
score = classifier.score(x_test, y_test)
print(score)
```