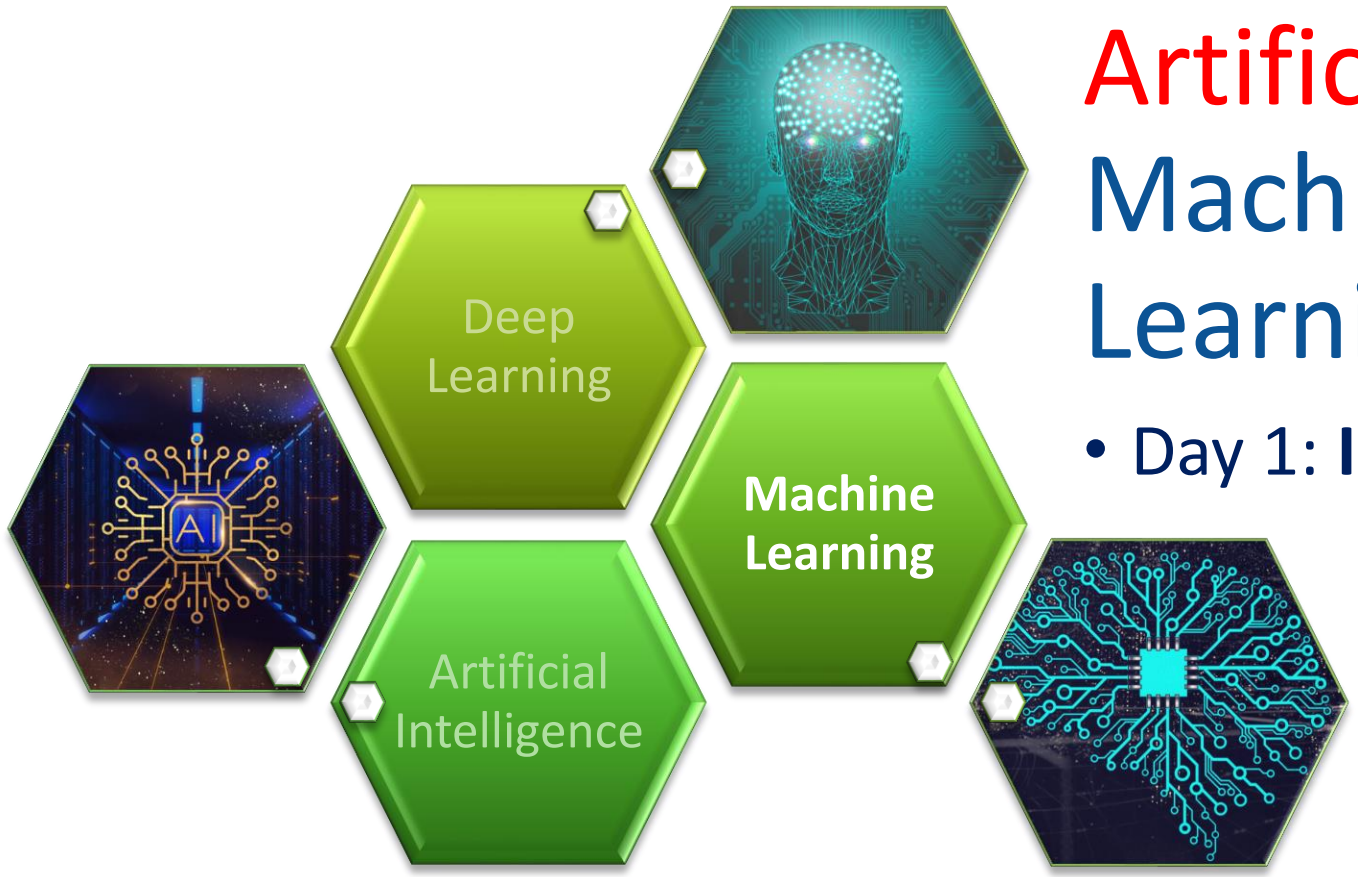


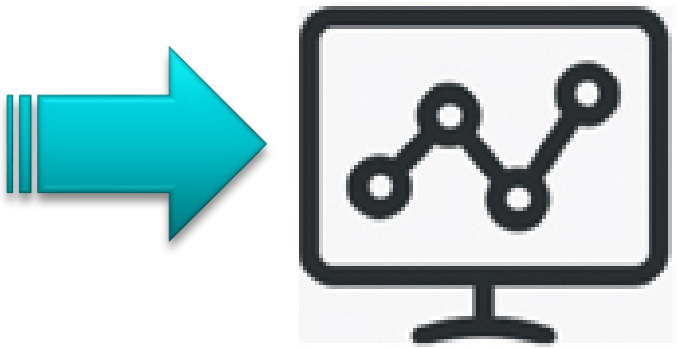
Bootcamp on Artificial Intelligence: Machine Learning and Deep Learning

- Day 1: **Introduction to Machine Learning**

Outline: Day 1

- Introduction to Machine Learning
- Classification and Clustering, classifier surfaces
- Feature Engineering
- Measuring distances, statistical coefficient
- Confusion matrix
- Training errors

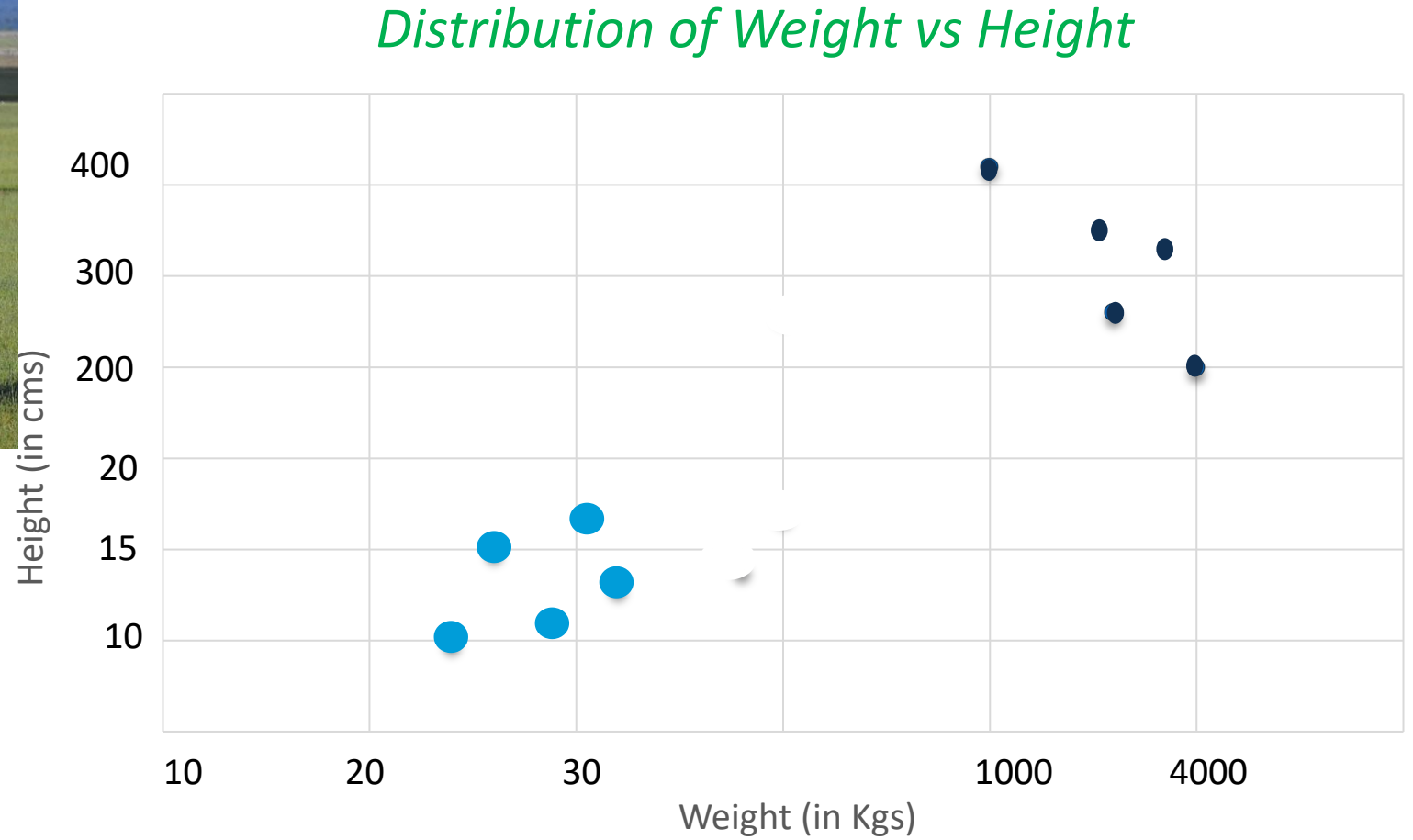
Make a Machine differentiate Elephants from Dogs?



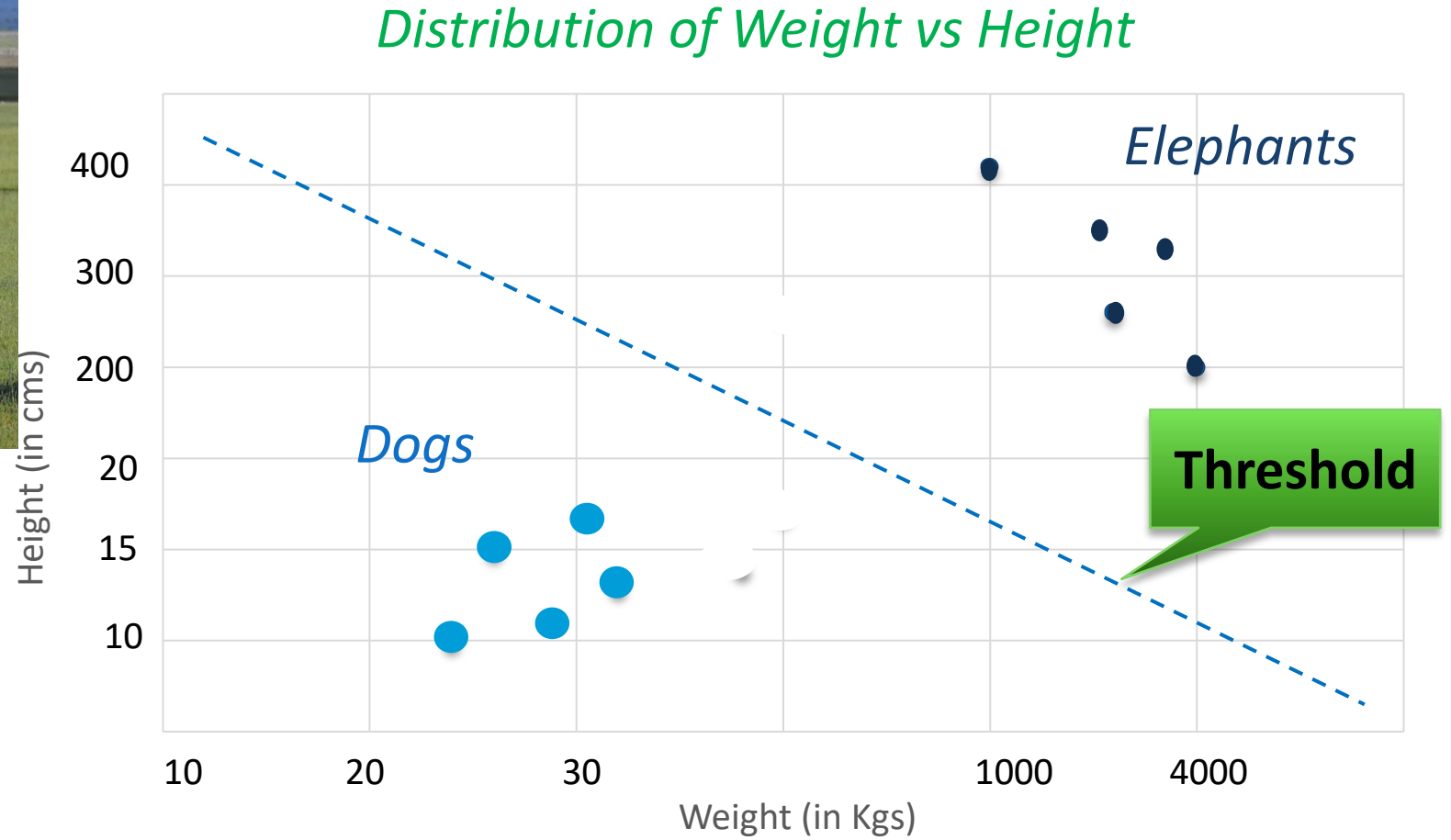
Machine

Elephant OR Dog?

Make a Machine differentiate Elephants from Dogs



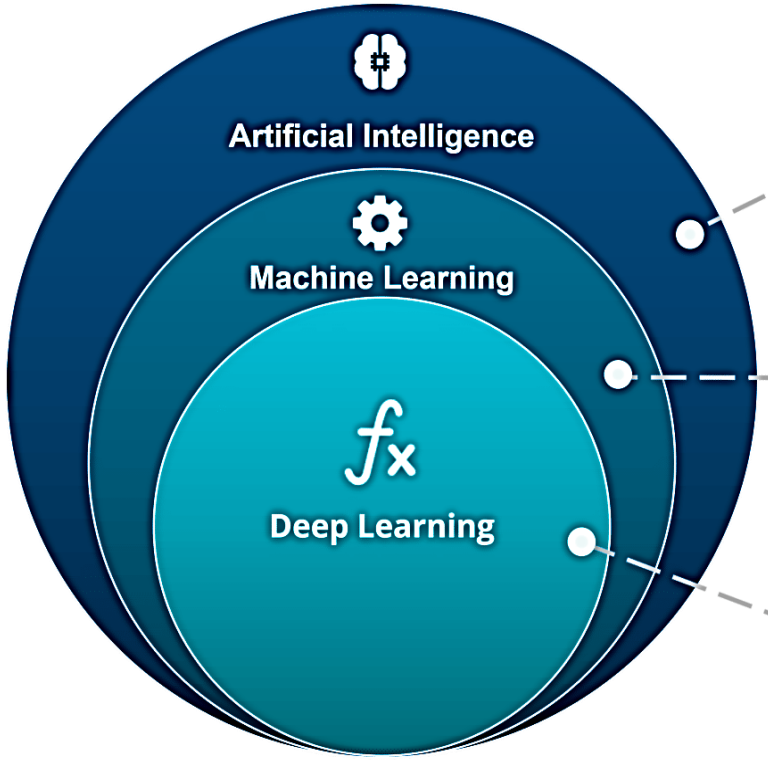
Make a Machine differentiate Elephants from Dogs: Simply Threshold



Differentiate 100 species of Moths with Simple Threshold?

Make Machine Intelligent !

AI is the science of making machines do things that would require intelligence if done by men



ARTIFICIAL INTELLIGENCE
A technique which enables machines to mimic human behaviour

MACHINE LEARNING
Subset of AI technique which use statistical methods to enable machines to improve with experience

DEEP LEARNING
Subset of ML which make the computation of multi-layer neural network feasible

Timeline

- 1950's
- 1960's
- 1970's
- 1980's
- 1990's
- 2000's
- 2010's

What is Machine Learning?

One of the Earliest definition of Machine Learning:

Arthur Samuel (1959):

"Field of study that gives computers the ability to learn without being explicitly programmed."

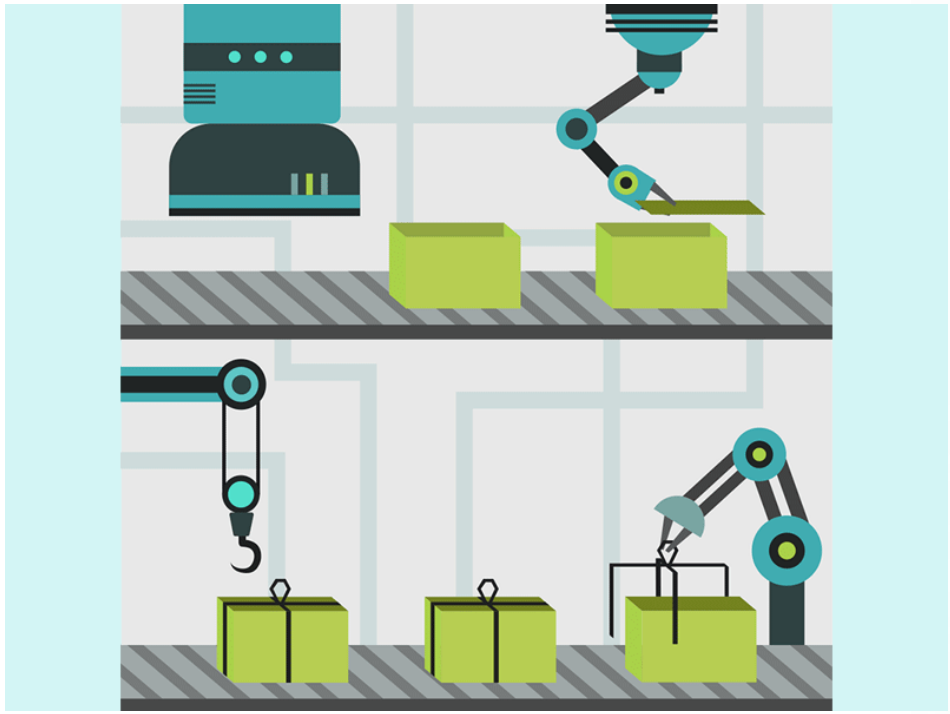
The Samuel Checkers-playing Program was among the world's first successful self-learning programs, and as such a very early demonstration of the fundamental concept of artificial intelligence.

***On February 24, 1956**, Arthur Samuel's Checkers program was developed for play on the IBM 701, In 1962, Self-proclaimed checkers master Robert Nealey played the game on an IBM 7094 computer. The computer won. It is still considered a milestone for artificial intelligence, as an example of the capabilities of an electronic computer.*

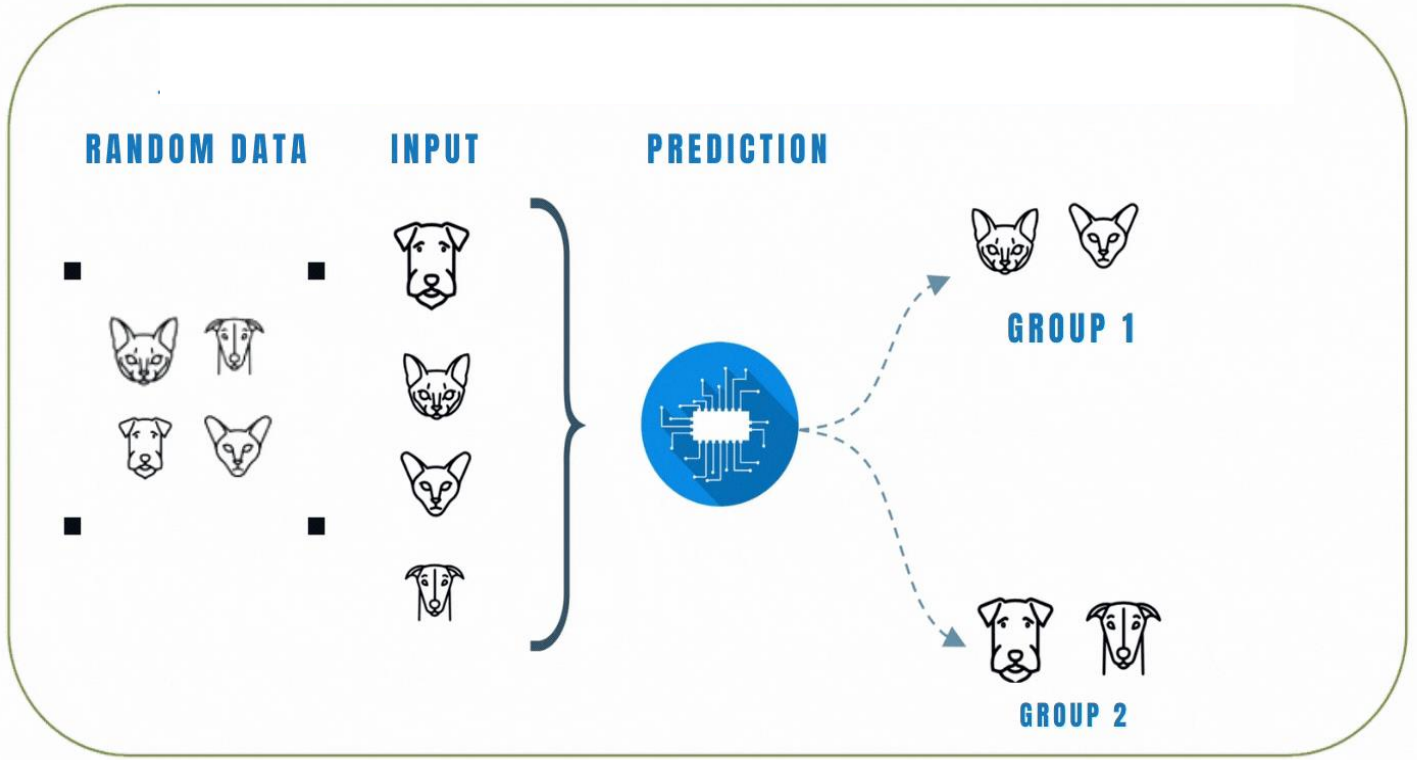
Courtesy: ibm.com

Machine Learning vs Process Automation

Process Automation



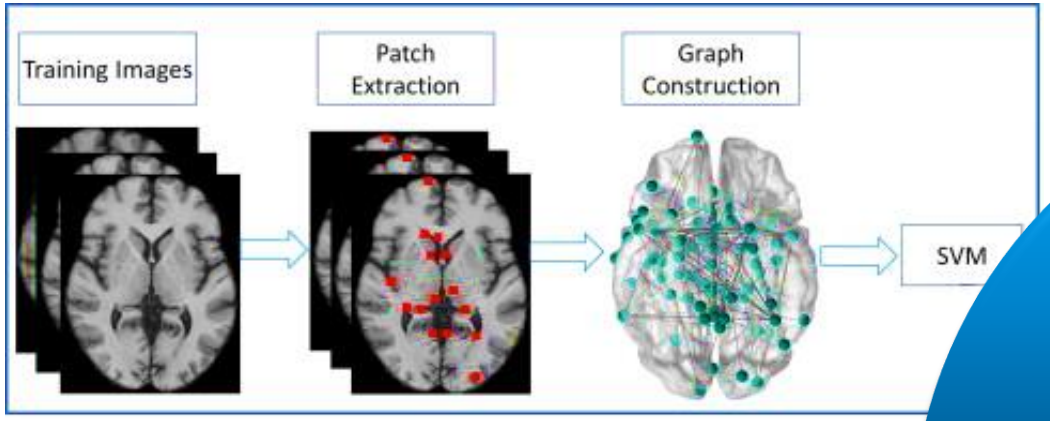
Machine Learning



Follow orders, Pre-programmed Rules to run a process, Monotonous, Repetitive process

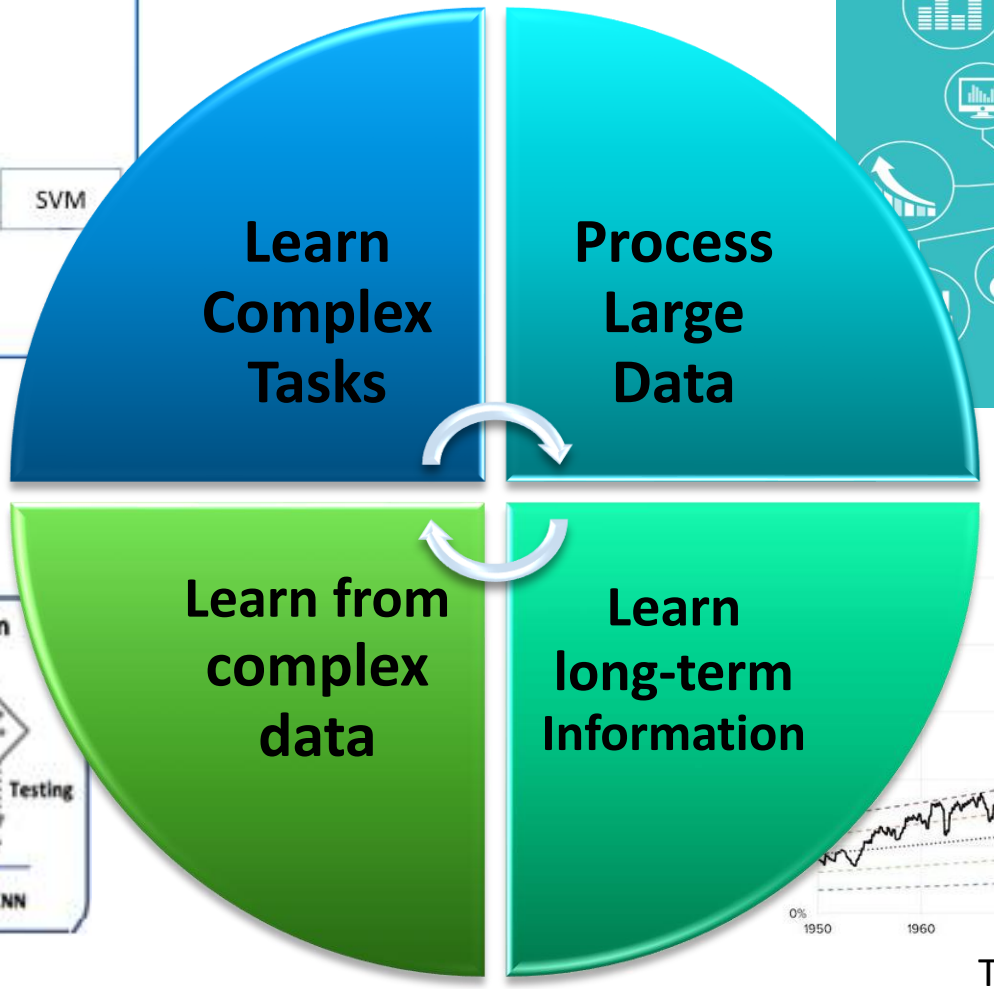
Mimic human-ability to think and do, Machine seeks patterns, adapts with experience

Machine Learning imparts abilities beyond human capability!

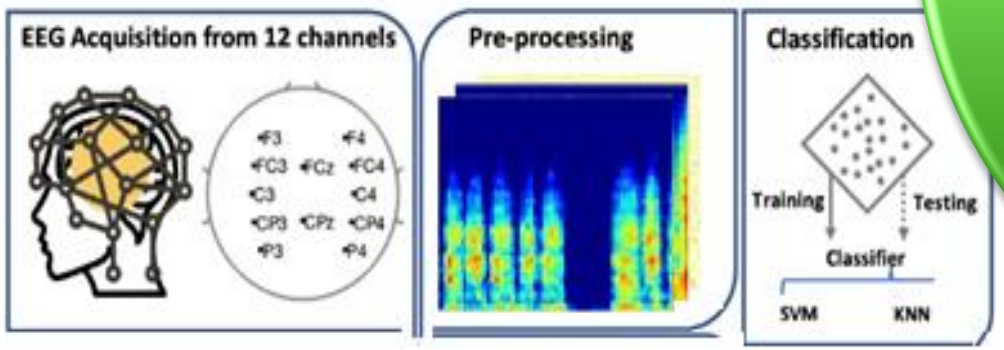


Automatic Disease Detection

https://doi.org/10.1016/j.media.2014.04.006



Process 1000's of Images Simultaneously



Auto-analyse EEG (brain) signals

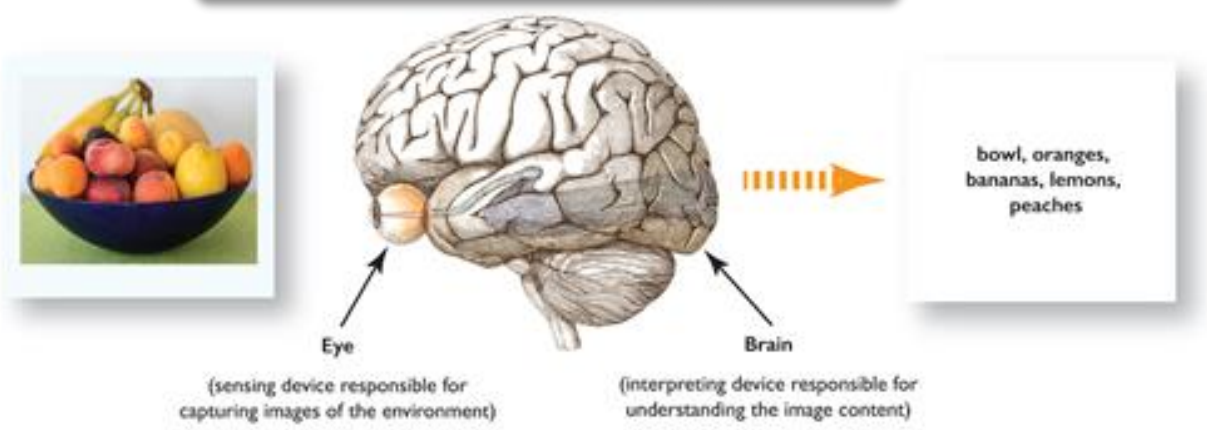
https://doi.org/10.1007/978-3-030-63823-8_6

Trend Analysis

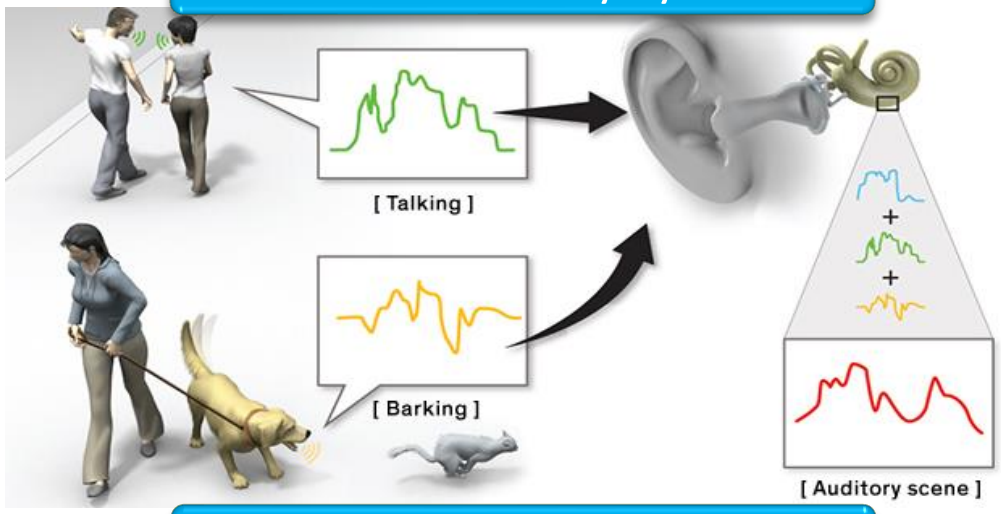
Human Perception vs. Machine Learning

Capture Data -> Process -> Interpret

Human Vision System

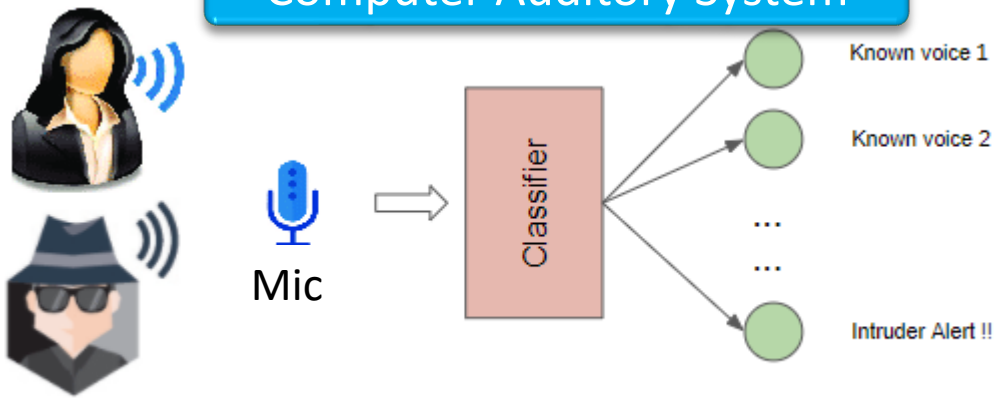


Human Auditory System



Computer Vision System

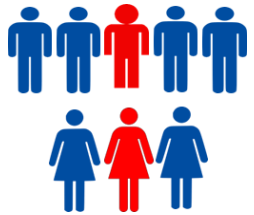
Computer Auditory System



Consider an Example of Computer Vision

Aim: Early Detection of Osteoporosis

- Osteoporosis is a chronic disease characterized by progressive bone loss and changes at microstructural levels resulting in increasing bone weakness and fragility.



Worldwide, 1 in 3 women and 1 in 5 men will experience osteoporotic fractures in their lifetime.

- Current Practice: Conventionally, Dual-Energy X-ray Absorptiometry (DEXA) is used, which is costly and is not available in small towns and villages. DEXA machine which is available only to 0.26/million population in India.

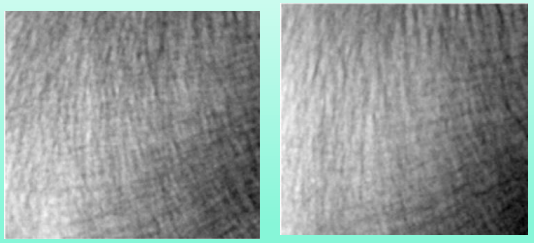
Question: Can we assess bone quality parameters on X-ray radiographs?

Answer: Yes ! Using Computer Vision on X-ray Images

Human Eye: Views Spatial Information in 2D

Computer Vision: Can go Beyond Human Vision!

Similar Visual Information
Computationally Different



Control Subject | Osteoporosis Patient

- Domain Transformations
- Computation Algorithms
- Computer Vision

Optimizations and Parameter Tuning for final Framework

Final Automatic Diagnostic tool based on the most optimized performance.

Classification of the Trabecular Bone Structure of Osteoporotic Patients using Machine Vision

Features Extraction

Table: Features (Principal components) and their corresponding *p-values* (after reduction using PCA)

Features	Healthy subjects (mean ± std. deviation)	Osteoporotic patients (mean ± std. deviation)	p- values
PC 1	(-4.02E-006) ± (6.47E-006)	(4.01E-006) ± (6.85E-006)	1.69E-006
PC 2	(-1.80E-006) ± (4.06E-006)	(1.80E-006) ± (2.83E-006)	2.24E-005
PC 3	(-2.62E-008) ± (1.10E-007)	(2.61E-008) ± (1.96E-007)	0.0013
PC 4	(-1.63E-012) ± (8.32E-011)	(1.63E-012) ± (9.52E-011)	0.8663
PC 5	(-4.79E-012) ± (1.64E-011)	(4.79E-012) ± (2.40E-011)	0.6834
PC 6	(7.84E-014) ± (2.78E-012)	(-7.82E-014) ± (2.63E-012)	0.8436

Table: Linear combinations of the original features that generate the principal components PC1 and PC2

Features		Principal Components (PCs)	
		PC1	PC2
Statistical Features	Mean	-0.5160	-0.4609
	Std. deviation	-0.3014	0.8533
Texture Features	Contrast	0.8017	0.0268
	Correlation	0.0052	-0.1397
	Energy	0.0052	-0.1397
	Homogeneity	0.0052	-0.1397

Classification Methods

Supervised Machine Learning

The discriminatory feature vectors (first and second PCs) were used to separate healthy X-ray images from osteoporotic ones using different classifiers. The four most popular classification algorithms, namely **Support Vector Machine** (SVM), **Naive Bayes classifier** , **Artificial Neural Network** (ANN) and ***k*-Nearest Neighbors** (*k*-NN) classifier, were applied to the first two principal components.

Bone X-ray image database considered in this study

Bone X-ray image	#Samples for training	#Samples for testing	Total samples
Healthy Subjects	40	47	87
Osteoporotic Patients	40	47	87
Total	80	94	174

Performance of the classifiers with feature processing

Classifiers	Sensitivity (%)	Specificity (%)	Accuracy (%)
SVM	100	95.74	97.87
Naive Bayes	97.87	93.61	95.74
k-NN	95.74	97.87	96.80
ANN	97.87	95.74	96.80

Computers in Biology and Medicine 91 (2017) 148–158

Contents lists available at ScienceDirect

Computers in Biology and Medicine

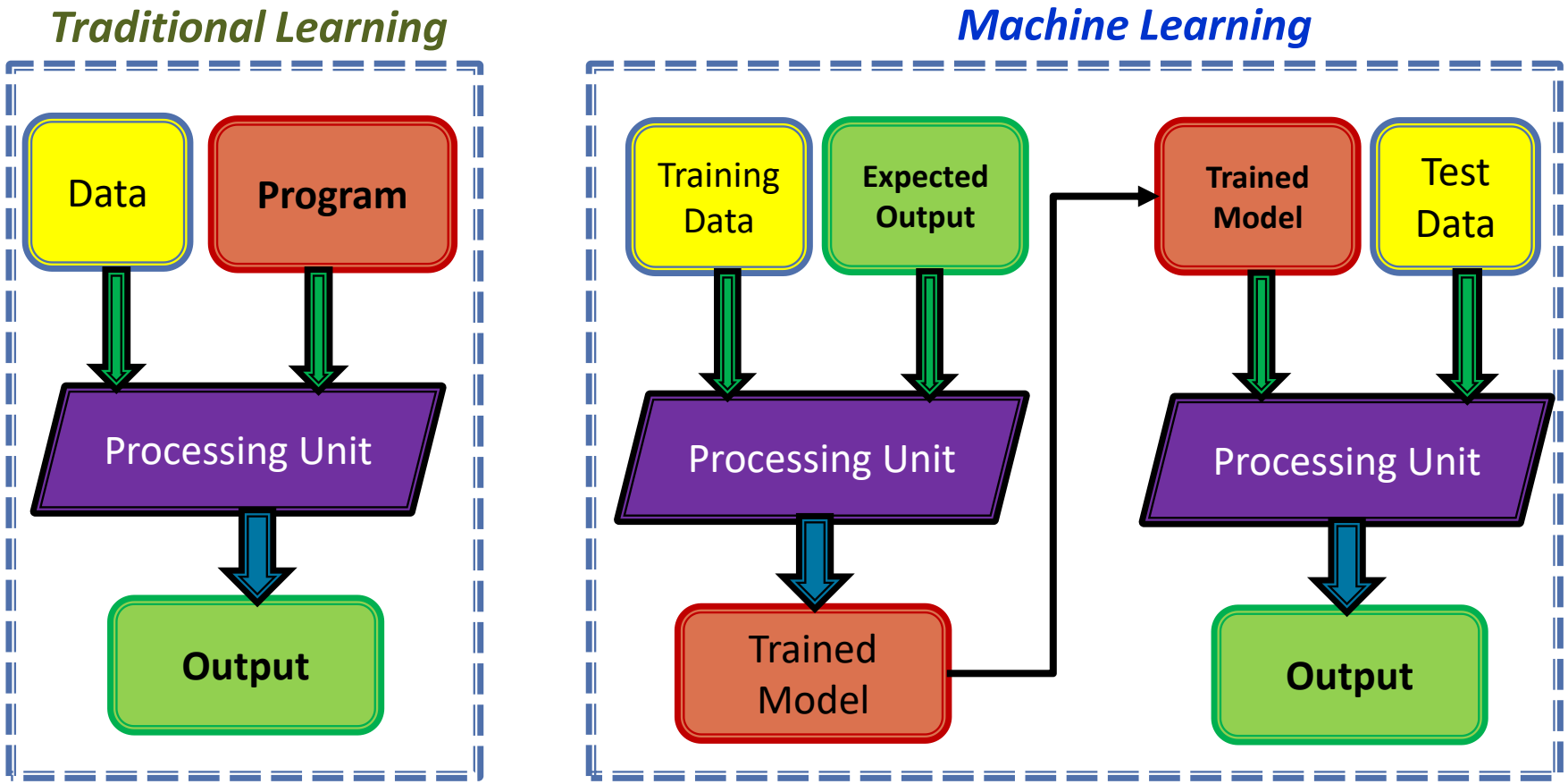
journal homepage: www.elsevier.com/locate/compbiomed

Classification of the trabecular bone structure of osteoporotic patients using machine vision

Anushikha Singh [a], Malay Kishore Dutta [a,*], Rachid Jennane [b], Eric Lespessailles [b,c]

[a] Department of Electronics & Communication Engineering, Amity University, Noida, Uttar Pradesh, India
[b] Univ. Orléans, I3MTO Laboratory, EA 4708, 45067 Orléans, France
[c] Hospital of Orleans, 45067 Orléans, France

Traditional Learning vs. Machine Learning



Formal Definition of Machine Learning

"A computer program is said to learn from experience E with respect to some task T and performance measure P if its performance at tasks in T, as measured by P, improves with experience E."

Tom M. Mitchel (1998)

T Task
- What is to be done? Prediction/ Clustering

E Experience
- **Data– Images, time-series, predictors**

P Performance
- **Quantitative measure of performance.**

Example of Machine Learning: Email Classification

Experience/Data

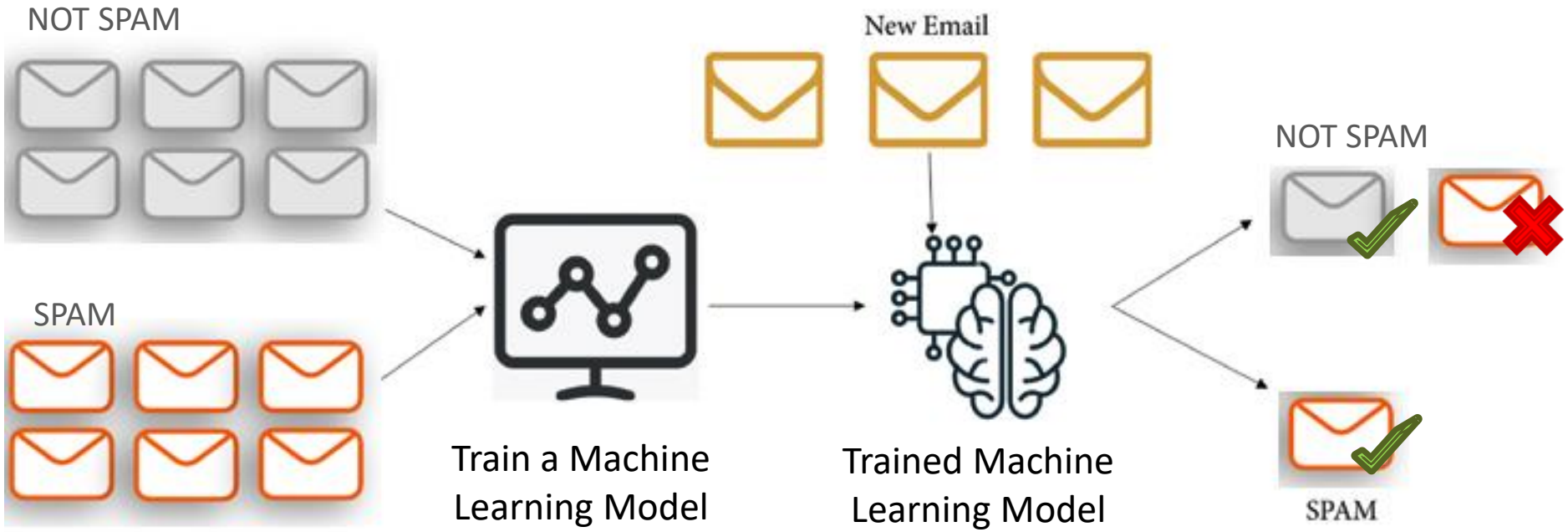
Lots of labelled e-mails

Task

Classify as Spam/Not Spam

Performance

$$\frac{\text{Correct Prediction}}{\text{Total Number of Prediction}} = \frac{2}{3}$$



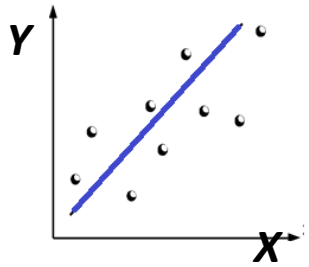
Example of Machine Learning: Score Prediction

Experience/Data

Labelled Data: Lectures attended vs. Score

	Lectures Attended	Score In Chemistry
Student 1	3	35
Student 2	7	45
Student 3	16	75
Student 4	14	60

Students in Class



X ***Y***

Train ML Model

Task

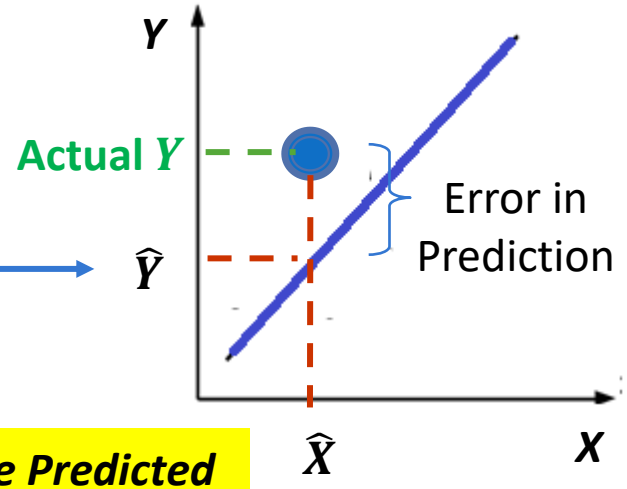
Predict
Score in Chemistry

New Value of X

Trained ML Model

Performance

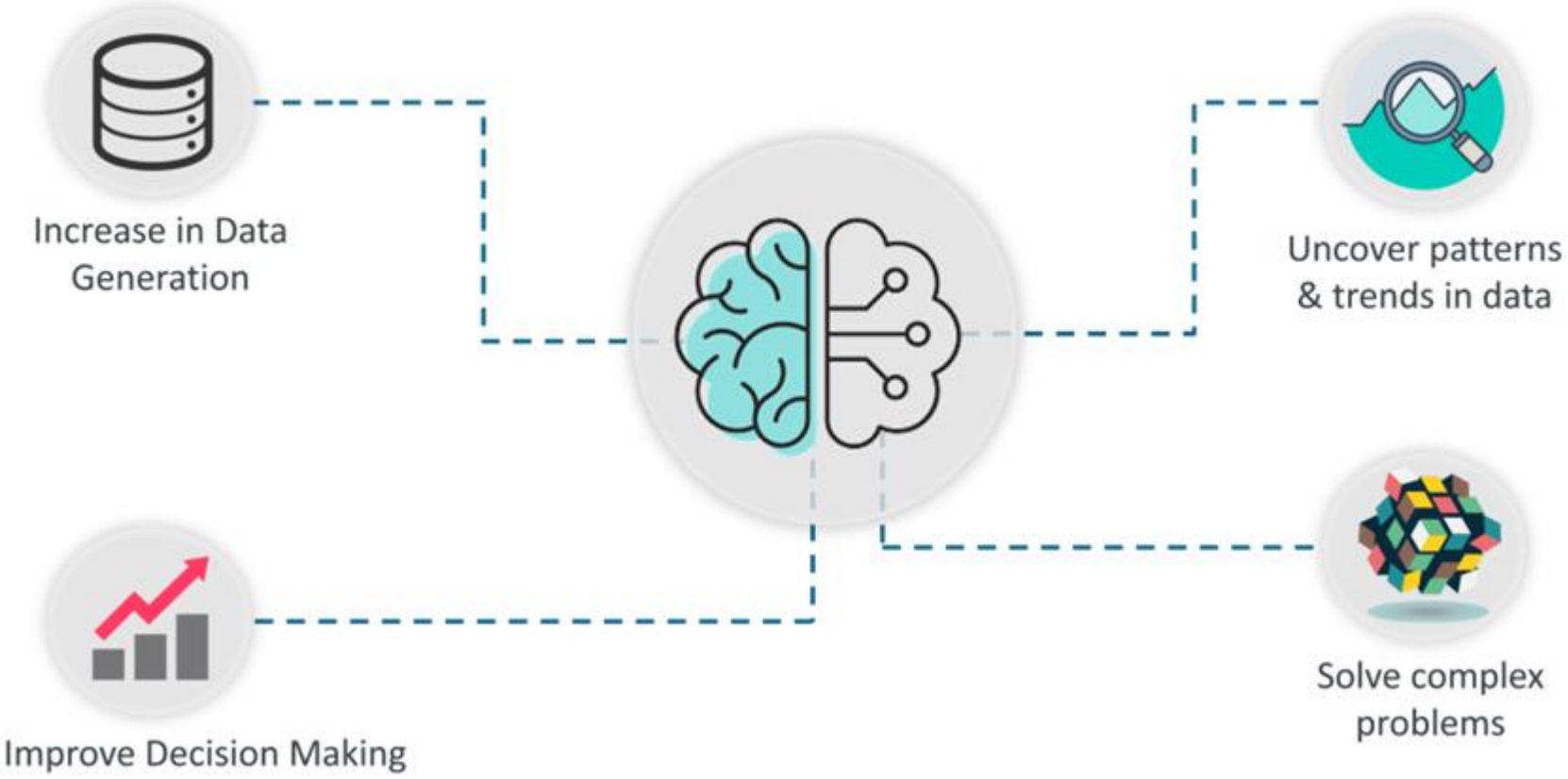
Error in Prediction



$\hat{Y}$ = Score Predicted by Model

Need for Machine Learning

Structure, analyze, interpret large amount of data



Find hidden patterns , dig beneath the surface

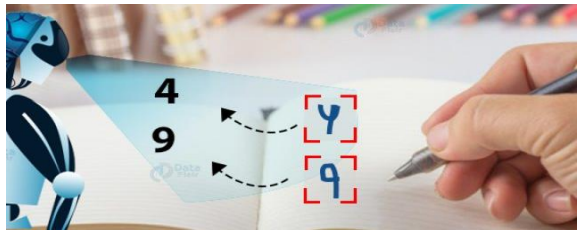
Forecast, predict risk accurately

Detecting genes linked to the deadly ALS disease to building self-driving cars

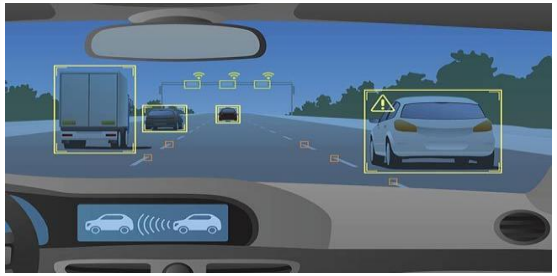
Few Examples of Machine Learning Application in Real World

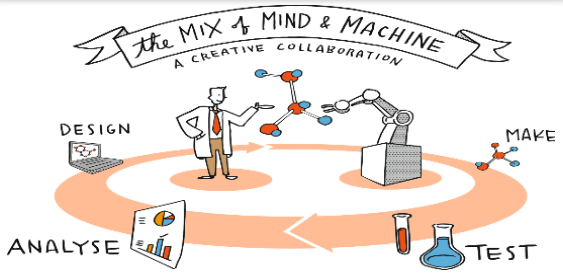
Voice Assistant

Recommendation System


Character Recognition

Facial Recognition/Detection


Driver Assistance System



Drug Discovery

Gaming (eg. AlphaGo)

Finance (Stock Prediction)


Medical Diagnosis

Types of Machine Learning

Machine Learning

Supervised Learning

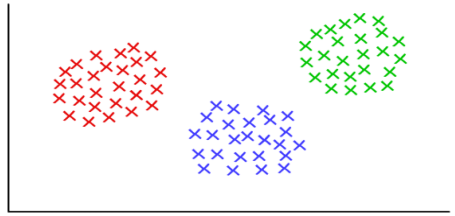
Task Driven
Predict Future Values

- Classification (Categorical Labels)
- Regression (Continuous Labels)

Predict Score in Chemistry

Unsupervised Learning

Data Driven
Clustering

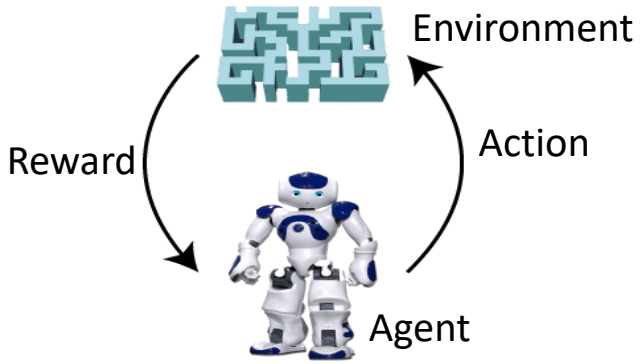


(No defined Labels)

Cluster based on Number of Legs

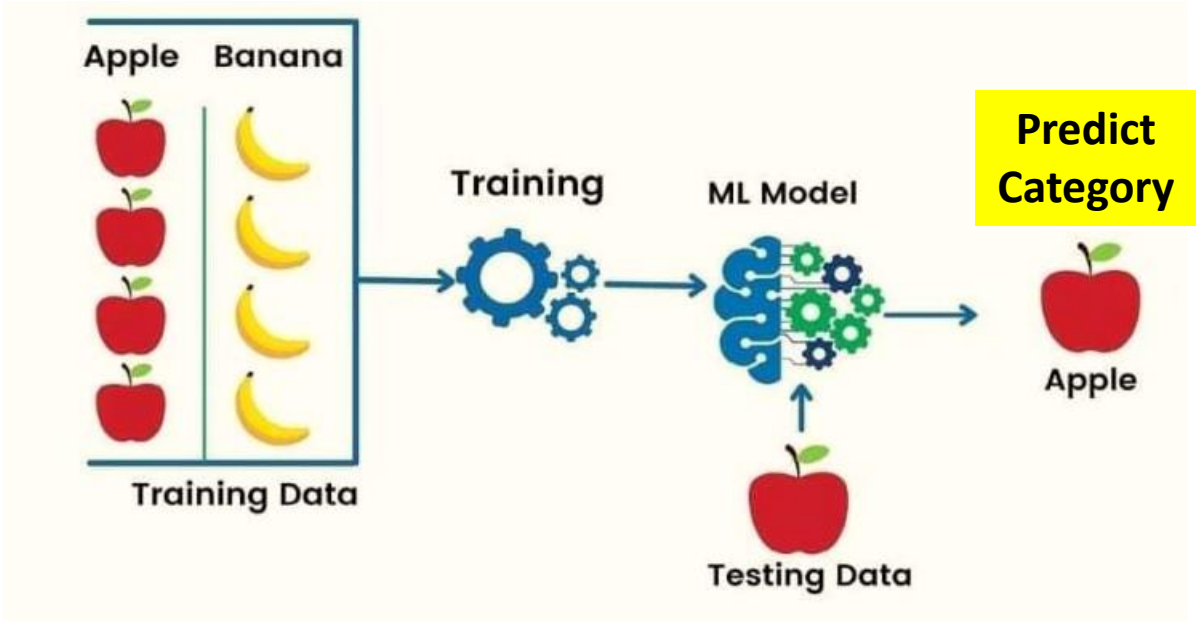
Reinforcement Learning

Learning from the mistakes



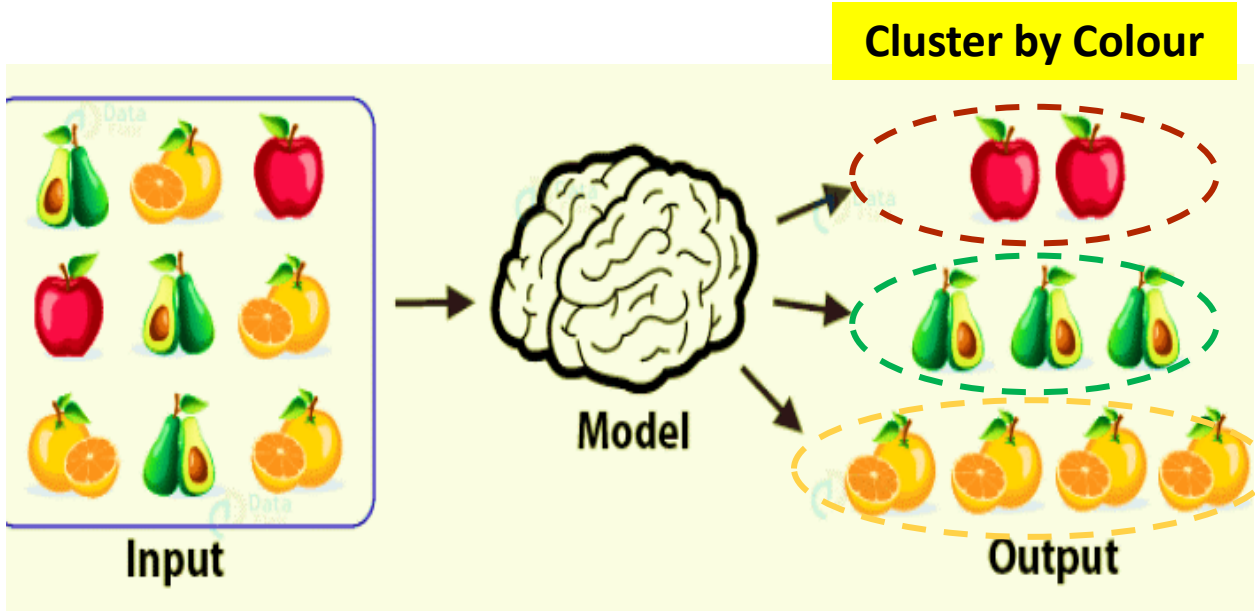
Supervised Learning

- Training Data is Labelled
- Task driven
- Classification/Regression



Unsupervised Learning

- Training Data is Not Labelled
- Data driven
- Clustering, Dimensionality Reduction

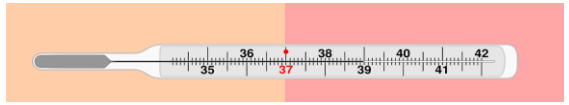


Example of Classification vs. Regression

Parameters	Classification	Regression
Output type	Discrete	Continuous
Trying to find	A boundary	Best Fit Line
Evaluation	Accuracy	Sum of squared errors

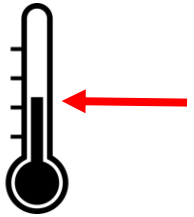
Examples

Will it be cold or Hot tomorrow?



Cold Hot

What is the temperature going to be tomorrow?



42°C
(Prediction)

Example of Classification vs. Regression

Students in Class

Classification → **Pass or Fail?**

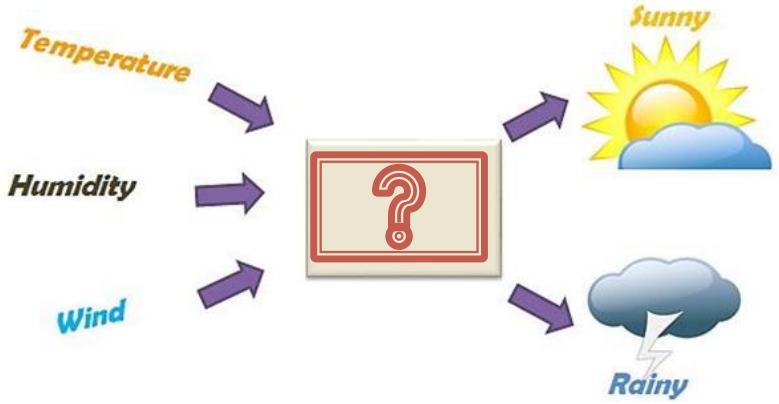
Student Prediction

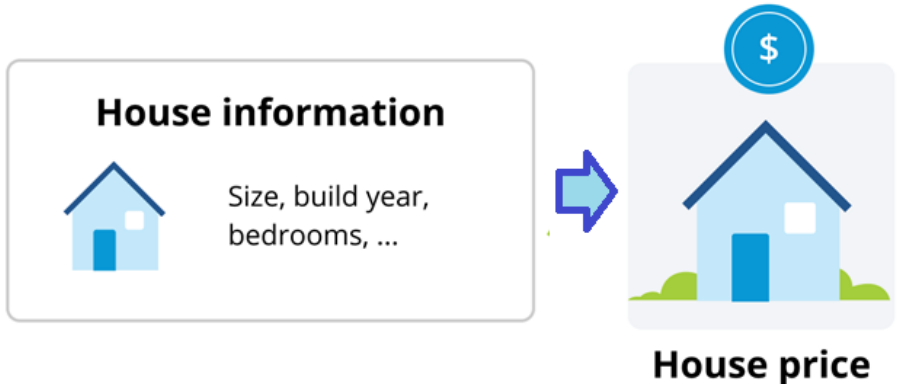
Regression → **Percentage?**

Student's marks Prediction

Q : Which ML technique is applicable?

1. Determine whether it will be rainy or sunny depending on temperature, humidity and wind observed on previous rainy and sunny days
2. Predict house price based on information related to the house, such as Size, build year, number of bedrooms.
3. Segment an image according to intensity (gray scale value)

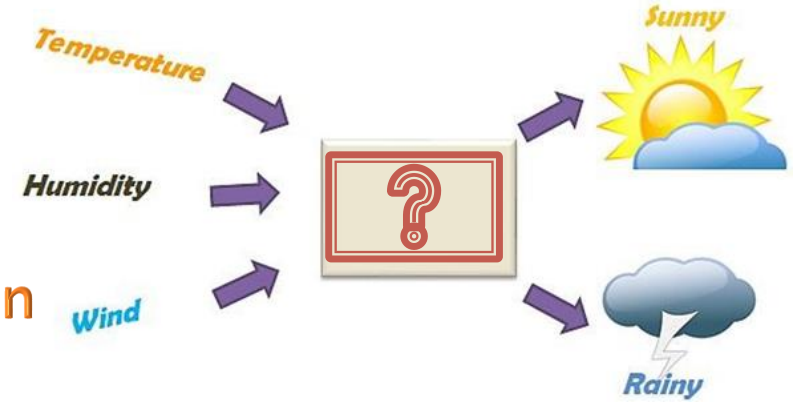




Q : Which ML technique is applicable?

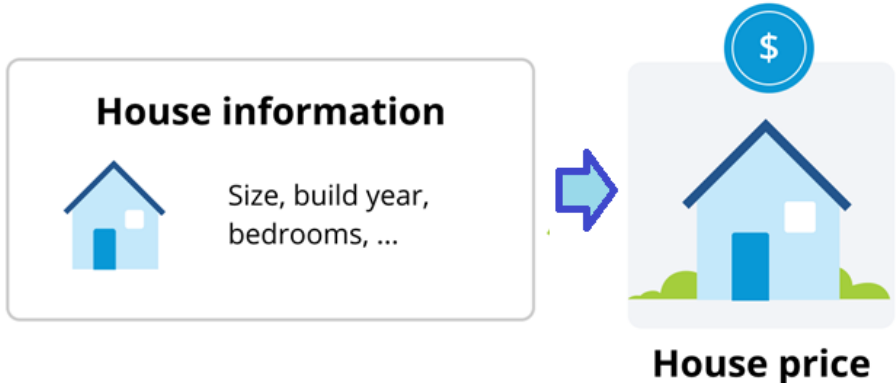
1. Determine whether it will be rainy or sunny depending on temperature, humidity and wind observed on previous rainy and sunny days

Classification



2. Predict house price based on information related to the house, such as Size, build year, number of bedrooms.

Regression



3. Segment an image according to intensity (gray scale value)

Clustering

How does Machine Learning Work?

Define Objectives

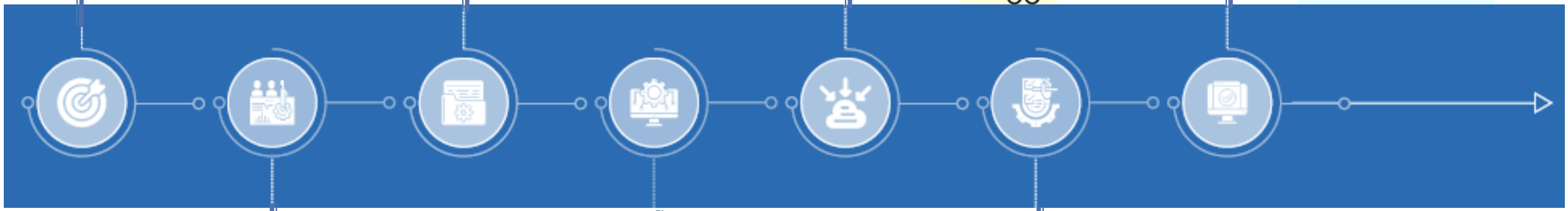
Eg. Classify email as Spam/Not Spam

Prepare Data

NOT SPAM SPAM

Train Model

Integrate Model



Collect Data

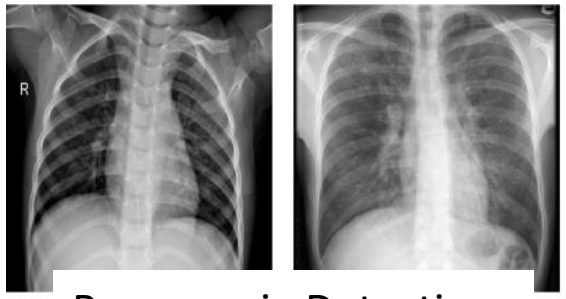
Select Algorithm

Test Model

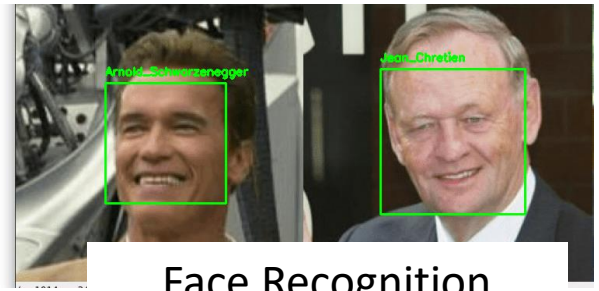
New Email

Collect Data

Images



Pneumonia Detection

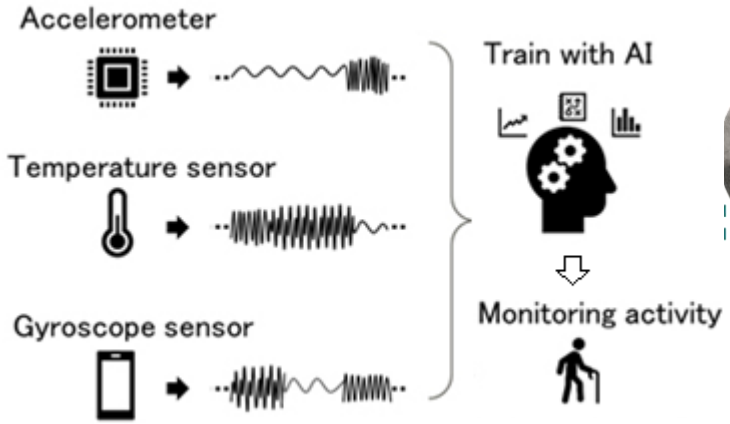


Face Recognition

Video

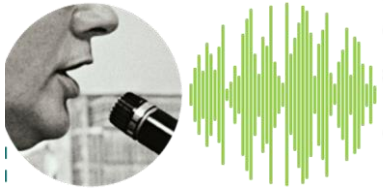
Activity Detection

Sensor data



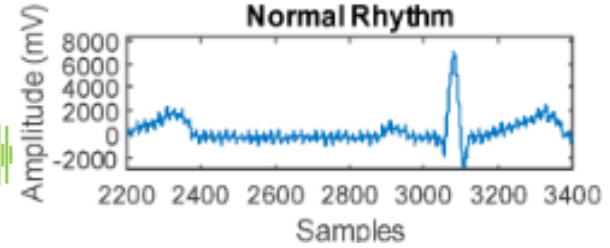
Activity Detection

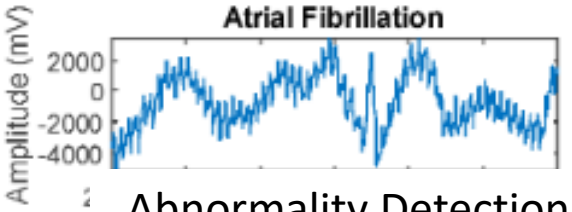
Speech



Speech Recognition

Biomedical (ECG)





Abnormality Detection

Demographic

Population Analysis

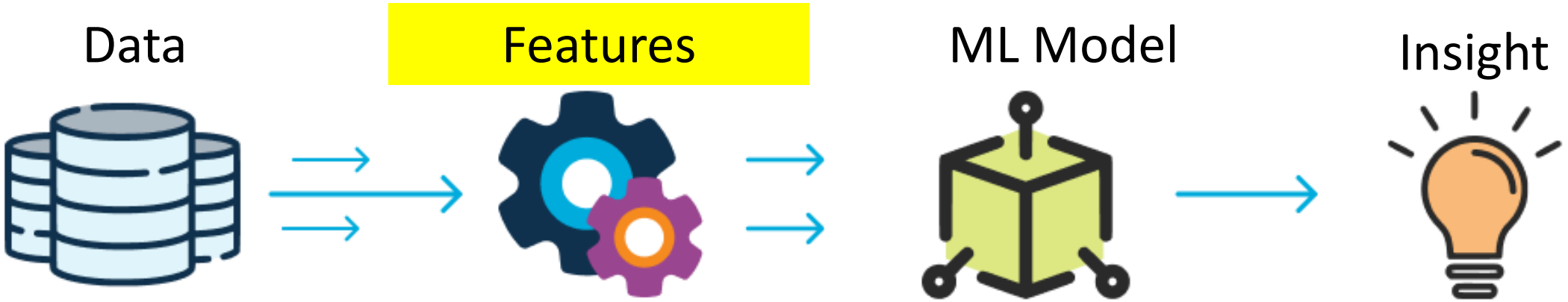
Other Tabular Data

Score Prediction	Lectures Attended	Previous CGPA	Final Exam Score
Student 1	3	15	35
Student 2	7	25	45
Student 3	16	55	75
Student 4	14	50	60

Data Cleaning

Remove Unwanted Observations	• Remove Duplicate observation • Remove redundant/irrelevant values
Missing Data Handling	• Fix unknown/ missing values using imputation • Remove incomplete entries
Structure error solving	• Fix problems with mislabelled data • Fix problems such as same attribute with different name
Outlier Management	• Unwanted Values that do not fit the dataset

How to give Data to the Model ?



- **Data**: collection of examples/ observations/ instances/ samples
- **Features**: Descriptor of the object. Feature is derived from the data and it represents the data. Features are properties/ attributes/ parameters that describe the data

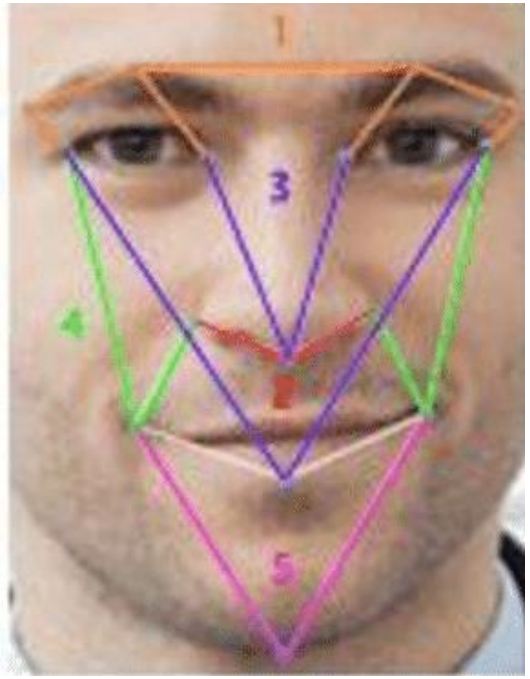
cs231n.stanford.edu/reports/2016/pdfs/010_Report.pdf

Feature Representation

- Feature Vector: $\boldsymbol{x}_i = [x_{i1} \quad x_{i2} \quad \cdots \quad x_{iM}]$ is a $1 \times M$ vector of features for i^{th} observation,
  M=Dimensionality of data (Number of features)

Features

1. Distance between eyes
2. Width of nose
3. Depth of eye sockets
4. Shape of cheekbones
5. Length of jaw-line



Feature Representation

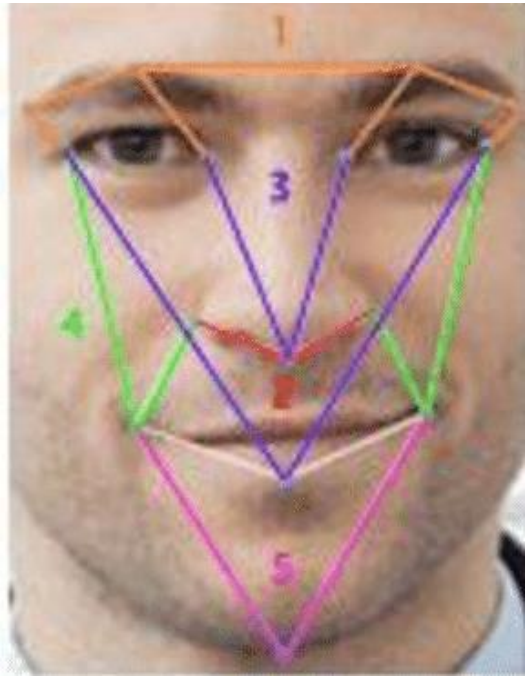
- Feature Vector: $\boldsymbol{x}_i = [x_{i1} \quad x_{i2} \quad \cdots \quad x_{iM}]$ is a $1 \times M$ vector of features for i^{th} observation,
  M=Dimensionality of data (Number of features)
- Feature Matrix: $X = \begin{bmatrix} x_{11} & x_{12} & \cdots x_{1M} \\ \vdots & & \\ x_{N1} & x_{N2} & x_{NM} \end{bmatrix}$

N=Size of data (Number of Observations)

- Target Label $Y = \begin{bmatrix} y_1 \\ \vdots \\ y_N \end{bmatrix}$ Eg. $Y = \begin{bmatrix} \text{John} \\ \vdots \\ \text{Jane} \end{bmatrix}$

Features

1.Distance between eyes
2.Width of nose
3.Depth of eye sockets
4.Shape of cheekbones
5.Length of jaw-line



Feature Representation

- **Feature Space**: Each observation is a point in M −dimensional feature space

Eg: 2D feature space

Feature Engineering Example: Classification of Animals

Viper

Rohu

Python

Cobra

Crocodile

Golden Poison Frog

Emerald tree boas

Feature Engineering Example

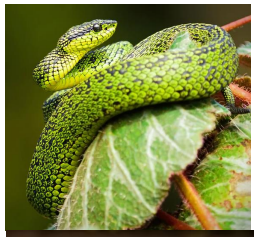
Viper

	FEATURES					LABEL
Name	Egg-laying	Scales	Poisonous	Cold-blooded	#legs	Reptile
Viper	True	True	True	True	0	True

- Initial model:

  Not enough information to classify new image

Feature Engineering Example



Viper

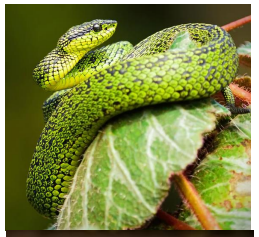
Cobra

	FEATURES					LABEL
Name	Egg-laying	Scales	Poisonous	Cold-blooded	#legs	Reptile
Viper	True	True	True	True	0	True
Cobra	True	True	True	True	0	True

Initial model has following features:

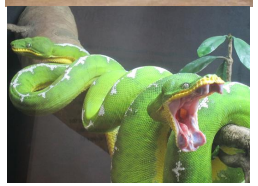
- Egg laying
- Scales
- Poisonous
- Cold-blooded
- No of Legs (in above 2 cases: 0)
- Reptile

Feature Engineering Example



Viper

Cobra



Emerald tree boas

	FEATURES					LABEL
Name	Egg-laying	Scales	Poisonous	Cold-blooded	#legs	Reptile
Viper	True	True	True	True	0	True
Cobra	True	True	True	True	0	True
Emerald tree boas	False	True	False	True	0	True

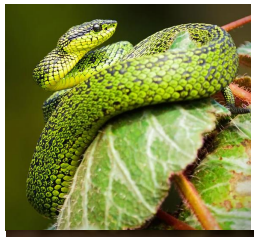
Now, model has following features:

- ❖Scales
- ❖Cold-blooded
- ❖No Legs

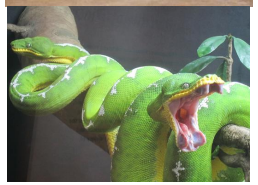
Emerald is reptile, but doesn't FIT in the developed model

➢ as not satisfying all the features

Feature Engineering Example


Viper


Cobra


Emerald tree boas


Crocodile

		FEATURES				LABEL
Name	Egg-laying	Scales	Poisonous	Cold-blooded	#legs	Reptile
Viper	True	True	True	True	0	Yes
Cobra	True	True	True	True	0	Yes
Emerald tree boas	False	True	False	True	0	Yes
Crocodile	True	True	False	True	4	Yes

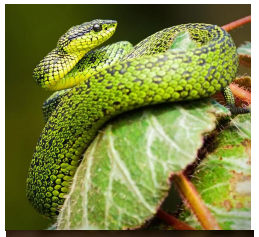
Now, features are as follows:

- Scales
- Cold-blooded
- Has 0 or 4 legs

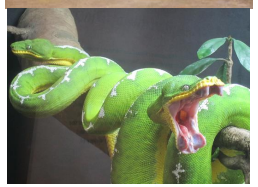
Crocodile is reptile, but doesn't FIT in the developed model

➢ as not satisfying all the features

Feature Engineering Example


Viper


Cobra


Emerald tree boas


Crocodile

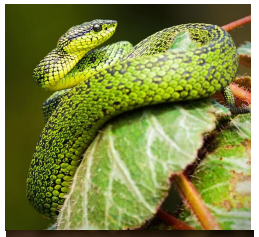

Golden Poison Frog

		FEATURES				LABEL
Name	Egg-laying	Scales	Poisonous	Cold-blooded	#legs	Reptile
Viper	True	True	True	True	0	Yes
Cobra	True	True	True	True	0	Yes
Emerald tree boas	False	True	False	True	0	Yes
Crocodile	True	True	False	True	4	Yes
Golden Poison Frog	**True**	**False**	**True**	**False**	**4**	**No**

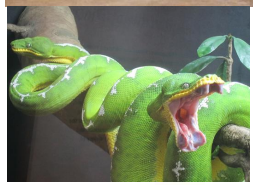
Now, features are as follows:

- Scales
- Cold-blooded
- Has 0 or 4 legs

Feature Engineering Example


Viper


Cobra


Emerald tree boas


Crocodile


Golden Poison Frog

	FEATURES					LABEL
Name	Egg-laying	Scales	Poisonous	Cold-blooded	#legs	Reptile
Viper	True	True	True	True	0	Yes
Cobra	True	True	True	True	0	Yes
Emerald tree boas	False	True	False	True	0	Yes
Crocodile	True	True	False	True	4	Yes
Golden Poison Frog	**True**	**False**	**True**	**False**	**4**	**No**
Rohu	True	True	False	True	**0**	**No**

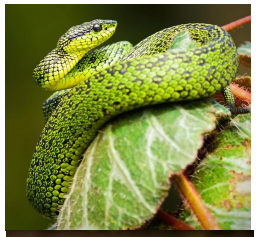

Rohu

Now, features are as follows:

- ❖Scales
- ❖Cold-blooded
- ❖Has 0 or 4 legs

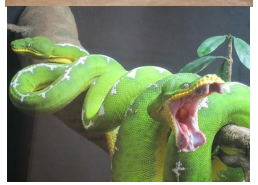
Features of Rohu (Fish) meet the features of current model
But Rohu is not a REPTILE

Feature Engineering Example



Viper

Cobra



Emerald tree boas

Crocodile

Golden Poison Frog

		FEATURES				LABEL
Name	Egg-laying	Scales	Poisonous	Cold-blooded	#legs	Reptile
Viper	True	True	True	True	0	Yes
Cobra	True	True	True	True	0	Yes
Emerald tree boas	False	True	False	True	0	Yes
Crocodile	True	True	False	True	4	Yes
Golden Poison Frog	**True**	**False**	**True**	**False**	**4**	**No**
Rohu	True	True	False	True	**0**	**No**
Python	True	True	False	True	**0**	Yes

Now, features are as follows:

- Scales
- Cold-blooded
- Has 0 or 4 legs

Features of Rohu and Python are **exactly same,** but their **labels** are **different**
No easy way to add or remove features that help to classify correctly

Rohu

Python

Example

	FEATURES				LABEL
Name	Egg-laying	Scales	Poisonous	Cold-blooded	#legs	Reptile
Viper	True	True	True	True	0	Yes
Cobra	True	True	True	True	0	Yes
Emerald tree boas	False	True	False	True	0	Yes
Crocodile	True	True	False	True	4	Yes
Golden Poison Frog	**True**	**False**	**True**	**False**	**4**	**No**
Rohu	True	True	False	True	**0**	**No**
Python	True	True	False	True	**0**	Yes

Good model has considered following features:

- ❖Cold-blooded
- ❖Scales

No model is perfect !

- ➢ No false negatives

(anything classified as not reptile is correctly labelled)

- ➢ Some false positives

(may incorrectly label some animals as Reptile)

Need to Measure Distances between Features

Feature engineering

- Decide which features to include in the model

(As using all features merely add noise to the learning process)

Second step is to define…

Distance

- It helps to measure distances between training data points and new instances.

- It helps decide how to weigh relative importance of different dimensions in feature vector, which impacts definition of distance.

Measuring Distance Between Animals

- In considered example, each animal consist:
  - four binary features (Egg laying, scales, poisonous, cold-blooded)
  - one integer feature (leg-0 to 4)

- One way to learn to separate reptiles from non-reptile is to measure the distance between pairs of data points (animals) , and use that:
  - To **cluster** nearby examples into a common class (unlabelled data),

or

  - To find a **classifier surface** in feature space that optimally separates different groups of data points from other groups

Viper = [1,1,1,1,0]

Emerald tree boas = [0,1,0,1,0]

Golden Poison Frog= [1,0,1,0,4]

Convert Given Dataset into set of feature vectors

Measuring Distance Between Animals

- Aim: Separate reptiles from non-reptile

- Approach 1:
- measure the distance between pairs of data points (animals)
- Then **Cluster** nearby examples into a common class (unlabelled data)

Minkowski Difference

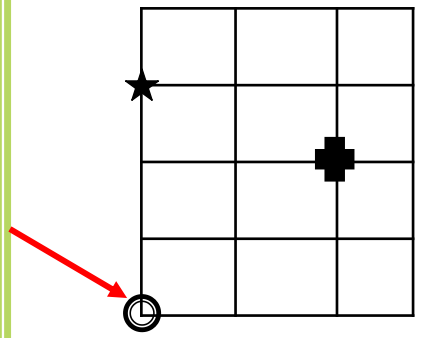
$$dist(X1, X2, p) = \sum_{k=1}^{len} abs\ (X1_k - X2_k)^p)^{1/p}$$

To measure distance between datapoints

p = 1: Manhattan Distance
p = 2: Euclidean Distance

Is circle closer to star or cross?

- Euclidean distance
  - Cross-2.8
  - Star-3
- Manhattan Distance
  - Cross-4
  - Star-3



Typically Euclidean distance is used; Manhattan may be appropriate if different dimensions are not comparable

Minkowski Difference

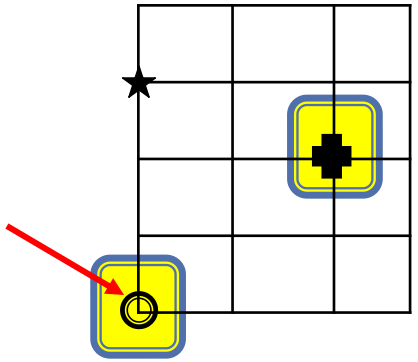
$$dist(X1, X2, p) = \sum_{k=1}^{len} abs\ (X1_k - X2_k)^p)^{1/p}$$

p = 1: Manhattan Distance
p = 2: Euclidean Distance

Need to measure distance between feature vectors

Is circle closer to star or cross?

- Euclidean distance
  - Cross-2.8
  - Star-3
- Manhattan Distance
  - Cross-4
  - Star-3



Typically Euclidean distance is used; Manhattan may be appropriate if different dimensions are not comparable

$$dist(\circ, +, 2) = \sqrt{(2-0)^2 + (2-0)^2} = \sqrt{8} = 2.8$$

Minkowski Difference

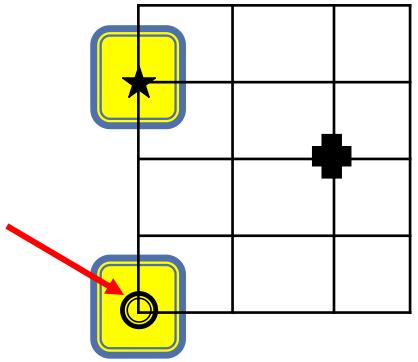
$$dist(X1, X2, p) = \sum_{k=1}^{len} abs\ (X1_k - X2_k)^p)^{1/p}$$

p = 1: Manhattan Distance
p = 2: Euclidean Distance

Need to measure distance between feature vectors

Is circle closer to star or cross?

- Euclidean distance
  - Cross-2.8
  - Star-3
- Manhattan Distance
  - Cross-4
  - Star-3



Typically Euclidean distance is used; Manhattan may be appropriate if different dimensions are not comparable

$$dist(\circ, \star\ 2) = \sqrt{(0-0)^2 + (3-0)^2} = \sqrt{9} = 3$$

Minkowski Difference

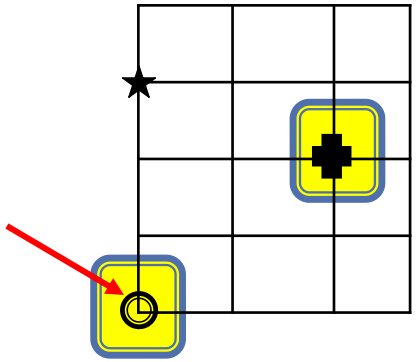
$$dist(X1, X2, p) = \sum_{k=1}^{len} abs\ (X1_k - X2_k)^p)^{1/p}$$

p = 1: Manhattan Distance
p = 2: Euclidean Distance

Need to measure distance between feature vectors

Is circle closer to star or cross?

- Euclidean distance
  - Cross-2.8
    - Star-3
- Manhattan Distance
  - Cross-4
    - Star-3



$$dist(\circ, +, 1) = |(2 - 0) + (2 - 0)| = 4$$

Typically Euclidean distance is used; Manhattan may be appropriate if different dimensions are not comparable

Minkowski Difference

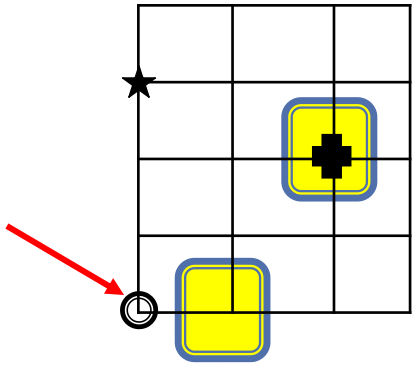
$$dist(X1, X2, p) = \sum_{k=1}^{len} abs\ (X1_k - X2_k)^p)^{1/p}$$

p = 1: Manhattan Distance
p = 2: Euclidean Distance

Need to measure distance between feature vectors

Is circle closer to star or cross?

- Euclidean distance
  - Cross-2.8
    - Star-3
- Manhattan Distance
  - Cross-4
    - Star-3



Typically Euclidean distance is used; Manhattan may be appropriate if different dimensions are not comparable

$$dist(\circ, +, 1) = |(2 - 0) + (2 - 0)| = 4$$

Minkowski Difference

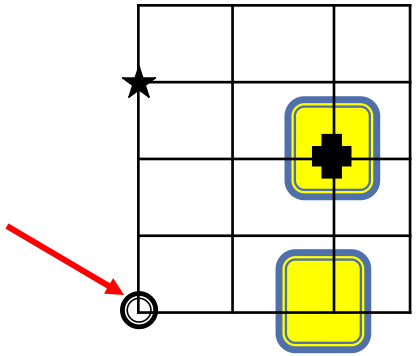
$$dist(X1, X2, p) = \sum_{k=1}^{len} abs\ (X1_k - X2_k)^p)^{1/p}$$

p = 1: Manhattan Distance
p = 2: Euclidean Distance

Need to measure distance between feature vectors

Is circle closer to star or cross?

- Euclidean distance
  - Cross-2.8
    - Star-3
- Manhattan Distance
  - Cross-4
    - Star-3



Typically Euclidean distance is used; Manhattan may be appropriate if different dimensions are not comparable

$$dist(\circ, +, 1) = |(2 - 0) + (2 - 0)| = 4$$

Minkowski Difference

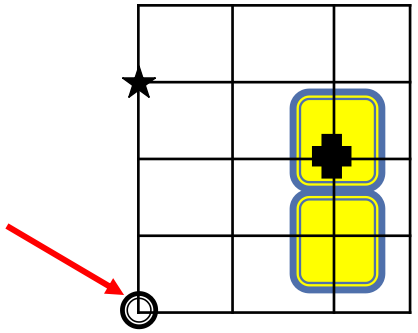
$$dist(X1, X2, p) = \sum_{k=1}^{len} abs\ (X1_k - X2_k)^p)^{1/p}$$

p = 1: Manhattan Distance
p = 2: Euclidean Distance

Need to measure distance between feature vectors

Is circle closer to star or cross?

- Euclidean distance
  - Cross-2.8
    - Star-3
- Manhattan Distance
  - Cross-4
    - Star-3



$$dist(\circ, +, 1) = |(2 - 0) + (2 - 0)| = 4$$

Typically Euclidean distance is used; Manhattan may be appropriate if different dimensions are not comparable

Minkowski Difference

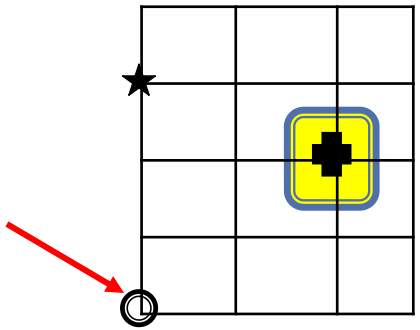
$$dist(X1, X2, p) = \sum_{k=1}^{len} abs\ (X1_k - X2_k)^p)^{1/p}$$

p = 1: Manhattan Distance
p = 2: Euclidean Distance

Need to measure distance between feature vectors

Is circle closer to star or cross?

- Euclidean distance
  - Cross-2.8
    - Star-3
- Manhattan Distance
  - Cross-4
    - Star-3



$$dist(\circ, +, 1) = |(2 - 0) + (2 - 0)| = 4$$

Typically Euclidean distance is used; Manhattan may be appropriate if different dimensions are not comparable

Minkowski Difference

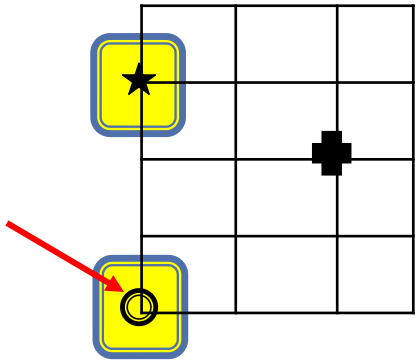
$$dist(X1, X2, p) = \sum_{k=1}^{len} abs\ (X1_k - X2_k)^p)^{1/p}$$

p = 1: Manhattan Distance
p = 2: Euclidean Distance

Need to measure distance between feature vectors

Is circle closer to star or cross?

- Euclidean distance
  - Cross-2.8
  - Star-3
- Manhattan Distance
  - Cross-4
  - Star-3



$$dist(\circ, \star\ 1) = |(0 - 0) + (3 - 0)| = 3$$

Typically Euclidean distance is used; Manhattan may be appropriate if different dimensions are not comparable

Minkowski Difference

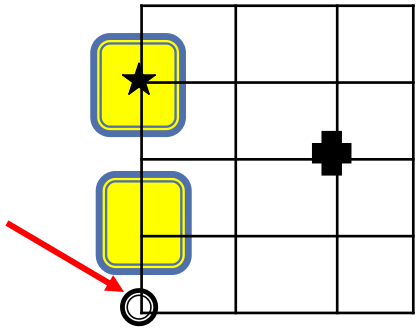
$$dist(X1, X2, p) = \sum_{k=1}^{len} abs\ (X1_k - X2_k)^p)^{1/p}$$

p = 1: Manhattan Distance
p = 2: Euclidean Distance

Need to measure distance between feature vectors

Is circle closer to star or cross?

- Euclidean distance
  - Cross-2.8
  - Star-3
- Manhattan Distance
  - Cross-4
  - Star-3



$$dist(\circ, \star\ 1) = |(0 - 0) + (3 - 0)| = 3$$

Typically Euclidean distance is used; Manhattan may be appropriate if different dimensions are not comparable

Minkowski Difference

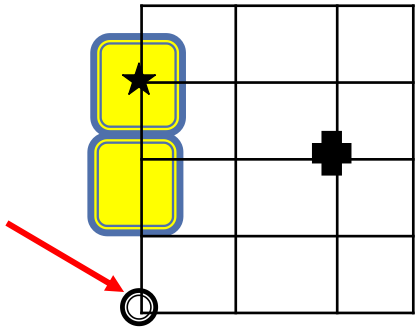
$$dist(X1, X2, p) = \sum_{k=1}^{len} abs\ (X1_k - X2_k)^p)^{1/p}$$

p = 1: Manhattan Distance
p = 2: Euclidean Distance

Need to measure distance between feature vectors

Is circle closer to star or cross?

- Euclidean distance
  - Cross-2.8
  - Star-3
- Manhattan Distance
  - Cross-4
  - Star-3



$$dist(\circ, \star\ 1) = |(0-0) + (3-0)| = 3$$

Typically Euclidean distance is used; Manhattan may be appropriate if different dimensions are not comparable

Minkowski Difference

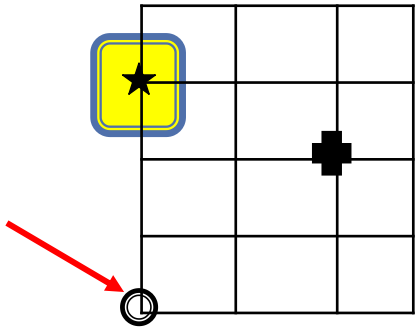
$$dist(X1, X2, p) = \sum_{k=1}^{len} abs\ (X1_k - X2_k)^p)^{1/p}$$

p = 1: Manhattan Distance
p = 2: Euclidean Distance

Need to measure distance between feature vectors

Is circle closer to star or cross?

- Euclidean distance
  - Cross-2.8
  - Star-3
- Manhattan Distance
  - Cross-4
  - Star-3



$$dist(\circ, \star\ 1) = |(0-0) + (3-0)| = 3$$

Typically Euclidean distance is used; Manhattan may be appropriate if different dimensions are not comparable

Euclidean Distance Between Animals

Feature Vector = {Egg-laying, Scales, Poisonous, Cold-blooded, #legs}

Viper = [1,1,1,1,0]

Emerald tree boas = [0,1,0,1,0]

Golden Poison Frog= [1,0,1,0,4]

Viper

Emerald tree boas

Golden Poison Frog

Euclidean Distance Between Animals

Viper = [1,1,1,1,0]
Emerald tree boas = [0,1,0,1,0]
Golden Poison Frog= [1,0,1,0,4]

	Viper	Emerald Tree Boas	Golden Poison Frog
Viper	--	1.414	4.243
Emerald Tree Boas	1.414	--	4.472
Golden Poison Frog	4.243	4.472	--

Example:

$$dist(Vip\ ETB, 2) = \sqrt{(1-0)^2 + (1-1)^2 + (1-0)^2 + (1-1)^2 + (0-0)^2}$$
$$= \sqrt{2} = 1.414$$

Euclidean Distance Between Animals

Viper = [1,1,1,1,0]
Emerald tree boas = [0,1,0,1,0]
Golden Poison Frog= [1,0,1,0,4]

	Viper	Emerald Tree Boas	Golden Poison Frog
Viper	--	1.414	4.243
Emerald Tree Boas	1.414	--	4.472
Golden Poison Frog	4.243	4.472	--

From the Table:
Both snakes are reasonably close to each other.
While golden poison frog is fairly distance away from them

Add a new animal

```
crocodile = Animal (‘crocodile’, [1,1,0,1,4] )
Animals.append(crocodile)
compareAnimals(animals, 3)
```

Add a new animal

crocodile = Animal ('crocodile', [1,1,0,1,4])
Animals.append(crocodile)
compareAnimals(animals, 3)

Viper = [1,1,1,1,0]
Emerald tree boas = [0,1,0,1,0]
Golden Poison Frog= [1,0,1,0,4]

	Viper	Emerald Tree Boas	Golden Poison Frog	Crocodile
Viper	--	1.414	4.243	4.123
Emerald Tree Boas	1.414	--	4.472	4.123
Golden Poison Frog	4.243	4.472	--	1.732
Crocodile	4.123	4.123	1.732	

Here comes the Manhattan Distance into picture

Two snakes are closer, but crocodile is closer to golden poison

- Crocodile differs from frog in 3 features, from boas in only 2 features
- Other features are binary: true (1) or false (0)
- While "legs" dimension is disproportionately large consist from 0-4

Using Binary Features

In binary features, feature "leg" : either legs(1) or no legs (0)

Viper = [1,1,1,1,0]
Emerald tree boas = [0,1,0,1,0]
Golden Poison Frog= [1,0,1,0,1]
Crocodile = [1,1,0,1,1]

	Viper	Emerald Tree Boas	Golden Poison Frog	Crocodile
Viper	--	1.414	1.732	1.414
Emerald Tree Boas	1.414	--	2.236	1.414
Golden Poison Frog	1.732	2.236	--	1.732
Crocodile	1.414	1.414	1.732	

Now crocodile is closer to snakes than it is to dart frog --- make more sense

Features Engineering Matters !

Issues of Concern When Model Learns

- Learned model depends on following considerations:
  - ➢ To measure distance between data points
  - ➢ Choice of right set of feature vectors
  - ➢ Constraints on complexity of model
    - ✓ Specified number of clusters (in case of Uncluttered data)
    - ✓ Complexity of separating surface
    - ✓ Want to avoid over fitting problem (each data point is in its own cluster, or a complex separating surface)

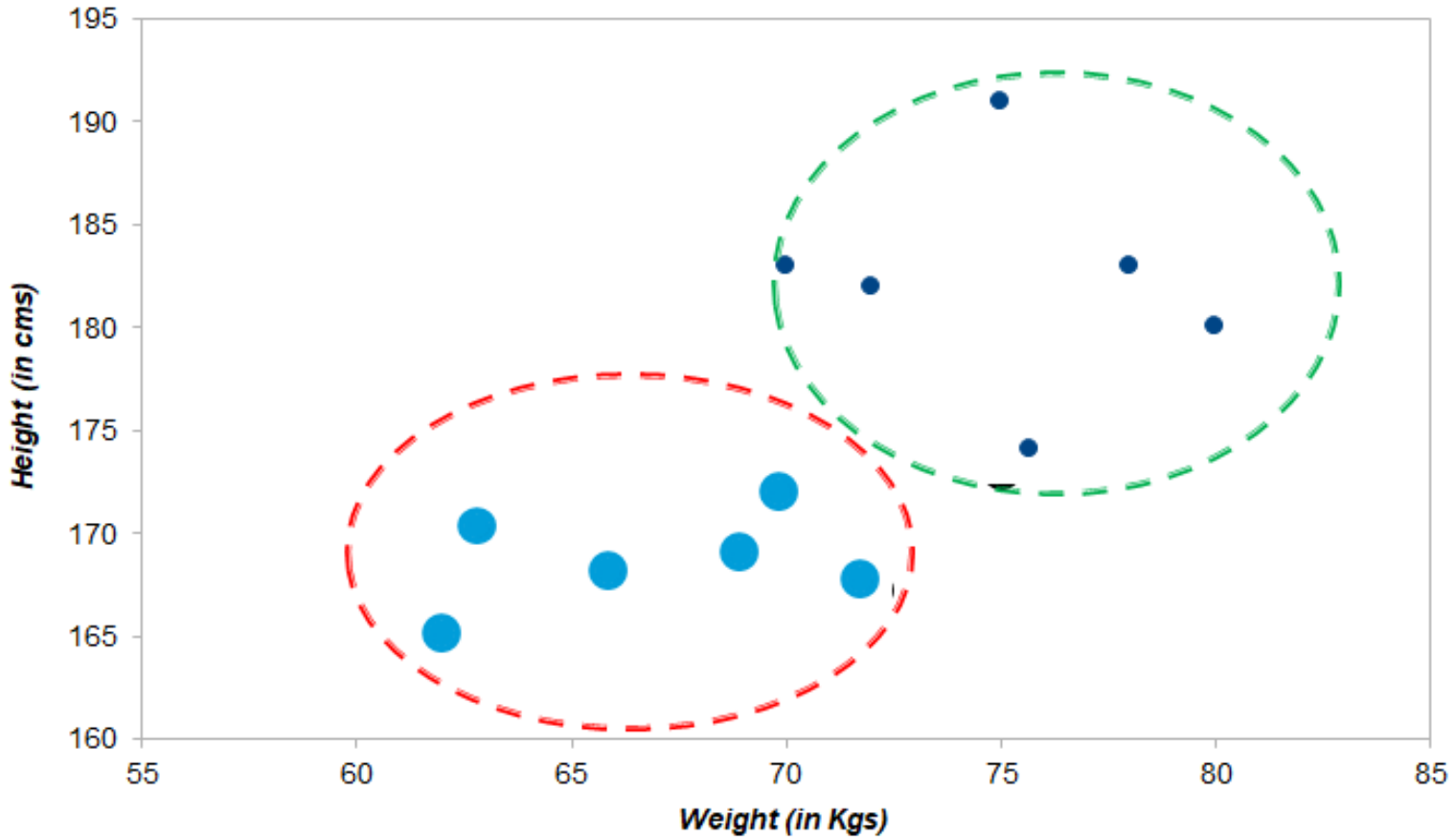
Clustering approaches

- Suppose there are “k” different groups in our training data, but don’t know their labels (here k=2)
  - Pick k samples (at random) from the training set of data.
  - Minimize the distance between remaining samples from the selected data points.
  - Find a new median sample data point in each cluster.
  - Repeat until no change is observed.

- ***Issues:***
  - How do we decide on the best number of clusters?
  - How do we select the best features, the best distance metric?

Clustering using unlabelled Data

Distribution of Weight vs Height



Approach 2: Classification approaches

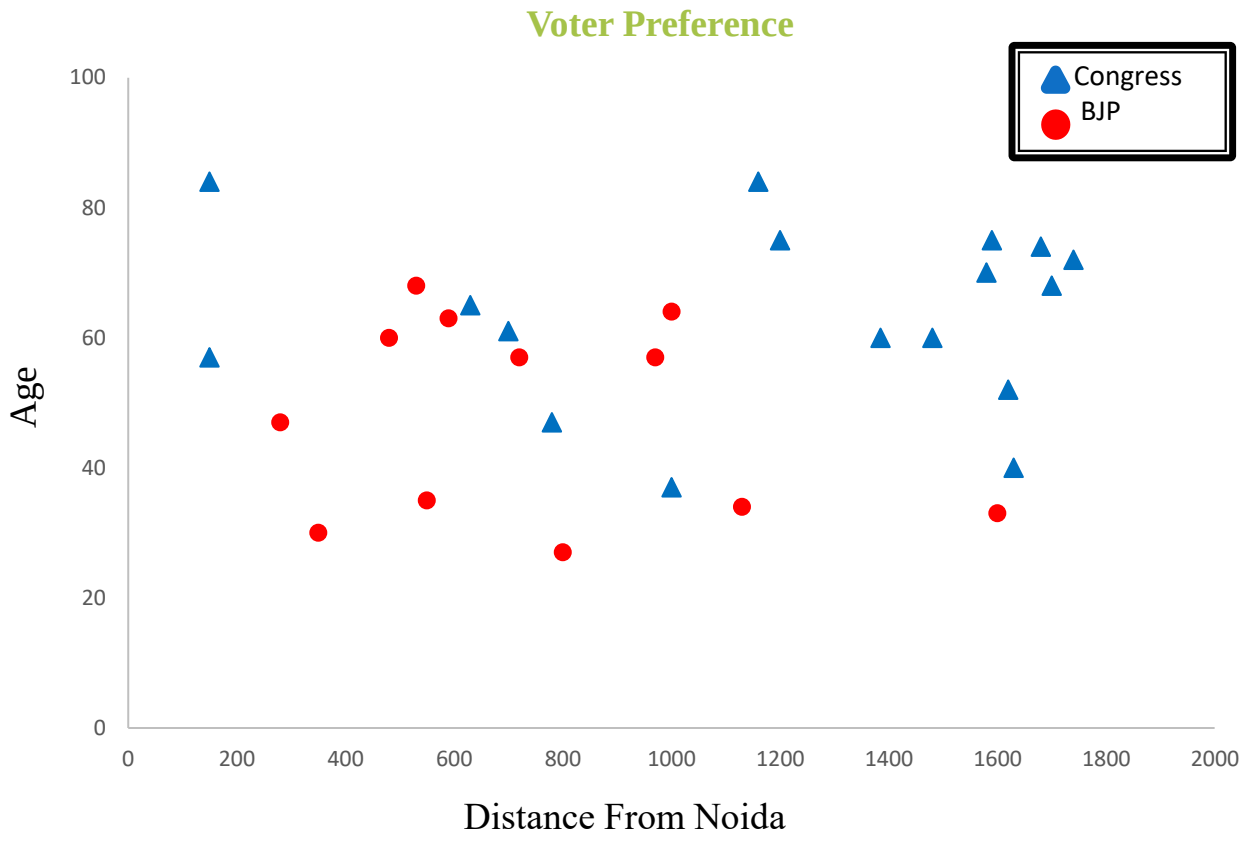
- Want to find boundaries in feature space that separate different classes of labelled data points
  - Look for simple surface (e.g. best line or plane) that separates classes
  - Look for more complex surfaces (subject to constraints) that separate classes
  - Use voting schemes
    - Find k nearest training examples, use majority vote to select label
- ***Issues:***
  - How do we avoid over-fitting to data?
  - How do we measure performance?
  - How do we select best features?

Classification

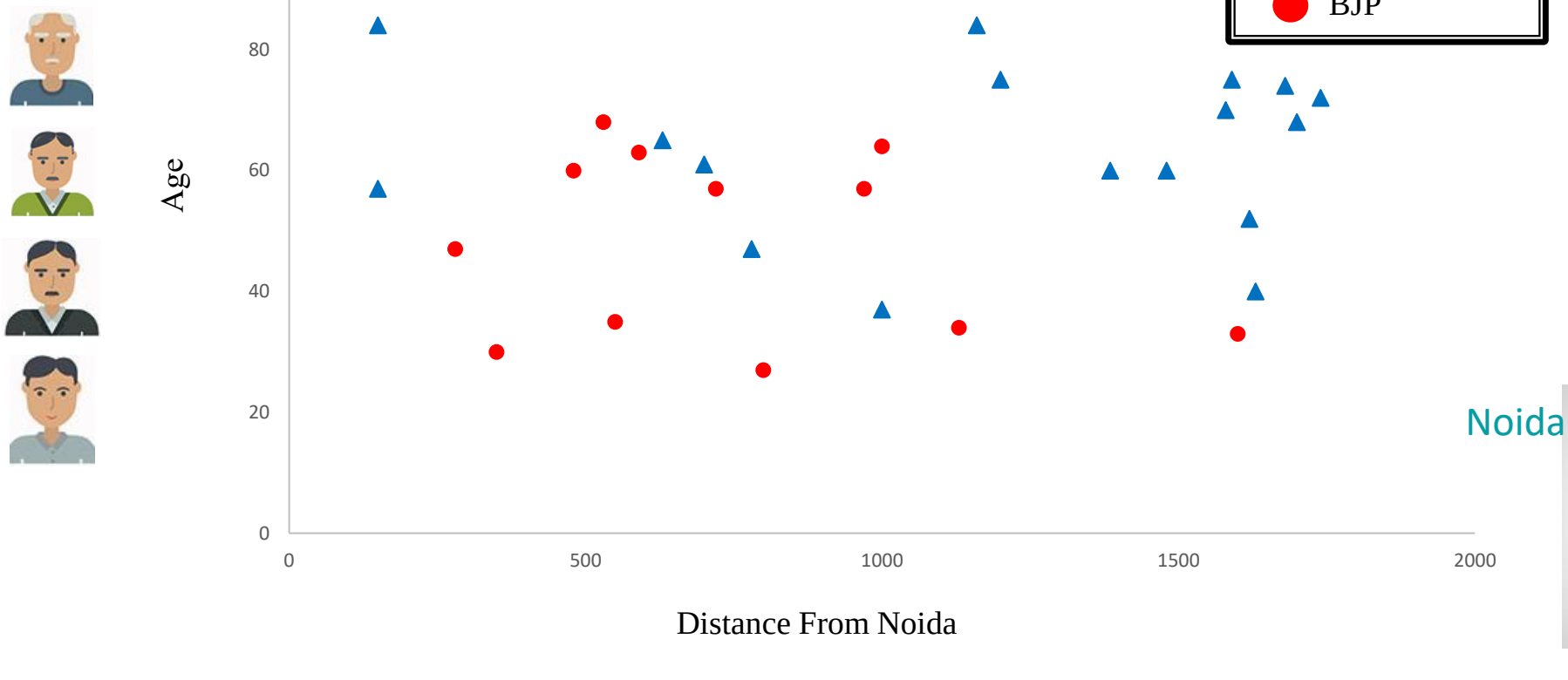
- Attempt to minimize error on training data
  - Similar to fitting a curve to data
- Evaluate on test data

Cluster Voter Preference by

1. Age, and
2. Distance from Noida

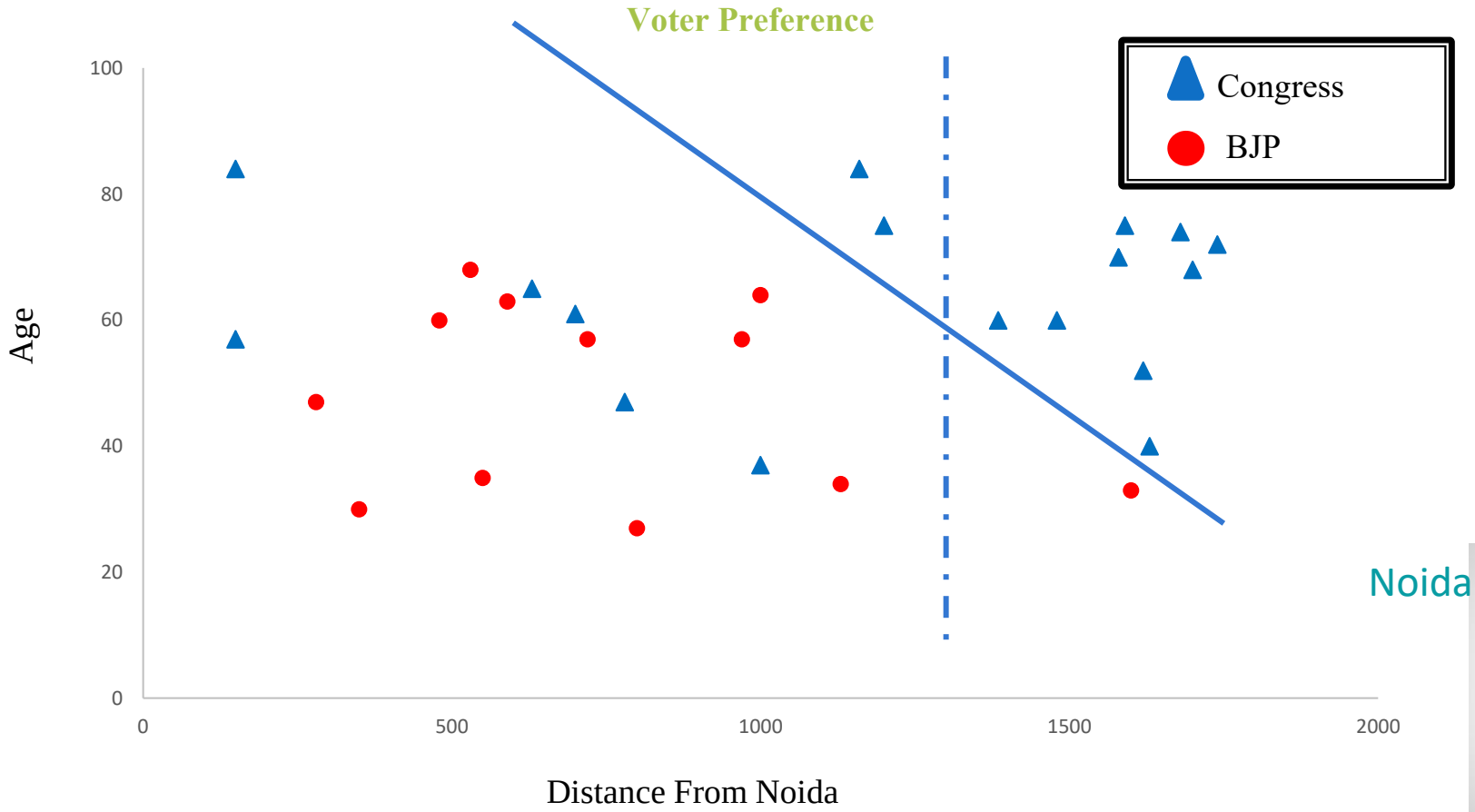


Random Division of Data into Training and Test Set

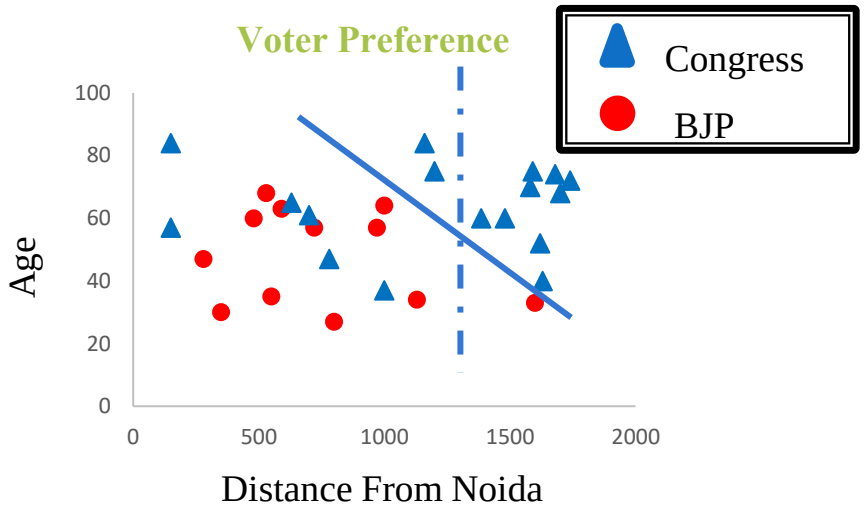


Possible Models for Training Set



Confusion Matrices (Training Error)



Predicted BJP

Actually BJP

	Pos	Neg
Pos	12	6
Neg	0	11

Solid Line

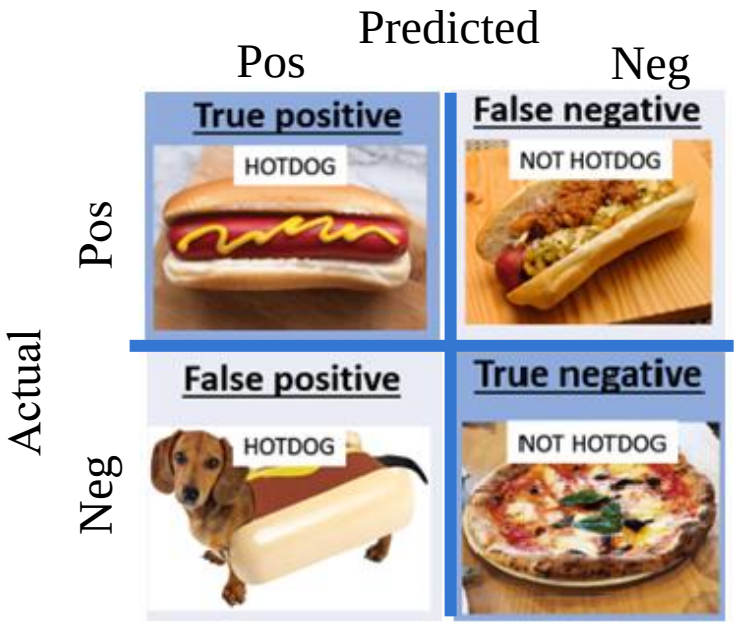
Predicted BJP

Actually BJP

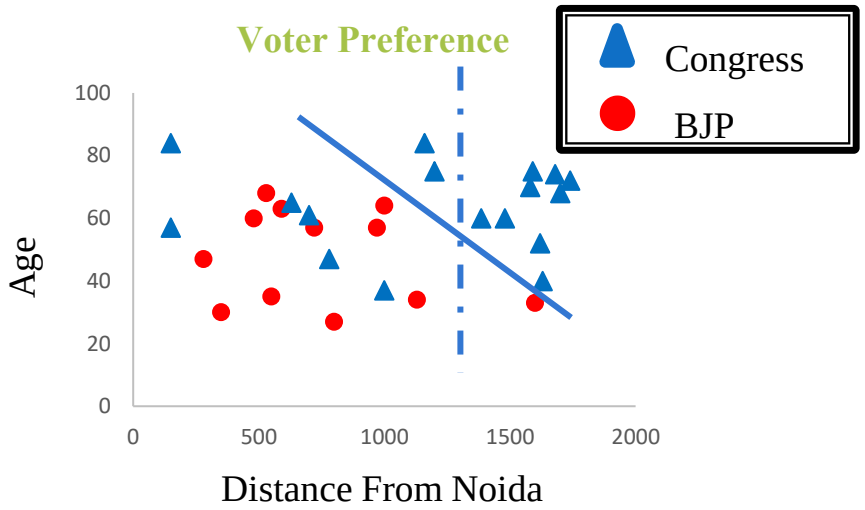
	Pos	Neg
Pos	11	6
Neg	1	11

Dashed Line

- **True Positive**: Actual value was Positive, the Model predicts Positive.
- **True Negative**: Actual value was Negative, the Model predicts Negative.
- **False Negative**: Actual value was Positive, the Model predicts Negative.
- **False Positive**: Actual value was Negative, the Model predicts Positive.



Training Accuracy of Models



Predicted BJP

Actually BJP	Pos	Neg
Pos	12	6
Neg	0	11

Solid Line

Predicted BJP

Actually BJP	Pos	Neg
Pos	11	6
Neg	1	11

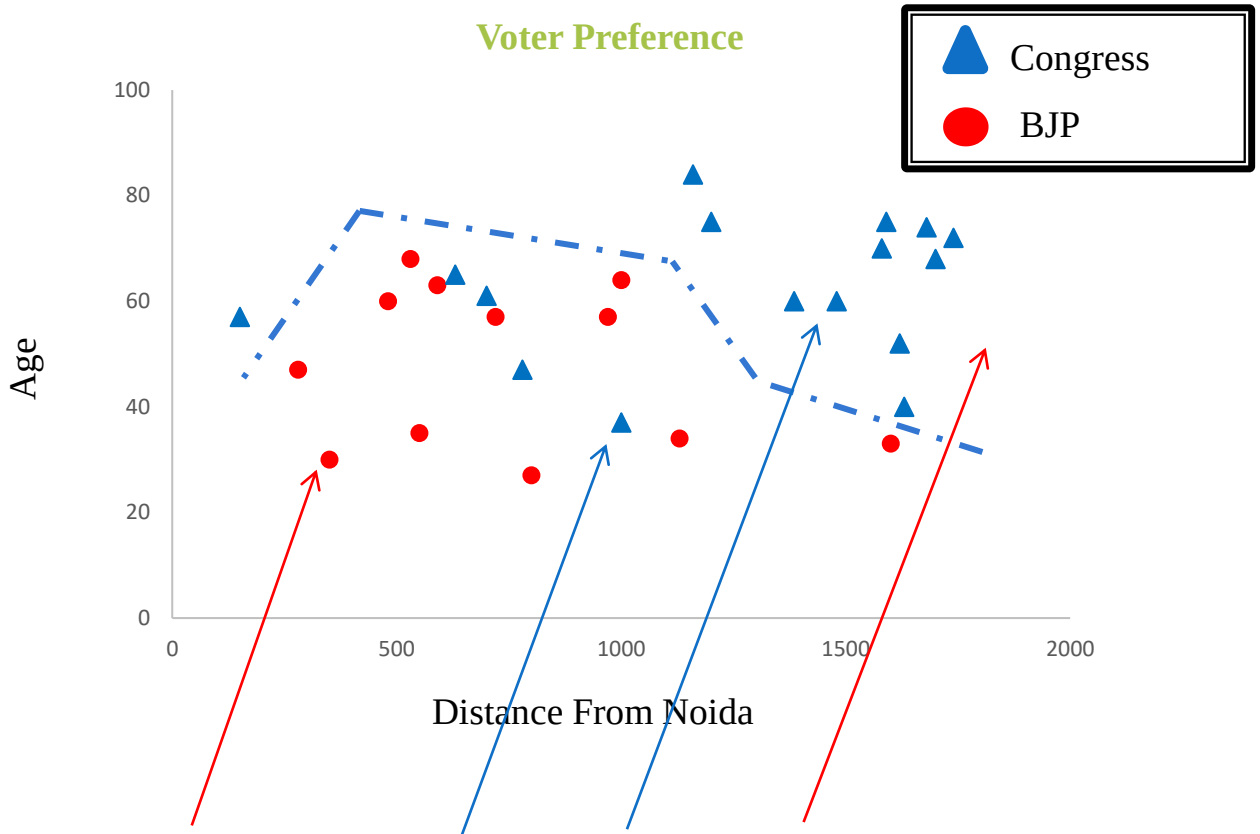
Dashed Line

$$Accuracy = \frac{true\ positive + true\ negative}{true\ positive + true\ negative + false\ positive + false\ negative}$$

For Solid Line, Accuracy= 0.85

For Dashed Line, Accuracy = 0.88

More Complex Model



TP= 12 FP= 4, TN = 12 FN=0

Accuracy = 24/26 = 0.92

Other Statistical Measures

$$Positive\ Predictive\ Value = \frac{true\ positive}{true\ positive + false\ positive}$$

- ❖Solid line model: 0.48
- ❖Dashed line model: 0.67
- ❖Complex model: 0.61

- One can also use following parameters:

$$Sensitivity = \frac{true\ positive}{true\ positive + false\ negative}$$

Percentage correctly found

$$Specificity = \frac{true\ negative}{true\ negative + false\ positive}$$

Percentage correctly rejected

Advantages and Limitations of Machine learning

Advantages

- Easily Identifies Trends and Patterns
- Reduced Human Intervention
- Continuous Improvement
- Handle Multi-dimensional Data
- Wide range of Applications

Limitations

- Data Acquisition
- Time and Resources
- Interpretation of Results
- High error-Susceptibility

Summary

- Machine learning methods provide a way of building models from datasets.
- Supervised learning uses labelled data while Unsupervised learning tries to infer from unlabelled data
- Unsupervised clustering tries to infer latent relation between the training set examples by clustering them into nearby groups
- Feature engineering requires us to decide which features to include in the model (Choice of features influence the results of ML model.)
- Choice of distance metrics between examples also influence the results.

References

1. Machine Learning: A Probabilistic Perspective" by Kevin Murphy, published by MIT Press, 2012.
2. "Pattern Recognition and Machine Learning" by Christopher M. Bishop, published by Springer, 2006.
3. "Python Machine Learning" by Sebastian Raschka and Vahid Mirjalili, published by Packt Publishing, 2015.
4. "Machine Learning Yearning" by Andrew Ng, published by Goodfellow Publishers, 2018.
5. "Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow" by Aurélien Géron, published by O'Reilly Media, 2019.
6. "Applied Predictive Modeling" by Max Kuhn and Kjell Johnson, published by Springer, 2013.
7. "Reinforcement Learning: An Introduction" by Richard S. Sutton and Andrew G. Barto, published by MIT Press in 2018.

Thank You !

Any Questions