Stats5 Seminar: Machine Learning

Winter 2018

Professor Padhraic Smyth
Departments of Computer Science and Statistics
University of California, Irvine

Class Organization

- Meet weekly for 40 minute seminar with 5-10 minute discussion
- 8 topics (with guest speakers), weeks 2 through 9
  - You are encouraged to ask questions during and after the talks
- Intro and wrap-up talks in weeks 1 and 10
- Class Web site is at www.ics.uci.edu/~smyth/courses/stats5
  - Slides and related materials will be posted during the quarter

Schedule of Lectures

Date	Speaker	Department Or Organization	Topic
Jan 9	Padhraic Smyth	Computer Science	Introduction to Data Science
Jan 16	Padhraic Smyth	Computer Science	Classification Algorithms in Machine Learning
Jan 23	Michael Carey	Computer Science	Databases and Data Management
Jan 30	Sameer Singh	Computer Science	Statistical Natural Language Processing
Feb 6	Zhaoxia Yu	Statistics	An Introduction to Cluster Analysis
Feb 13	Erik Sudderth	Computer Science	Computer Vision and Machine Learning
Feb 20	John Brock	Cylance, Inc	Data Science and CyberSecurity
Feb 27	Video Lecture (Kate Crawford)	Microsoft Research and NYU	Bias in Machine Learning
Mar 6	Matt Harding	Economics	Data Science in Economics and Finance
Mar 13	Padhraic Smyth	Computer Science	Review: Past and Future of Data Science

Submission of Review Forms (Weeks 2 to 10)

- Submit Review forms for Lectures 2 through 10
  - Available at http://www.ics.uci.edu/~smyth/courses/stats5/Forms/

- Review forms will be available online at the start of each class
  - A few relatively short questions based on the lecture that day
  - Needs to be submitted to EEE by 12:15 for each lecture
  - Bring your laptop or other device

- Requirements to pass the class
  - Attend and submit review form for least 8 lectures for weeks 2 through 10

    (allowed to miss one if you need to for some reason)

- No final exam: pass/fail based on attendance and review forms

Outline of Today's Topic

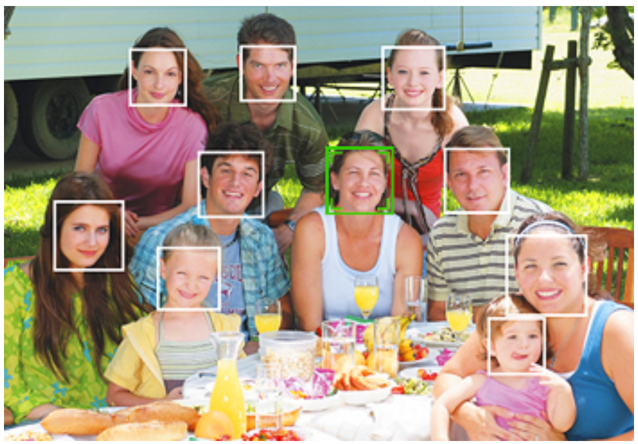
- What is machine learning?
- Classification algorithms
- Examples from image and sequence classification
- Conclusions and discussion

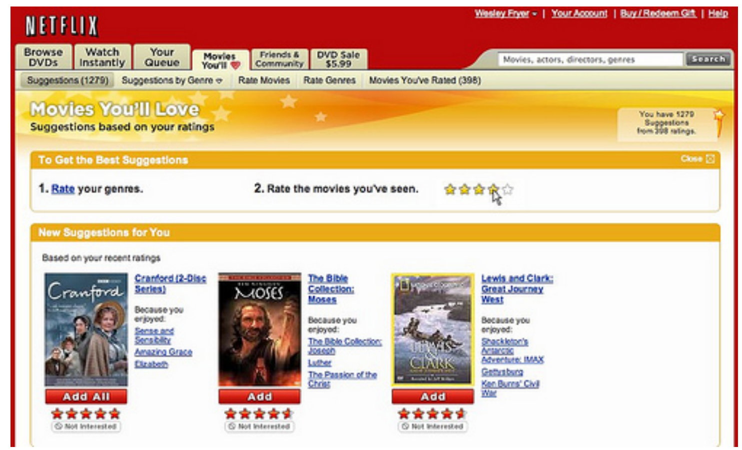
[Acknowledgement to Professor Alex Ihler for various slides and figures in this lecture]

What is Machine Learning?

Machine learning (ML)

- Learning models from data
- Making predictions (or decisions)
- Getting better with experience (data)
- Problems whose solutions are “hard to describe”

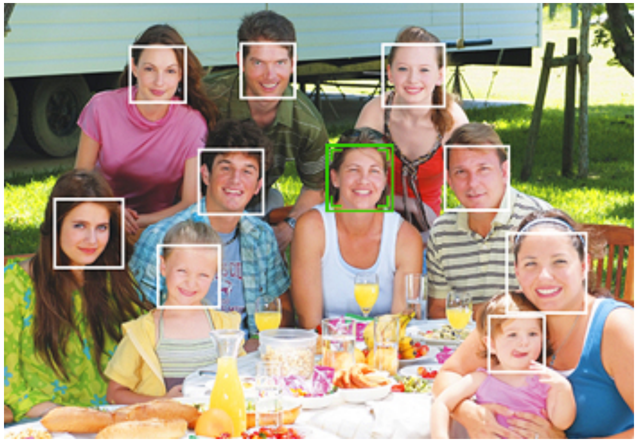


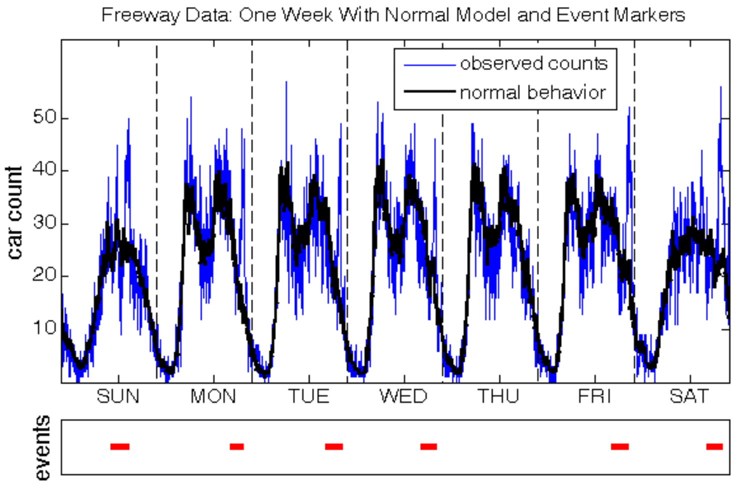


Types of machine learning problems

- Supervised learning
  - "Labeled" training data
  - Every example has a desired target value (a "known answer")
  - Reward predictions close to target; penalize predictions with large errors

  - Classification: a discrete-valued prediction
  - Regression: a continuous-valued prediction





amazon alexa prize

Home | About | FAQs | Contest Rules | 2017 Results | 2017 Proceedings

The Alexa Prize

Over $3.5 Million to Advance Conversational Artificial Intelligence

December 2017 - November 2018

The application period for the 2018 Alexa Prize is now closed. Participants will be announced on February 1, 2018 and the competition will officially begin.

2018 Alexa Prize

The way humans interact with machines is at an inflection point and conversational artificial intelligence (AI) is at the center of the transformation. Alexa, the voice service that powers Amazon Echo, enables customers to interact with the world around them in a more intuitive way using only their voice.

Types of machine learning problems

- **Supervised learning**
  - “Labeled” training data
  - Every example has a desired target value (a “best answer”)
  - Reward prediction being close to target

  - Classification: a discrete-valued prediction
  - Regression: a continuous-valued prediction

  - Recommender systems

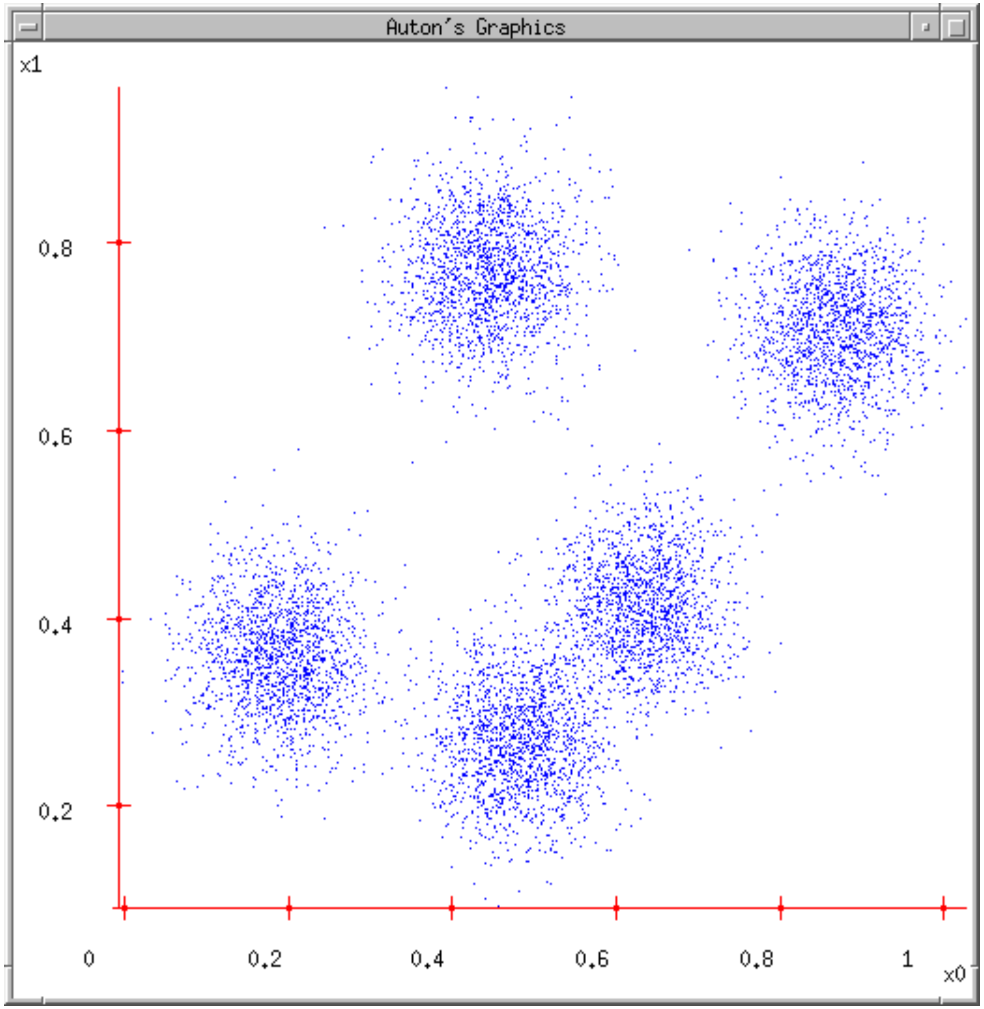
users

movies	1	2	3	4	5	6	7	8	9	10	11	12
1	1		3		?	5			5		4	
2			5	4			4			2	1	3
3	2	4		1	2		3		4	3	5	
4		2	4		5			4			2	
5			4	3	4	2					2	5
6	1		3		3			2			4	

Types of machine learning problems

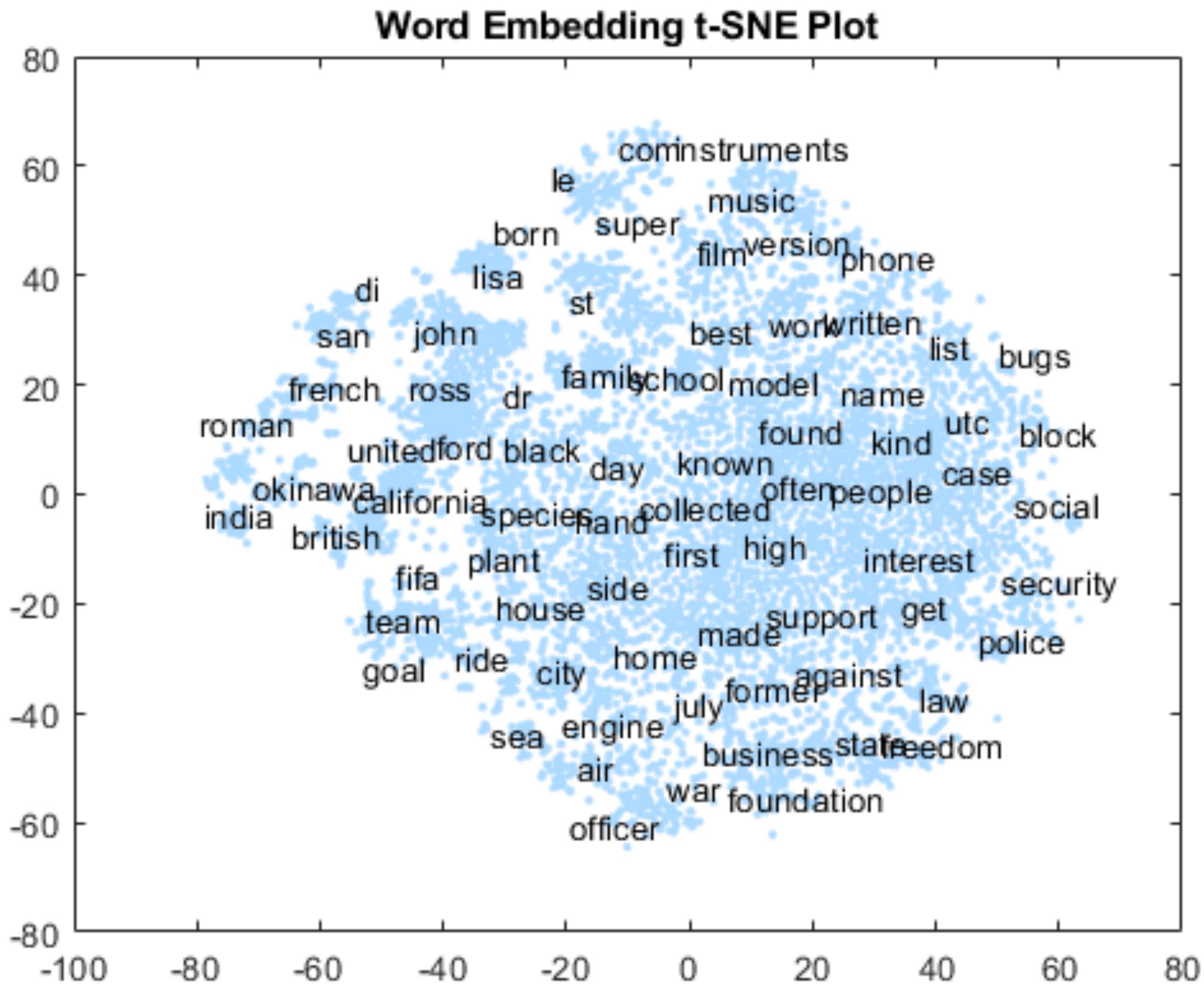
- **Supervised learning**
  - Training data has labels or target values
- **Unsupervised learning**
  - Training data has no labels or target values
  - Interested in discovering natural structure in data
  - Often used in exploration of data, e.g., in science, in business
  - Example:
    - Clustering customers or medical patients into groups
    - Discovering a numerical representation of words or movies

Data in 2 Dimensions with 5 Clusters

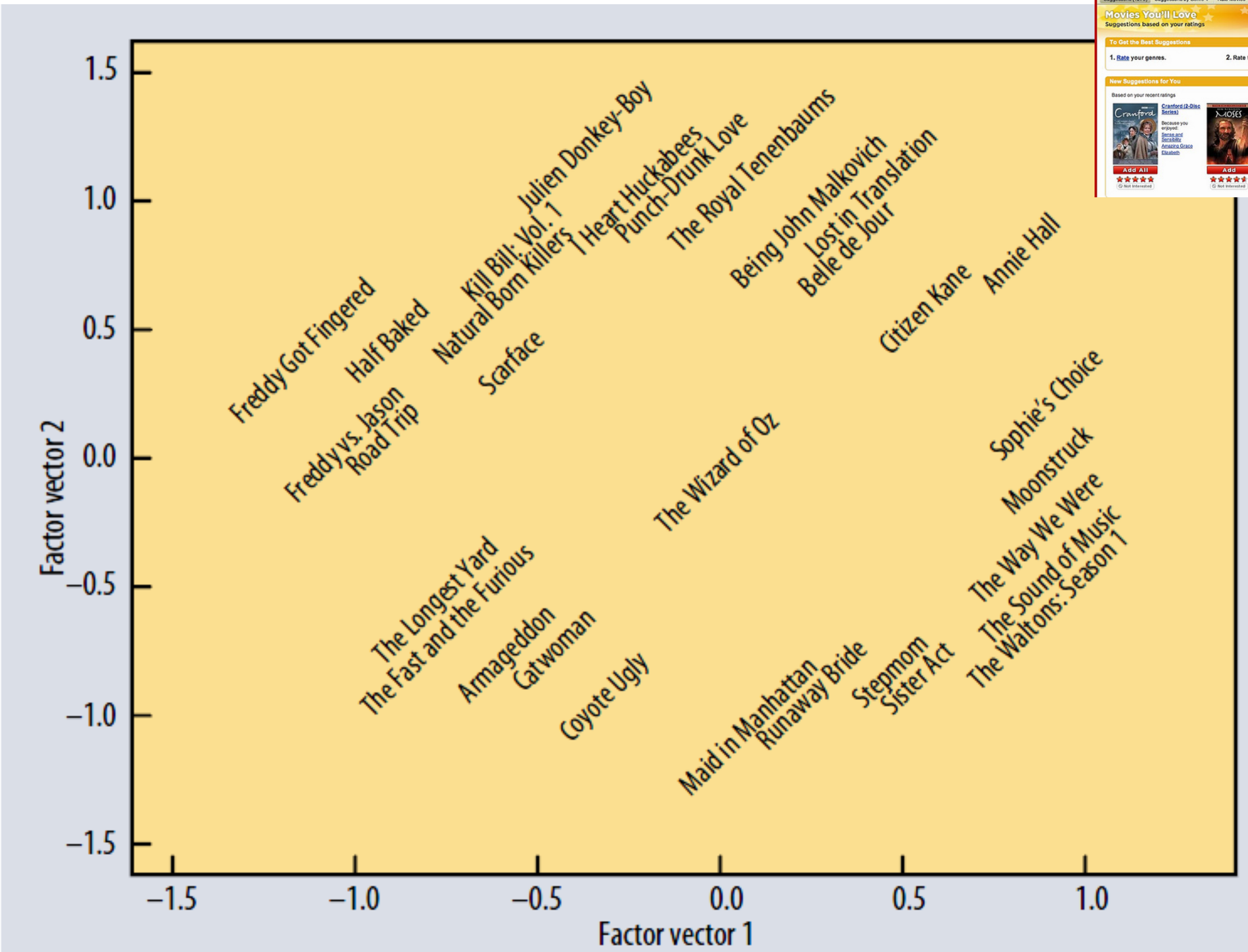


See Lecture by Prof Zhaoxia Yu later this quarter on Clustering Algorithms

Embeddings of Words as Vectors



From: https://www.mathworks.com/help/examples/textanalytics/



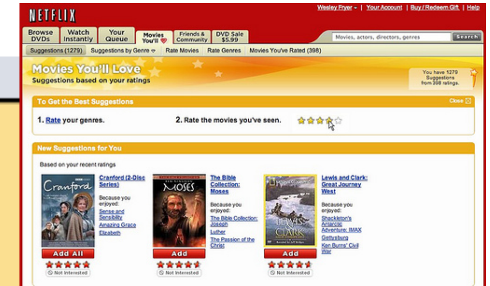
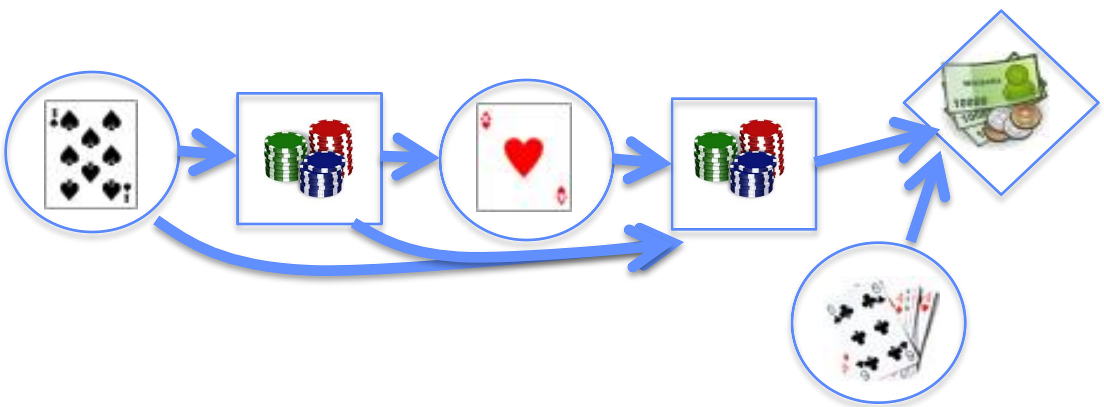


Figure from Koren, Bell, Volinksy, IEEE Computer, 2009

Types of machine learning problems

- **Supervised learning**
- **Unsupervised learning**

- **Reinforcement learning**
  - Algorithm gets indirect feedback on its progress (rather than correct/incorrect)
  - E.g., a program learning to play chess, or Go, or a video game
  - E.g., an autonomous vehicle learning how to navigate a city
  - Mathematical models for delayed reward, credit assignment, explore/exploit



Daily Report: AlphaGo Shows How Far Artificial Intelligence Has Come

Bits
By PUI-WING TAM   MAY 23, 2017

Classification using Supervised Learning

Learning a Classification Model

Training Data

Patient ID	Zipcode	Age	....	Test Score	Diagnosis
18261	92697	55		83	1
42356	92697	19		99	1
00219	90001	35		21	0
83726	24351	0		35	0

Learning algorithm learns a function that takes values on the left to predict the value (diagnosis) on the right

Making Predictions with a Classification Model

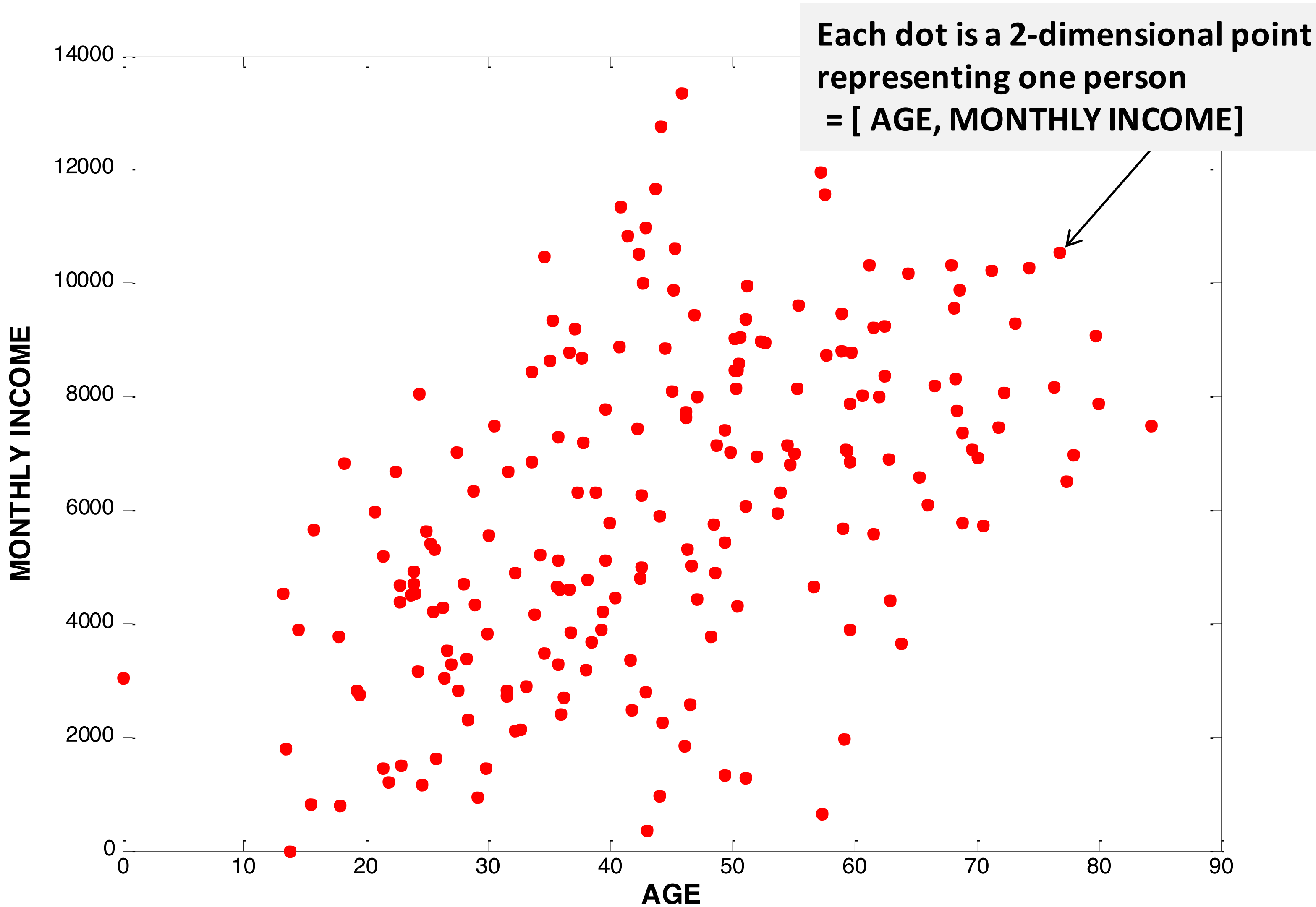
Training Data

Patient ID	Zipcode	Age	....	Test Score	Diagnosis
18261	92697	55		83	1
42356	92697	19		99	1
00219	90001	35		21	0
83726	24351	0		35	0

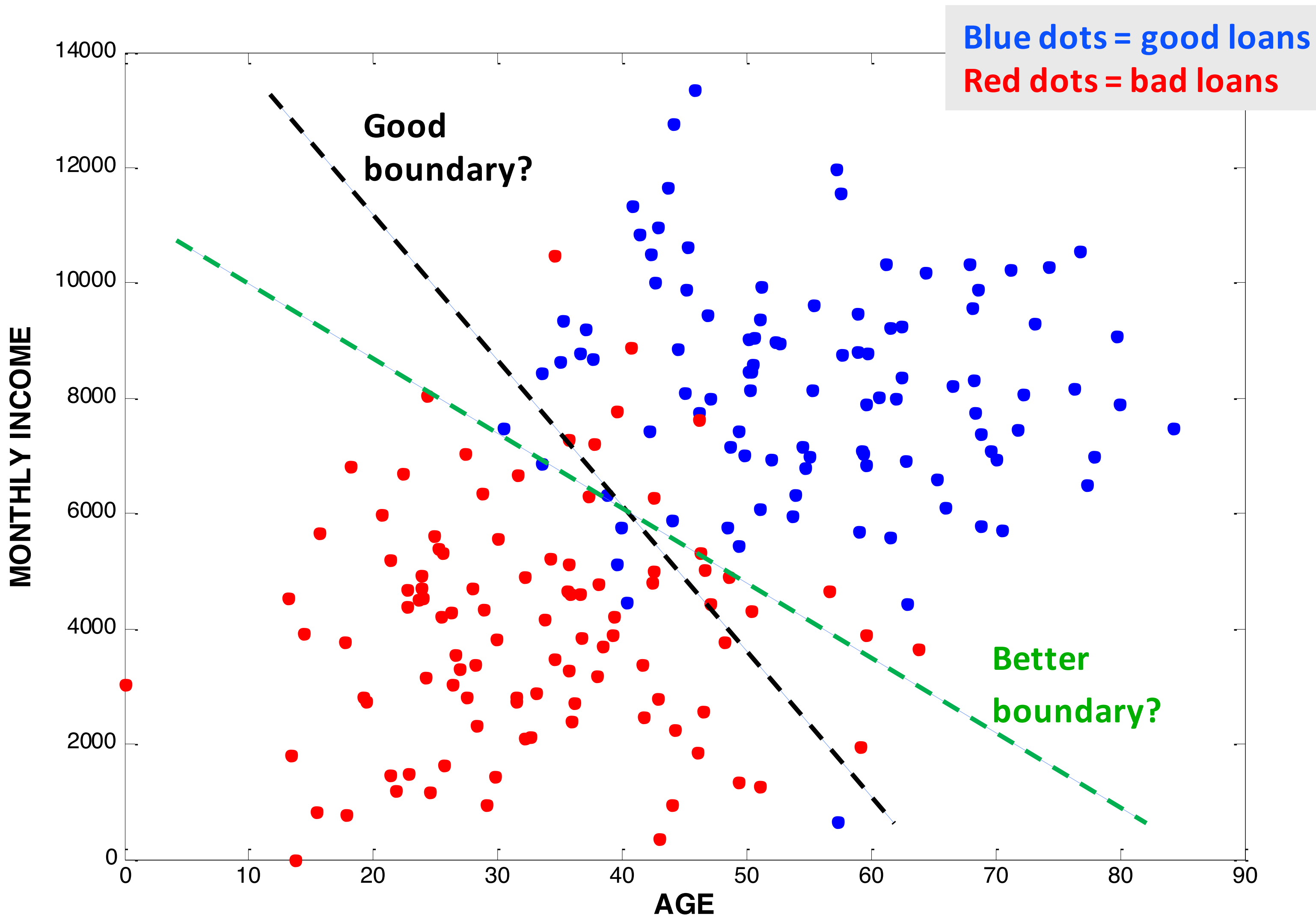
Test Data

12837	92697	40		70	??
72623	92697	32		44	??

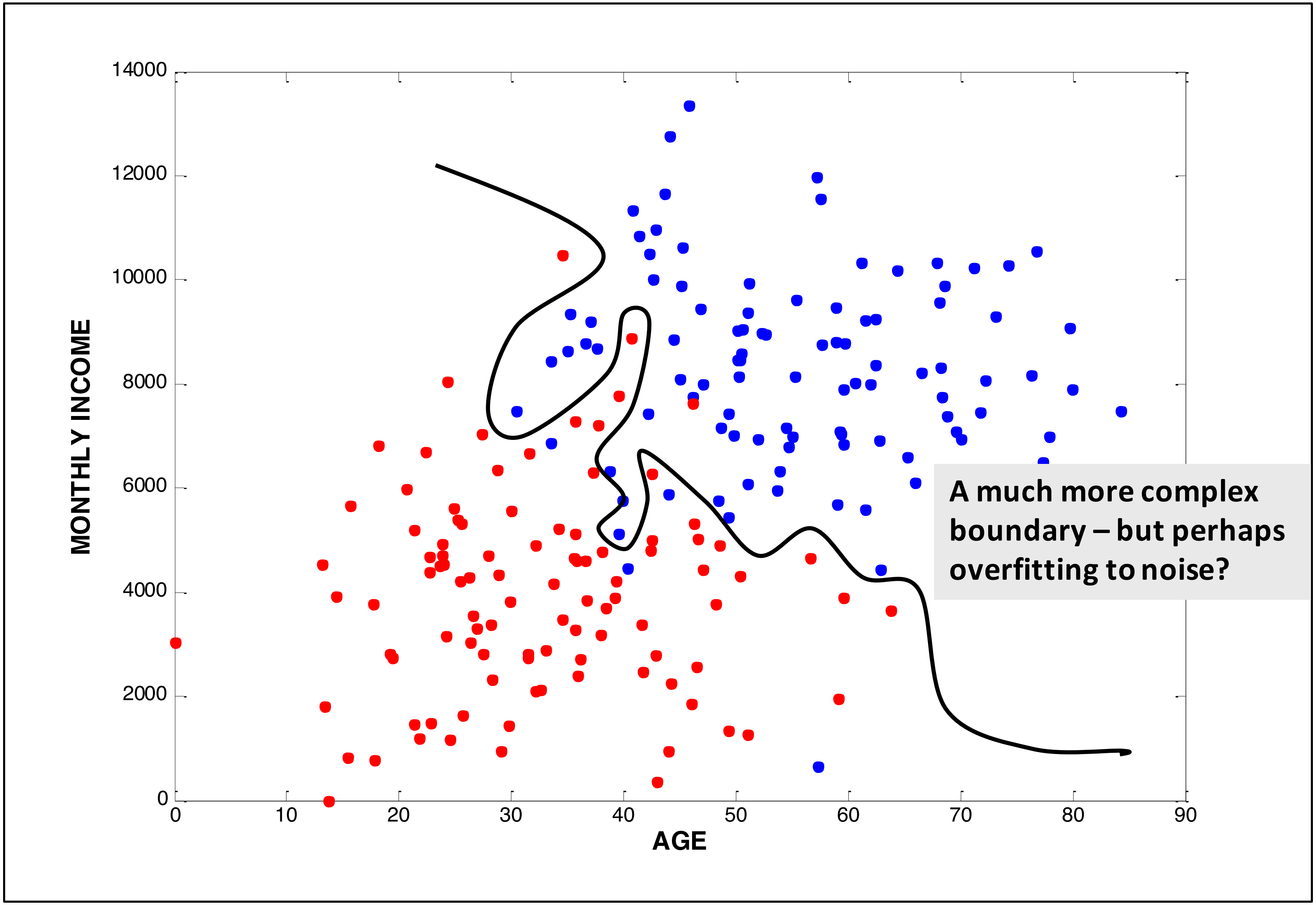
We can then use the model to make predictions when target values are unknown



UCIrvine
University of California, Irvine



UCIrvine
University of California, Irvine



UCIrvine
University of California, Irvine

Basic Concepts

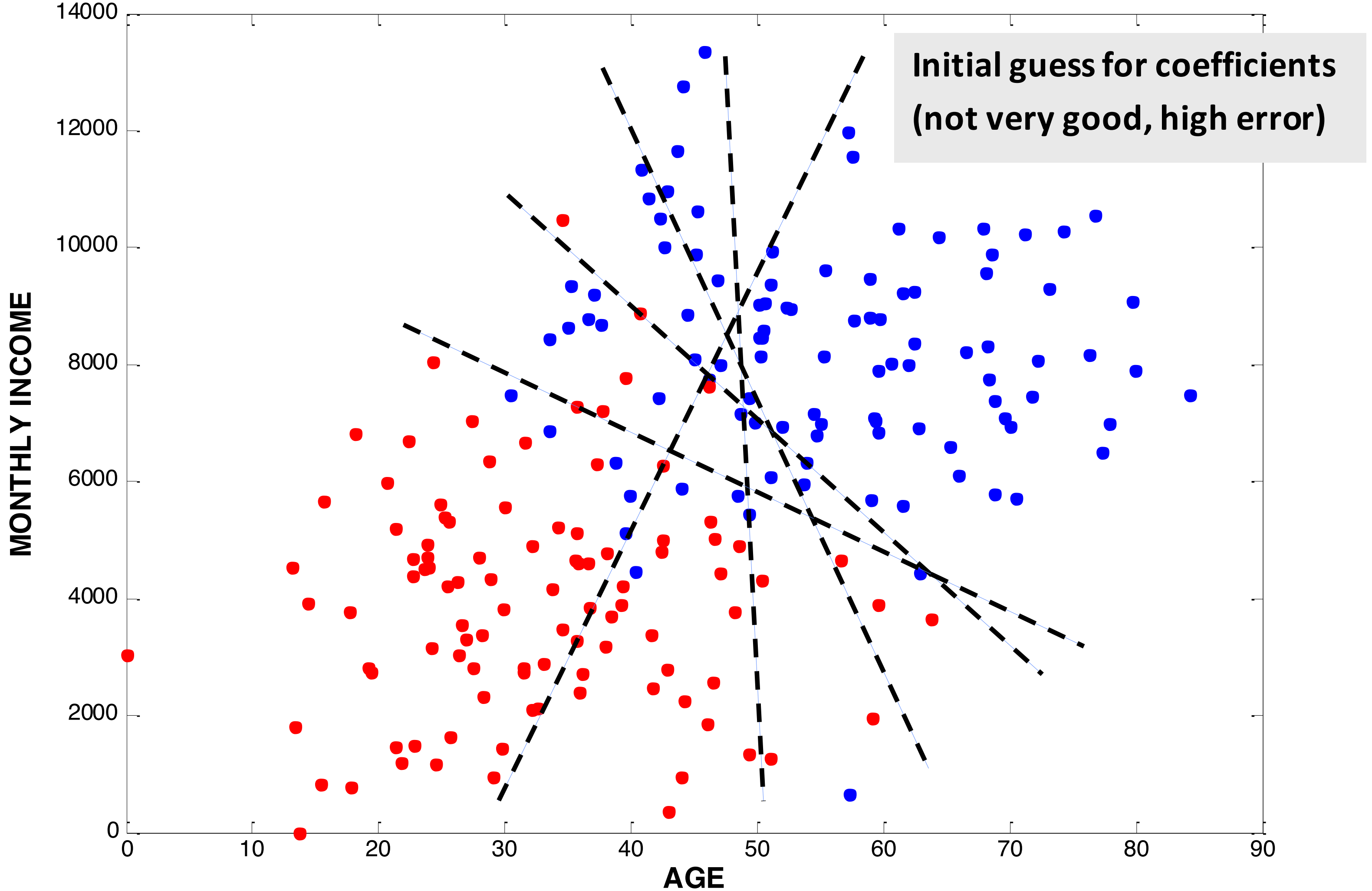
- **The curve represents a classifier (a model, a predictor)**
  - Points on one side of the line get classified as one class
  - Points on the other side get classified as the other class
  - Once we know the curve we can take new points and classify them

- **The curve is represented internally by a set of coefficients**
  - These are also known as "parameters" or "weights"

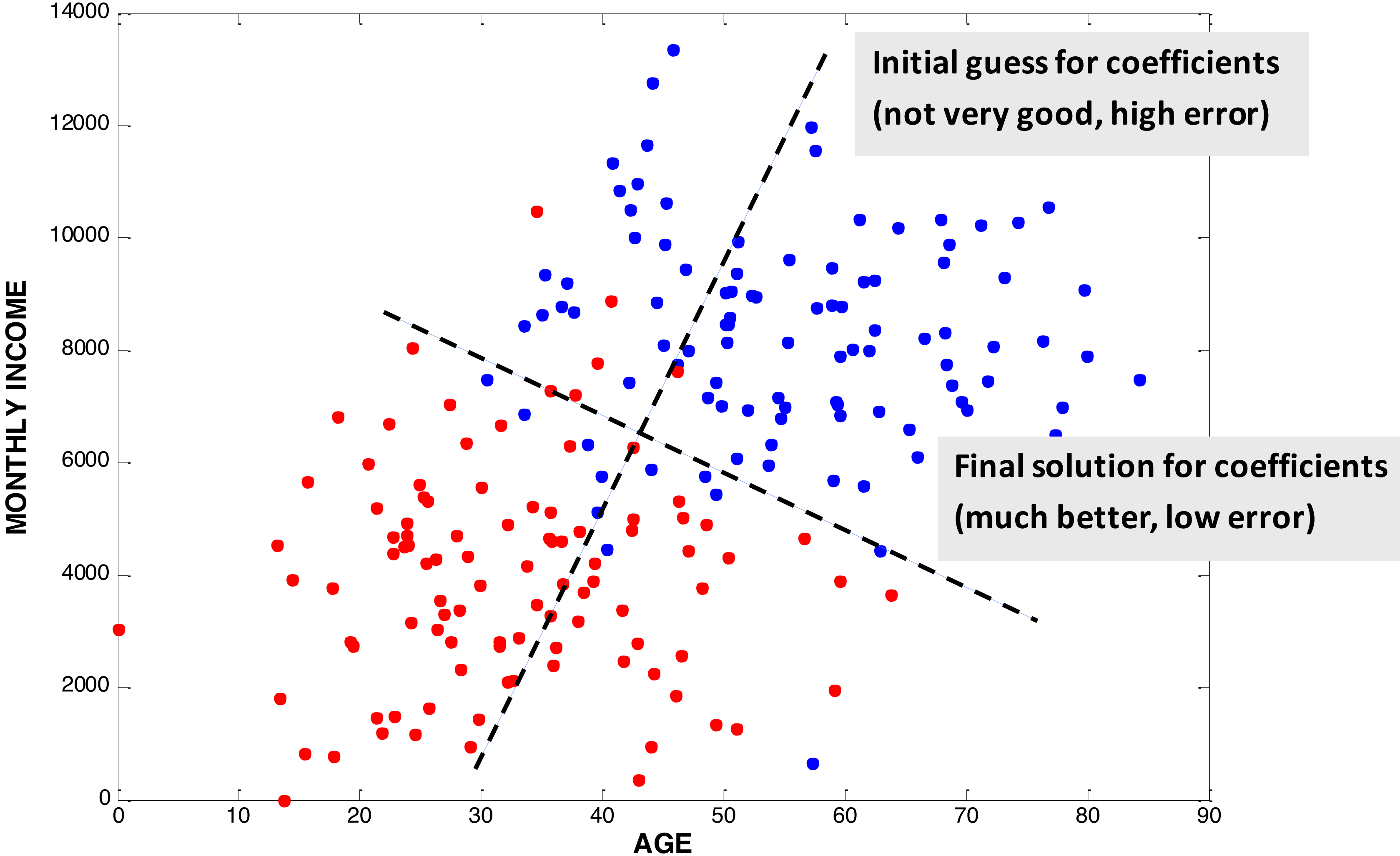
- **The algorithm systematically adjusts the coefficients on training data to reduce the error as much as it can**

- **This process of finding the weights is known as "learning a model"**

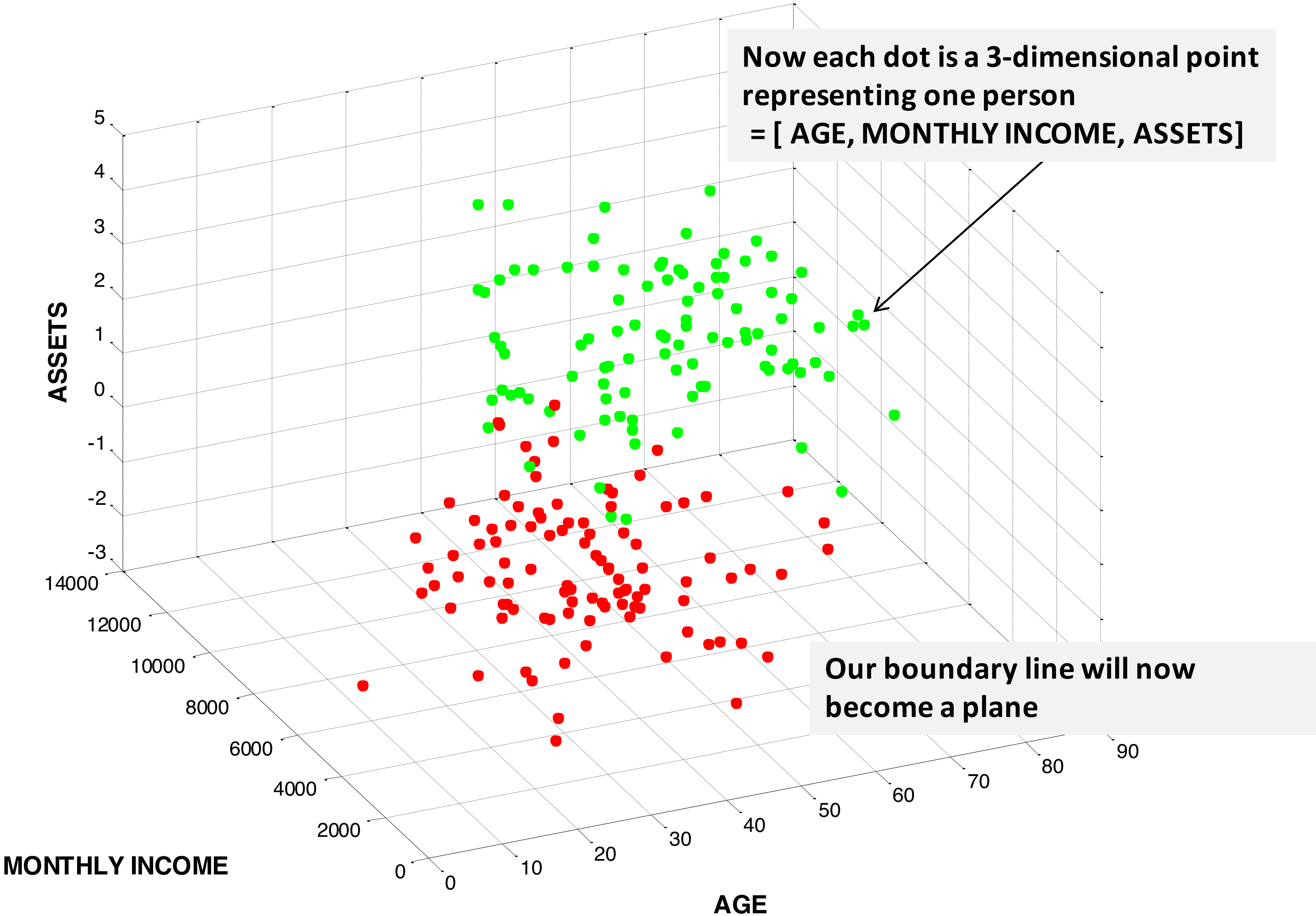
- **Foundational ideas are from statistics and optimization**



Initial guess for coefficients
(not very good, high error)
MONTHLY INCOME
AGE
14000
12000
10000
8000
6000
4000
2000
0
0
10
20
30
40
50
60
70
80
90



UCIrvine
UNIVERSITY OF CALIFORNIA, IRVINE


Now each dot is a 3-dimensional point representing one person
= [AGE, MONTHLY INCOME, ASSETS]
Our boundary line will now become a plane
ASSETS
5
4
3
2
1
0
-1
-2
-3
MONTHLY INCOME
14000
12000
10000
8000
6000
4000
2000
0
AGE
0
10
20
30
40
50
60
70
80
90

How Does this Work in Practice?

- **We use computer algorithms to search for the best line or curve**

- **These search algorithms are quite simple**
  1. Start with an initial random guess for coefficients
  2. Change the coefficients slightly to reduce the error
     (can use calculus to do this)
  3. Move to the new coefficients
  4. Keep repeating until "convergence"

- **This search can be done 10, 100, 1000, or 1 million "dimensions"**
  …. with 10's of millions of examples

- **This search process is at the core of machine learning algorithms**

Key Points

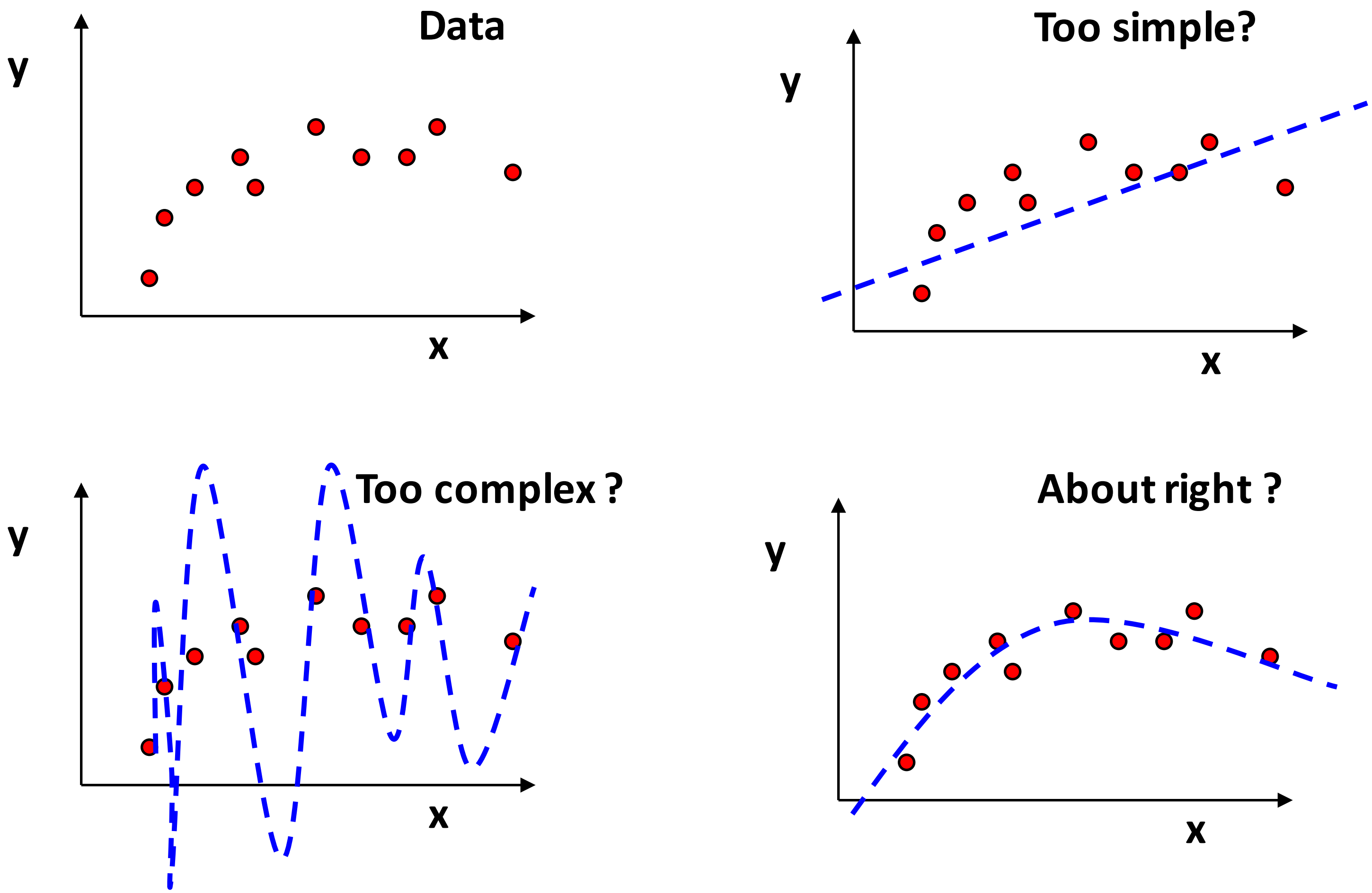
- We represent our training data as points in a multi-dimensional space
  - How do we obtain the labels for the data points?

- We want to find a boundary curve that can separate points into two classes

- The curves are represented by sets of coefficients (or weights)

- Machine learning algorithms use search (or optimization) to automatically find the coefficients with the lowest error on the training data

If the Model is too Complex it can Overfit

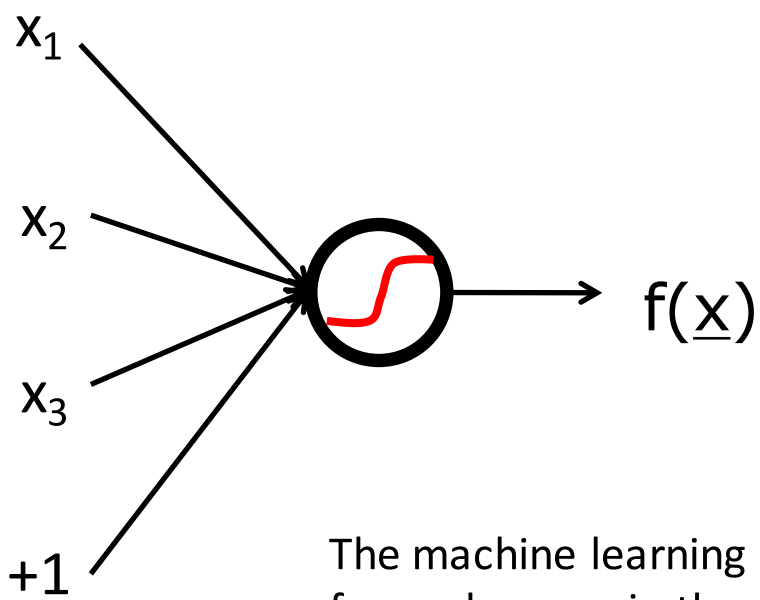


Neural Network Classifiers

Machine Learning Notation

Features	$\underline{x}$	*e.g., pixel inputs (usually a multidimensional vector)*
Targets	y	*e.g., true label for an image: "cat" or "no cat"*
Predictions	$\hat{y}$	*e.g., model's prediction given inputs, e.g., "cat"*
Error	$e(y , \hat{y})$	*e.g., e = 0 if prediction matches target, 1 otherwise*
Parameters	θ	*e.g., weights, coefficients specifying the model*

Example: A Simple Linear Model

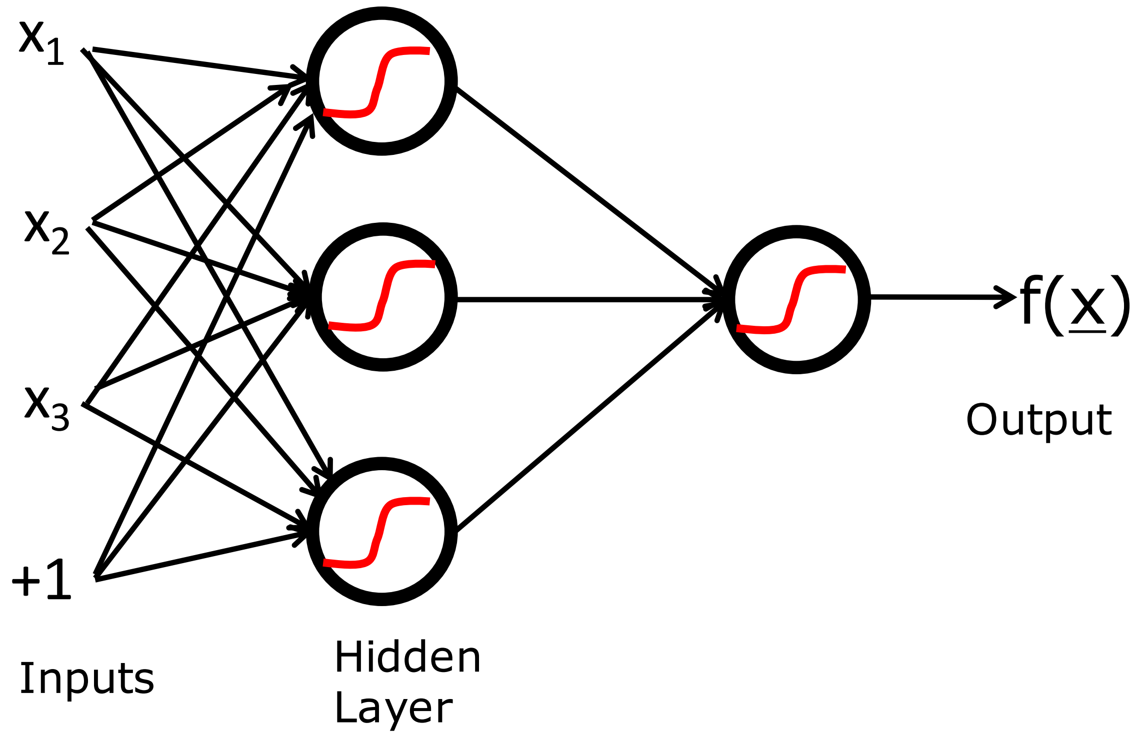


The machine learning algorithm will learn a weight for each arrow in the diagram

This a simple model: one weight per input

A Simple Neural Network

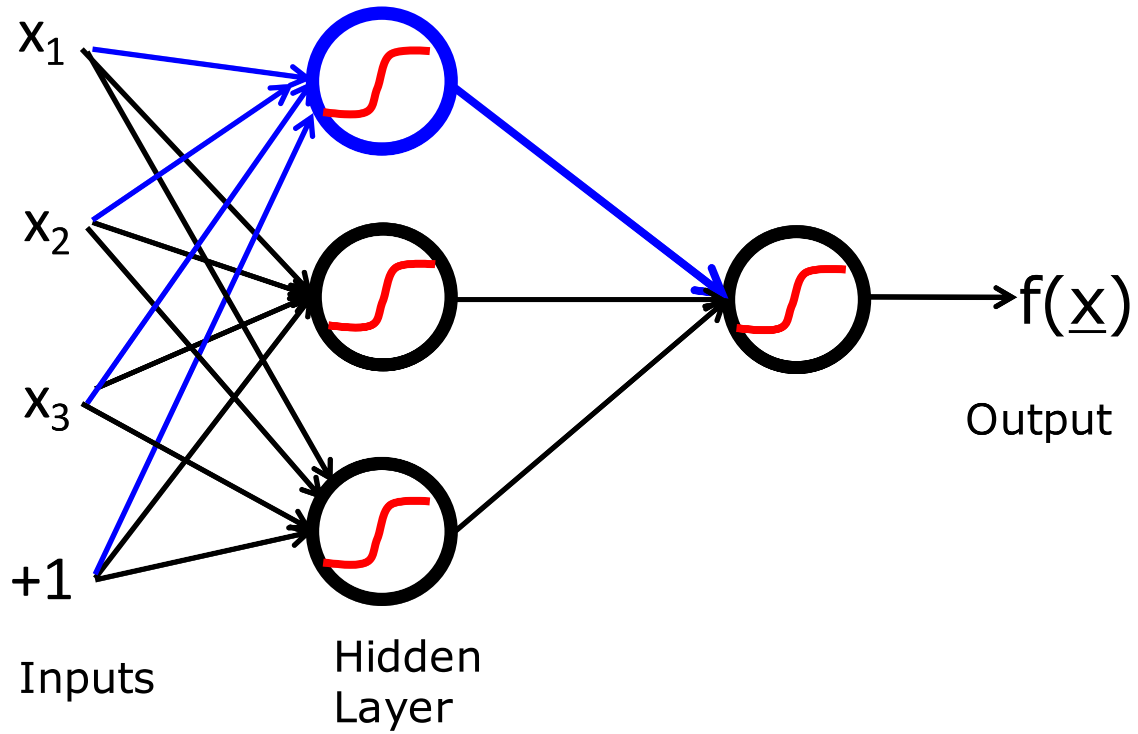
Here the model learns 3 different functions and then combines the outputs of the 3 to make a prediction



This is more complex and has more parameters than the simple model

A Simple Neural Network

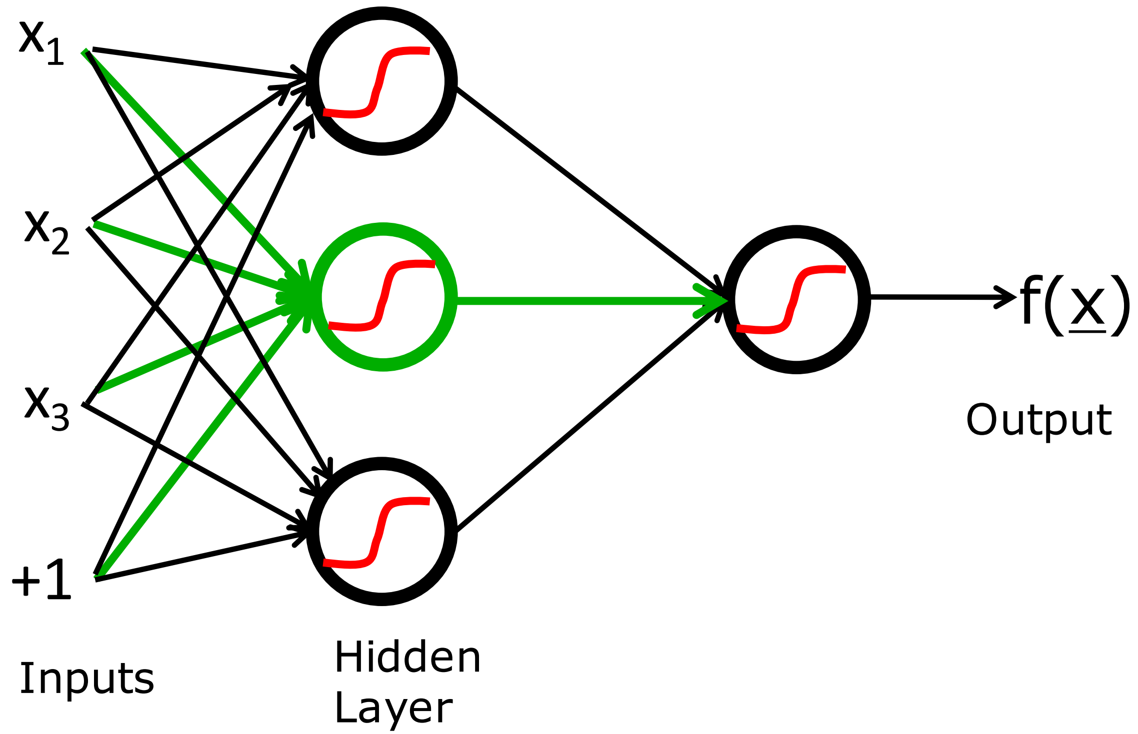
Here the model learns 3 different functions and then combines the outputs of the 3 to make a prediction



This is more complex and has more parameters than the simple model

A Simple Neural Network

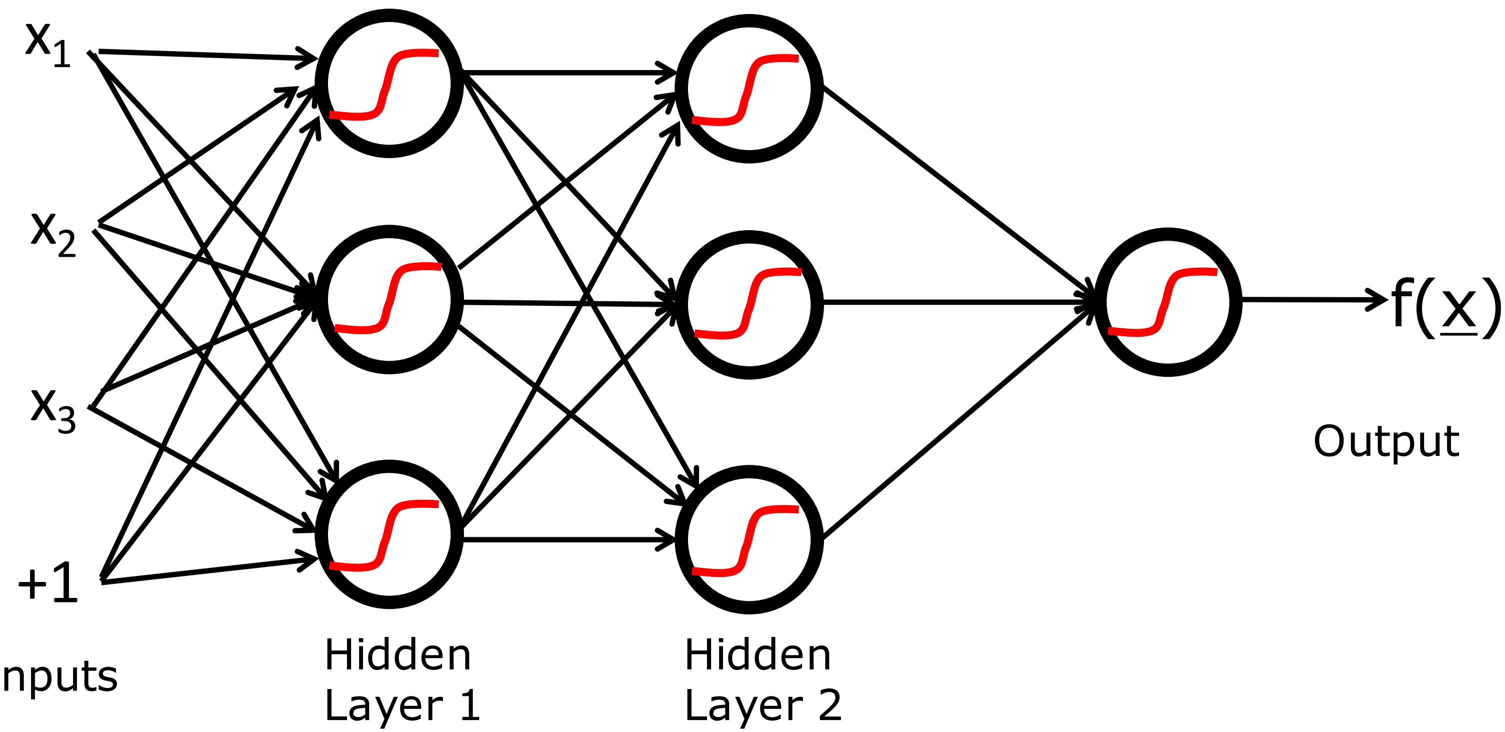
Here the model learns 3 different functions and then combines the outputs of the 3 to make a prediction



This is more complex and has more parameters than the simple model

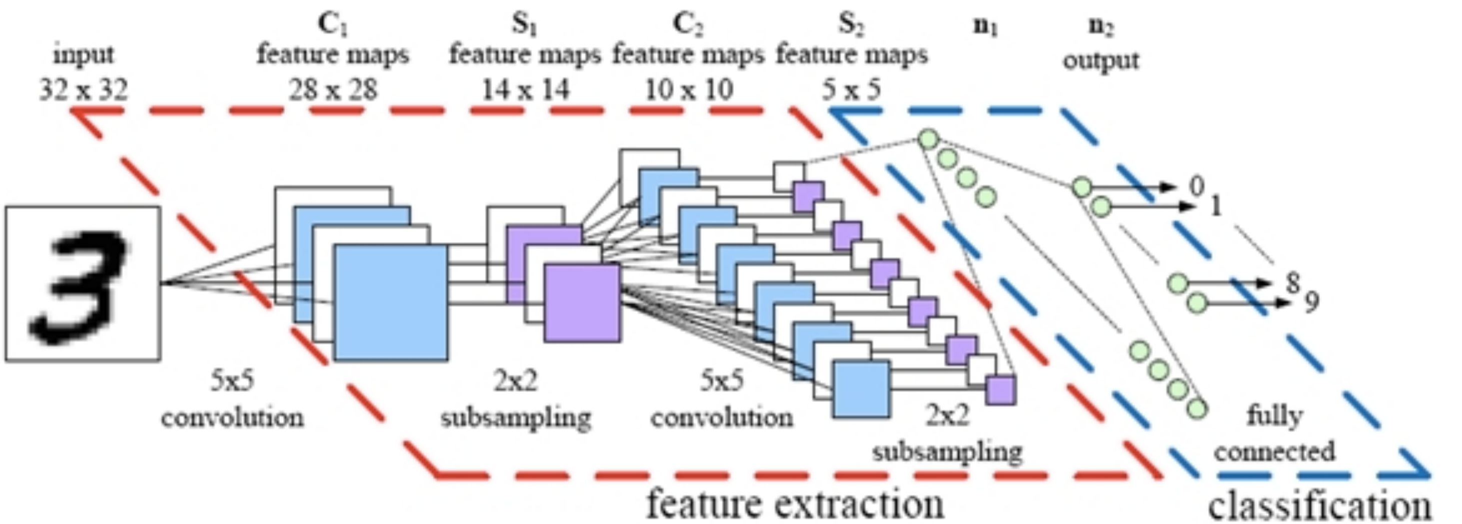
Deep Learning: Models with More Hidden Layers

We can build on this idea to create "deep models" with many hidden layers



Very flexible and complex functions

Example of a Network for Image Recognition



Mathematically this is just a function (a complicated one)

Figure from http://parse.ele.tue.nl/

A Brief History of Neural Networks…

- The Perceptron Era: 1950s and 60s
  - Great optimism with perceptrons (linear models)....
  - ...until Minsky, 1969: perceptrons had limited representation power
  - Hard problems require hidden layers....but there was no training algorithm

- The Backpropagation Era: Late 1980s to mid-90's
  - Invention of backpropagation – training of models with hidden layers
  - Wild enthusiasm (in the US at least)....NIPS conference, funding, etc
  - Mid 1990's: enthusiasm dies out: training deep NNs is hard

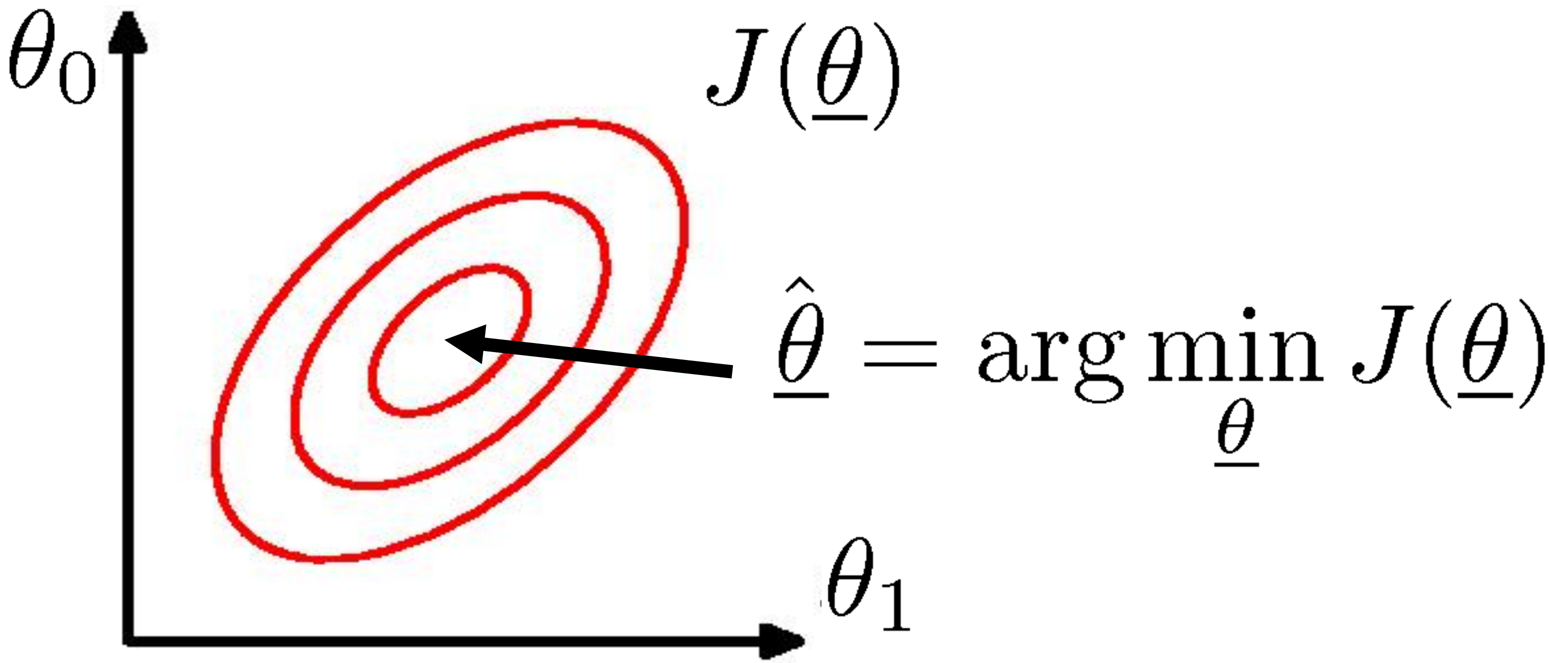
- The Deep Learning Era: 2010-present
  - 3rd wave of neural network enthusiasm
  - What happened since mid 90's?
    - Much larger data sets
    - Much greater computational power
    - Fast optimization techniques

Learning via Gradient Descent

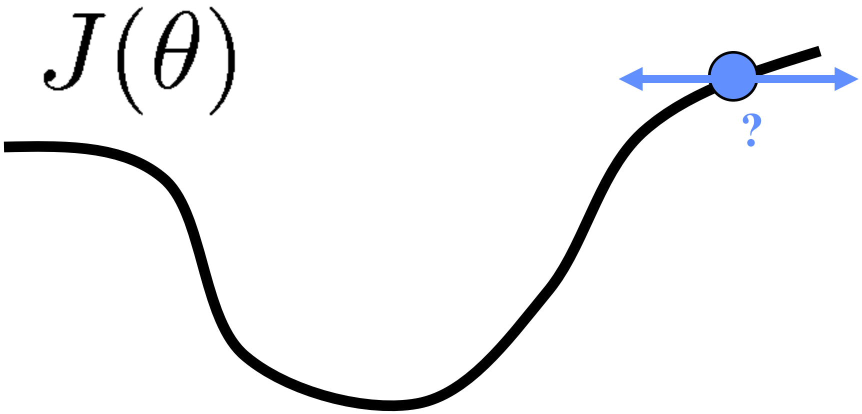
Finding good parameters

- Want to find parameters θ which minimize our error…

- Think of a cost "surface": error residual for that θ…

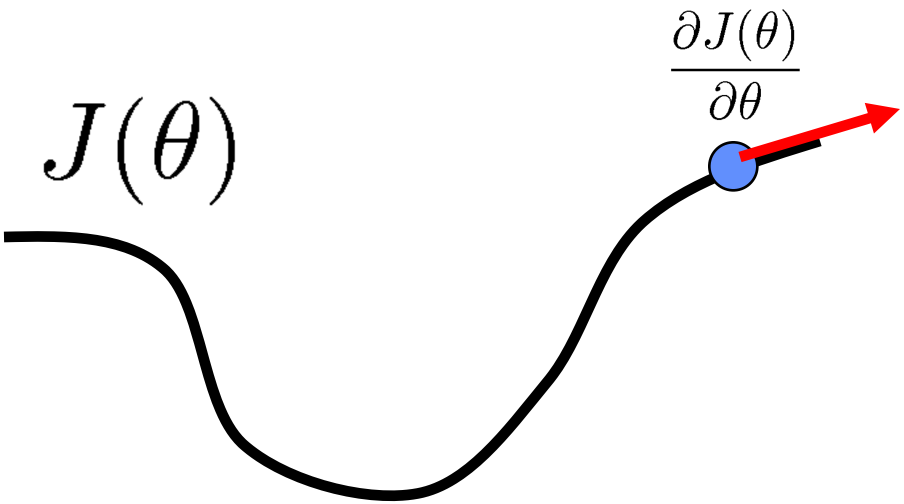


Gradient descent



- How to change θ to improve J(θ)?
- Choose a direction in which J(θ) is decreasing

Gradient descent

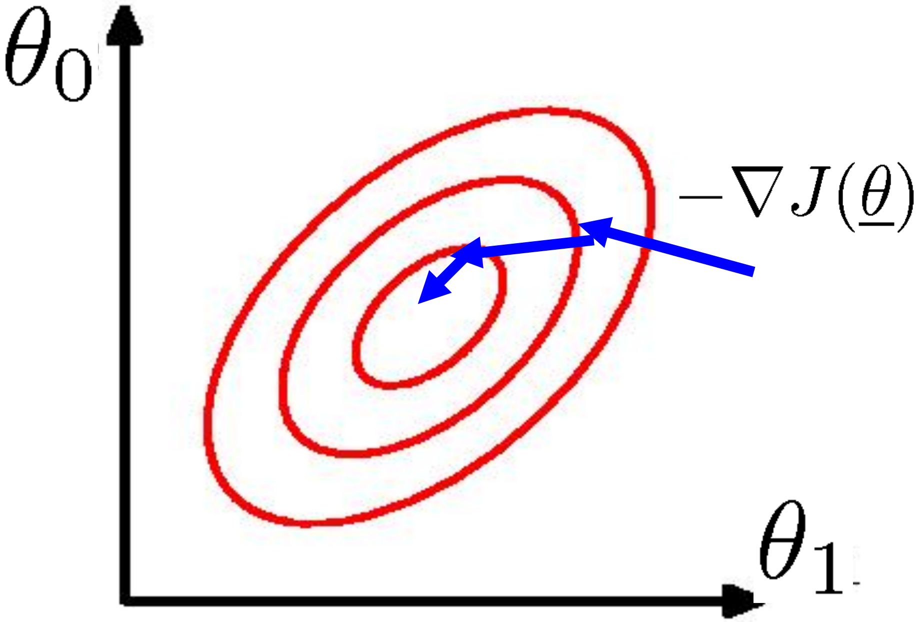


- How to change θ to improve J(θ)?
- Choose a direction in which J(θ) is decreasing

- Derivative $\frac{\partial J(\theta)}{\partial \theta}$

- Positive => increasing
- Negative => decreasing

Gradient descent in more dimensions



- Gradient vector

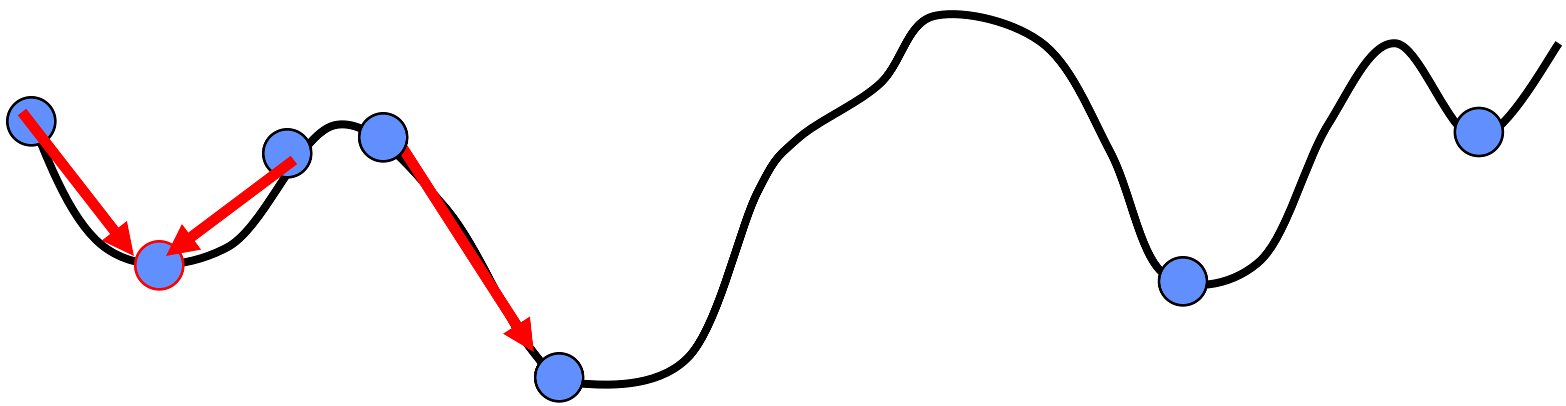
$$\nabla J(\underline{\theta}) = \left[\frac{\partial J(\underline{\theta})}{\partial \theta_0} \quad \frac{\partial J(\underline{\theta})}{\partial \theta_1} \quad \ldots \right]$$

- Indicates direction of steepest ascent

  (negative = steepest descent)

Comments on gradient descent

- Simple and general algorithm
  - Usable in broad variety of models
- Local minima
  - Sensitive to starting point



UCIrvine
University of California, Irvine

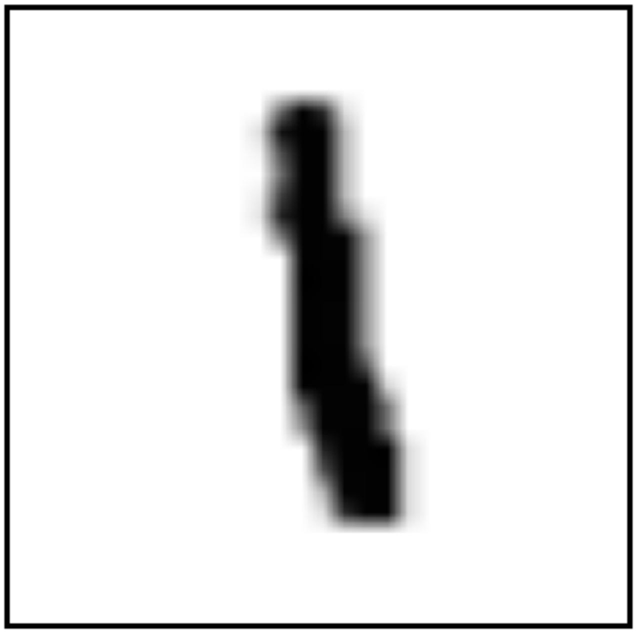
Image Classification Examples

Example: Classifying Handwritten Digits

What the data looks like to the human eye

Inputs: pixel values from each image
Output: 10 possible classes (0, 1, …, 9)

Pixel Inputs Represented Numerically



$$\simeq \begin{bmatrix} 0&0&0&0&0&0&0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&.6&.8&0&0&0&0&0&0\\ 0&0&0&0&0&0&.7&1&0&0&0&0&0&0\\ 0&0&0&0&0&0&.7&1&0&0&0&0&0&0\\ 0&0&0&0&0&0&.5&1&.4&0&0&0&0&0\\ 0&0&0&0&0&0&0&1&.4&0&0&0&0&0\\ 0&0&0&0&0&0&0&1&.4&0&0&0&0&0\\ 0&0&0&0&0&0&0&1&.7&0&0&0&0&0\\ 0&0&0&0&0&0&0&1&1&0&0&0&0&0\\ 0&0&0&0&0&0&0&.9&1&.1&0&0&0&0\\ 0&0&0&0&0&0&0&.3&1&.1&0&0&0&0\\ 0&0&0&0&0&0&0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0&0&0&0&0&0&0 \end{bmatrix}$$

From https://www.tensorflow.org/get_started/mnist/beginners

Example: Classifying Handwritten Digits

Classification Accuracy has gone from 93% to 99.9% in the past 10 years

Examples of Errors made by the Neural Network Classifier

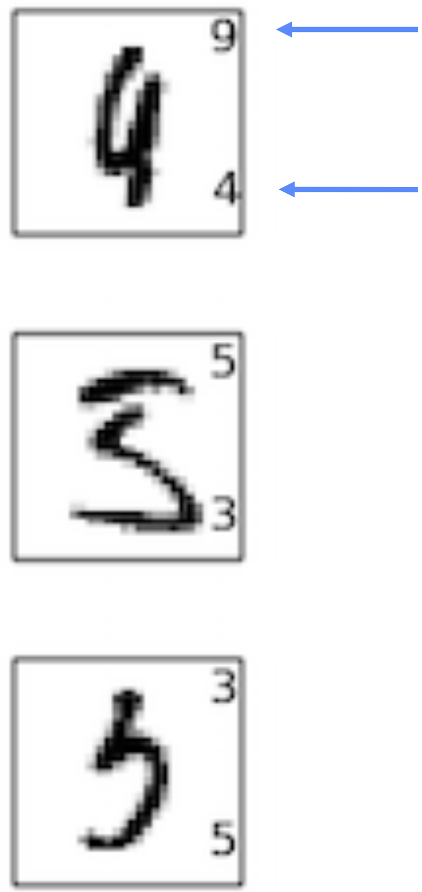
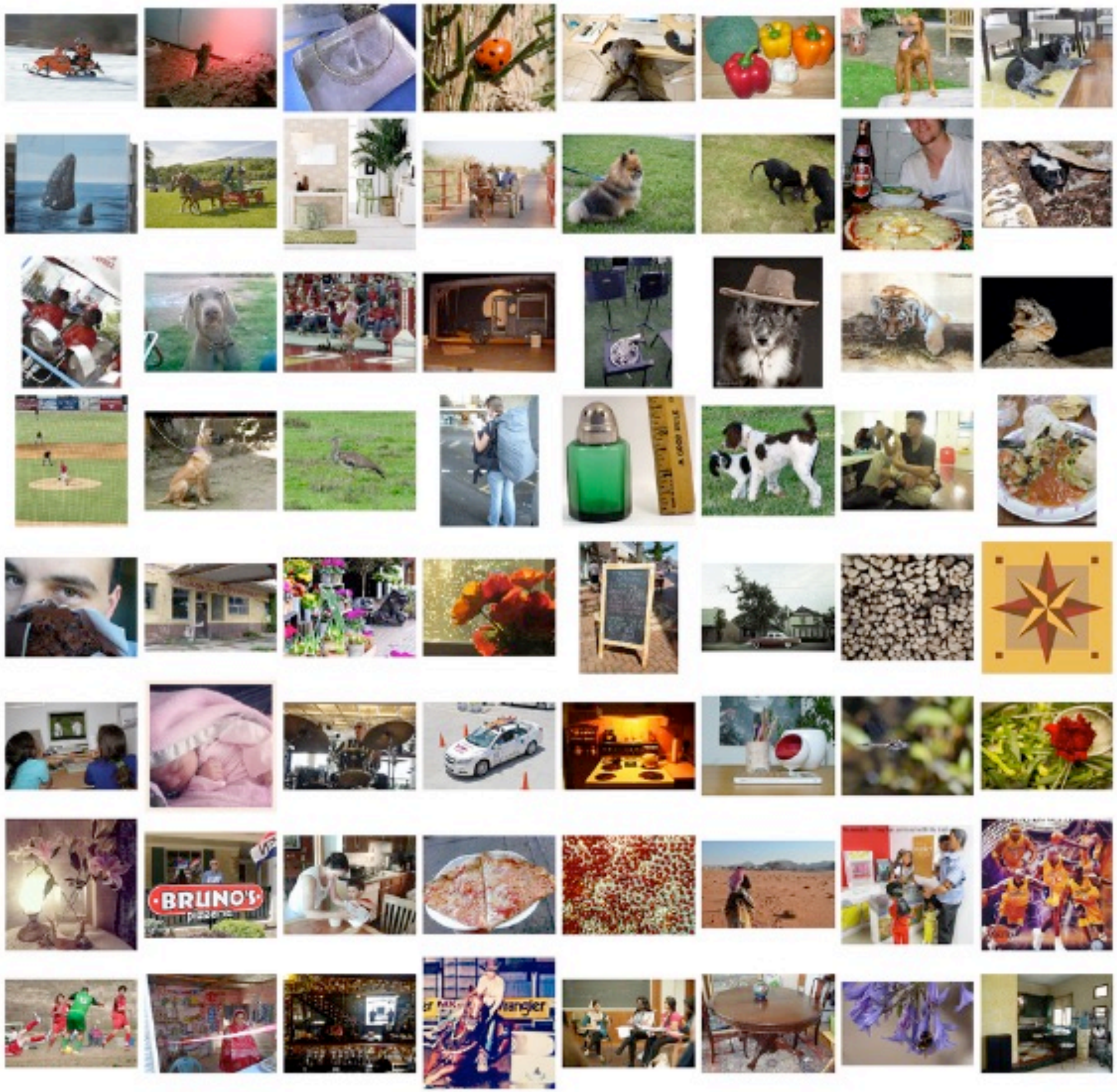


Image from http://neuralnetworksanddeeplearning.com/chap6.html

Russakovsky et al, ImageNet Large Scale Visual Recognition Challenge, 2015



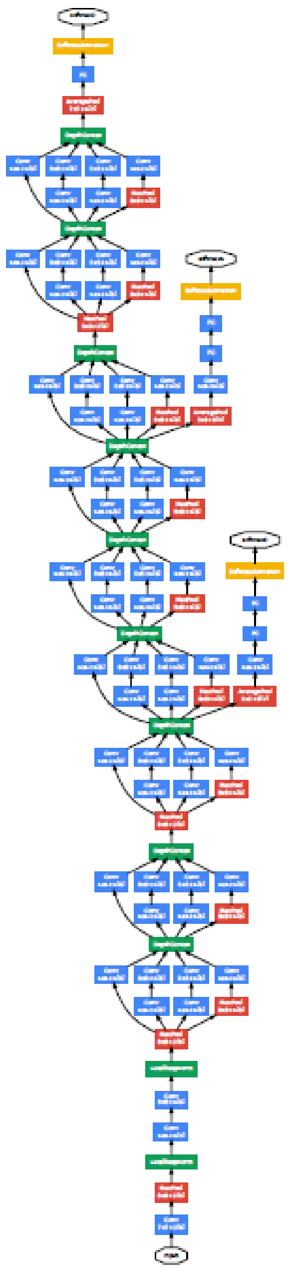


Figure 3: GoogLeNet network with all the bells and whistles

Deep Network architecture for GoogLeNet network, 27 layers

Training data
inputs x = raw pixel values
labels y = values from 1 to 1000

Trained on millions of images

How is network structure determined?
Essentially trial-and-error (expensive!)

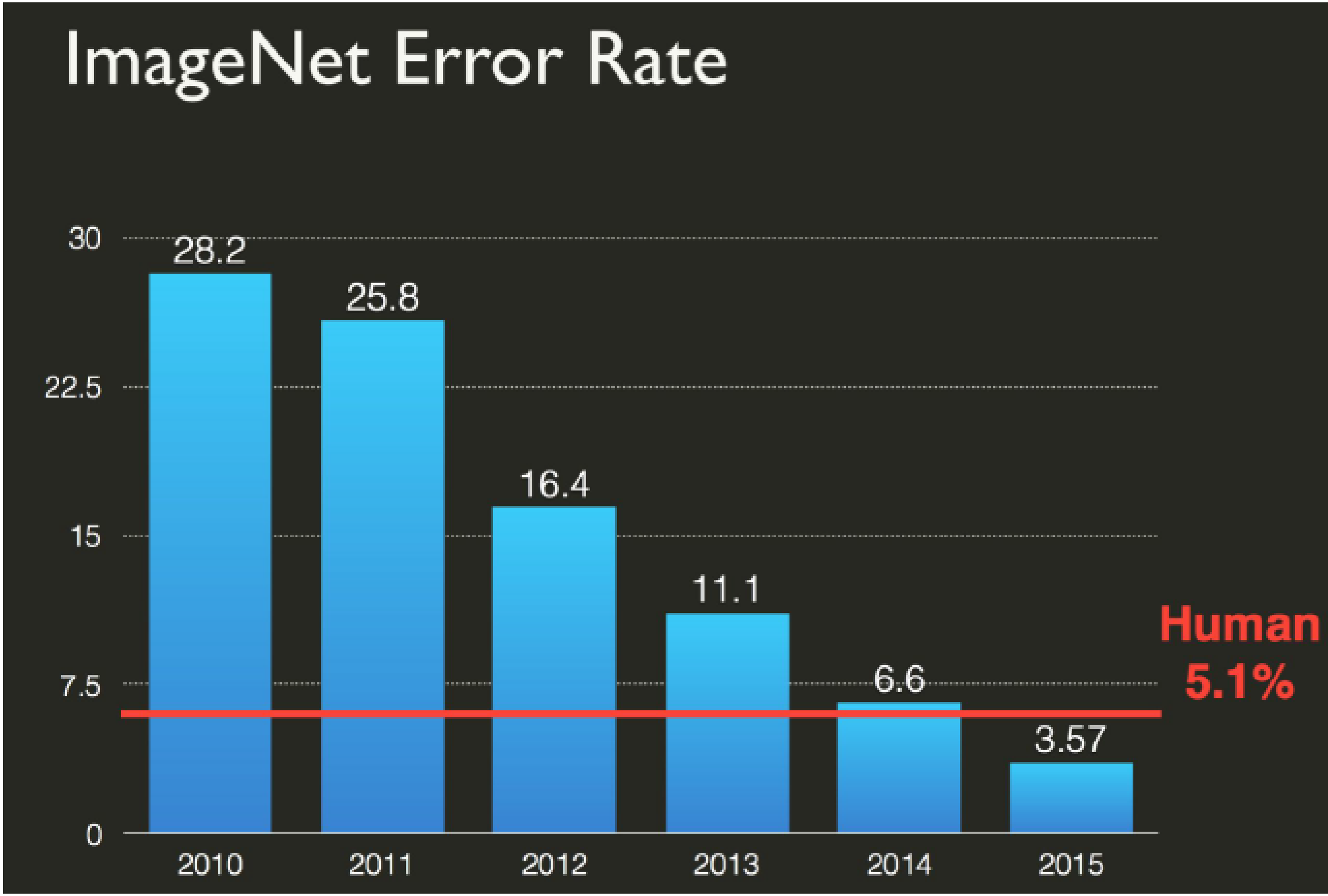


Figure from Kevin Murphy, Google, 2016

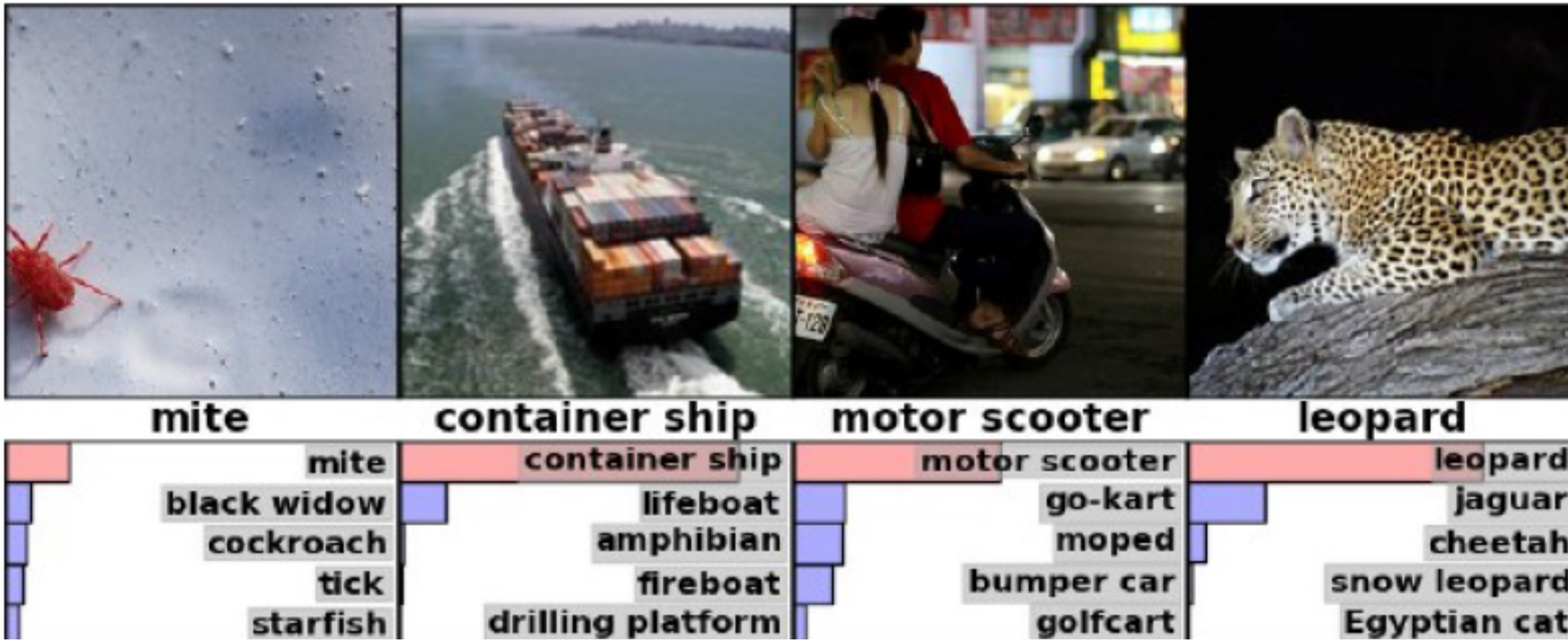


Figure from Krizhevsky, Sutskever, Hinton, 2012

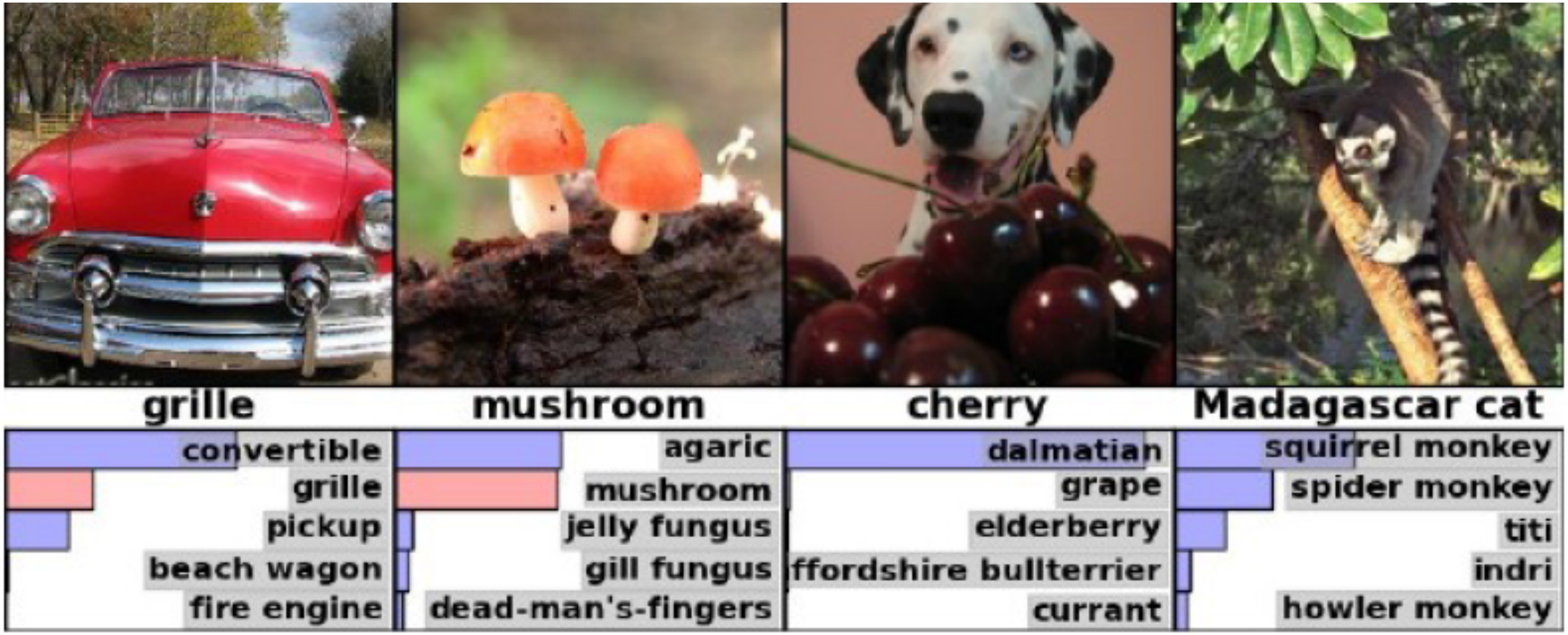
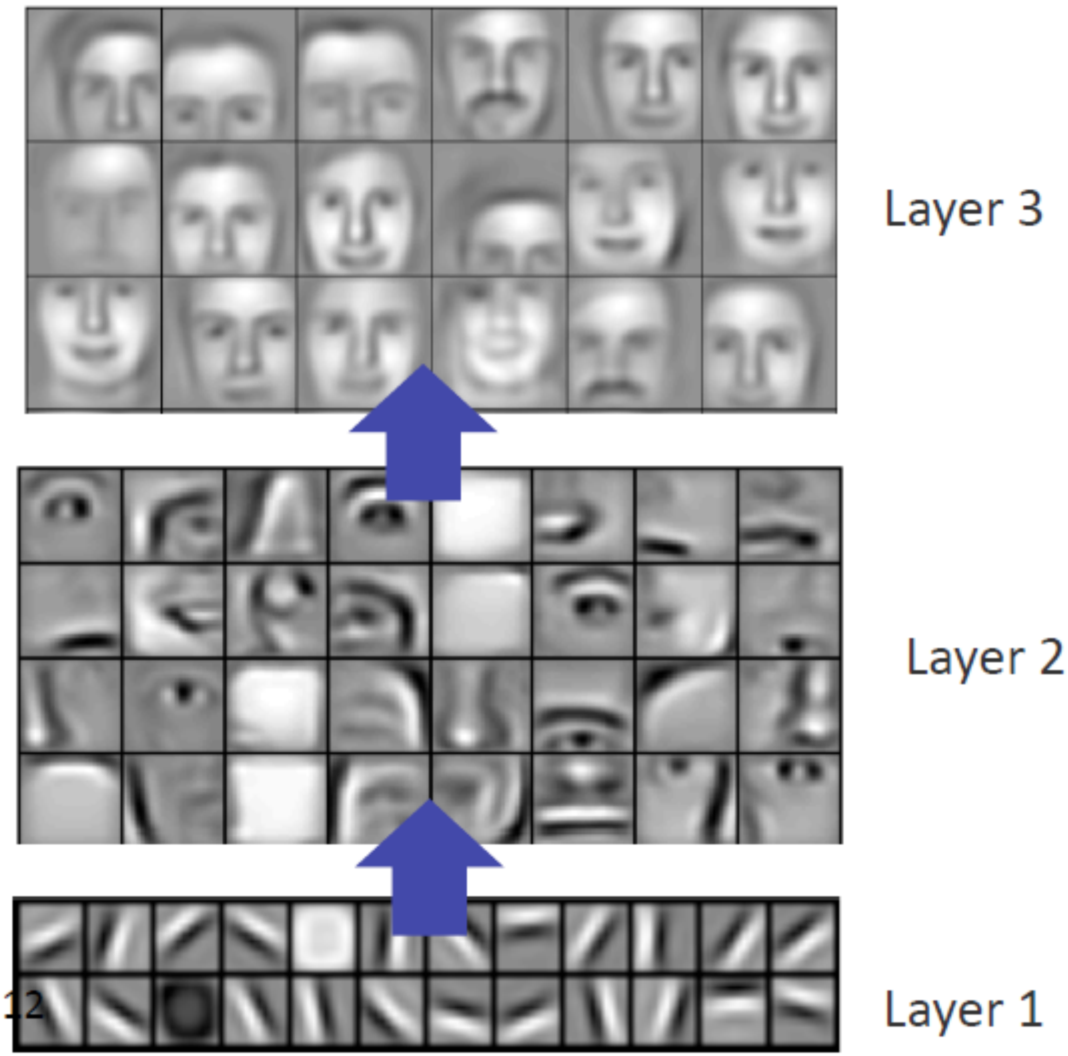


Figure from Krizhevsky, Sutskever, Hinton, 2012

Figure from Lee et al., ICML 2009



Sequence Prediction Examples

Learning by Predicting what's Next

- Examples
  - Predict the next word a person will type or speak, given words up to this point
  - Predict the value of the Dow Jones tomorrow afternoon, given history

- We can use the same general methodologies as before
  - Model now uses past data to predict next event

- Applications
  - Speech recognition
  - Auto-suggest in human typing
  - Machine translation
  - Consumer modeling
  - Chatbots
  - …and more

Example: Predicting the Next Character

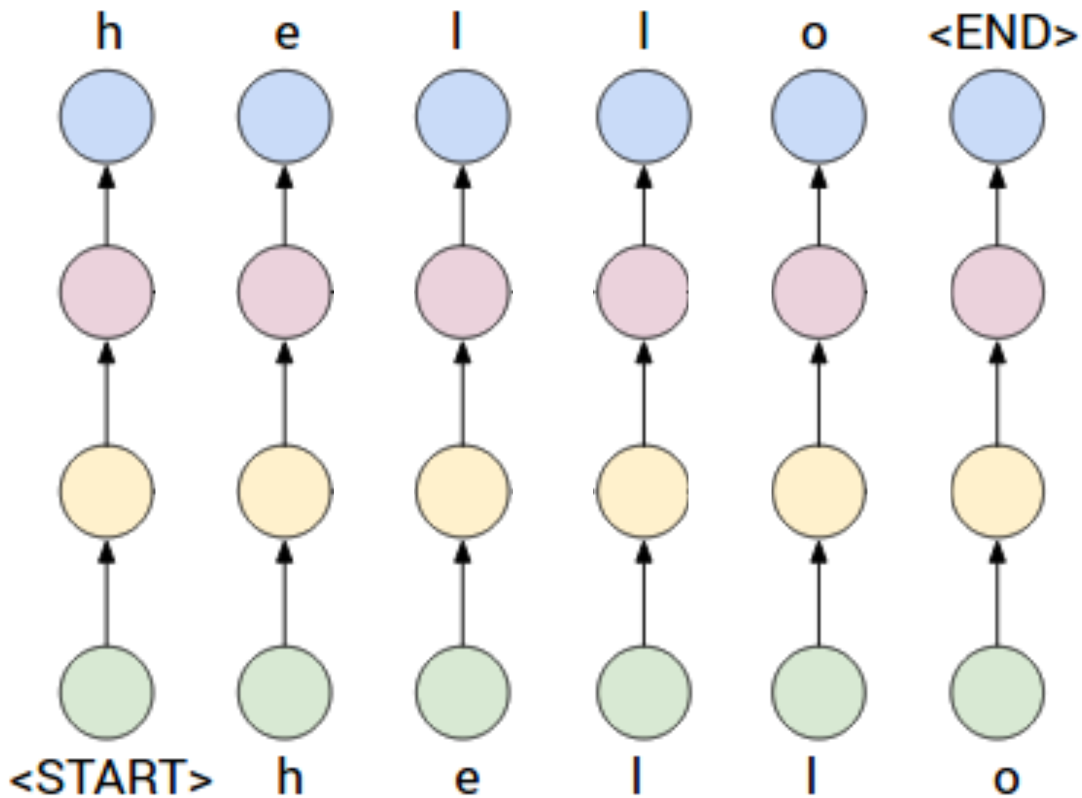


Figure from http://cs.stanford.edu/people/karpathy/recurrentjs/

Example: Predicting Characters with a Recurrent Network

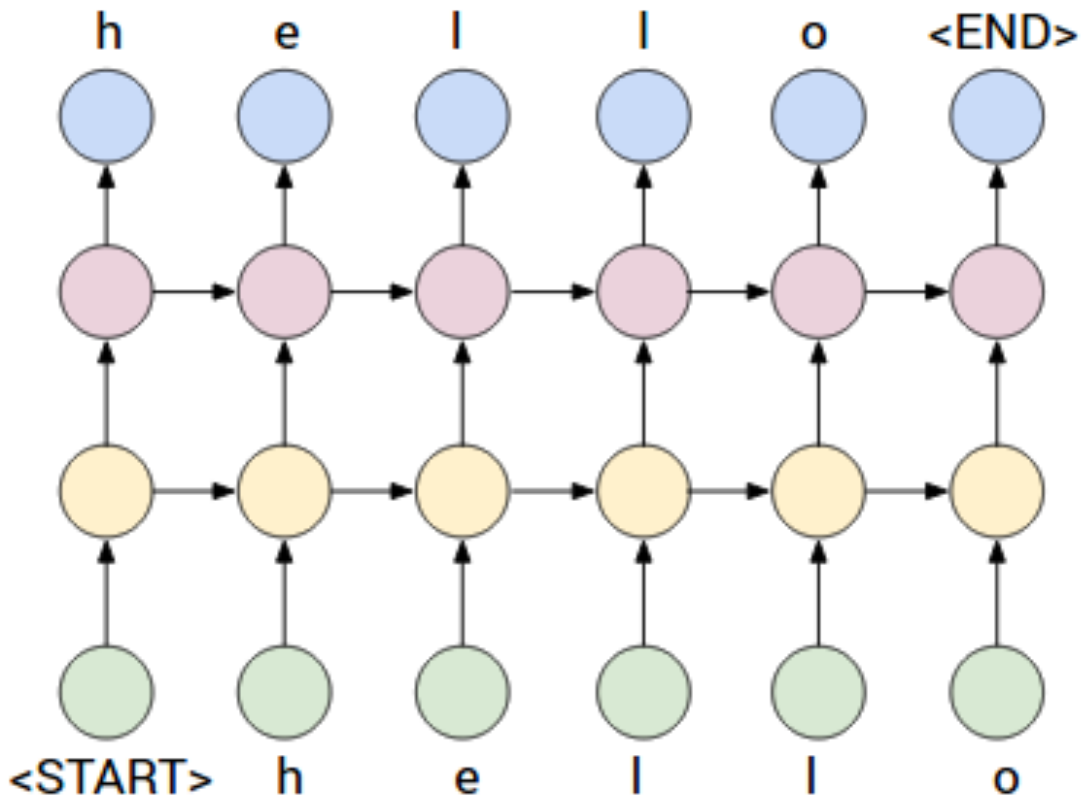


Figure from http://cs.stanford.edu/people/karpathy/recurrentjs/

Output from a Model Learned on Shakespeare

KING LEAR:
O, if you were a feeble sight, the courtesy of your law,
Your sight and several breath, will wear the gods
With his heads, and my hands are wonder'd at the deeds,
So drop upon your lordship's head, and your opinion
Shall be against your honour.

Second Senator:
They are away this miseries, produced upon my soul,
Breaking and strongly should be buried, when I perish
The earth and thoughts of many states.

DUKE VINCENTIO: Well, your wit is in the care of side and that.

Examples from "The Unreasonable Effectiveness of Recurrent Neural Networks", Andrej Kaparthy, blog, http://karpathy.github.io/2015/05/21/rnn-effectiveness/

Output from a Model Learned on Cooking Recipes

```
MMMMM----- Recipe via Meal-Master (tm) v8.05

      Title: CARAMEL CORN GARLIC BEEF
 Categories: Soups, Desserts
      Yield: 10 Servings

      2 tb Parmesan cheese, ground
    1/4 ts Ground cloves
           -- diced
      1 ts Cayenne pepper

  Cook it with the batter. Set aside to cool. Remove the peanut oil in a small saucepan and
  pour into the margarine until they are soft. Stir in a
  a mixer (dough). Add the chestnuts, beaten egg whites, oil, and salt
  and brown sugar and sugar; stir onto the boqtly brown it.

  The recipe from an oiled by fried and can. Beans, by Judil Cookbook, Source: Pintore,
  October, by Chocolates, Breammons of Jozen, Empt.com

MMMMM
```

From https://gist.github.com/nylki/1efbaa36635956d35bcc

Output from a Model Learned on Source Code

```
/*
 * Increment the size file of the new incorrect UI_FILTER group information
 * of the size generatively.
 */
static int indicate_policy(void)
{
  int error;
  if (fd == MARN_EPT) {
    /*
     * The kernel blank will coeld it to userspace.
     */
    if (ss->segment < mem_total)
      unblock_graph_and_set_blocked();
    else
      ret = 1;
    goto bail;
  }
  segaddr = in_SB(in.addr);
  selector = seg / 16;
  setup_works = true;
  for (i = 0; i < blocks; i++) {
    seq = buf[i++];
    bpf = bd->bd.next + i * search;
    if (fd) {
      current = blocked;
    }
  }
```

Examples from "The Unreasonable Effectiveness of Recurrent Neural Networks", Andrej Kaparthy, blog, http://karpathy.github.io/2015/05/21/rnn-effectiveness/

Output from a Model Learned on Mathematics Papers

For $\bigoplus_{n=1,\ldots,m}$ where $\mathcal{L}_{m_\bullet} = 0$, hence we can find a closed subset $\mathcal{H}$ in $\mathcal{H}$ and any sets $\mathcal{F}$ on X, U is a closed immersion of S, then $U \to T$ is a separated algebraic space.

Proof. Proof of (1). It also start we get

$$S = \mathrm{Spec}(R) = U \times_X U \times_X U$$

and the comparicoly in the fibre product covering we have to prove the lemma generated by $\coprod Z \times_U U \to V$. Consider the maps M along the set of points Sch_{fppf} and $U \to U$ is the fibre category of S in U in Section, **??** and the fact that any U affine, see Morphisms, Lemma **??**. Hence we obtain a scheme S and any open subset $W \subset U$ in $Sh(G)$ such that $\mathrm{Spec}(R') \to S$ is smooth or an

$$U = \bigcup U_i \times_{S_i} U_i$$

which has a nonzero morphism we may assume that f_i is of finite presentation over S. We claim that $\mathcal{O}_{X,x}$ is a scheme where $x, x', s'' \in S'$ such that $\mathcal{O}_{X,x'} \to \mathcal{O}'_{X',x'}$ is separated. By Algebra, Lemma **??** we can define a map of complexes $\mathrm{GL}_{S'}(x'/S'')$ and we win. □

To prove study we see that $\mathcal{F}|_U$ is a covering of $\mathcal{X}'$, and $\mathcal{T}_i$ is an object of $\mathcal{F}_{X/S}$ for $i > 0$ and $\mathcal{F}_p$ exists and let $\mathcal{F}_i$ be a presheaf of $\mathcal{O}_X$-modules on $\mathcal{C}$ as a $\mathcal{F}$-module. In particular $\mathcal{F} = U/\mathcal{F}$ we have to show that

$$\widetilde{M}^\bullet = \mathcal{I}^\bullet \otimes_{\mathrm{Spec}(k)} \mathcal{O}_{S,s} - i_X^{-1}\mathcal{F})$$

is a unique morphism of algebraic stacks. Note that

$$\mathrm{Arrows} = (Sch/S)^{opp}_{fppf}, (Sch/S)_{fppf}$$

and

$$V = \Gamma(S, \mathcal{O}) \longmapsto (U, \mathrm{Spec}(A))$$

is an open subset of X. Thus U is affine. This is a continuous map of X is the inverse, the groupoid scheme S.

Proof. See discussion of sheaves of sets. □

The result for prove any open covering follows from the less of Example **??**. It may replace S by $X_{spaces,étale}$ which gives an open subspace of X and T equal to S_{Zar}, see Descent, Lemma **??**. Namely, by Lemma **??** we see that R is geometrically regular over S.

Examples from “The Unreasonable Effectiveness of Recurrent Neural Networks”,
Andrej Kaparthy, blog, http://karpathy.github.io/2015/05/21/rnn-effectiveness/

Output from a Model Learned from US President Speeches

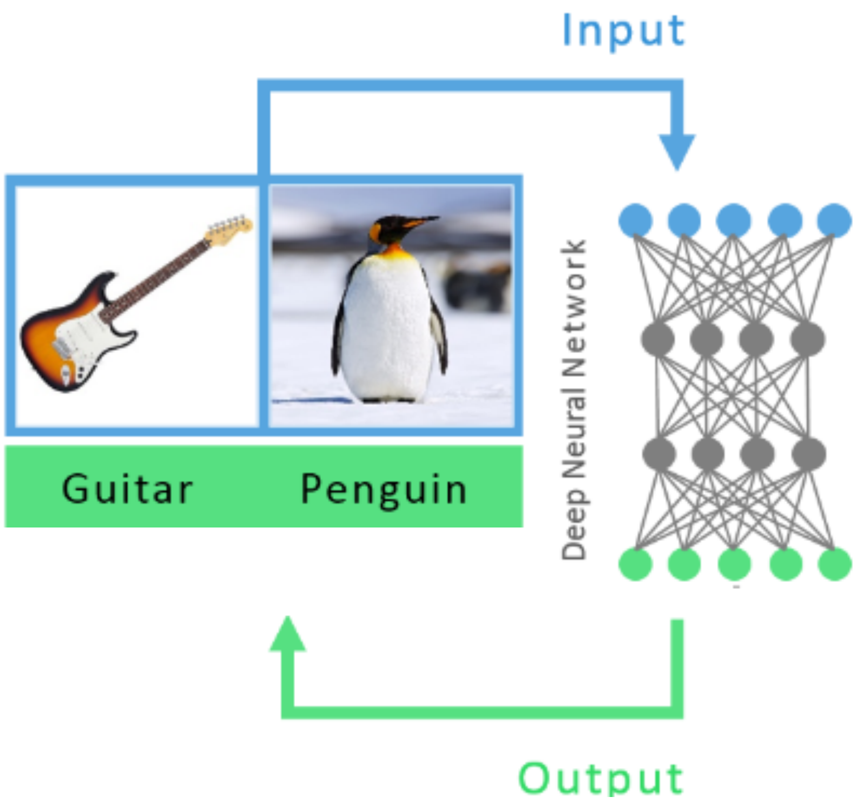
> *Good afternoon. God bless you.*
>
> *The United States will step up to the cost of a new challenges of the American people that will share the fact that we created the problem. They were attacked and so that they have to say that all the task of the final days of war that I will not be able to get this done. The promise of the men and women who were still going to take out the fact that the American people have fought to make sure that they have to be able to protect our part. It was a chance to stand together to completely look for the commitment to borrow from the American people. And the fact is the men and women in uniform and the millions of our country with the law system that we should be a strong stretcks of the forces that we can afford to increase our spirit of the American people and the leadership of our country who are on the Internet of American lives.*
>
> *Thank you very much. God bless you, and God bless the United States of America.*

From https://medium.com/@samim/

Limitations of Classification Algorithms

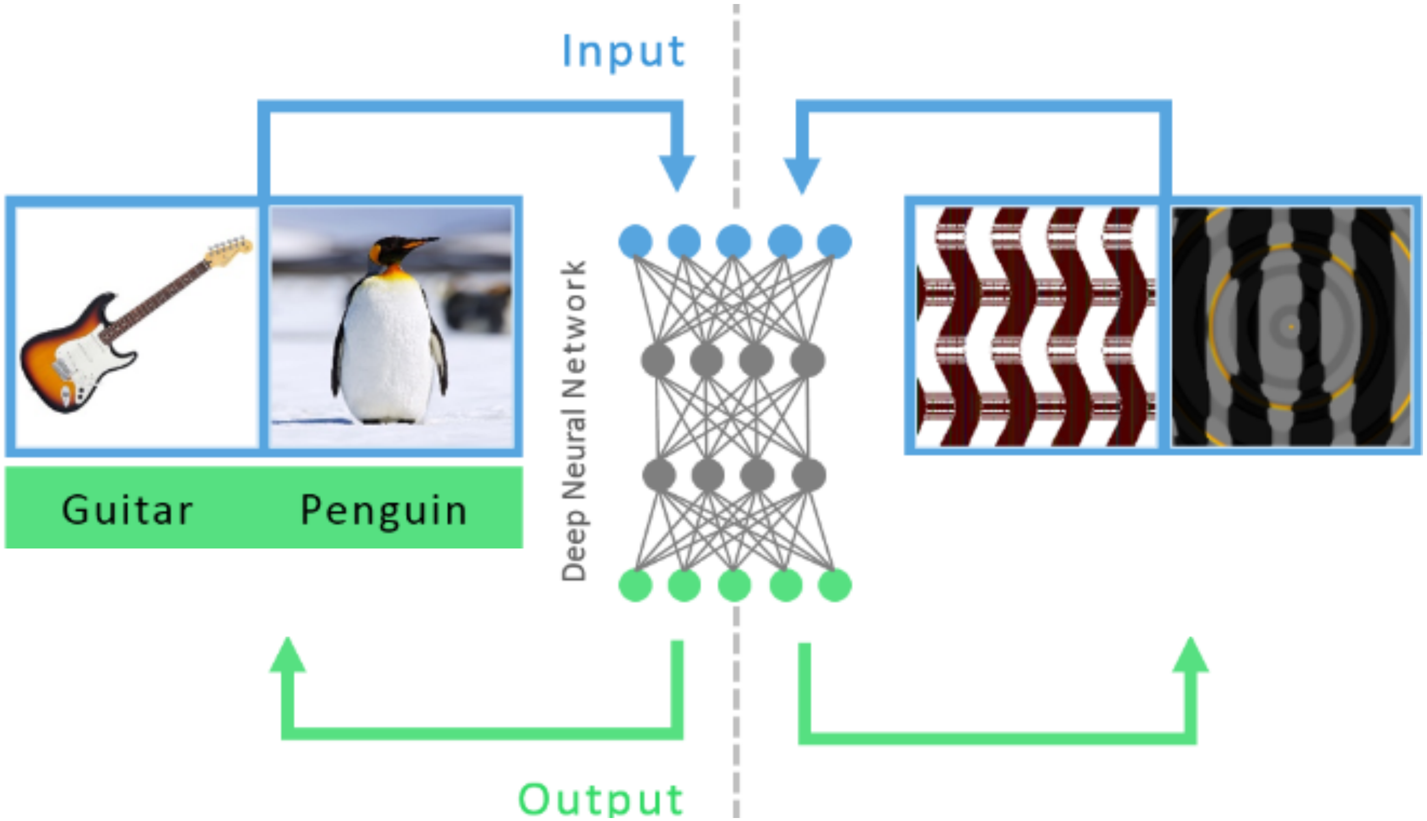
A Deep Neural Network for Image Recognition

From Nguyen, Yosinski, Clune, CVPR 2015



A Deep Neural Network for Image Recognition

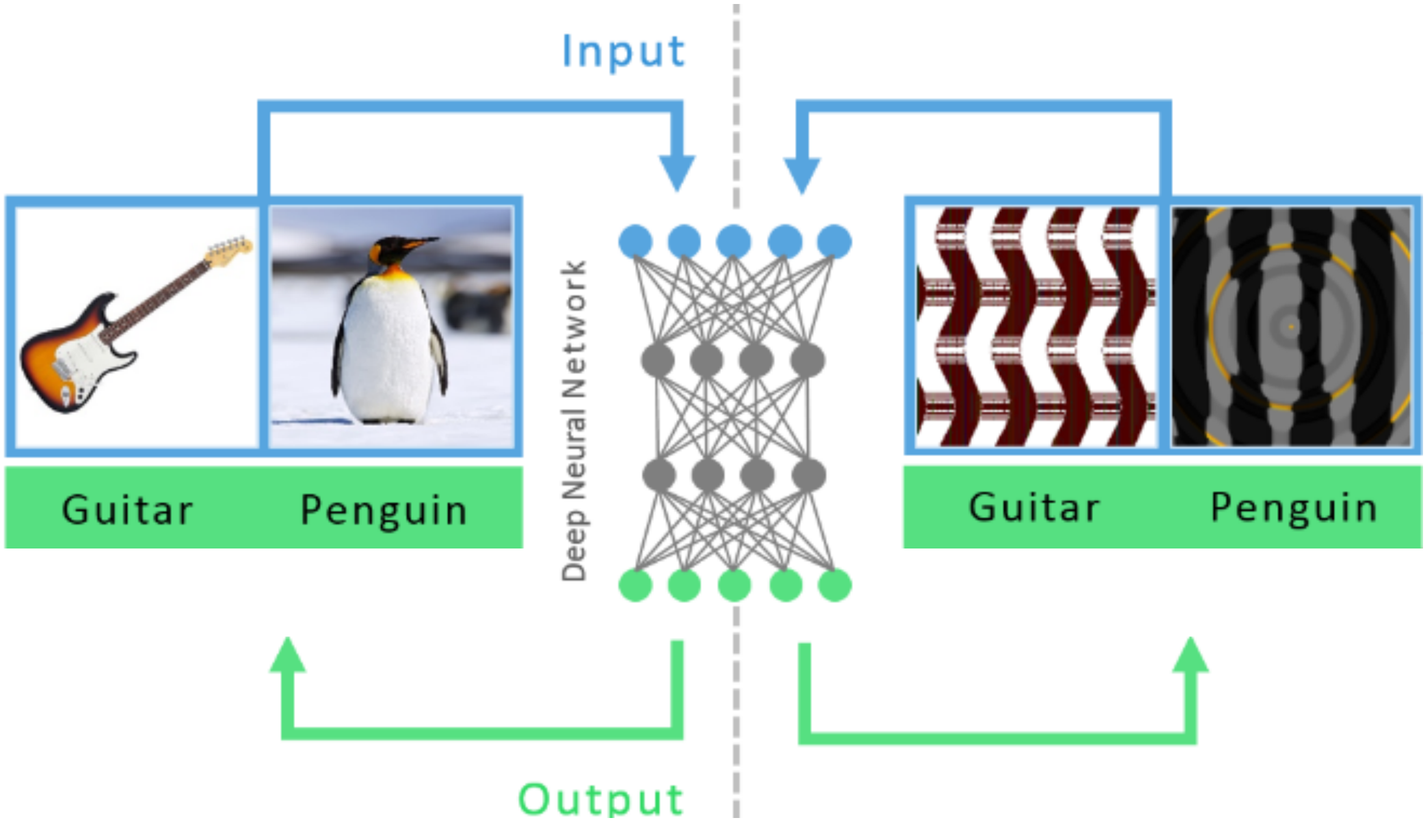
From Nguyen, Yosinski, Clune, CVPR 2015



Images used for Training **New Images**

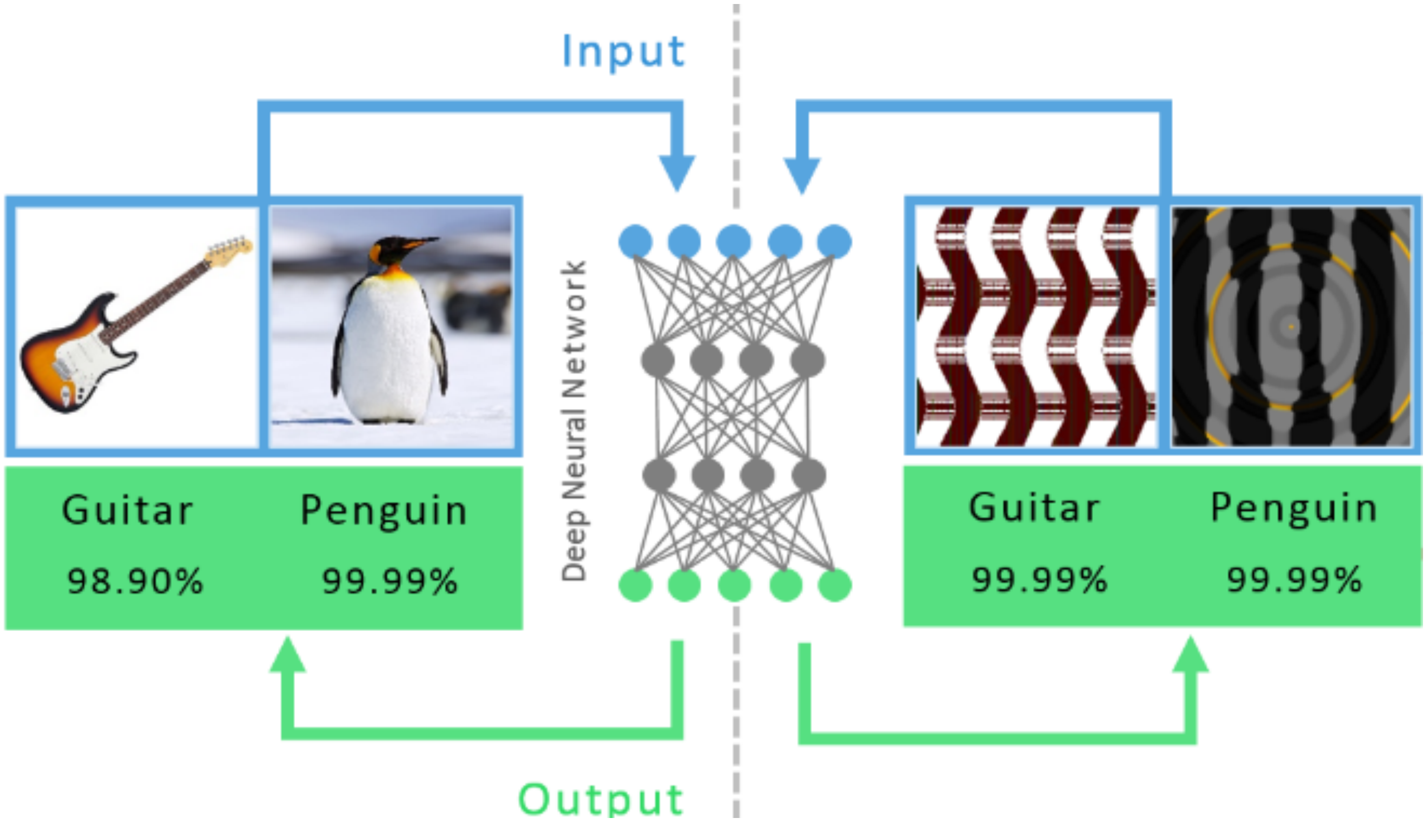
A Deep Neural Network for Image Recognition

From Nguyen, Yosinski, Clune, CVPR 2015



A Deep Neural Network for Image Recognition

From Nguyen, Yosinski, Clune, CVPR 2015

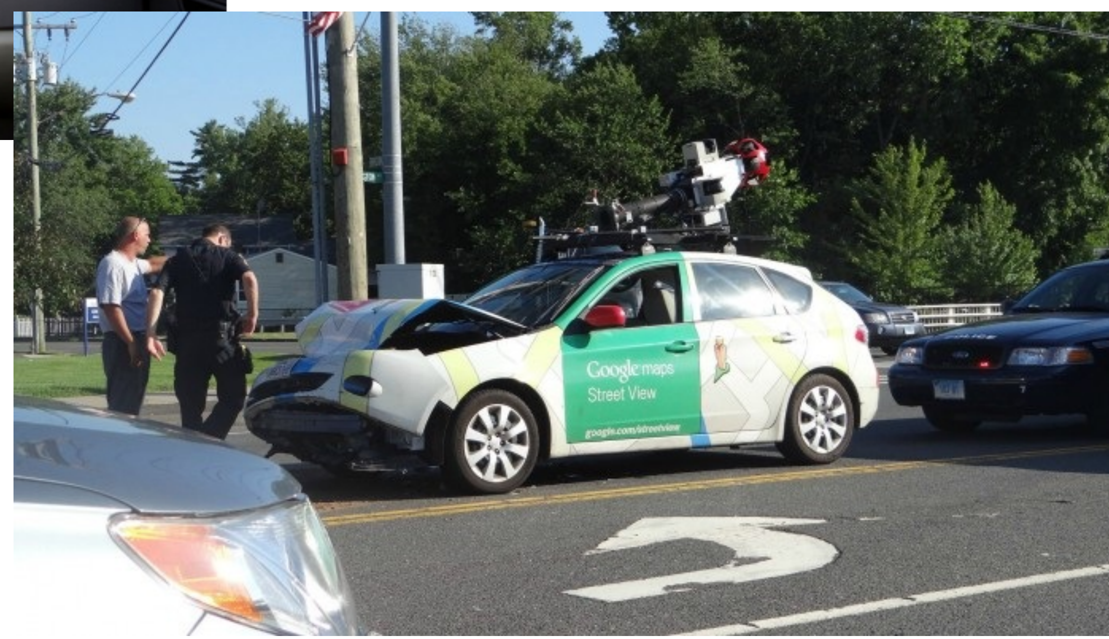


A Lesson of Tesla Crashes? Computer Vision Can't Do It All Yet

By **STEVE LOHR** SEPT. 19, 2016

International New York Times

Monday, September 19, 2016 | Today's Paper | Video | 57°F | S. & P. 500 +0.33% ↑



Schedule of Lectures

Date	Speaker	Department Or Organization	Topic
Jan 9	Padhraic Smyth	Computer Science	Introduction to Data Science
Jan 16	Padhraic Smyth	Computer Science	Machine Learning
Jan 23	Michael Carey	Computer Science	Databases and Data Management
Jan 30	Sameer Singh	Computer Science	Statistical Natural Language Processing
Feb 6	Zhaoxia Yu	Statistics	An Introduction to Cluster Analysis
Feb 13	Erik Sudderth	Computer Science	Computer Vision and Machine Learning
Feb 20	John Brock	Cylance, Inc	Data Science and CyberSecurity
Feb 27	Video Lecture (Kate Crawford)	Microsoft Research and NYU	Bias in Machine Learning
Mar 6	Matt Harding	Economics	Data Science in Economics and Finance
Mar 13	Padhraic Smyth	Computer Science	Review: Past and Future of Data Science