



# Hands on with TensorFlow

-- *Sri Harsha Gajavalli*



# Who are we?

We are a group of students from IIIT Sri City who have embarked on a mission to build an AI-powered environment. We are members of an open community and we organise various events to proliferate knowledge of AI among developers..

Stay tuned to us:

<https://www.meetup.com/AI-Developer-Community-Intel>

<https://github.com/AIDeveloperCommunity-Intel>

# Who am I?

---

Founder

Hacker

Engineer

Mentor

Community Lead

Trainer

Coder

FreeLancer

Student

Speaker

Guide

Entrepreneur

Developer

Programmer

# Learner



# Artificial Intelligence

“The Science and Engineering of making Intelligent Machines”



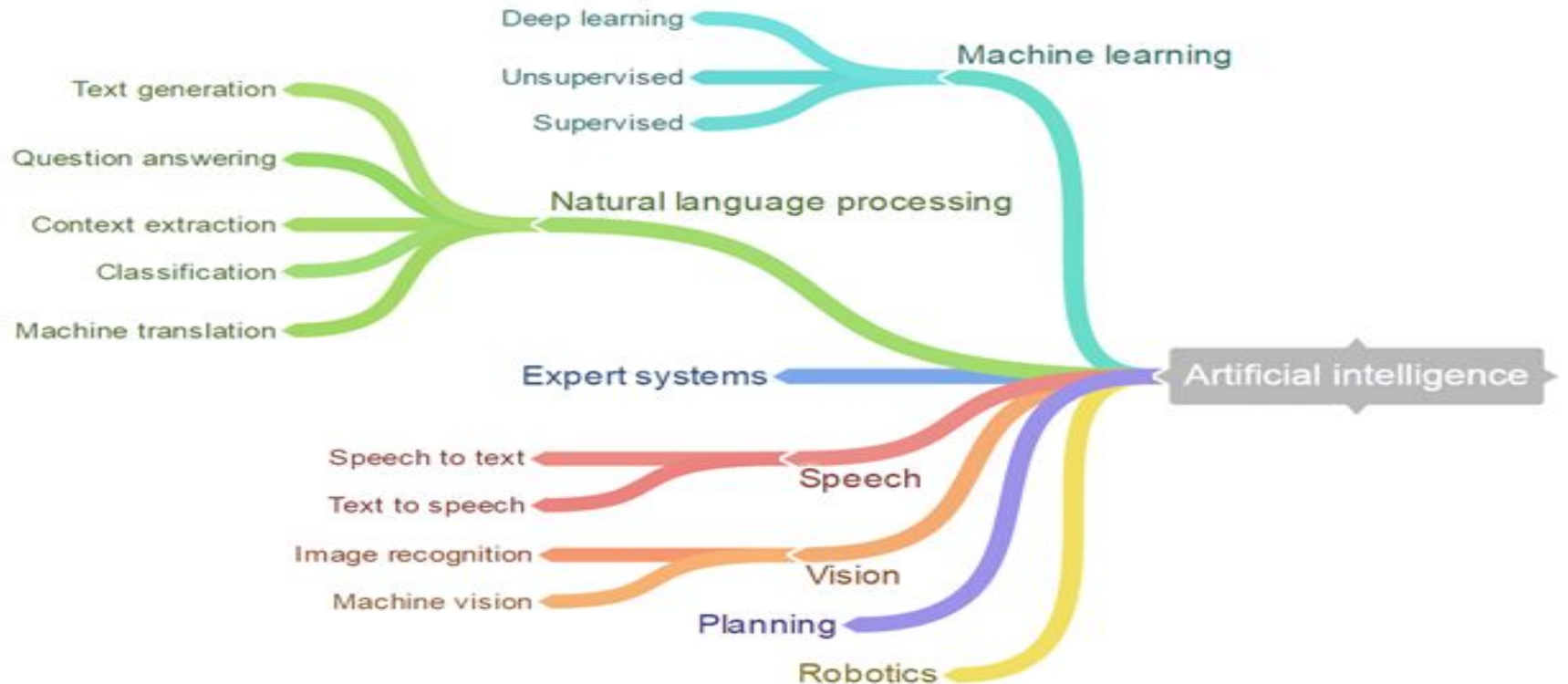
# AI?

Homo sapiens (Human beings) are able to control (and exploit) other species and nature because of their thinking capability.

We call programs intelligent if they exhibit behaviours that would be regarded intelligent if they were exhibited by human beings

--- Herbert Simon

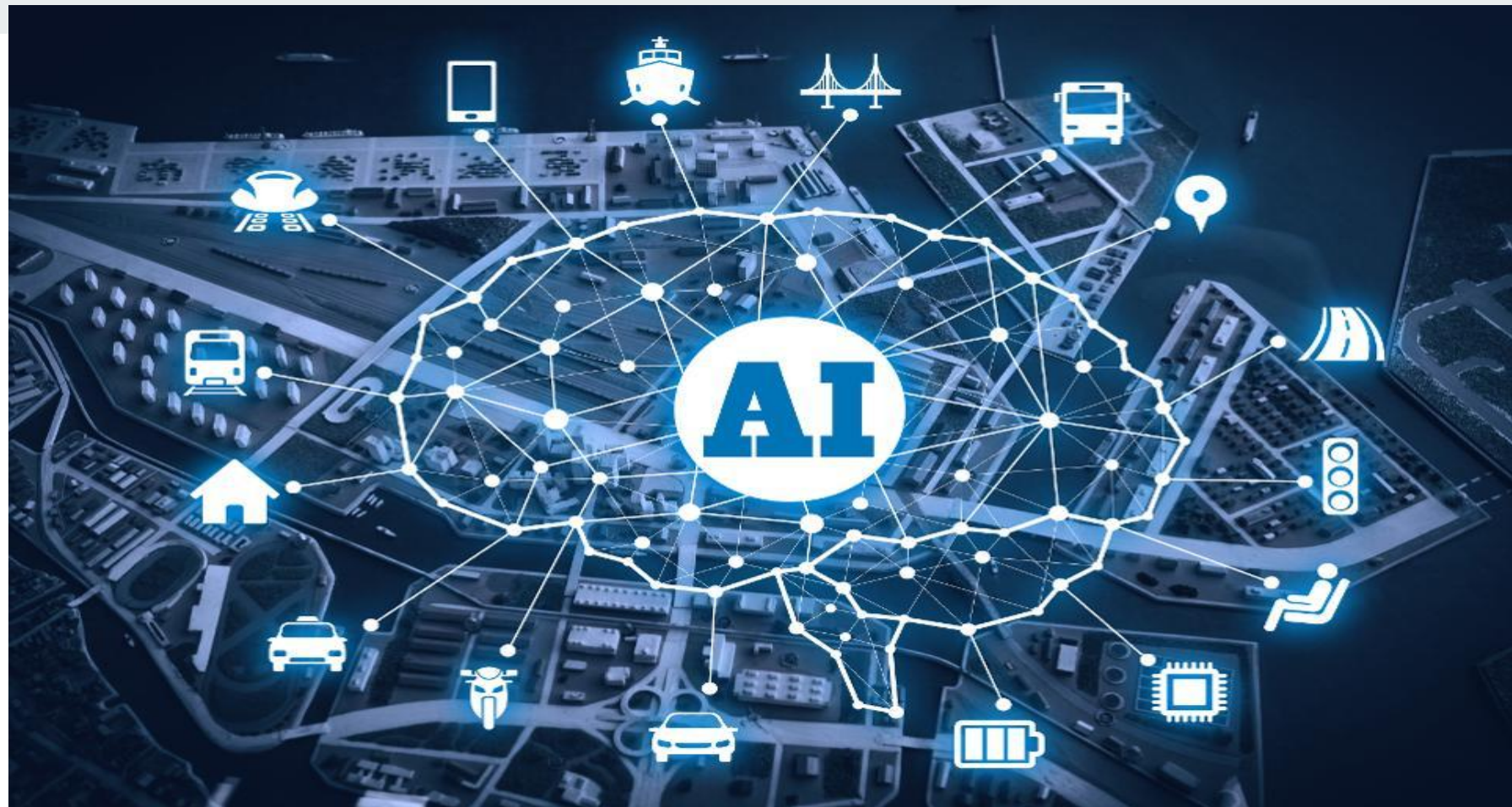
# Topics in AI:



# Where are we?











# Future Prospects

- Outperform Human in
  - Translating Languages
  - Writing High School essays
  - Driving a truck
  - Working in retail
  - Writing a best-selling book
  - Working as a surgeon
  - Automation of jobs



What is Machine Learning?  
What is Deep Learning?

---



# Machine Learning:

Ability of computers to learn without being explicitly programmed.

---- Wikipedia

**My View:** Making the machines more experienced by training with enough data instead of programming them explicitly to handle each and every scenario.



## **Deep Learning:**

**Deep Learning** is a subfield of machine learning concerned with algorithms inspired by the structure and function of the brain called artificial neural networks.



A Venn diagram consisting of three nested ellipses. The outermost ellipse is light blue and contains the text 'Field of Artificial Intelligence'. Inside it is a medium blue ellipse containing the text 'Field of Machine Learning'. Inside that is a small dark blue circle containing the text 'Deep Learning'. This illustrates that Deep Learning is a subset of Machine Learning, which is a subset of Artificial Intelligence.

**Field of Artificial Intelligence**

**Field of  
Machine Learning**

**Deep  
Learning**



What is this?





What is this?

Apple. Red Apple, Indeed



# What is this?

Apple. Red Apple, Indeed

## How do we know?



# What is this?

Apple. Red Apple, Indeed

## How do we know?

### Features:

- Color : Red
- Taste: Sweet
- Shape etc.,



Oooopss... What is this?



Oooopss... What is this?

Green Apple!

# Deductive vs Inductive Learning



- Deductive
  - Rules of the game are (hard coded) given ahead.
  - Eg: An algorithm to do multiplication of numbers is given. Given any two numbers you can apply this and get the answer.
- Inductive
  - We are given with examples (not the concept). We need to learn the mapping from i/p to o/p.
  - Supervised learning problems in AI comes under this



# Learning Strategies:

- Supervised
  - Classification, Regression, ...
- Unsupervised
  - Clustering, density estimation
- Reinforced
  - A robot navigating through obstacles
- Learn the good features (attributes)
  - Feature extraction





# Classification Problem?

- Let there are two classes of objects.
  - Class 1: Set of dog pictures
  - Class 2: Set of cat pictures
- Problem is –
  - Given a picture, you should say whether it is cat or dog.
  - For a human being it is easy..., but for a machine it is a non-trivial problem.

# Training/Learning Phase:



# Testing:



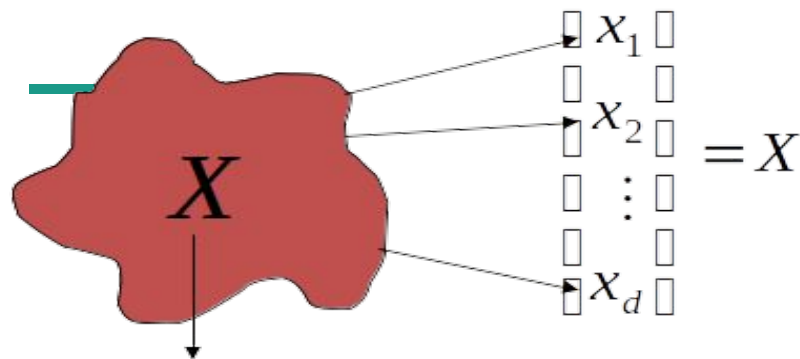
Dog

This picture as it  
is may not be in  
the training set

# What is learning?



- Child has learnt what is it that is common among dogs ... and, what is it that is common among cats... also, what are the distinguishing features/attributes.
- Child has learnt the pattern (regularity) behind all dogs and the pattern behind all cats.
- Child then recognized a test image as having a particular pattern that is unique to dogs.



$$X \in \mathcal{X}$$

$X$

$\mathcal{X}$

Class to which  $X$  belongs is  $y \in Y$

-Needs to be estimated, based on training set.

### Task

- To design a classifier (decision rule)  $f : \mathcal{X} \rightarrow Y$  which decides about the class label based on  $X$ .



# Feature extraction:

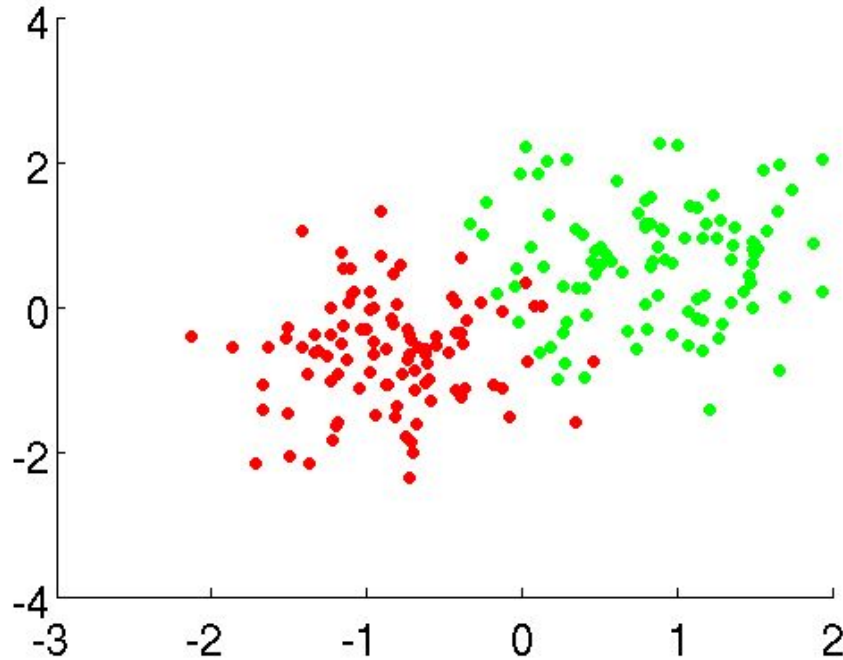
Good Features:

- Objects from same class have similar feature values
- Objects from different classes have different values

# Feature Space



Plot of total normalized data



Training Set is shown





# Learning Steps:

- **Feature extraction:**
  - This is an important step. Good features are needed.
  - This is a lower level step. Normally done by techniques like image processing, speech processing, video processing, etc.
- **Training set:**
  - Set of feature vectors along with their class labels.
  - An expert can see a few examples and give labels to them based on his experience.
- Build the classifier by using the training set.



# Classification Problem

- Given a training set, build the classifier.
- One has to evaluate, how good is the built classifier.
  - Of course, it has to agree with the training set
    - Is this 100% true?
  - But, it should do more than this.
  - The behavior of the classifier when it is asked to classify some thing which is not in the training set determines the quality.



## An easy, but bad classifier

- Remember the training set.
- See whether the given feature vector to be classified is available in the training set.
- If yes, then return the label of that training example.
- Else return a random class label.

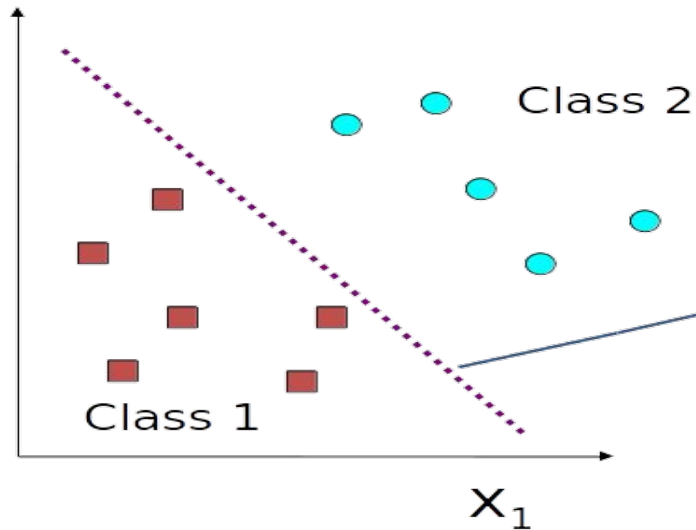
**This is called Rote learning**



## A few classifiers:

- KNN Classifier
- Decision Trees
- Random Forests
- Bayes Classifier, Naive Bayes classifier
- Artificial Neural Networks
- SVM
- .....

# Linear Classifier



## **Classifier:**

If  $f(x_1, x_2) < 0$  assign  
Class 1;

If  $f(x_1, x_2) > 0$  assign  
Class 2;

$$f(x_1, x_2) = w_1 x_1 + w_2 x_2 + b = 0$$



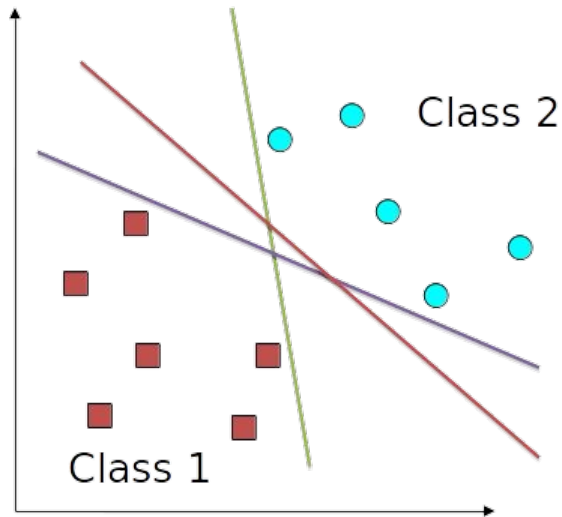
# Perceptron

- Perceptron is the name given to the linear classifier.
- If there exists a Perceptron that correctly classifies all training examples, then we say that the training set is linearly separable.
- In 1960s Rosenblatt gave an algorithm for Perceptron learning for linearly separable data.

# Perceptron



- For linearly separable data many classifiers are possible.



All being doing equally good on training set, which one is good on the unseen test set?



---

Let's get started with Neural  
Networks

*Get in Touch with me:*

@Github: <https://github.com/SriHarshaGajavalli>

@LinkedIn: <https://www.linkedin.com/in/sriharshagajavalli>

@Facebook: <https://facebook.com/harsha.gajavalli>

@Email: [sriharsha.g15@gmail.com](mailto:sriharsha.g15@gmail.com)