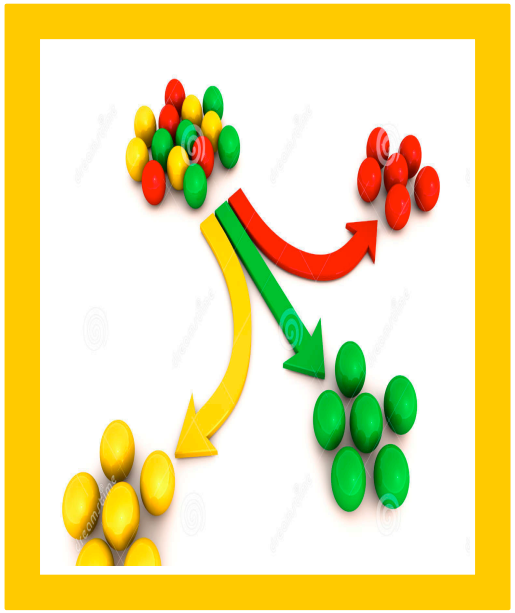


Closer look at Features and Classifiers

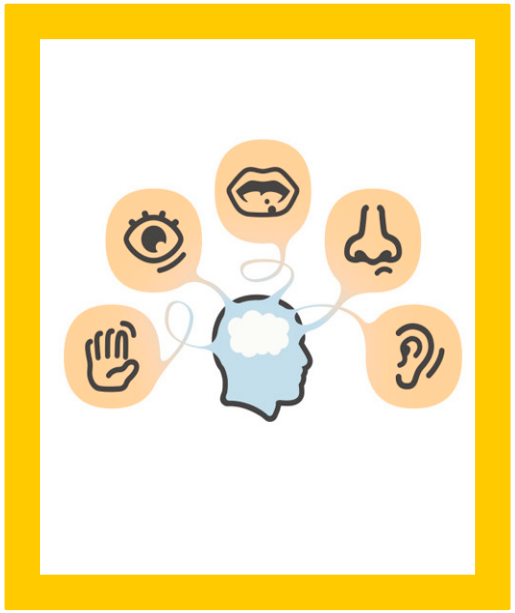
More on classifiers, classification and perception.

Plan

- Review and Lessons
- Case: Image Classification
  - Problem of Perception
- Four Possible Features
- Feature Normalization
  - Within Feature
  - Across Feature
- Design of a classifier
  - Peep into the Neural Nets
  - Gradient Descent for Linear
  - Perceptron Classifier
- Summary and Plans for Revision
- Spectrum of ML Problems (if time permits)

Review

- Data
  - Features, Transformations
  - Normalization
- Classifiers
  - KNN
  - Naïve Bayes
  - Linear
- Hyper parameters
  - K
  - Degree of the polynomial
- Training
  - Find parameters **w**
  - (Slowly getting into)
- Testing
  - Prediction
- Performance Metrics
  - Accuracy
  - Precision
  - Tradeoffs



AI: The Problem of Perception

Perception is a “human skill”. Can machines ever have?

Why is this hard?

194	210	201	212	199	213	215	195	178	158	182	209
180	189	190	221	209	205	191	167	147	115	129	163
114	126	140	188	176	165	152	140	170	106	78	88
87	103	115	154	143	142	149	153	173	101	57	57
102	112	106	131	122	138	152	147	128	84	58	66
94	95	79	104	105	124	129	113	107	87	69	67
68	71	69	98	89	92	98	95	89	88	76	67
41	56	68	99	63	45	60	82	58	76	75	65
20	43	69	75	56	41	51	73	55	70	63	44
50	50	57	69	75	75	73	74	53	68	59	37
72	59	53	66	84	92	84	74	57	72	63	42
67	61	58	65	75	78	76	73	59	75	69	50

Why is it challenging?



Occlusions/Truncations

View Point Variation









Intra class variations

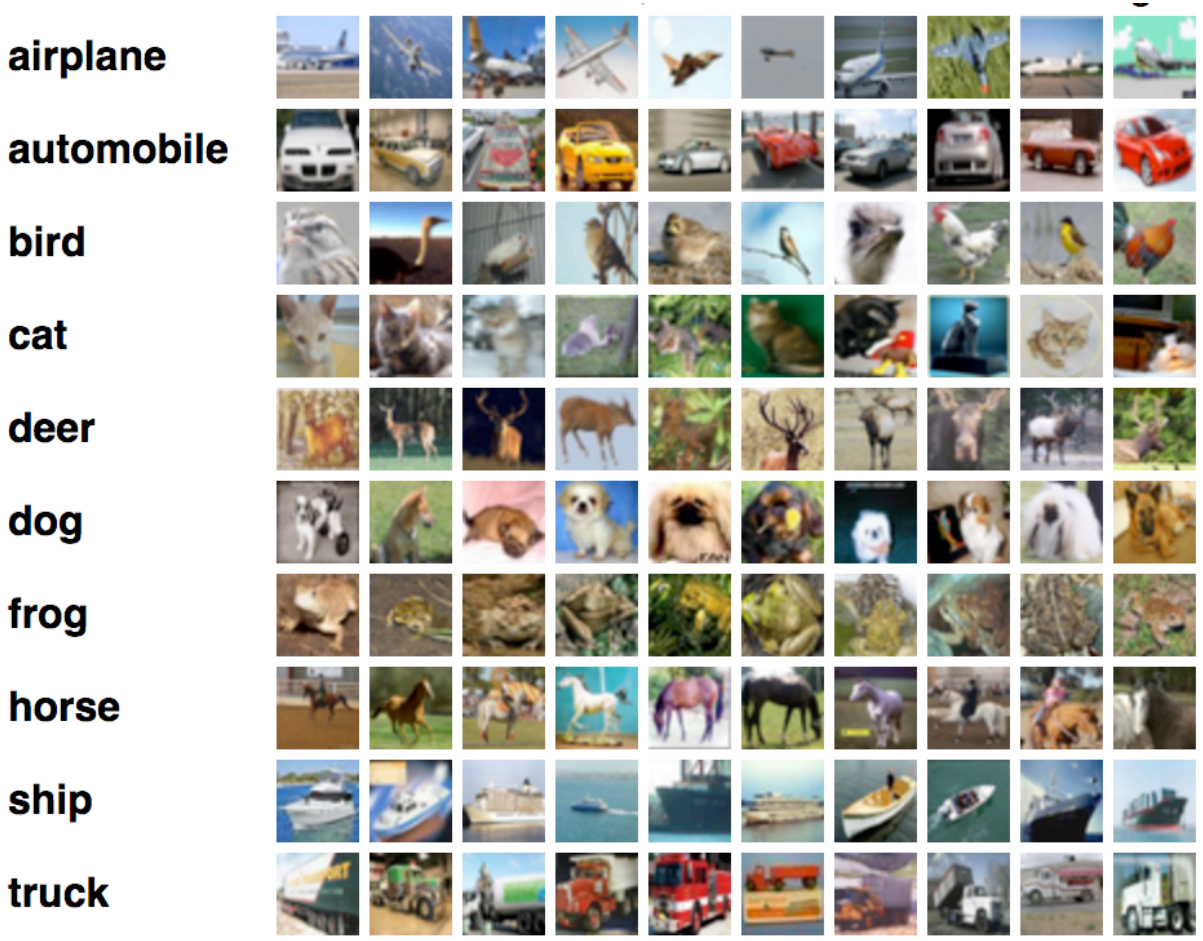





Inter class variations

Our Problem: CFAR-10

- 10 classes. 50K Train. 10K Test.



- Our "smaller" problem
  - Automobile Vs bird
  - Separate/recognize two classes
  - 10K Train 2K Test Samples

Naïve Attempt

- If (image has green/grass)
  - It is an animal
    - if (…)
- If (image has blue)
  - It is either airplane or bird
    - If (…)
- And so on ..

- Any hope of this working? ☺

Questions?

Possible Features: - 1 Handcrafting

MIN RED
MAX RED
MEAN RED
MIN GREEN
MAX GREEN
MEAN GREEN
MIN BLUE
MAX BLUE
MEAN BLUE

9 X 1 FEATURE VECTOR PER IMAGE

Concerns:

- **Too naïve to capture the visual content?**
- **Too small to represent information?**

Possible Features: - 2 Raw Data itself

FEATURE VECTOR
32 X 32 X 3 = 3072 DIMENSION PER IMAGE (d = 3072)

CONCERNS:

- Too big ?
- Lots of redundancy ?

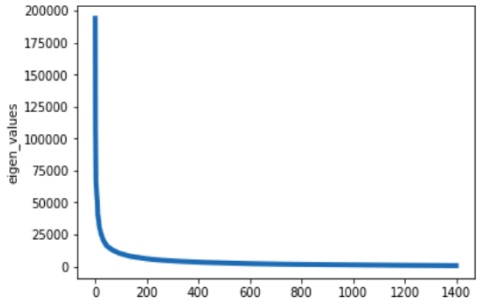
Possible Features: - 3 (PCA based)

K X I

=

M is a K X d matrix

d X I



Plot of Eigen values in decreasing order.

Usually only "K" (a very small are useful). Most of them are near zero or even zero.

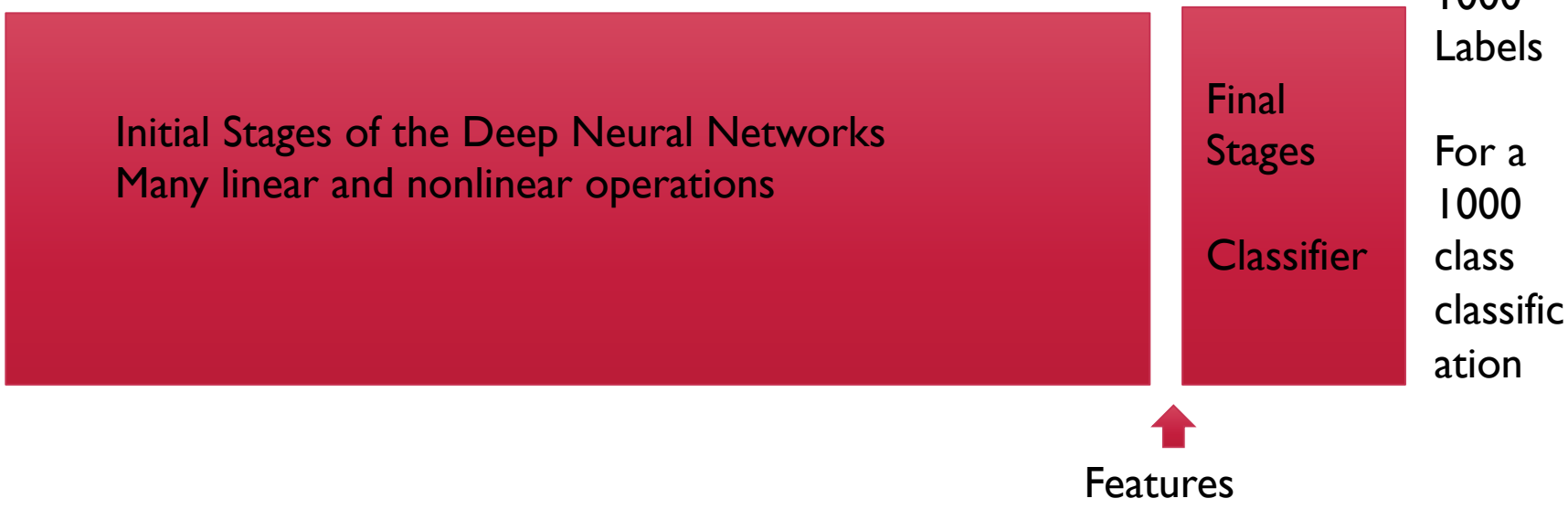
Let $\mathbf{x}_i$ be an image represented as a column vector.

Let $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, ---, \mathbf{x}_N]$ be zero mean a d X N matrix.
(Here d = 3072, N = 10K)
Let A = $\mathbf{XX}^T$ be an d X d Matrix. It also has d Eigen vectors each of d dimension.

Rows of matrix "M" are the selected K Eigen vectors of the matrix A.

Possible Features: – 4 (Deep Learned)

Deep Learning = End to End Learning (Raw data to labels)
Deep Learning = Feature Learning!!



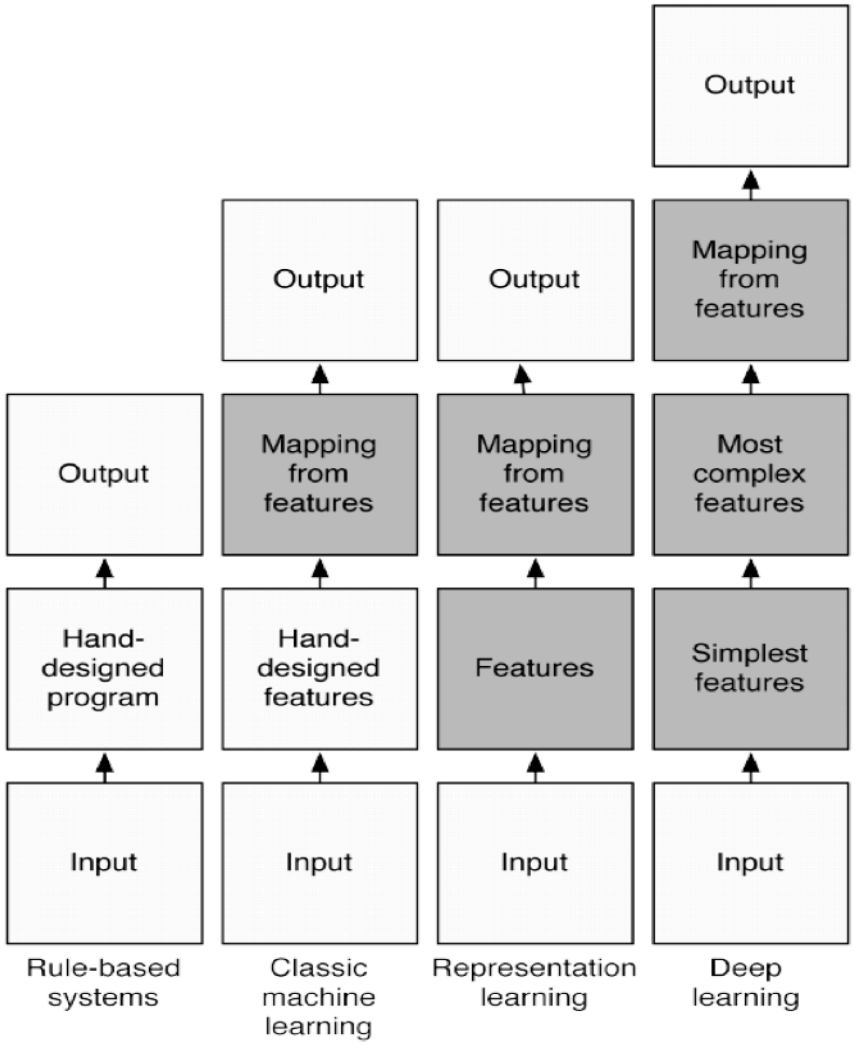
An intermediate representation from a popular "DeepNet", which was Designed and trained for solving a "general" 1000 class classification.

For our task (LAB3), we use a net with dimension = 512. Real numbers

Nets surfaces in 2012
The idea of pre-trained features popularized in 2015.

The best known class of features

What is Deep Learning?



Y. Bengio et al, ``Deep Learning", MIT Press, 2015

Questions?

Empirical Results (CFAR- 2 classes)

FEATURE	Dimension	ACCURACY
NAÏVE (Colour)	9	71.6
RAW DATA	3072	79.2
PCA (K=800)	800	82.4
DEEP FEATURE	512	99.5

Value of k	Accuracy
500	80.0%
800	82.4%
1100	82.6%
1400	81.3%
1700	81.65%
2000	82.8%
2300	83.2%
2600	83.2%
2900	83.2%

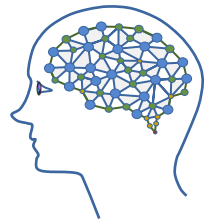
Extending to Multi-Class (10 classes)

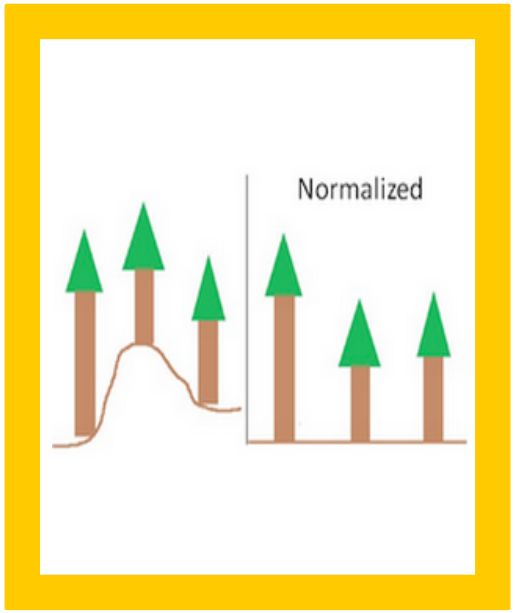
- Train ${}^{10}C_2 = 45$ linear classifiers
- Each of these 45 classifiers predict a label for the test sample.
- Find the majority label
- Assign/predict the label as majority-label.

CFAR-10 Full Results

Summary (Questions?)

- Handcrafting features/representation is not the most effective.
- Possible directions to create features:
  - Handcrafting (intuition driven)
  - Raw data
  - PCA (Statistics)
  - Learned Features (Deep Learning)





Normalizing Features

Normalizing within features

- Feature-wise Normalization

$$x_i' = \frac{x_i - \min(x_i)}{\max(x_i) - \min(x_i)}$$

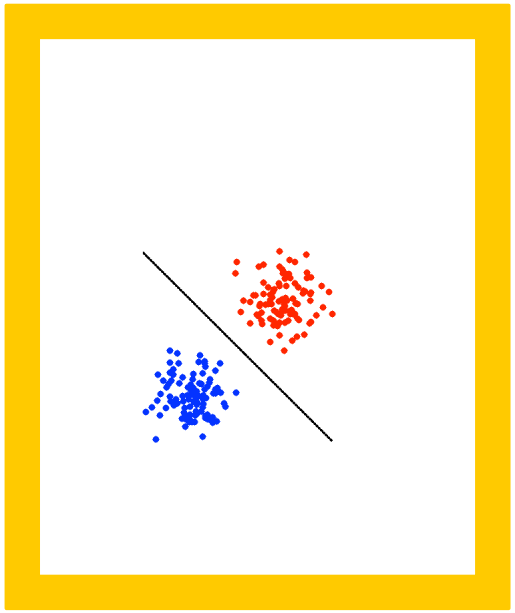
$$x_i' = \frac{x_i - \text{mean}(x_i)}{\textit{standarddeviation}(x_i)}$$

Normalizing across features

- Let **x** be a feature vector of dimension d.
- Normalize x such that
  - L1 Norm $||x||_1 = 1$ i.e., $\|\boldsymbol{x}\|_1 := \sum_{i=1}^{n} |x_i| .$
  - L2 Norm $||x||_2 = 1$ i.e., $|\mathbf{x}| = \sqrt{x_1^2 + x_2^2 + x_3^2} .$

Summary (Questions?)

- Hard to say what is the right normalization.
- Reasonable steps to do:
  - Feature level normalization (across samples)
  - Feature vector level normalization (within sample)
- If you had done feature level normalization (Kg Vs gram in Lab1), there would not have been any affect.
  - It only guarantees no change in results.



Training Linear Classifier

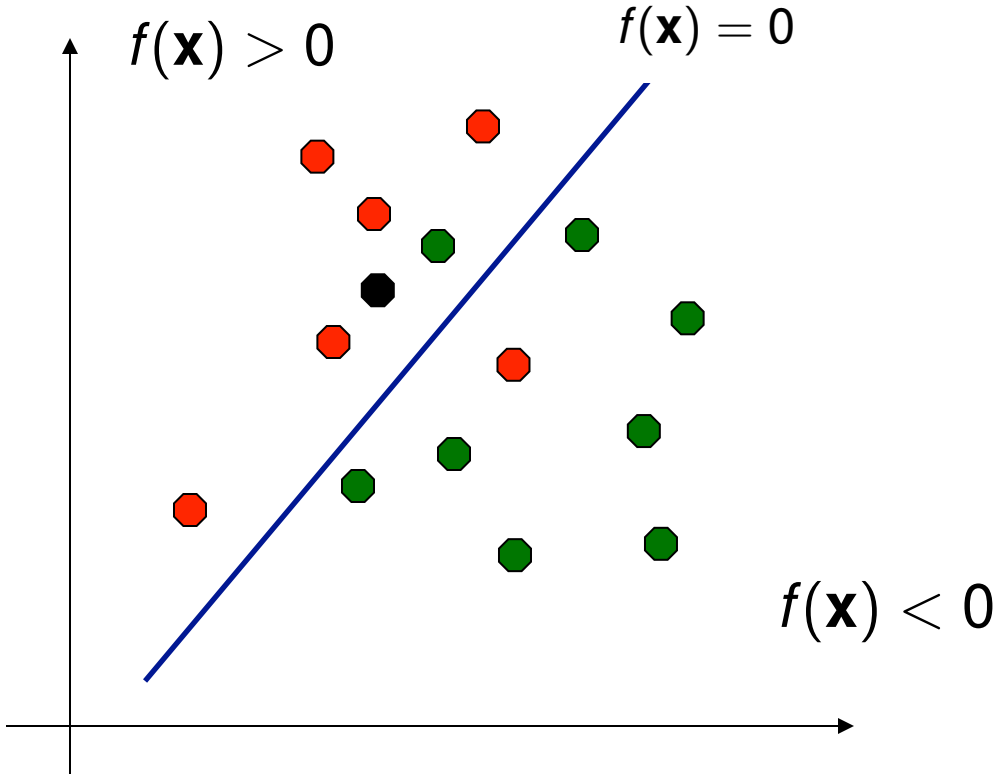
How do we now find the equarion of the line from the data?

Review: Linear Classification

$$f(\mathbf{x}) = w_1 x_1 + w_2 x_2 + w_3 x_3 + \ldots + w_d x_d$$

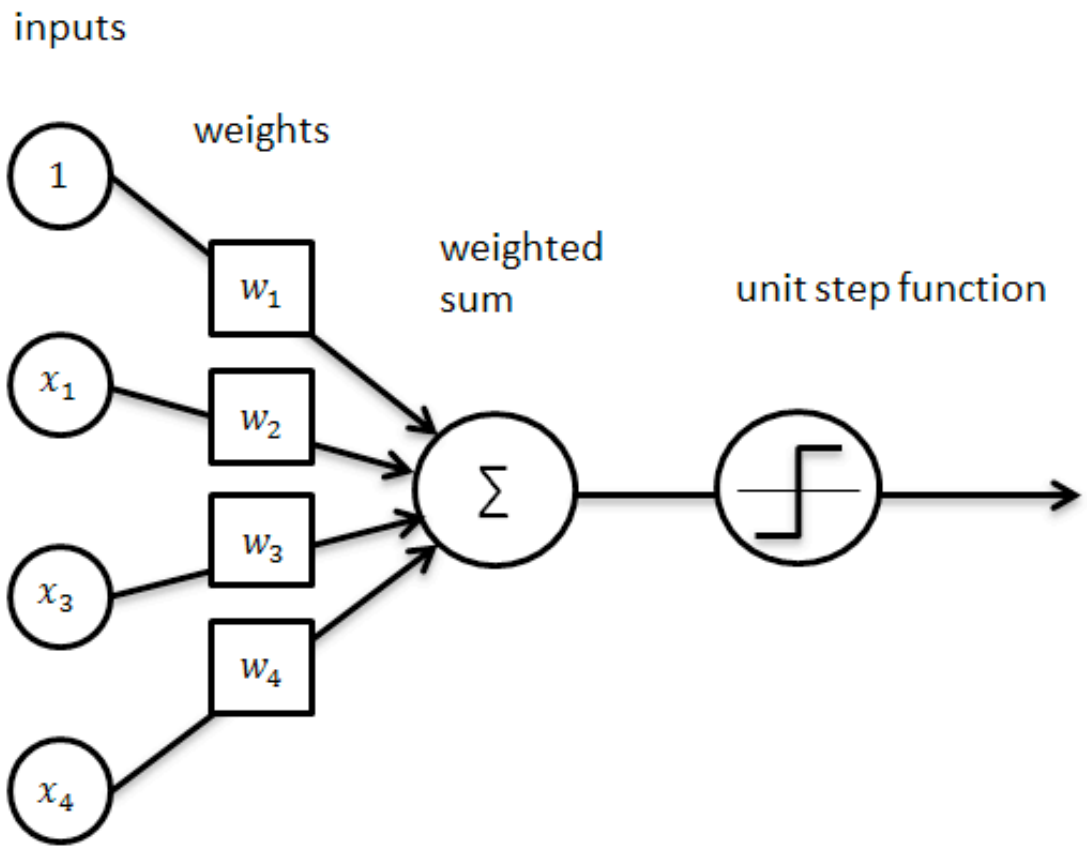
Feature Vector $\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_d \end{bmatrix}$

Parameters to be learned $\mathbf{w} = \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_d \end{bmatrix}$



Compactly: $f(\mathbf{x}) = \mathbf{w}^T \mathbf{x}$ or $\mathbf{w} \cdot \mathbf{x}$

A “Neural Network” Perspective



Predicted Label:

Sign ($\mathbf{w}^T\mathbf{x}$)

Formulating the Training Problem

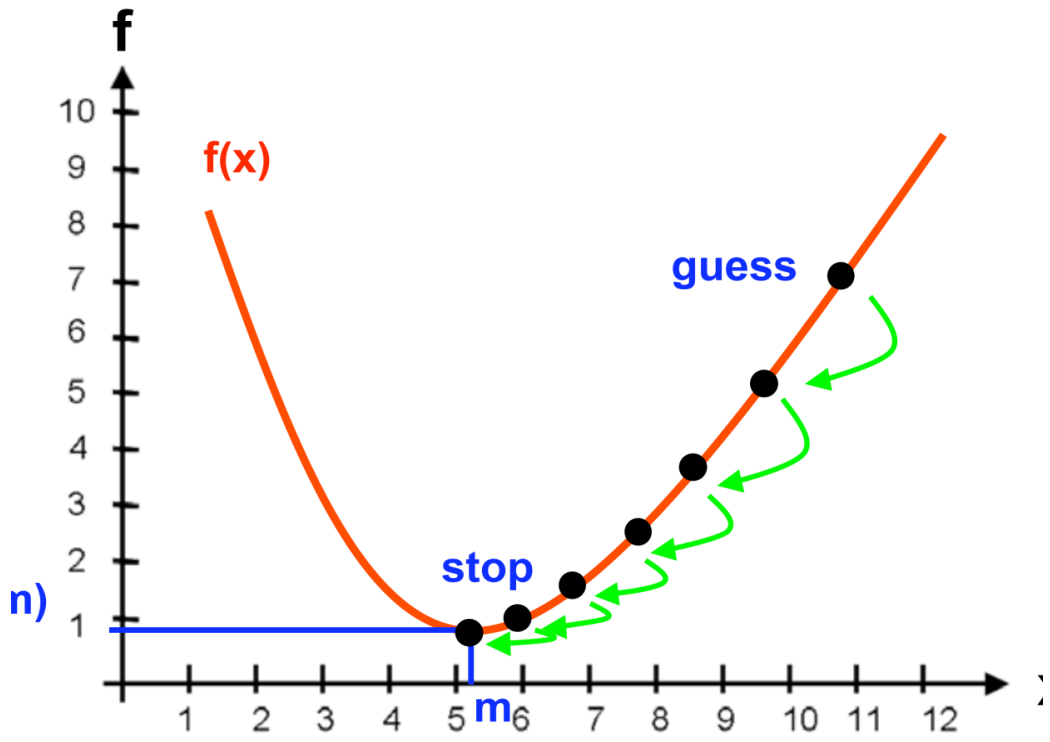
$$\text{Best } \mathbf{w} = \min_{\mathbf{w}} \mathcal{L}(Y, \hat{Y}) = \min_{\mathbf{w}} \mathcal{L}(f_w(X), Y)$$

$\mathcal{L}$ is called the Loss Function or Objective Function.

An Example Loss Function:

$\mathcal{L}(Y, \hat{Y})$ = Number of Misclassified Samples

Gradient Descent: Intuition



Problem: Minimize f (x).

1. Start with an initial guess.
2. Update your guess by moving along the negative slope (gradient).

$$\mathbf{w_{n+1}} = \mathbf{w_n} + \eta \Delta \mathbf{w}$$

Notes:

1. We do not have an analytic form of f(x) to differentiate and find the optima/best/minima.
2. We can differentiate "locally" with the help of

Error Minimization

Let t_i be the target and o_i is our prediction of our network.

$$\mathcal{L} = \sum_i (t_i - o_i)^2 = \sum_i (t_i - w^T x_i)^2$$

$$\Delta w = -\frac{\partial \mathcal{L}}{\partial w} = 2 \sum_i (t_i - o_i) x_i$$

$$w_{n+1} = w_n + \eta \sum_i (t_i - o_i) x_i$$

or sample wise $w_{n+1} = w_n + \eta (t_i - o_i) x_i$

Two questions

- When to stop?
  - When the gradient goes close to zero.
  - When no change in object function.

- How to choose the step size?
  - Start with large step size and reduce closer to convergence.

Perceptron Algorithm

- Initialize weight "randomly"
- For each training sample $\mathbf{x_i}$
  - Calculate prediction o_i
  - (Let true prediction be t_i)
  - Update weight **w**

$$\Delta w_i = (t_i - o_i)x_i$$

$$\mathbf{w_{n+1}} = \mathbf{w_n} + \eta \Delta \mathbf{w}$$

- Repeat

- Note:
  - If correctly classified,
    - $(t_i - o_i)$ will be zero.
- Act only for error samples.
- Will not converge for every data set (linear non separable).
- Note: This is not exactly what we derived few slides early. (shall discuss later.)

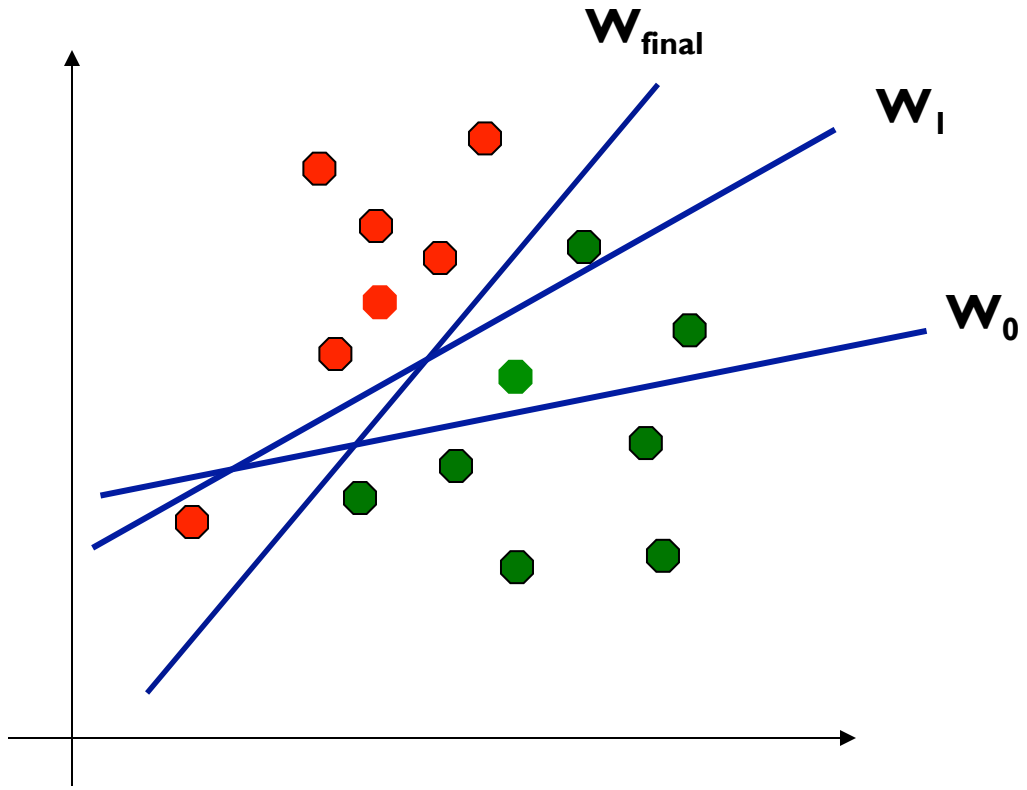
Linear Classification

We Start with $\mathbf{W_0}$

Three green and one red are misclassified.
All of them "pull" the line in the right direction.

$\mathbf{W_1}$ is the new line. This has one error. The error sample pull the line further in the right direction.

$\mathbf{W_{final}}$ is the final vector.



Summary

- Wide range of feature extraction schemes.
- Impressive results on image classification (CFAR).
- Introduction to Neural Networks and Gradient Descent
- Training problem in ML is often casted as an “optimization” problem.

Plans for the Revision Week

- We will have lectures and labs as scheduled
- Lectures: Focus
  - First Half: Clear doubts, revision
  - Second Half: Fill missing concepts/details.
- Labs: Tasks
  - Finish any pending tasks. Answer assessments and score.
  - (this is the last chance!!)
  - May be one or two new questions (on old content).

Space of ML Problems

- When Y is an an integer
  - **Classification**
- When Y is a real quantity
  - **Regression**
- When Y is a String, Graph, Set, Tree etc.
  - **Structured Prediction**

Thanks!!. Questions?