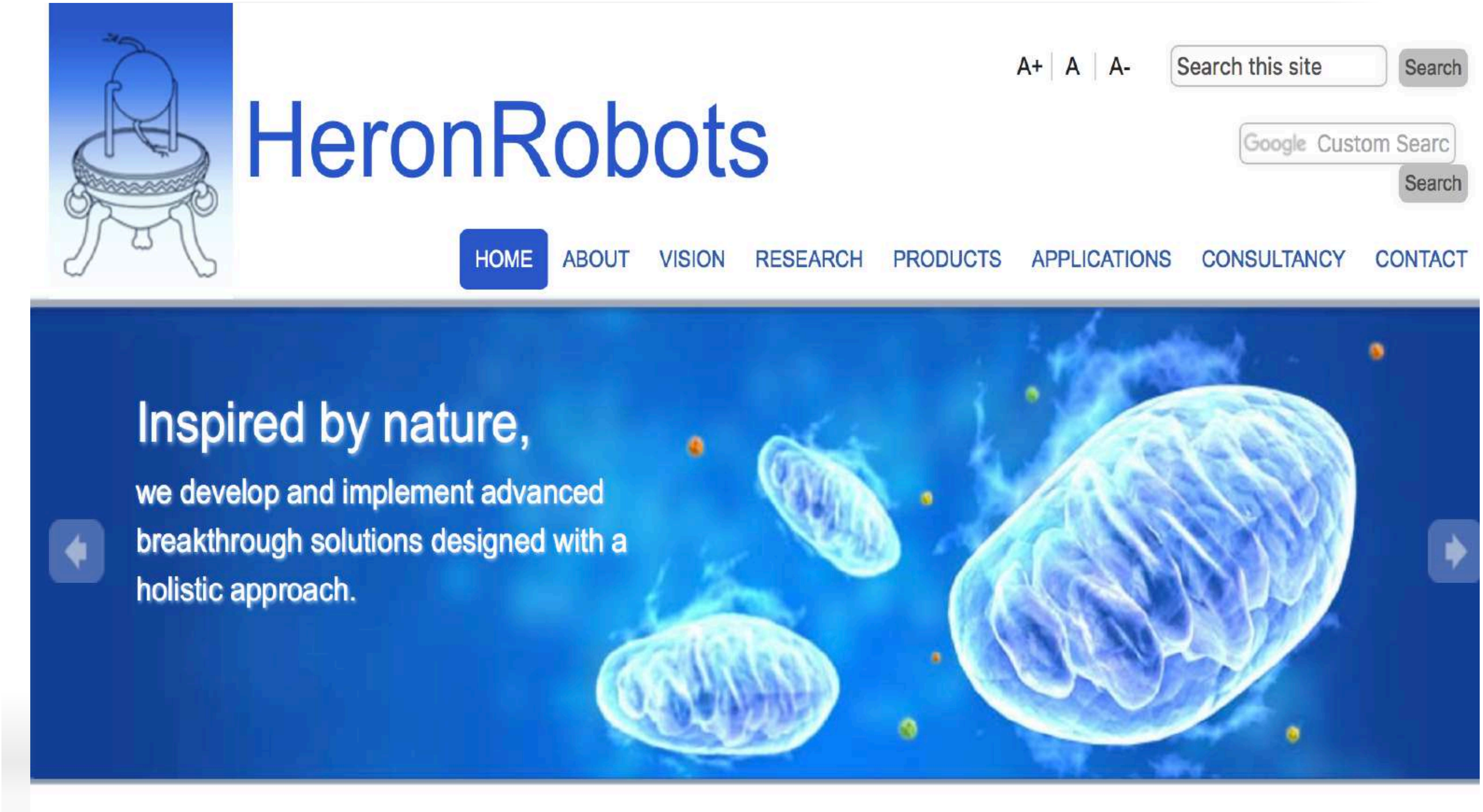
The Shanghai Lectures 2019

HeronRobots *Pathfinder Lectures*

Natural and Artificial Intelligence in Embodied Physical Agents


A+ A A-
Search this site
Search
HeronRobots
Google Custom Searc
Search
HOME
ABOUT
VISION
RESEARCH
PRODUCTS
APPLICATIONS
CONSULTANCY
CONTACT
Inspired by nature,
we develop and implement advanced breakthrough solutions designed with a holistic approach.

The ShanghAI Lectures

An experiment in global teaching

- Fabio Bonsignorio
- The ShanghAI Lectures and Heron Robots

欢迎您参与

“来自上海的人工智能系列讲座”

Lecture 5. ML, DL ..Object Recognition an Embodied AI view

Fabio Bonsignorio
The ShanghAI Lectures and Heron Robots

Crash Introduction to ML

Lecture slides adapted from *Deep Learning*
www.deeplearningbook.org

Ian Goodfellow

2016-09-26

Representations Matters

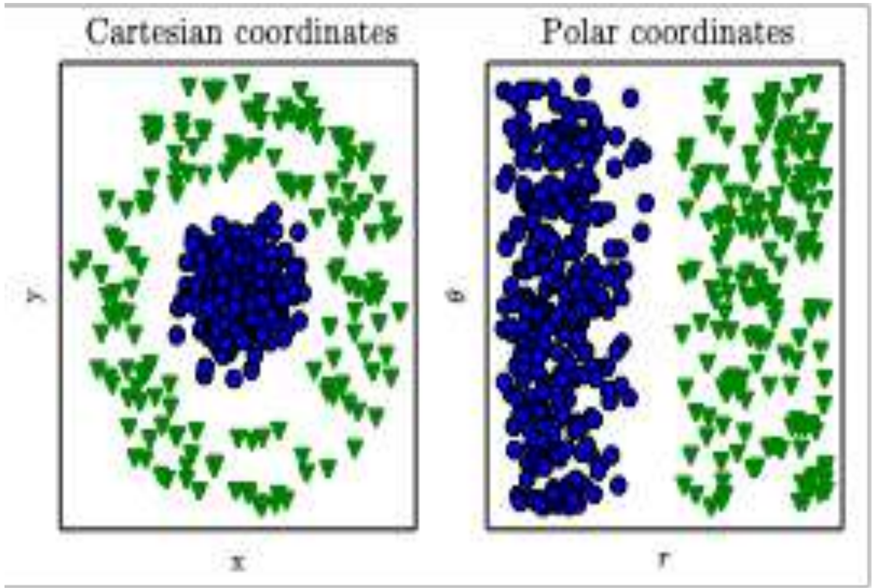


Figure 1.1

Depth: Repeated Composition

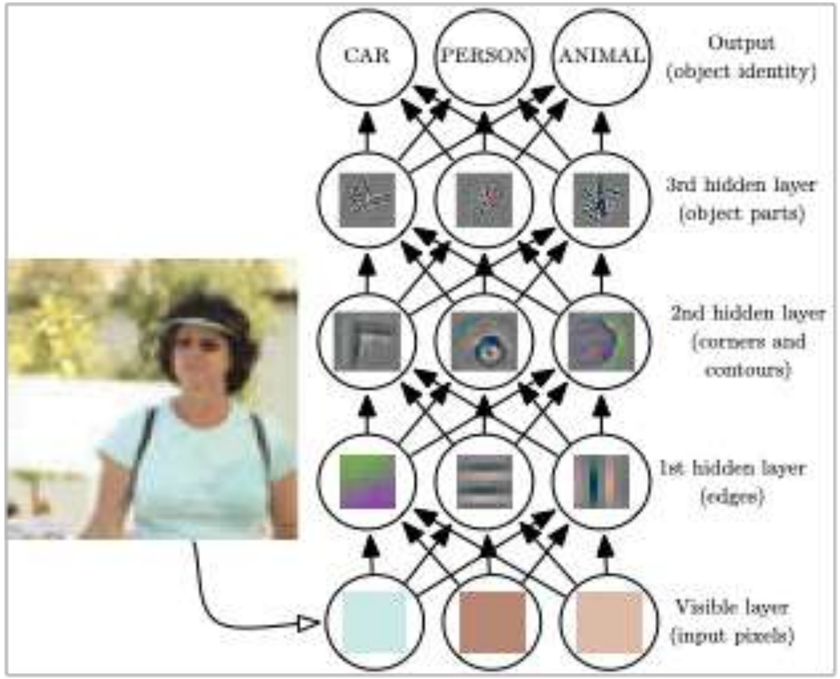


Figure 1.2

Computational Graphs

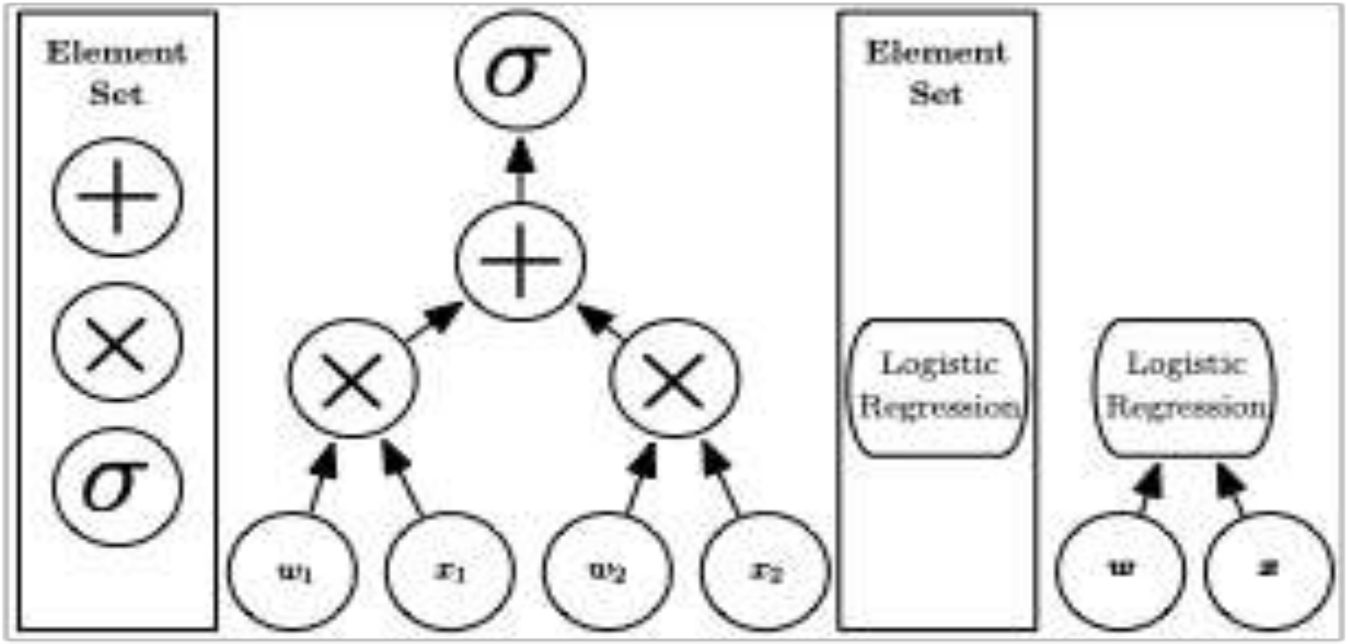


Figure 1.3

Machine Learning and AI

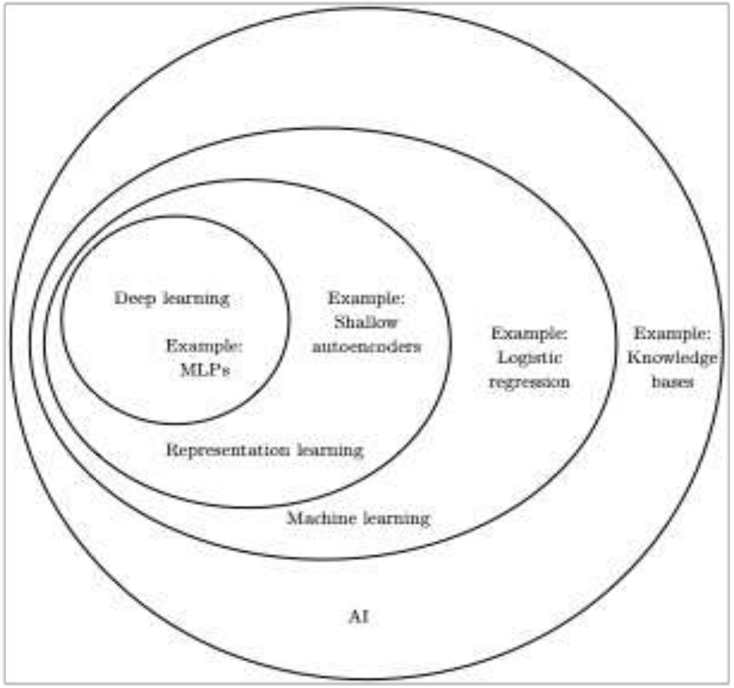


Figure 1.4

Historical Waves

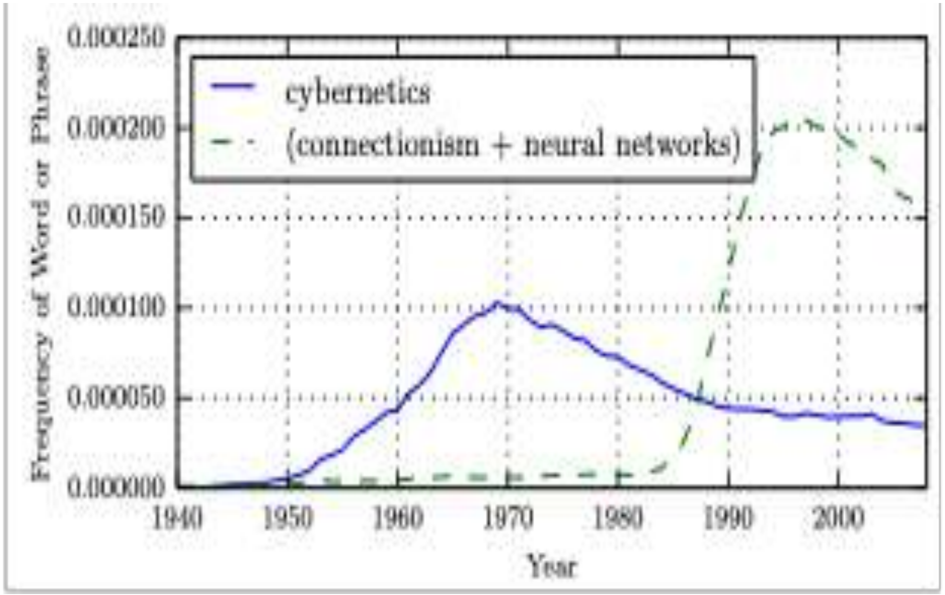


Figure 1.7

Historical Trends: Growing Datasets

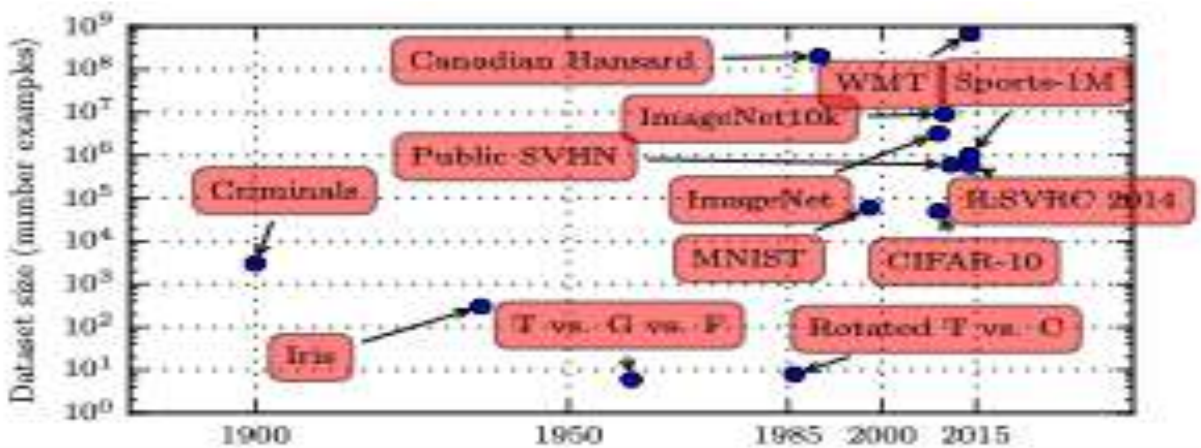


Figure 1.8

The MNIST Dataset

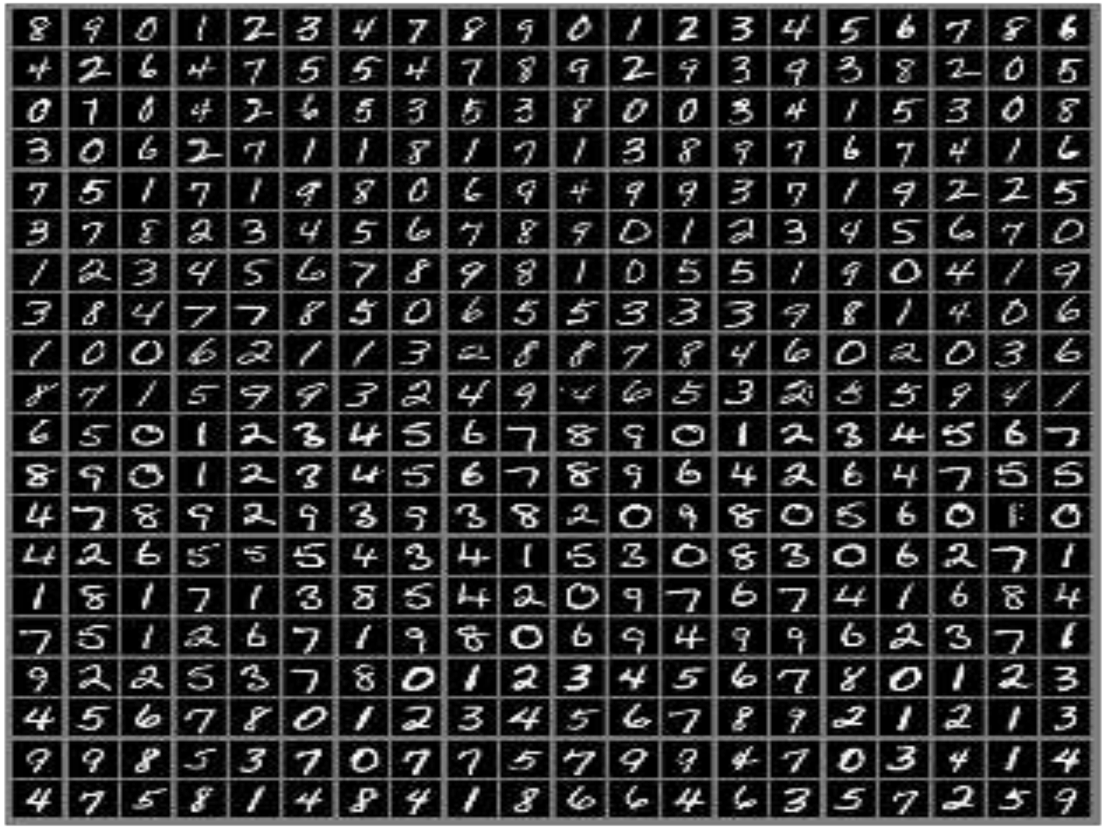


Figure 1.9

Connections per Neuron

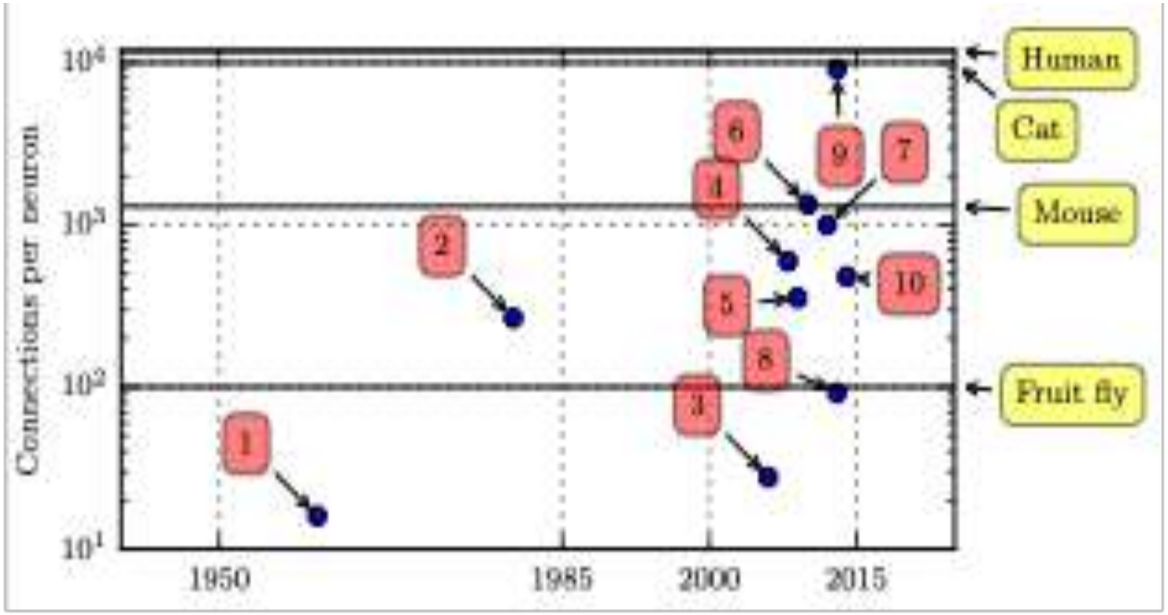


Figure 1.10

Number of Neurons

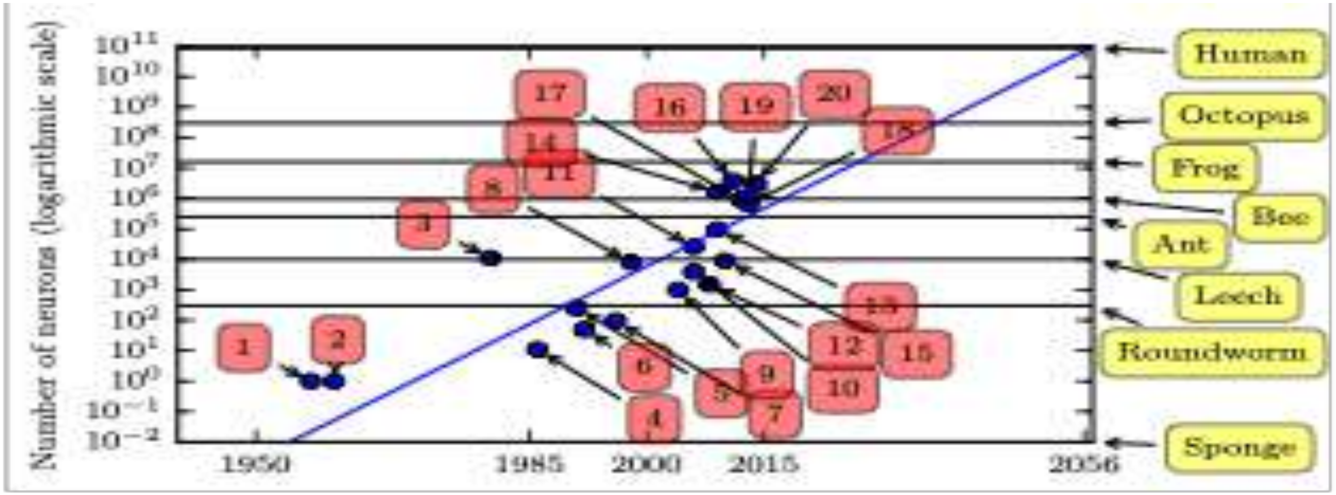


Figure 1.11

Solving Object Recognition

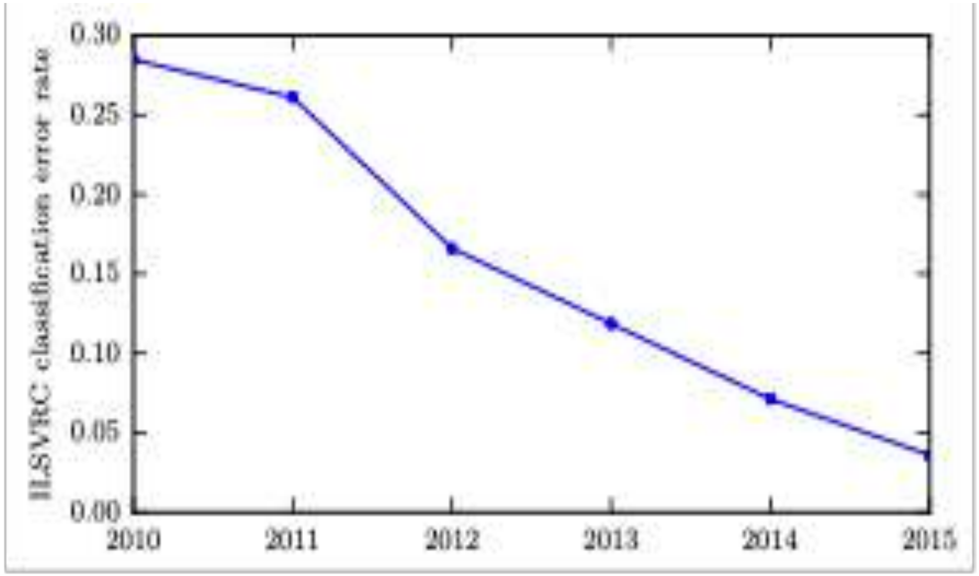


Figure 1.12

Numerical Computation for Deep Learning

Numerical concerns for implementations of deep learning algorithms

- Algorithms are often specified in terms of real numbers; real numbers cannot be implemented in a finite computer
    - Does the algorithm still work when implemented with a finite number of bits?
- Do small changes in the input to a function cause large changes to an output?
    - Rounding errors, noise, measurement errors can cause large changes
    - Iterative search for best input is difficult

Roadmap

- Iterative Optimization
- Rounding error, underflow, overflow

Iterative Optimization

- Gradient descent
- Curvature
- Constrained optimization

Gradient Descent

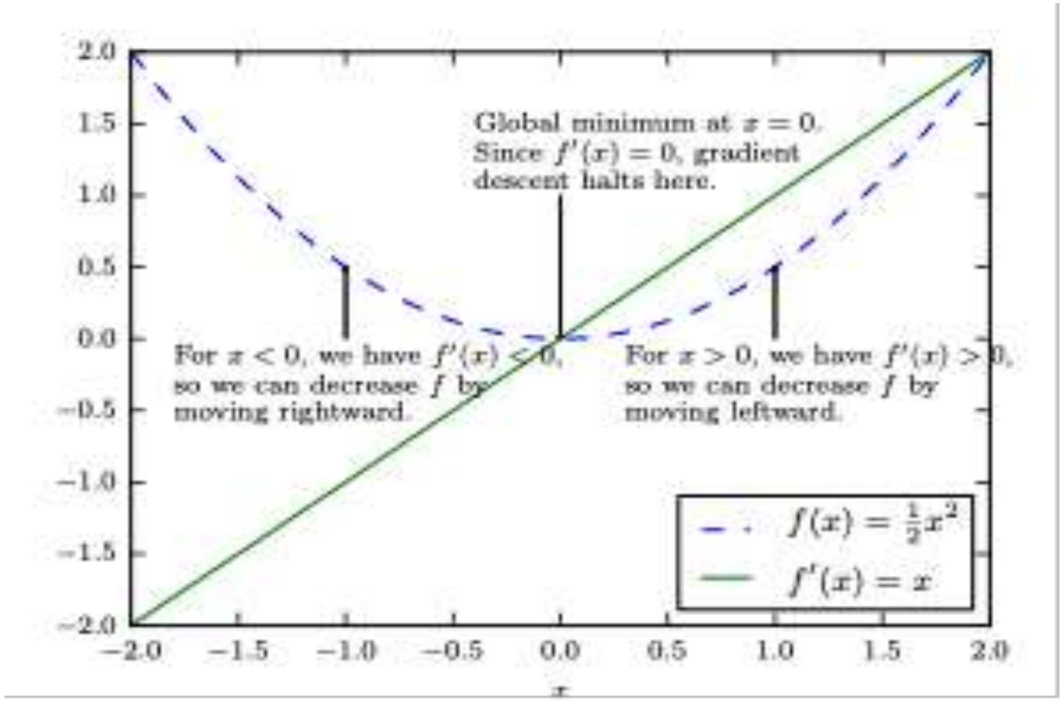


Figure 4.1

Approximate Optimization

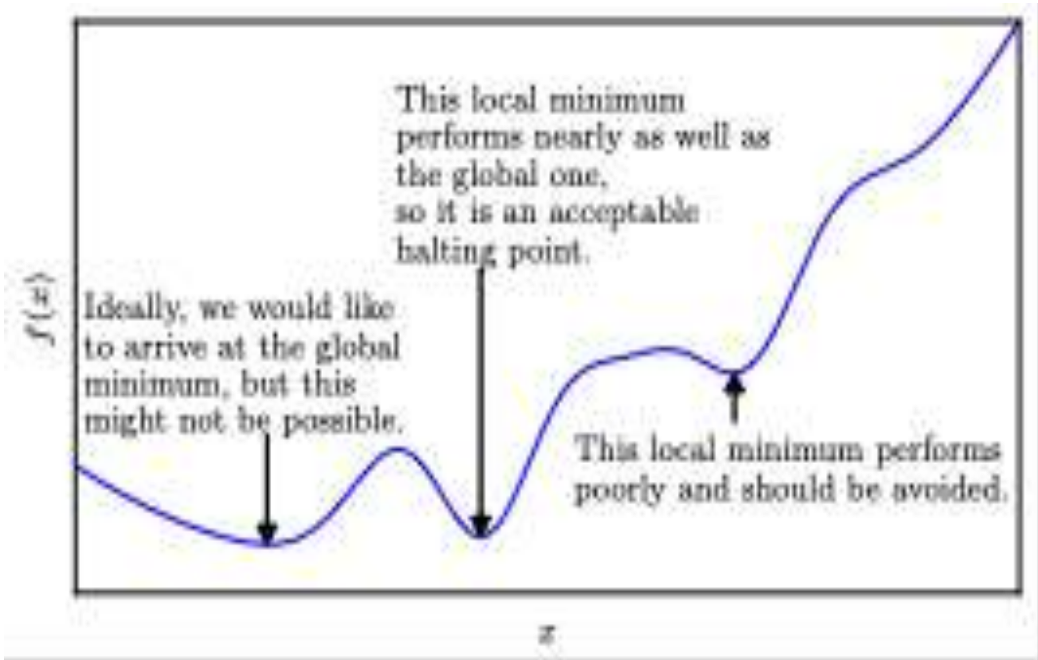
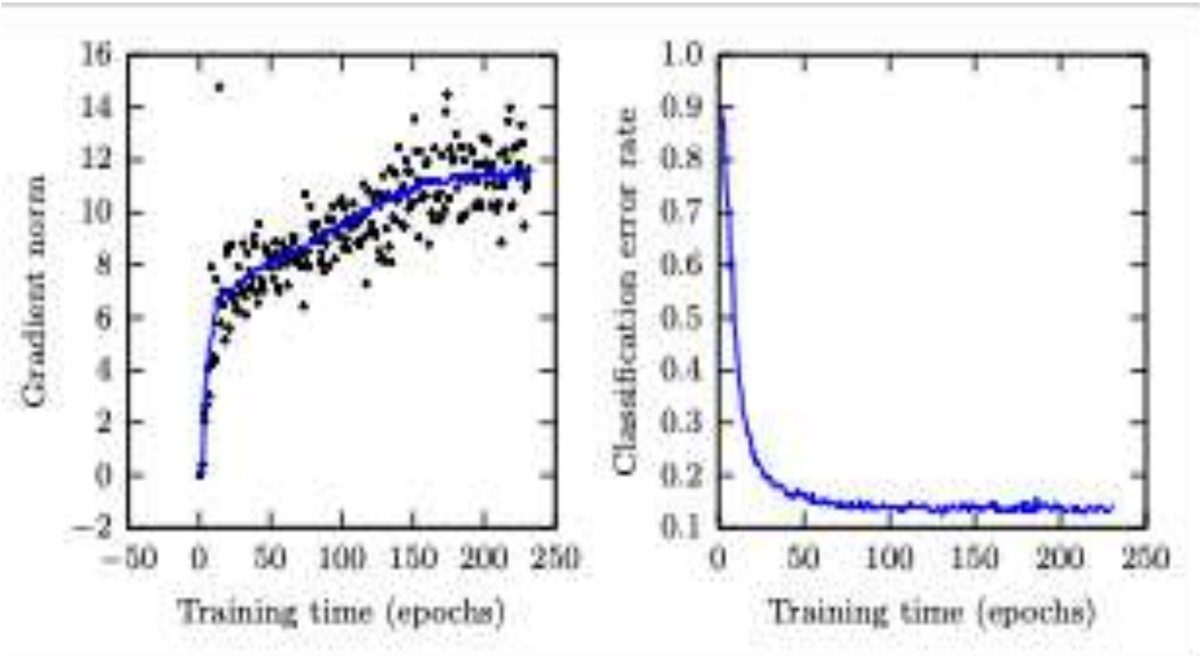


Figure 4.3

We usually don’t even reach a local minimum



Iterative Optimization

- Gradient descent
- Curvature
- Constrained optimization

Critical Points

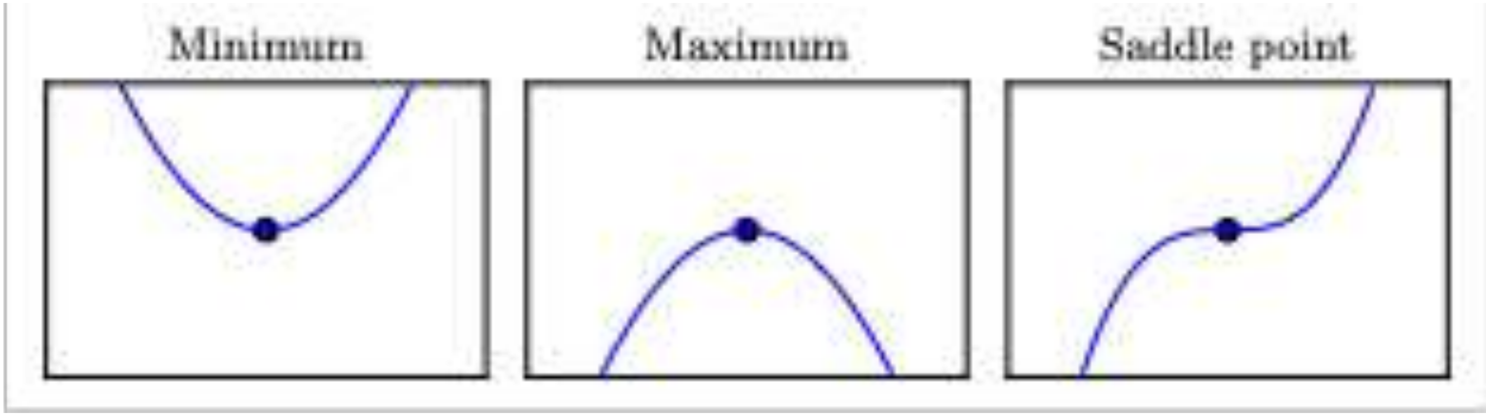


Figure 4.2

Saddle Points

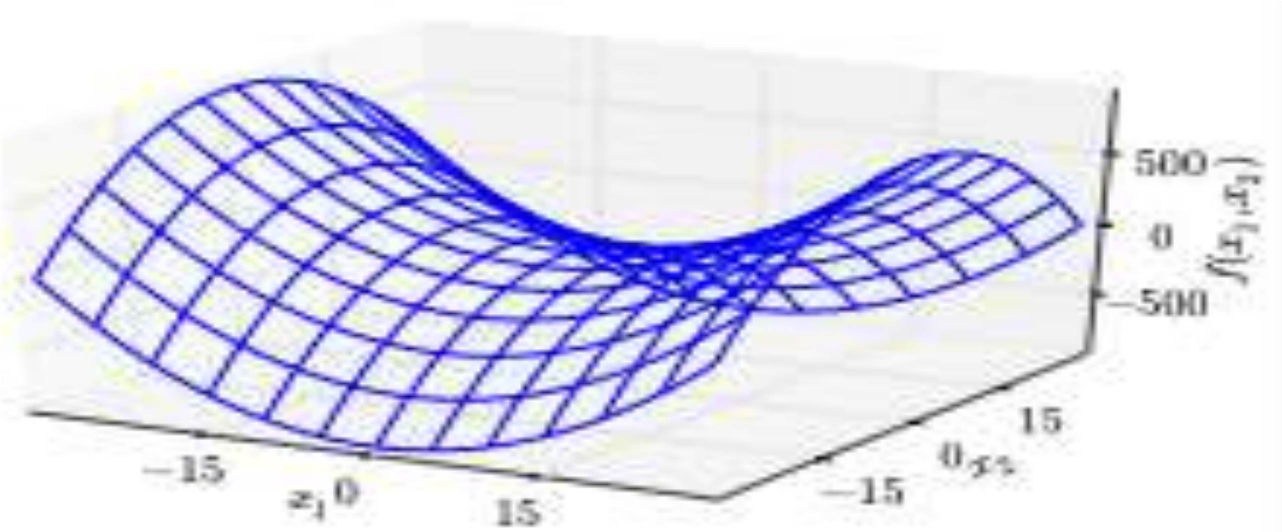


Figure 4.5

Saddle points attract Newton's method

(Gradient descent escapes, see Appendix C of "Qualitatively Characterizing Neural Network Optimization Problems")

Curvature

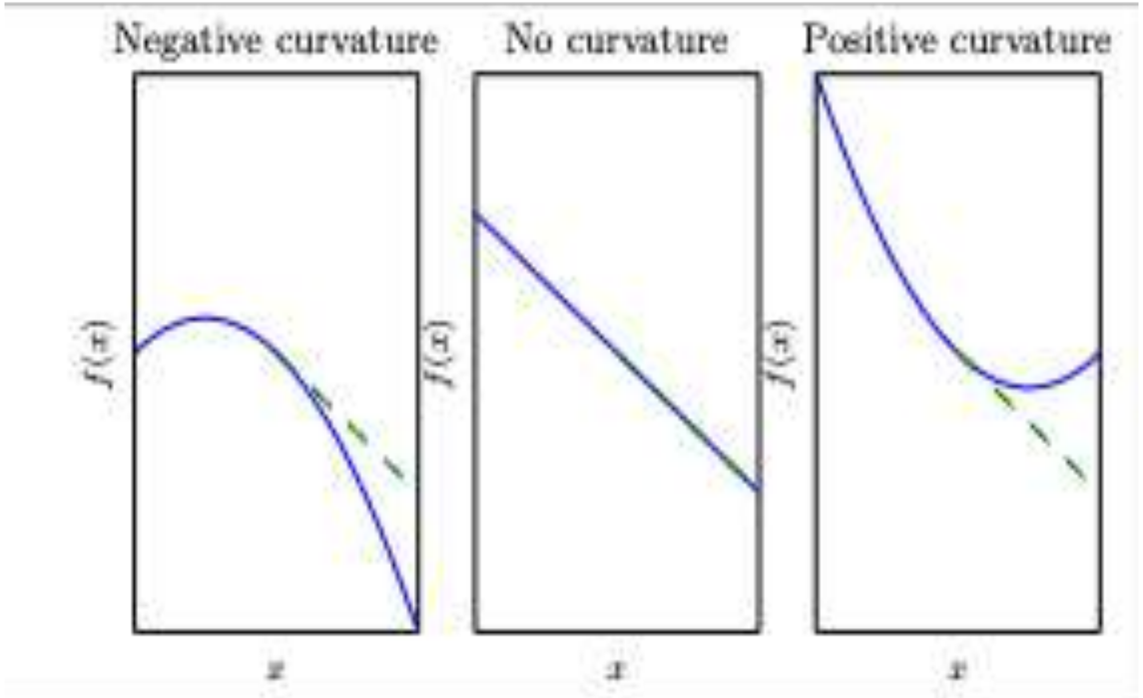
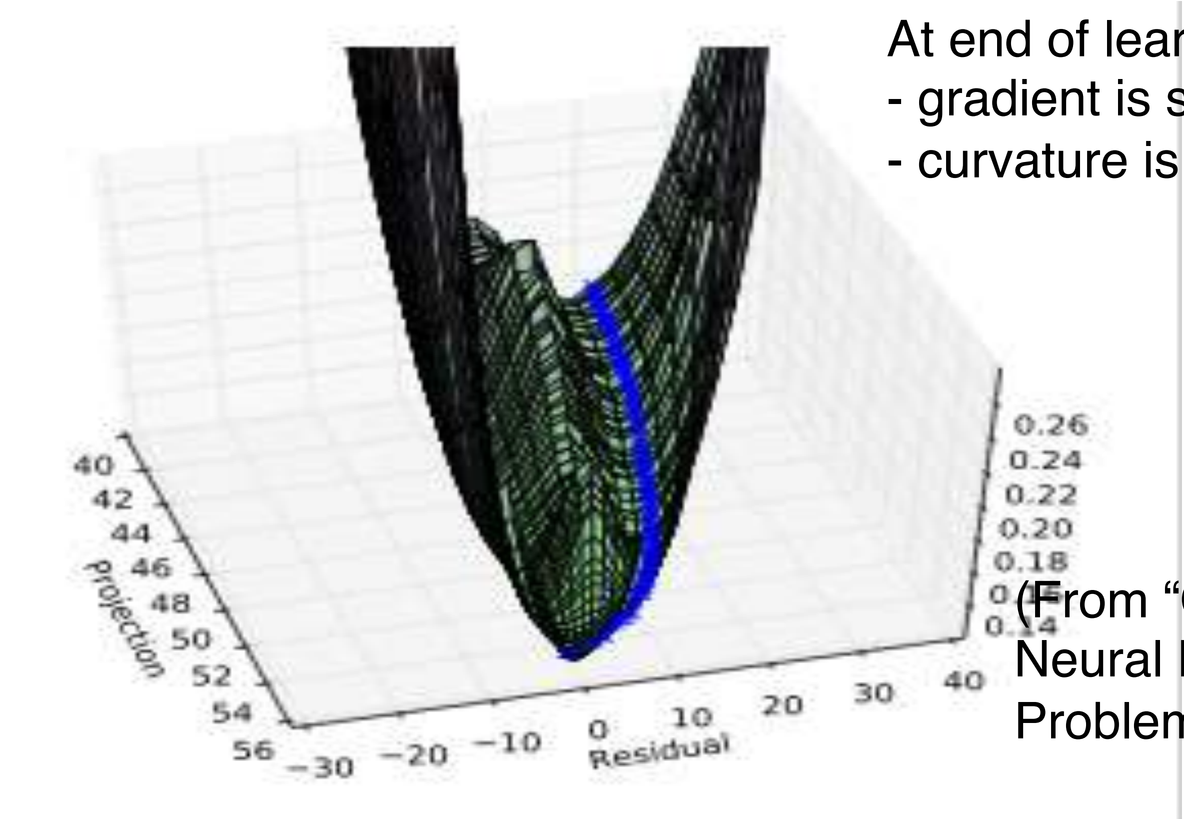


Figure 4.4

Neural net visualization



At end of learning:
- gradient is still large
- curvature is huge

(From "Qualitatively Characterizing Neural Network Optimization Problems")

Iterative Optimization

- Gradient descent
- Curvature
- Constrained optimization

Roadmap

- **Iterative Optimization**
- **Rounding error, underflow, overflow**

Numerical Precision: A deep learning super skill

- Often deep learning algorithms “sort of work”
  - Loss goes down, accuracy gets within a few percentage points of state-of-the-art
  - No “bugs” per se
- Often deep learning algorithms “explode” (NaNs, large values)
- Culprit is often loss of numerical precision

Rounding and truncation errors

- In a digital computer, we use **`float32`** or similar schemes to represent real numbers
- A real number *x* is rounded to **`x + delta`** for some small delta
- Overflow: large *x* replaced by **`inf`**
- Underflow: small *x* replaced by **`0`**

Bug hunting strategies

- If you increase your learning rate and the loss *gets stuck*, you are probably rounding your gradient to zero somewhere: maybe computing cross-entropy using probabilities instead of logits
- For correctly implemented loss, too high of learning rate should usually cause *explosion*

Machine Learning Basics

Linear Regression

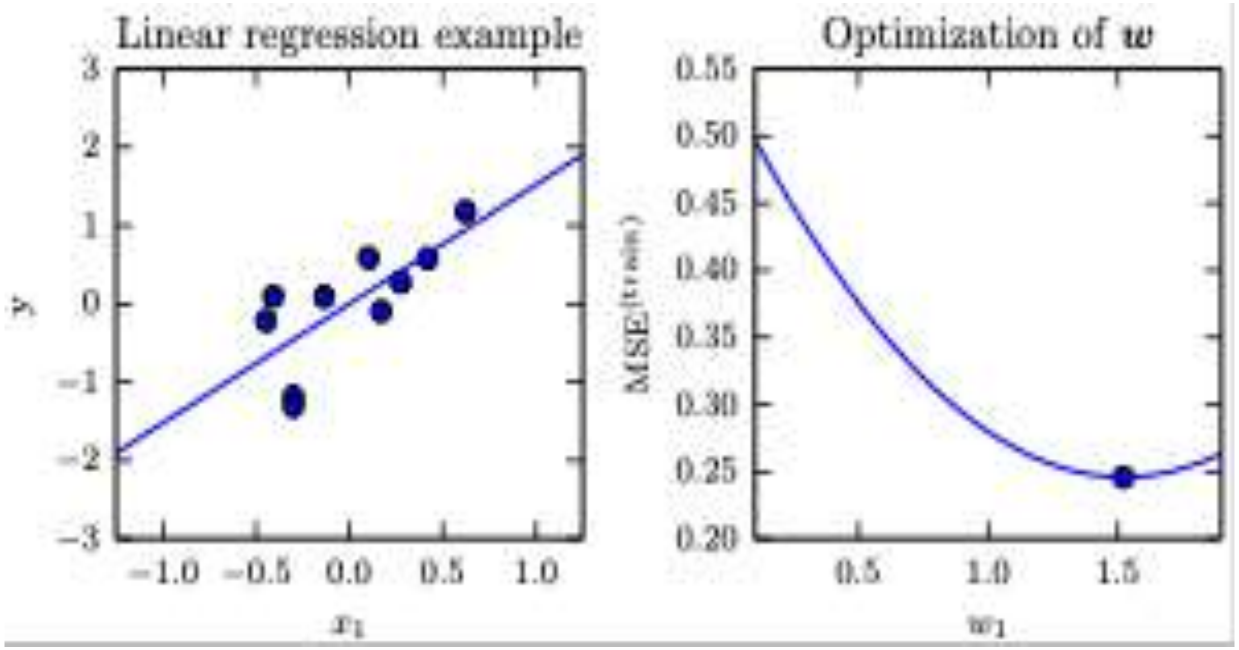


Figure 5.1

Underfitting and Overfitting in Polynomial Estimation

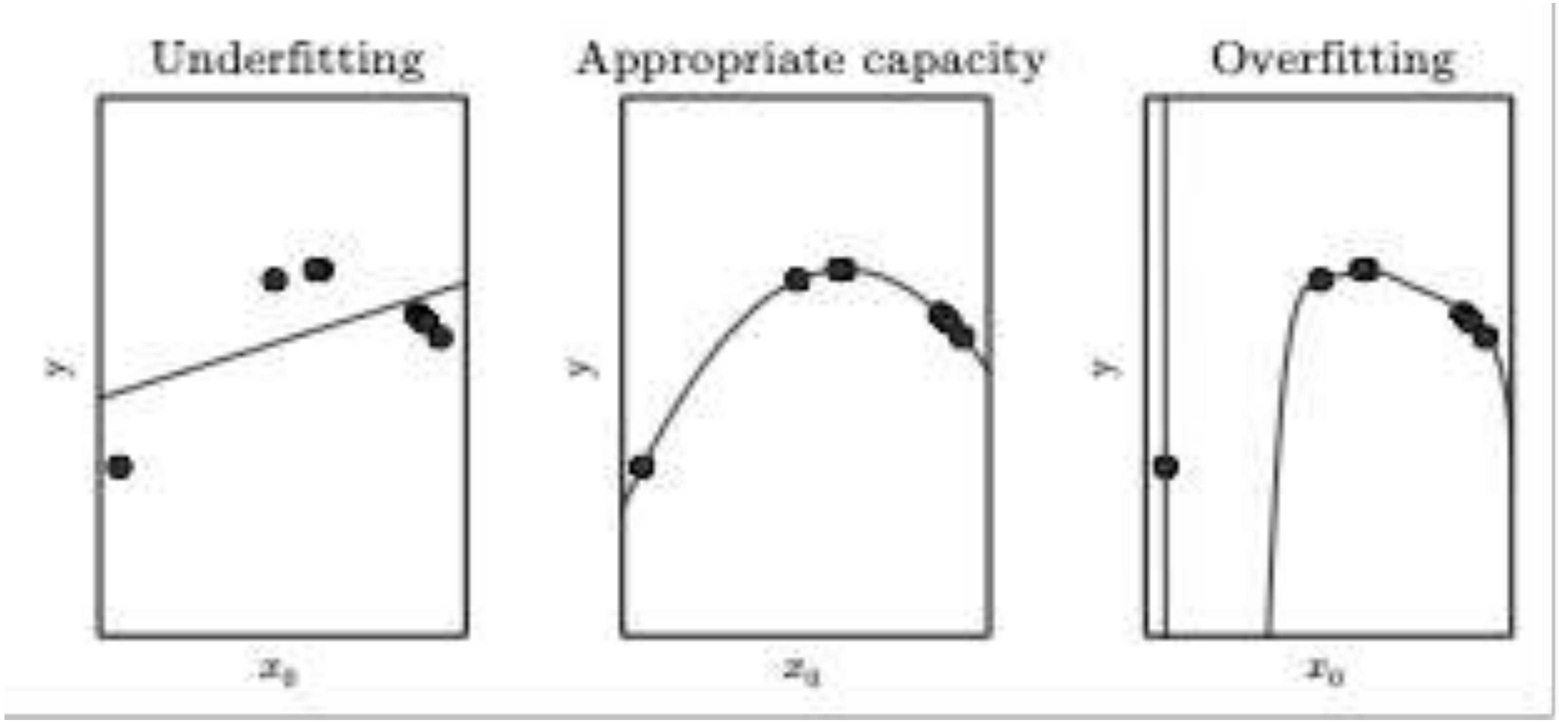


Figure 5.2

Generalization and Capacity

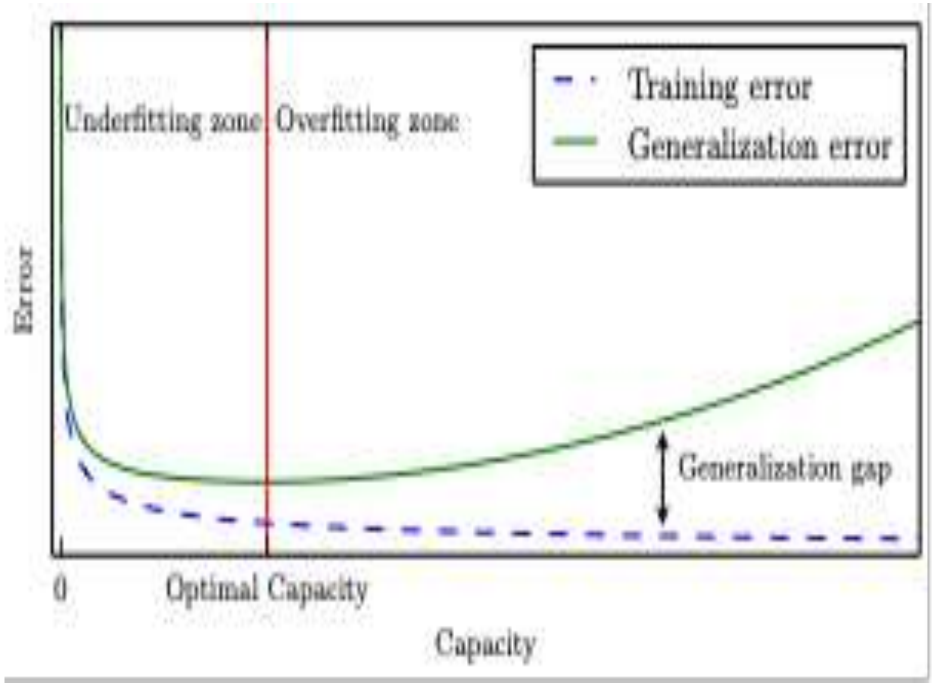
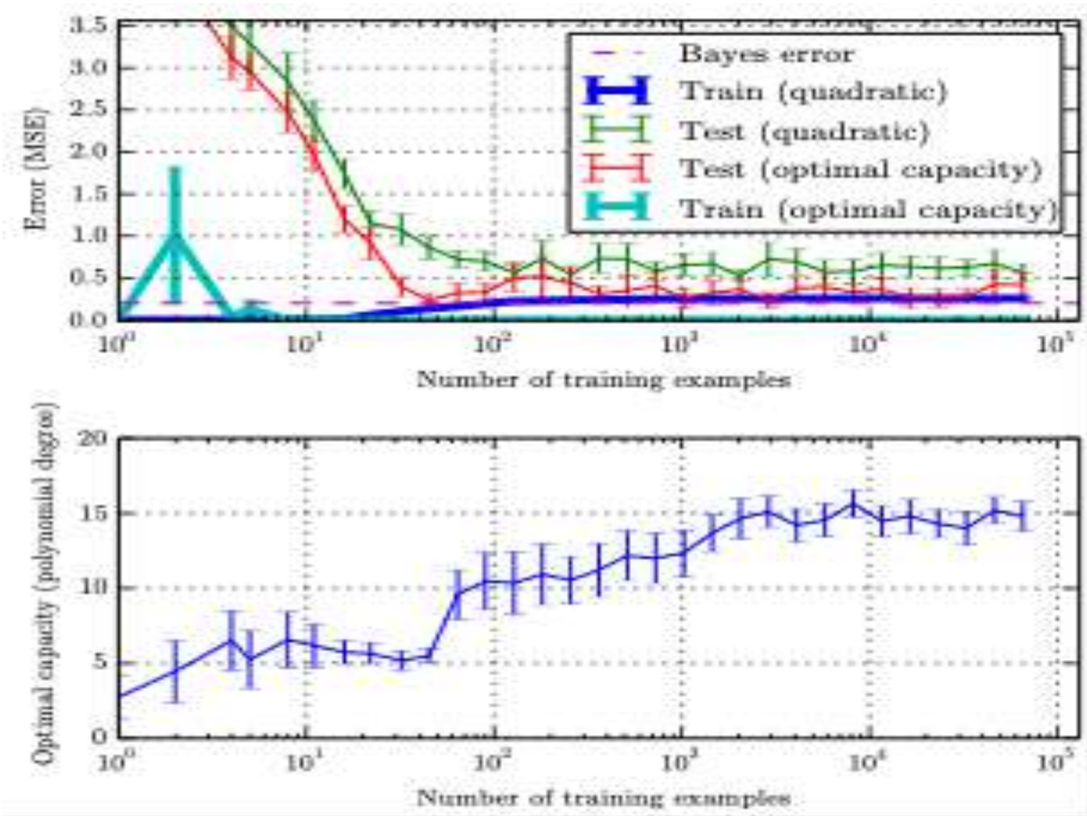


Figure 5.3

Training Set Size

Figure 5.4



Weight Decay

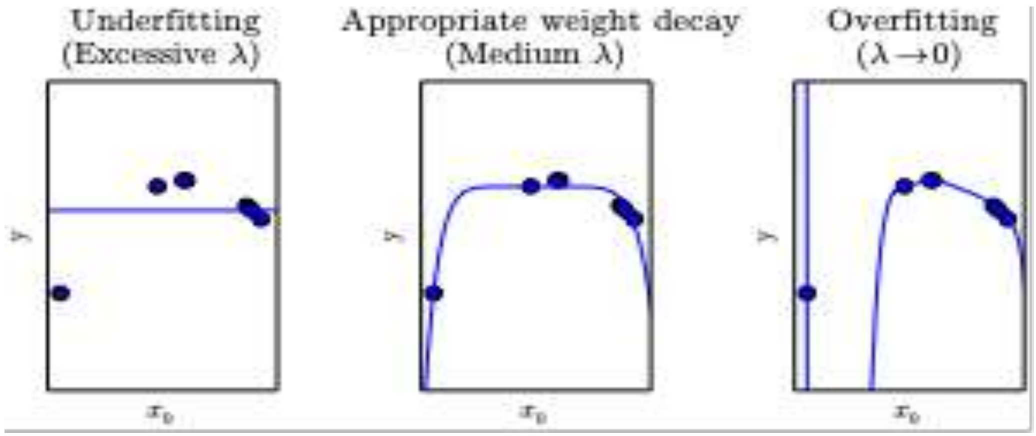


Figure 5.5

Bias and Variance

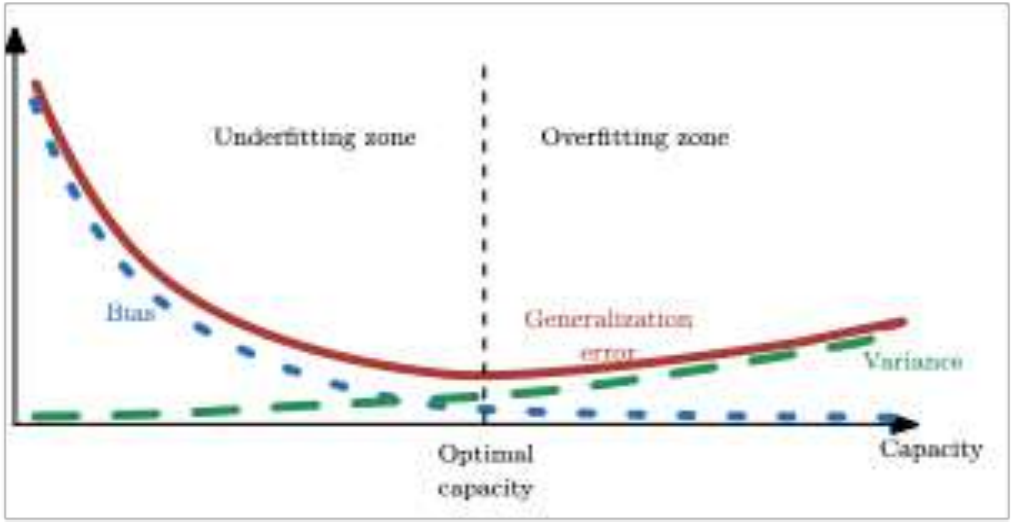
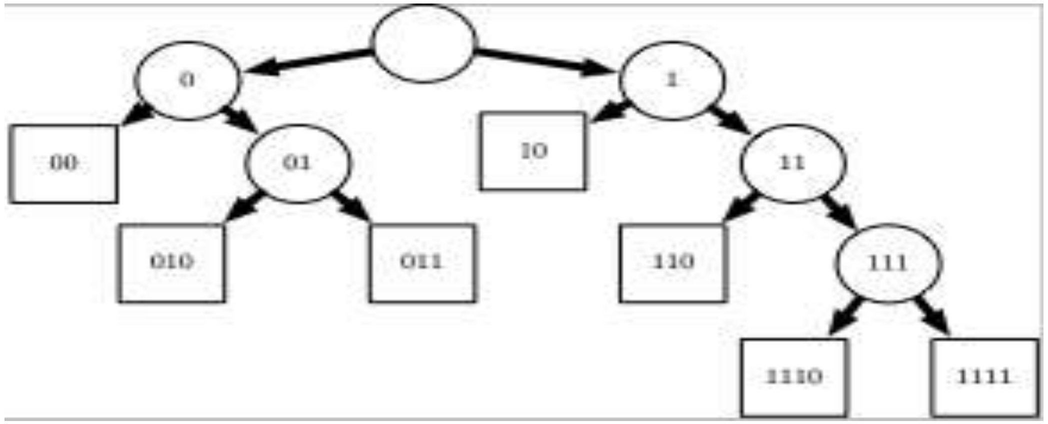


Figure 5.6

Decision Trees



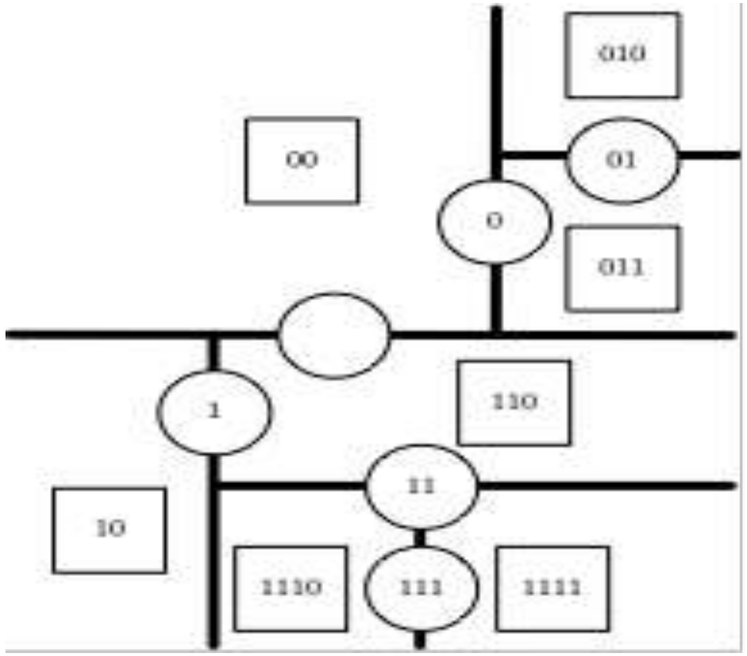


Figure 5.7

Principal Components Analysis

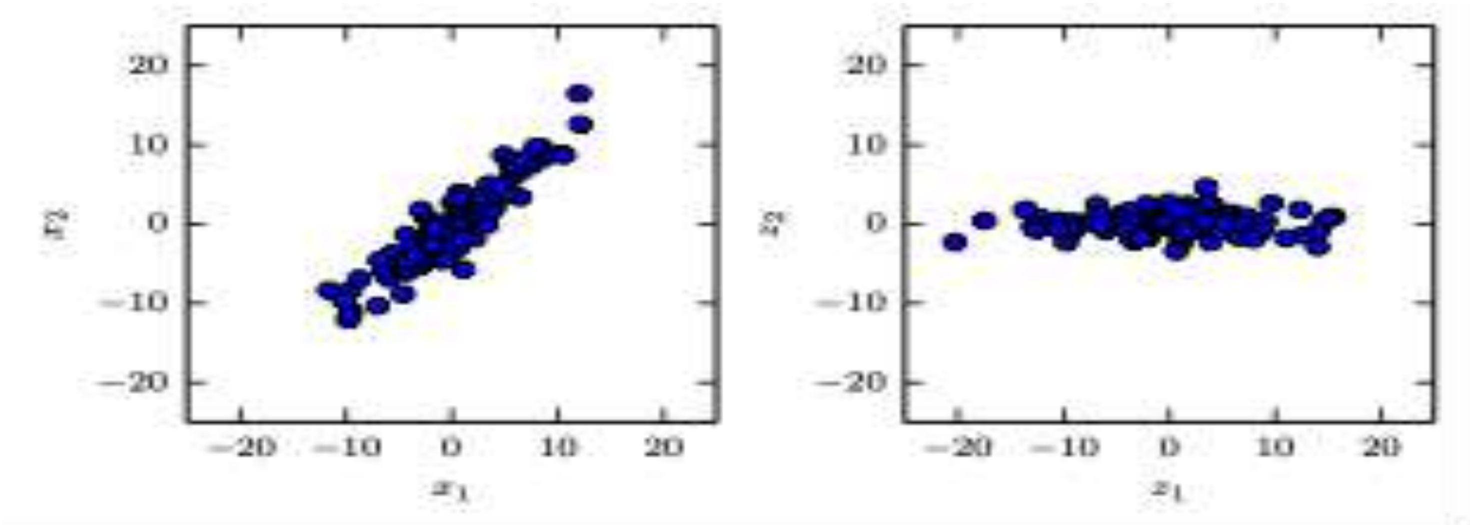


Figure 5.8

Curse of Dimensionality

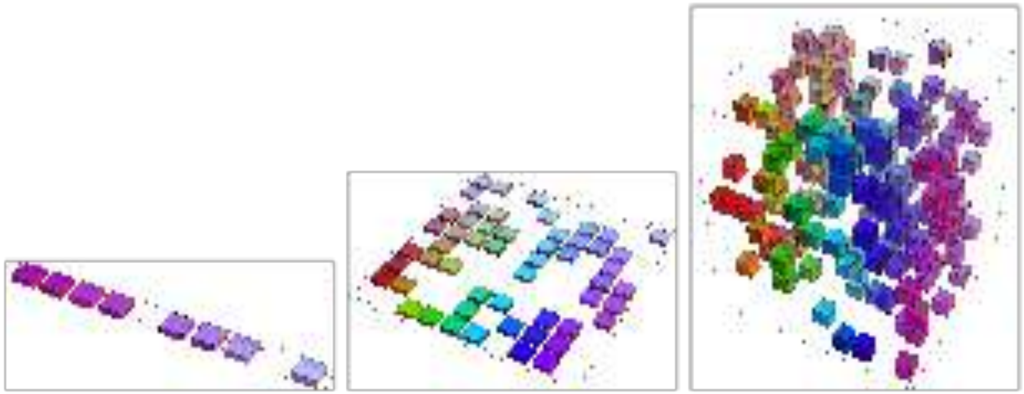


Figure 5.9

Nearest Neighbor

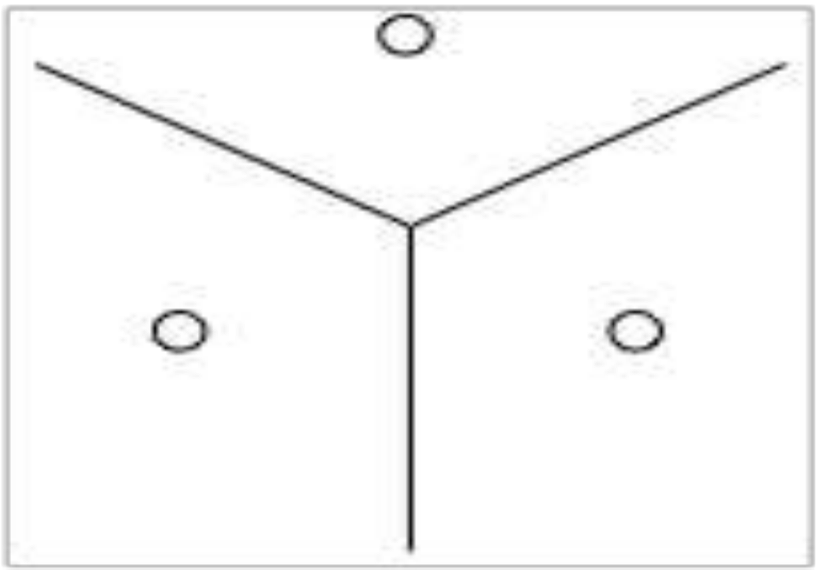


Figure 5.10

Manifold Learning

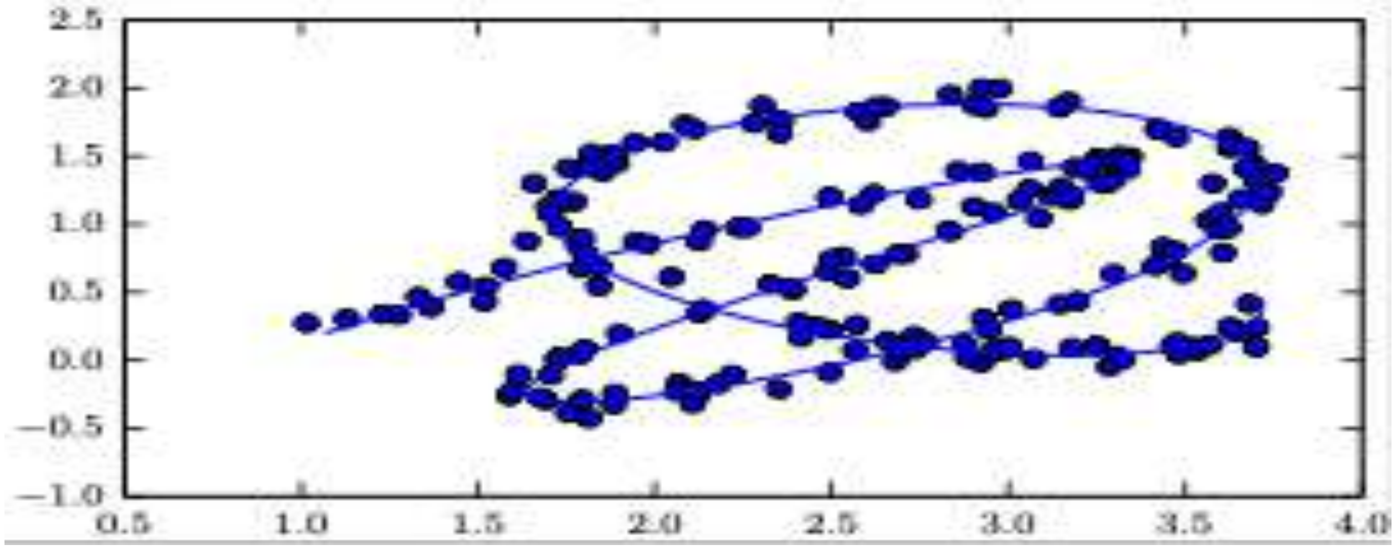


Figure 5.11

Convolutional Networks

Convolutional Networks

- Scale up neural networks to process very large images / video sequences
  - Sparse connections
  - Parameter sharing
- Automatically generalize across spatial translations of inputs
- Applicable to any input that is laid out on a grid (1-D, 2-D, 3-D, …)

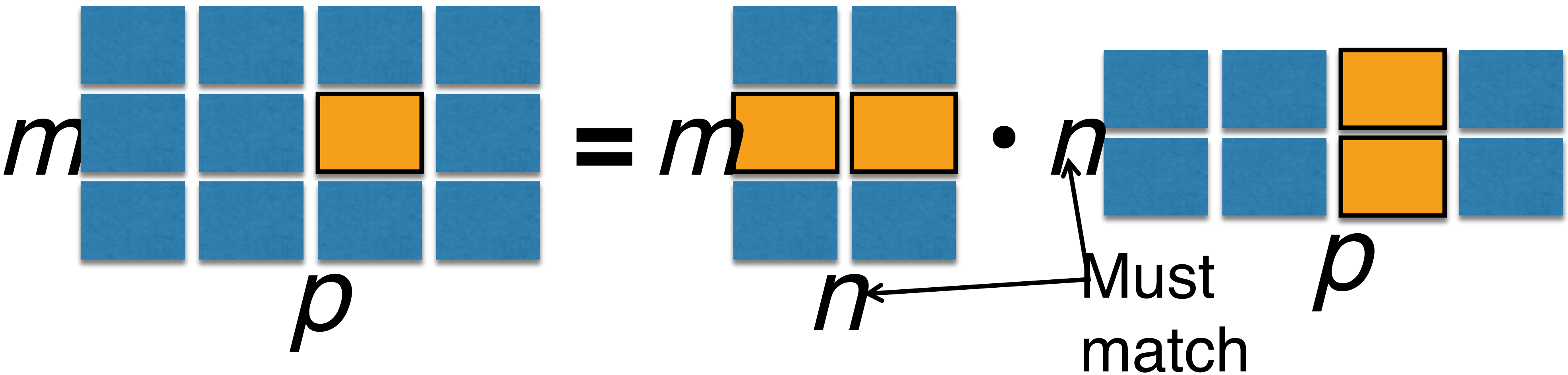
Key Idea

- Replace matrix multiplication in neural nets with convolution
- Everything else stays the same
  - Maximum likelihood
  - Back-propagation
  - etc.

Matrix (Dot) Product

$$\boldsymbol{C} = \boldsymbol{A}\boldsymbol{B}. \tag{2.4}$$

$$C_{i,j} = \sum_k A_{i,k} B_{k,j}. \tag{2.5}$$



Edge Detection by Convolution

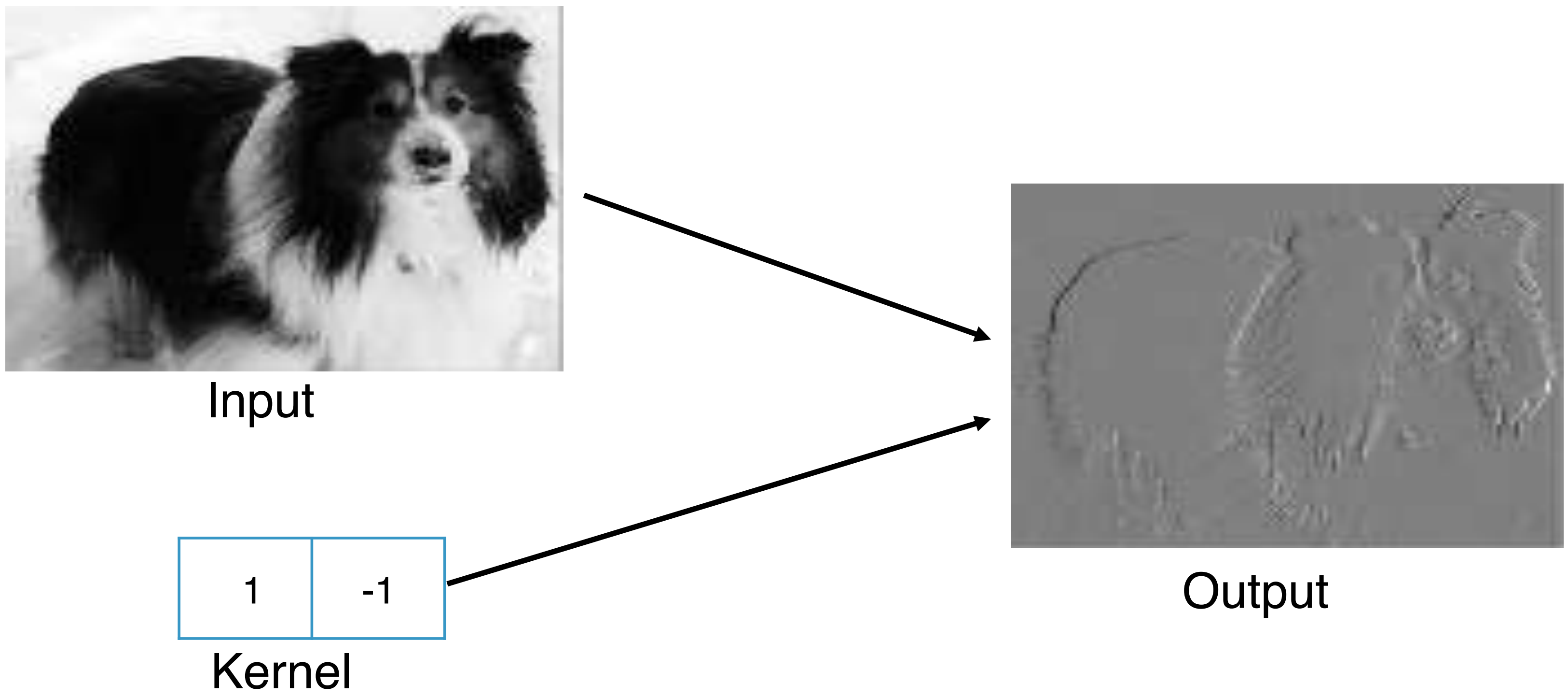
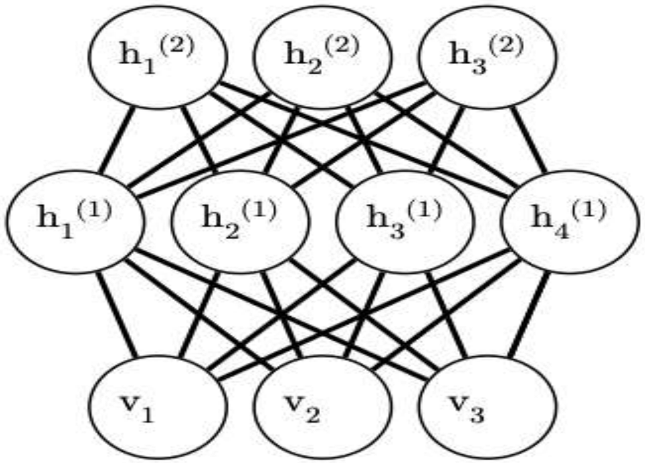


Figure 9.6

Practical Methodology

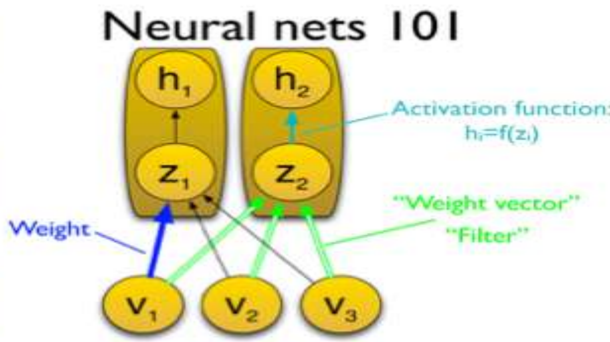
What drives success in ML?

Arcane knowledge of dozens of obscure algorithms?

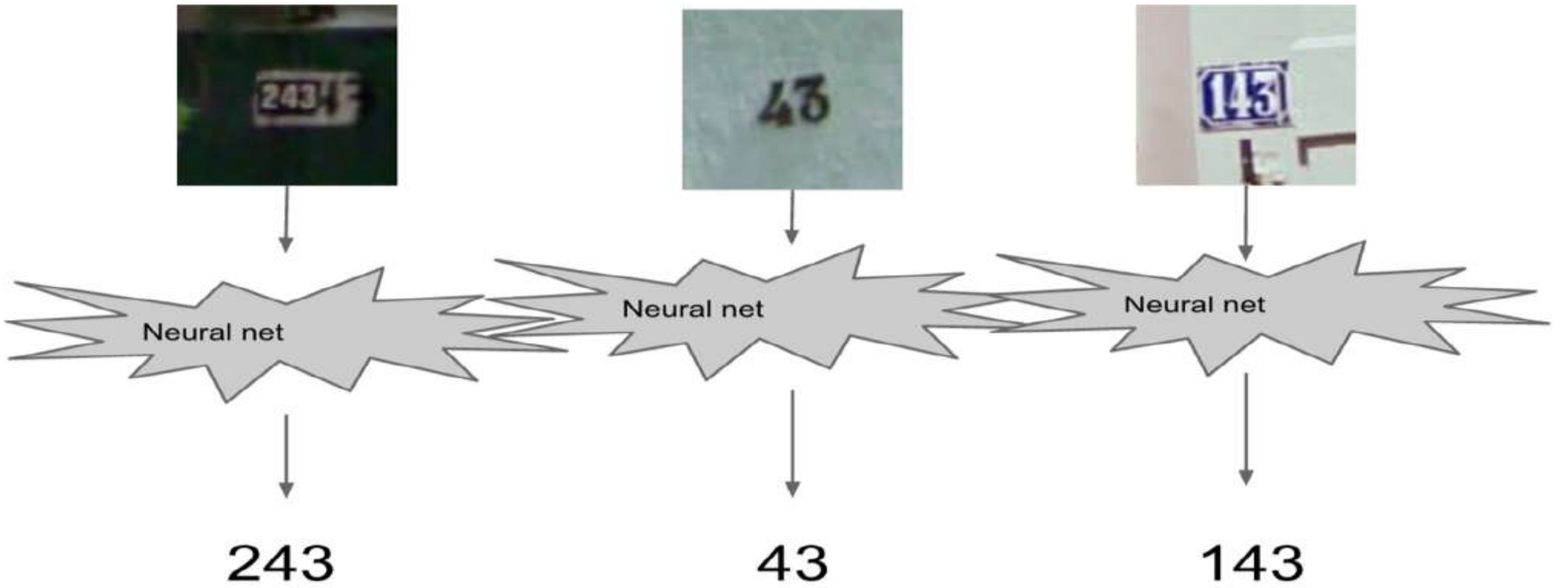


Mountains of data?

Knowing how to apply 3-4 standard techniques?



Example: Street View Address Number Transcription



(Goodfellow et al, 2014)

Three Step Process

- Use needs to define metric-based goals
- Build an end-to-end system
- Data-driven refinement

Identify Needs

- High accuracy or low accuracy?
- Surgery robot: high accuracy
- Celebrity look-a-like app: low accuracy

Choose Metrics

- Accuracy? (% of examples correct)
- Coverage? (% of examples processed)
- Precision? (% of detections that are right)
- Recall? (% of objects detected)
- Amount of error? (For regression problems)

End-to-end System

- Get up and running ASAP
- Build the simplest viable system first
- What baseline to start with though?
  - Copy state-of-the-art from related publication

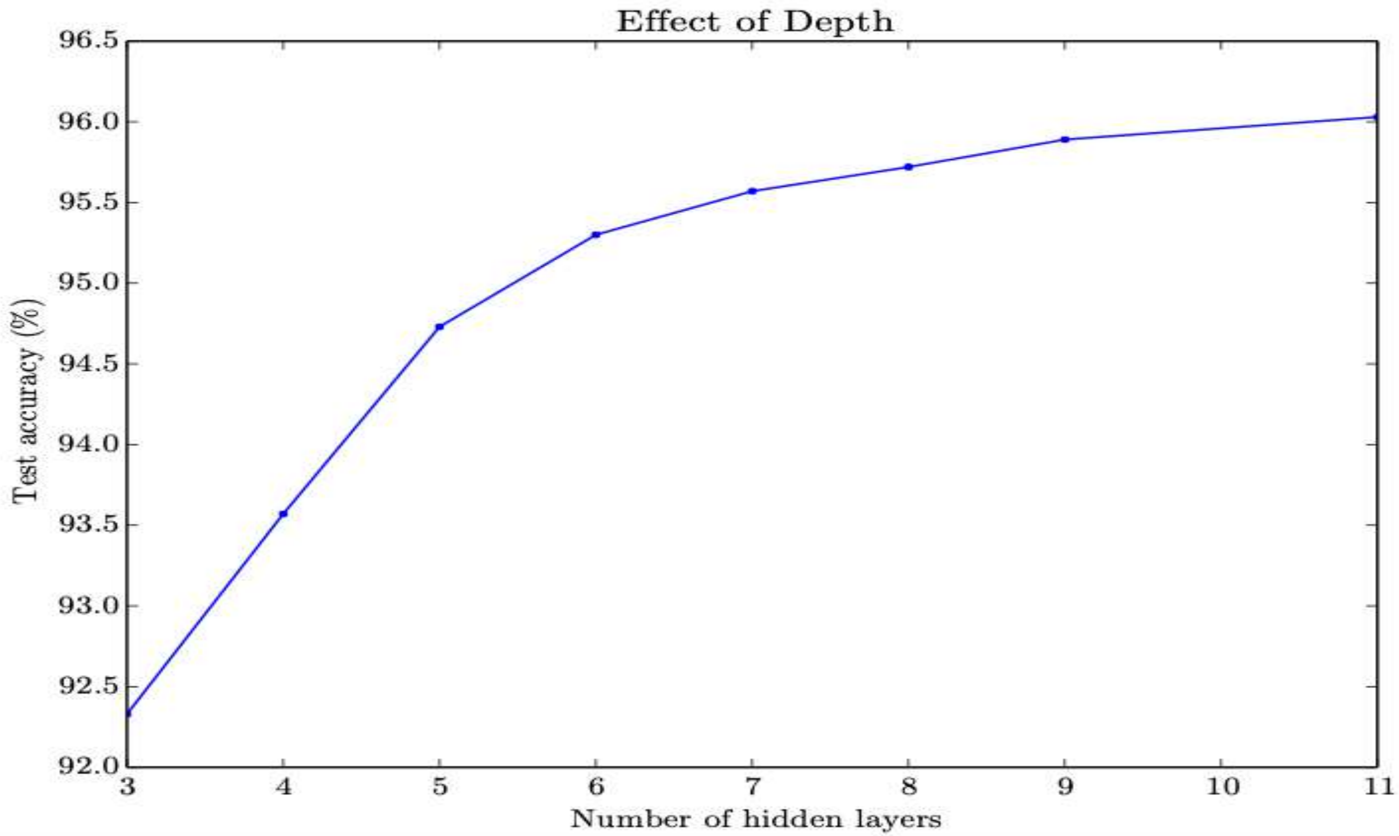
Deep or Not?

- Lots of noise, little structure -> not deep
- Little noise, complex structure -> deep
- Good shallow baseline:
  - *Use what you know*
  - Logistic regression, SVM, boosted tree are all good

Choosing Architecture Family

- No structure -> fully connected
- Spatial structure -> convolutional
- Sequential structure -> recurrent

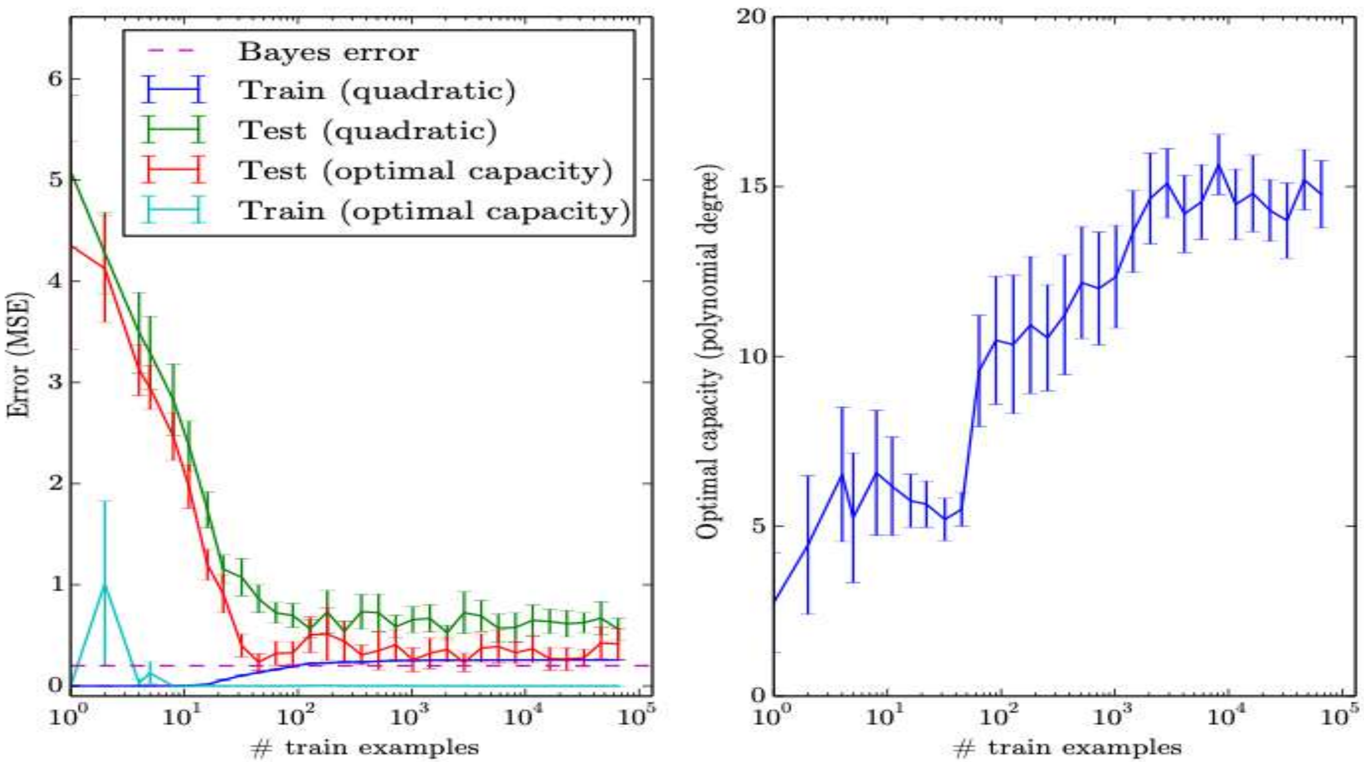
Increasing Depth



High Test Error

- Add dataset augmentation
- Add dropout
- Collect more data

Increasing Training Set Size



Tuning the Learning Rate

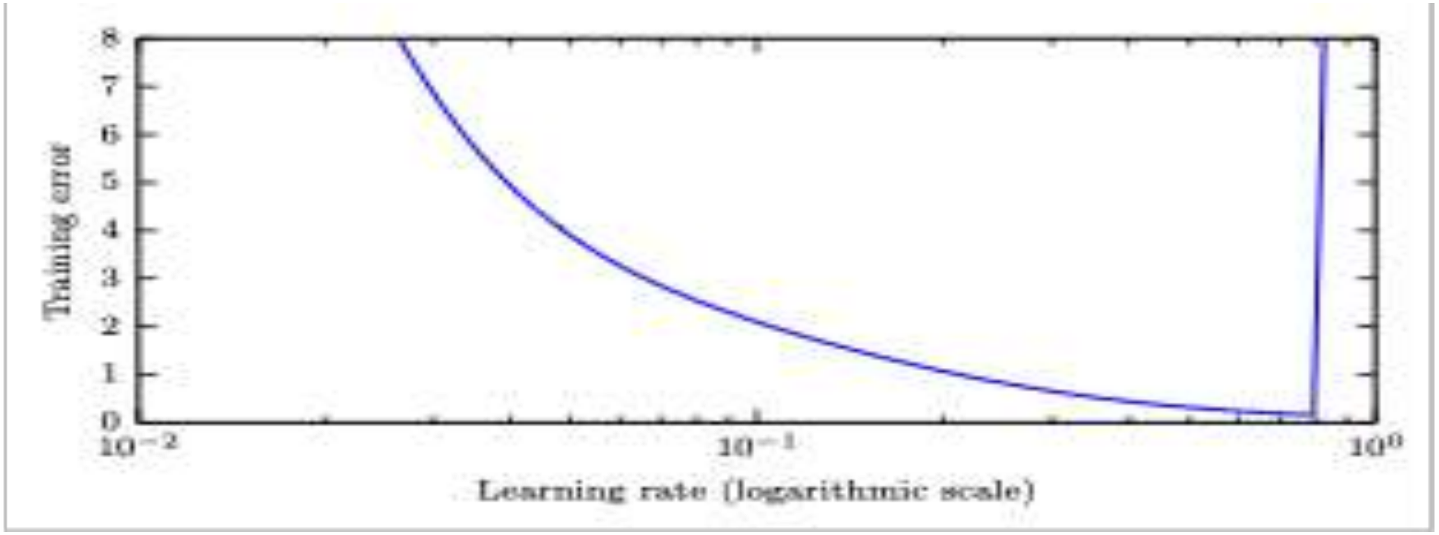


Figure 11.1

Monte Carlo Methods

Roadmap

- Basics of Monte Carlo methods
- Importance Sampling
- Markov Chains

Randomized Algorithms

	Las Vegas	**Monte Carlo**
Type of Answer	Exact	Random amount of error
Runtime	Random (until answer found)	Chosen by user (longer runtime gives lesss error)

Estimating sums / integrals with samples

$$s = \sum_{\boldsymbol{x}} p(\boldsymbol{x}) f(\boldsymbol{x}) = E_p[f(\mathbf{x})] \tag{17.1}$$

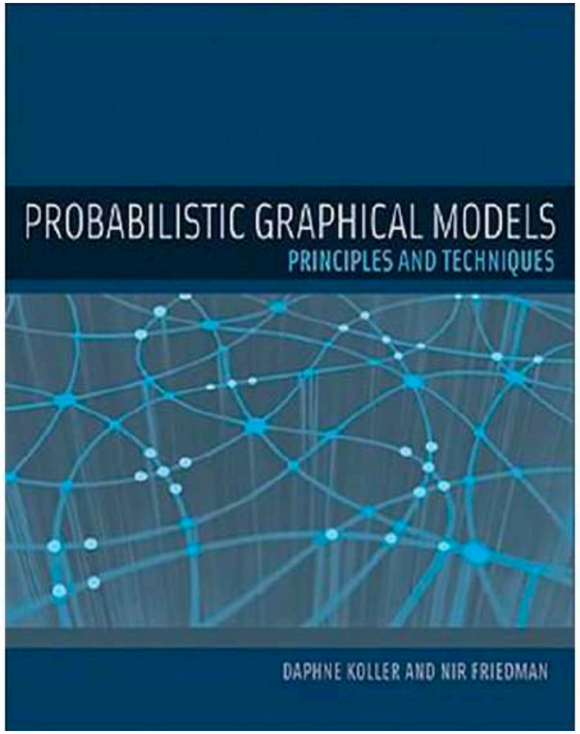
$$s = \int p(\boldsymbol{x}) f(\boldsymbol{x}) d\boldsymbol{x} = E_p[f(\mathbf{x})] \tag{17.2}$$

$$\hat{s}_n = \frac{1}{n} \sum_{i=1}^{n} f(\boldsymbol{x}^{(i)}). \tag{17.3}$$

Justification

- **Unbiased:**
  - **The expected value for finite *n* is equal to the correct value**
  - **The value for any specific *n* samples will have random error, but the errors for different sample sets cancel out**
- **Low variance:**
  - **Variance is *O*(1/*n*)**
  - **For very large *n*, the error converges "almost surely" to 0**

For more information…



Object Categorization

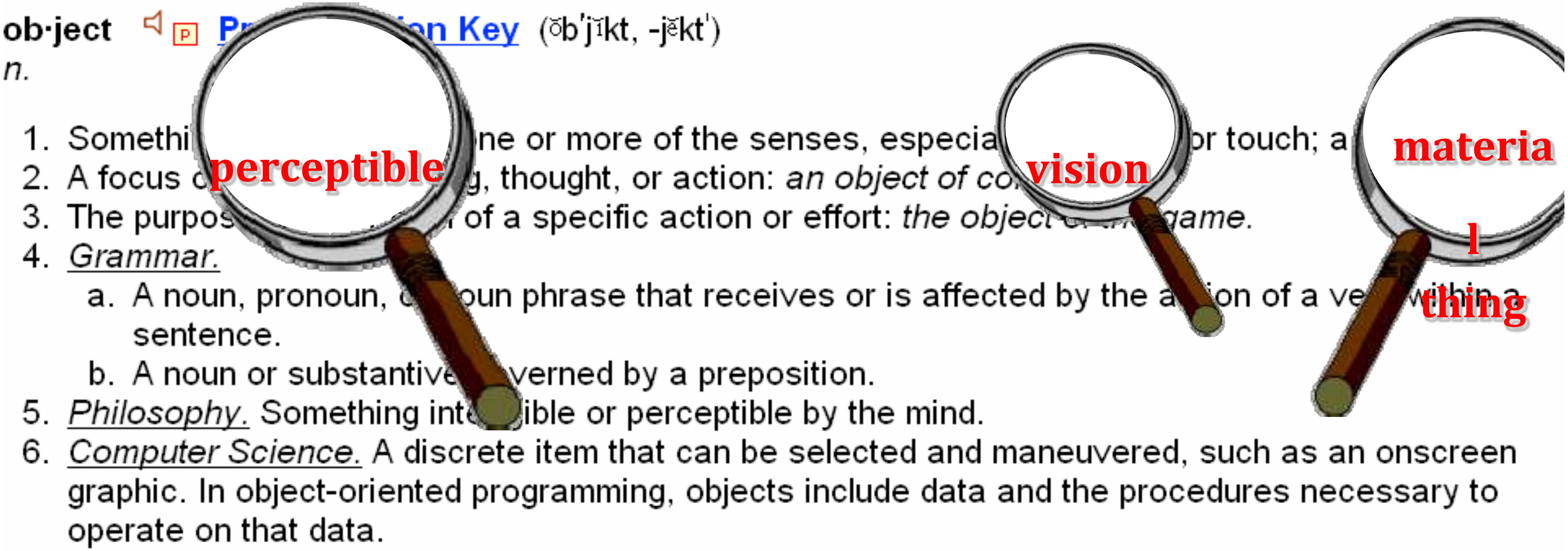
Lecture slides adapted from "Object Categorization an Overview and Two Models"
Fei Fei Li

Agenda

- Introduction to "Object Categorization"

- "Bag of Words" models

- Part-based models


ob·ject Pr… …on Key (ŏb'jĭkt, -jĕkt')
n.
1. Somethi… …ne or more of the senses, especia… …or touch; a
2. A focus o… …g, thought, or action: an object of co…
3. The purpos… …of a specific action or effort: the object … …game.
4. Grammar.
a. A noun, pronoun, o… …oun phrase that receives or is affected by the a… …ion of a ve… within a sentence.
b. A noun or substantive …verned by a preposition.
5. Philosophy. Something inte… …ible or perceptible by the mind.
6. Computer Science. A discrete item that can be selected and maneuvered, such as an onscreen graphic. In object-oriented programming, objects include data and the procedures necessary to operate on that data.
perceptible
vision
materia
l
thing

Plato said…

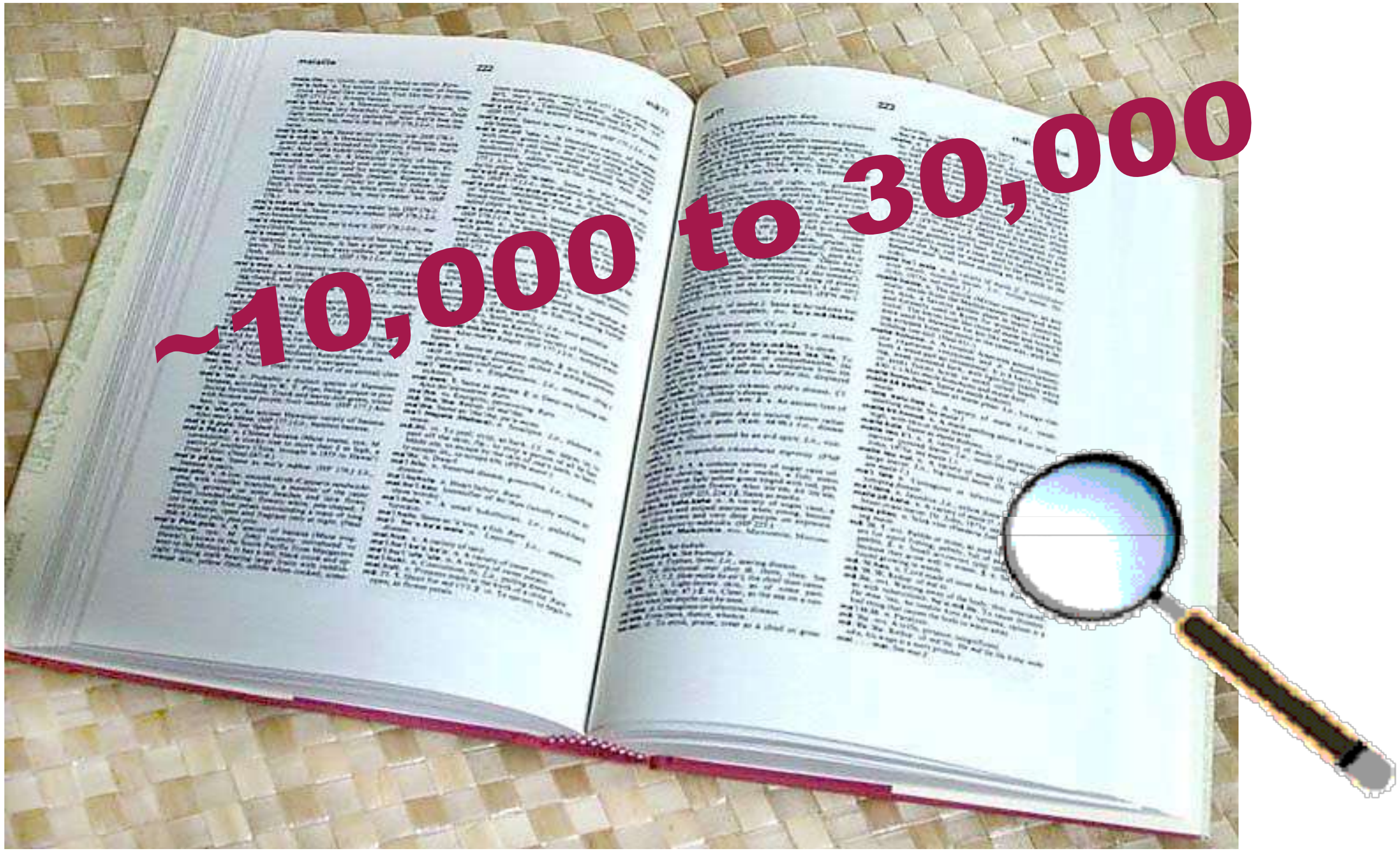
Ordinary objects are classified together if they `participate' in the same abstract Form, such as the Form of a Human or the Form of Quartz.

Forms are proper subjects of philosophical investigation, for they have the highest degree of reality.

Ordinary objects, such as humans, trees, and stones, have a lower degree of reality than the Forms.

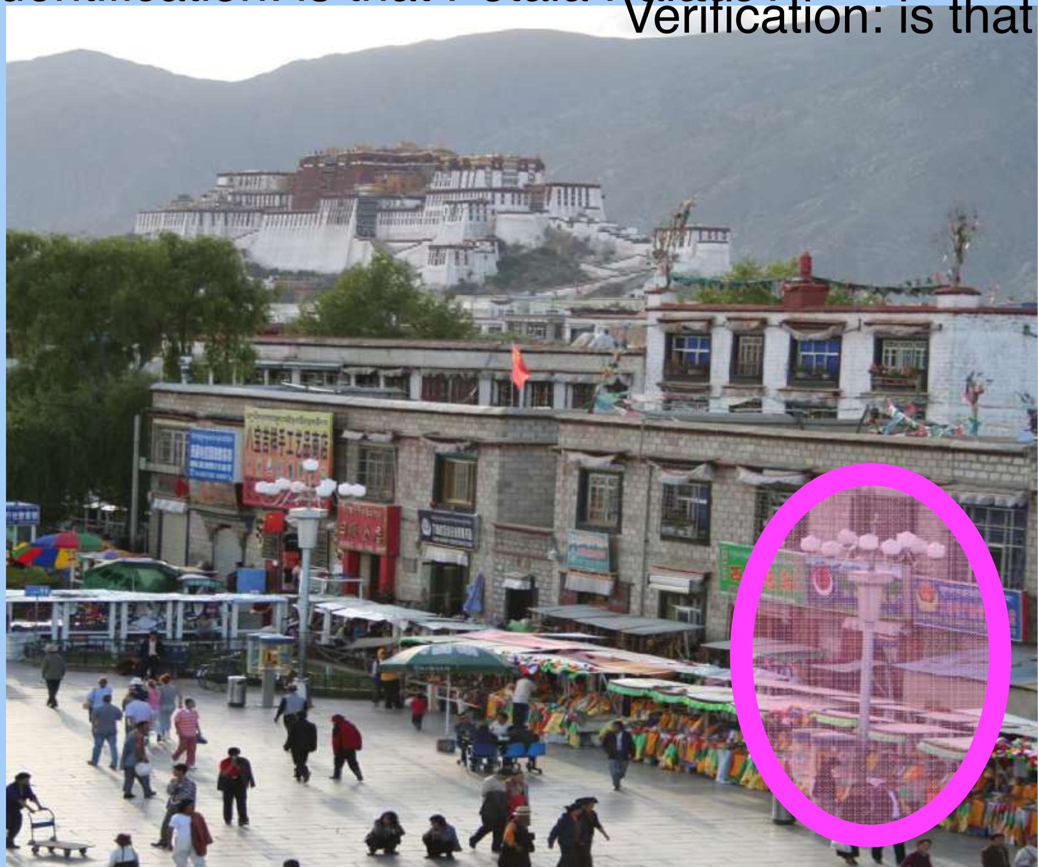
Fictions, shadows, and the like have a still lower degree of reality than ordinary objects and so are not proper subjects of philosophical enquiry.

How many object categories are there?



Identification: is that Potala Palace?

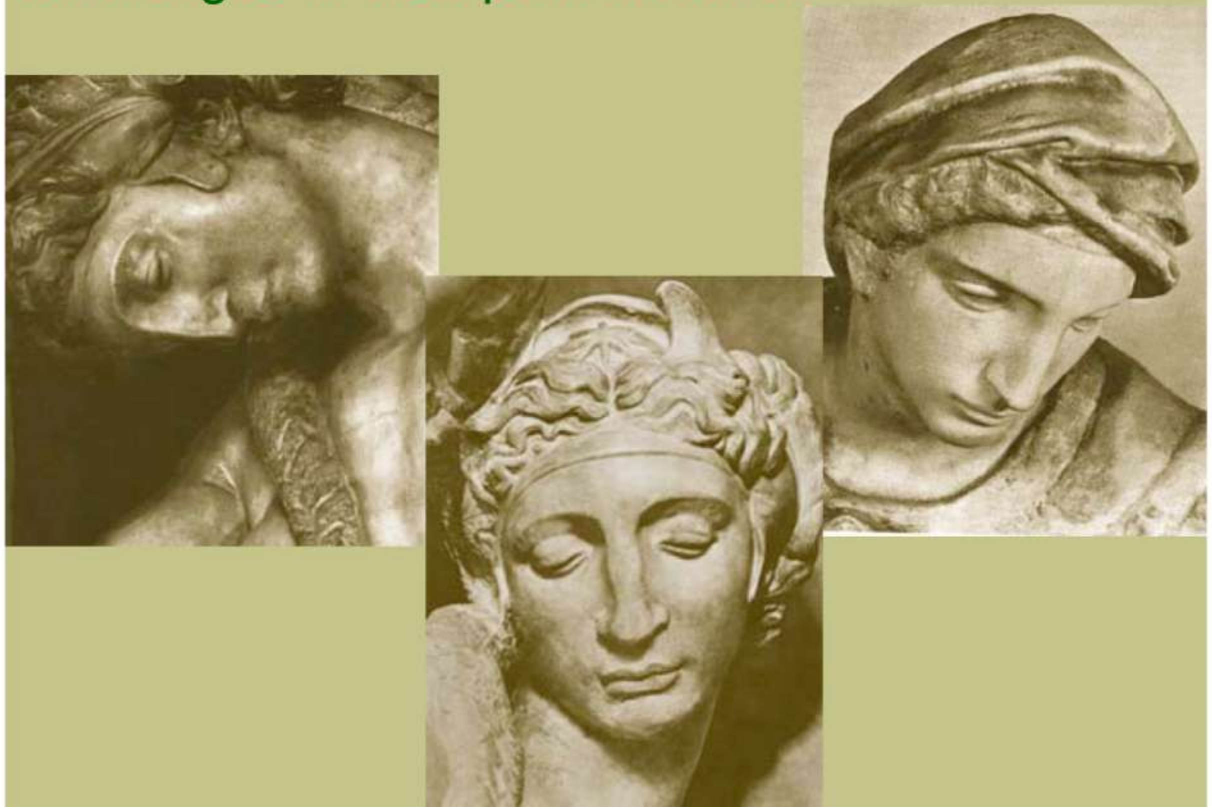
Verification: is that a lamp?



Detection: are there people?

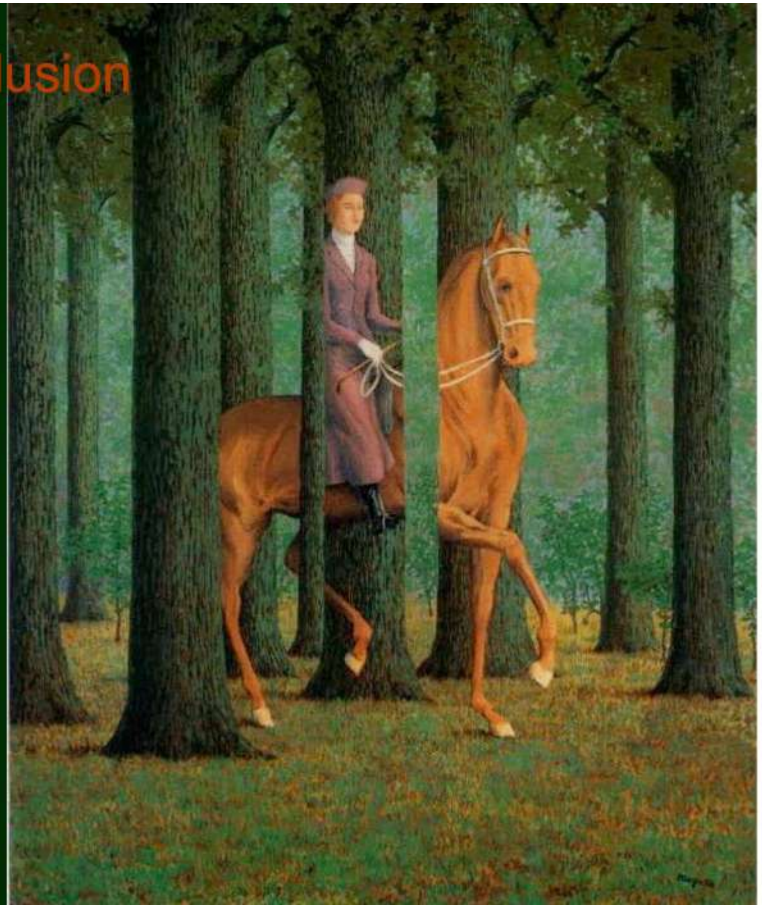
Object categorization

Challenges 1: view point variation



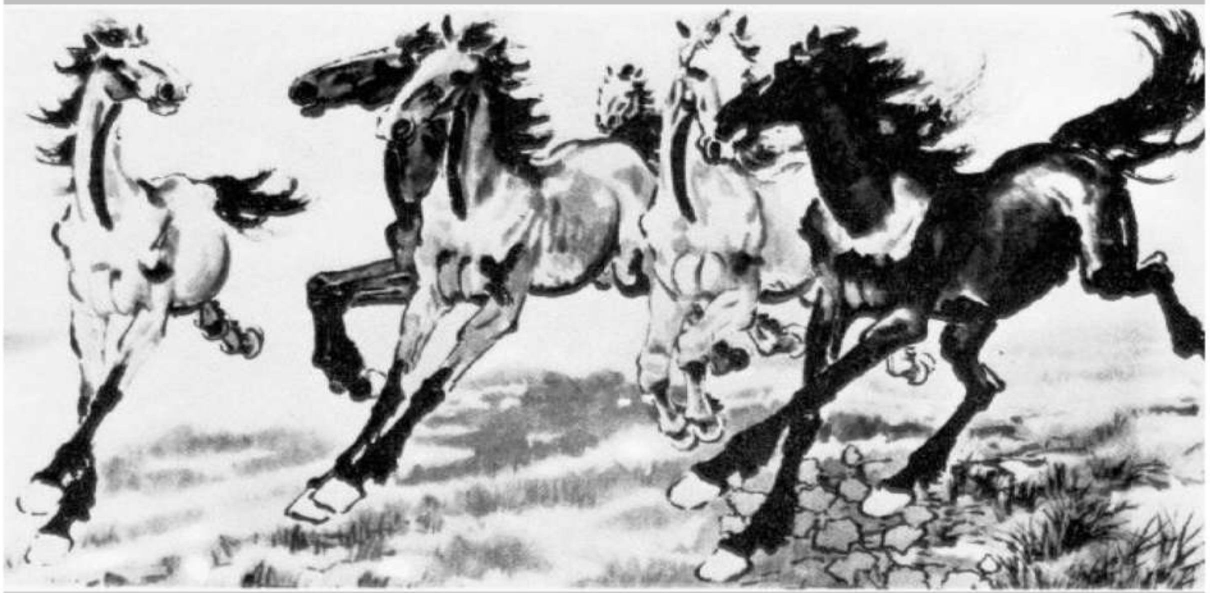
Challenges 2: illumination

Challenges 3: occlusion

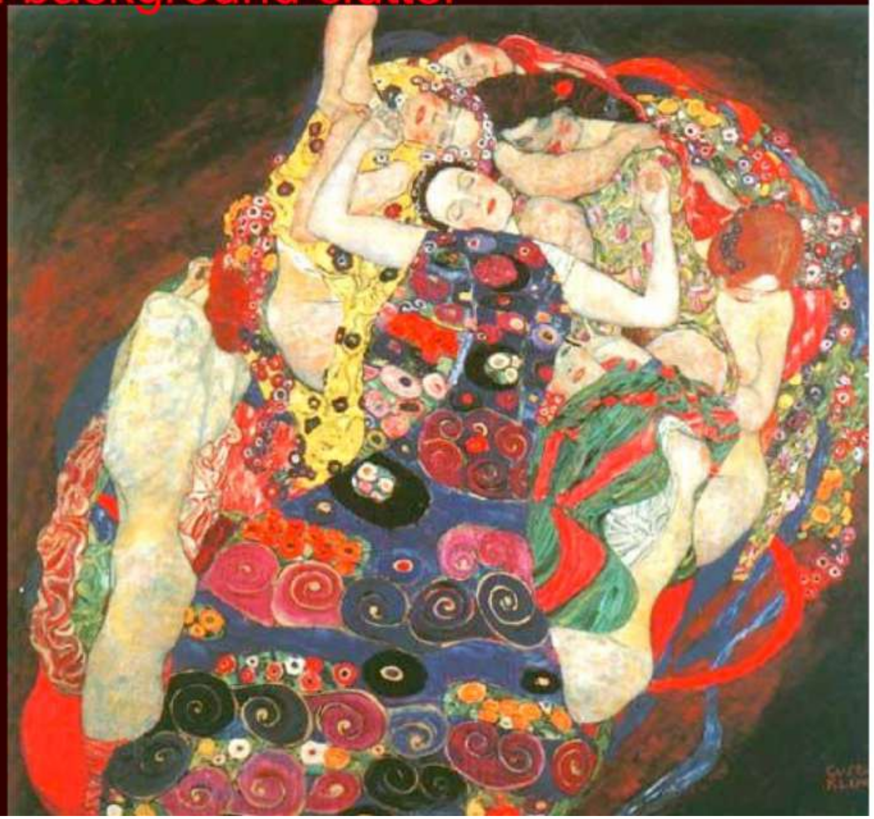


Challenges 4: scale

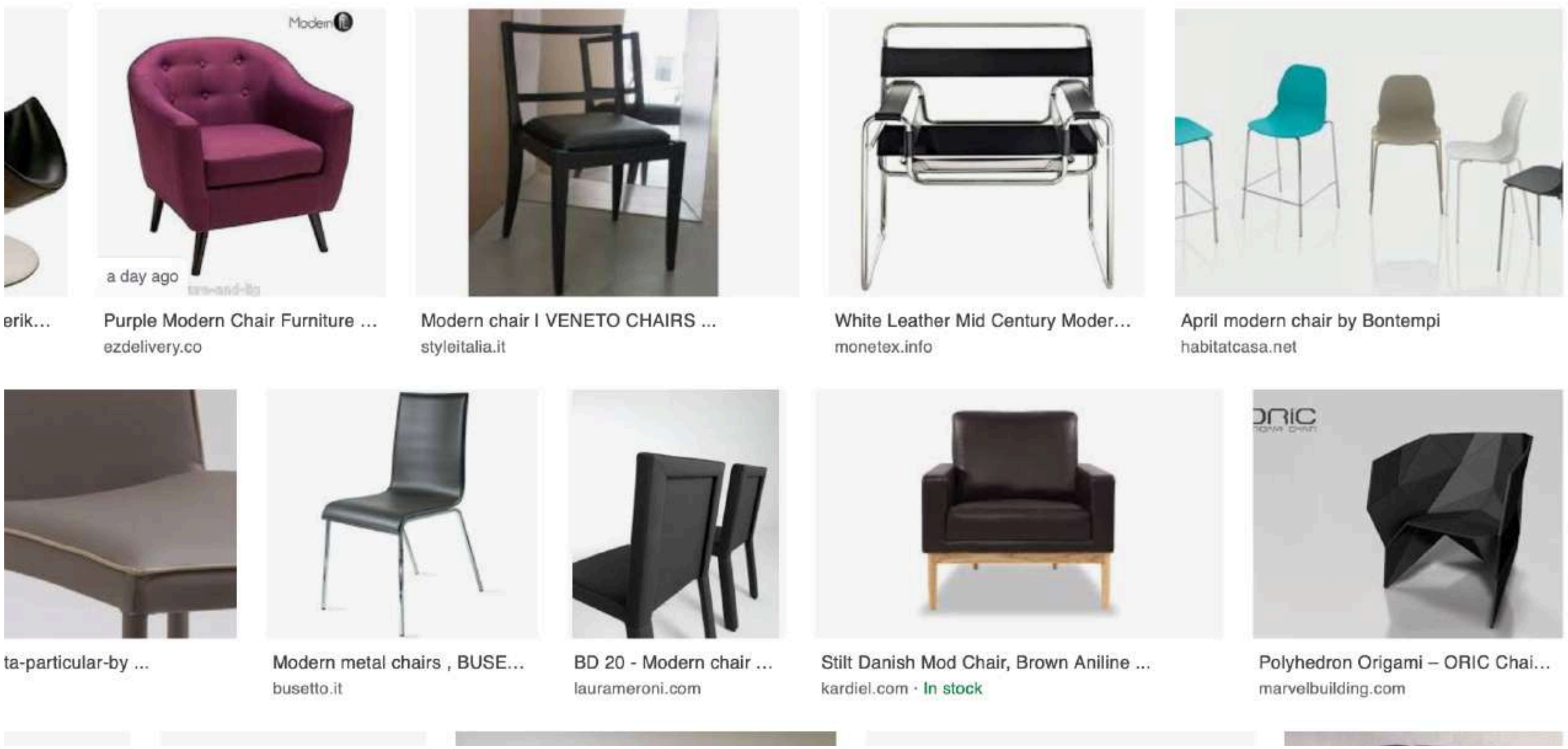
Challenges 5: deformation



Challenges 6: background clutter



Challenges 7: intra-class variation




~10,000 to 30,000

Three main issues

Representation

How to represent an object category

Learning

How to form the classifier, given training data

Recognition

How the classifier is to be used on novel data

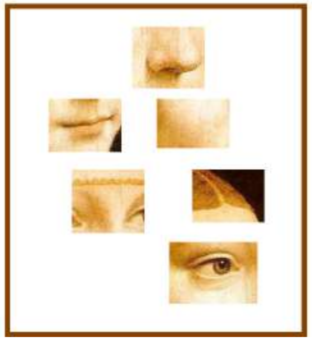
"Bag-of-words" models

Object

Bag of

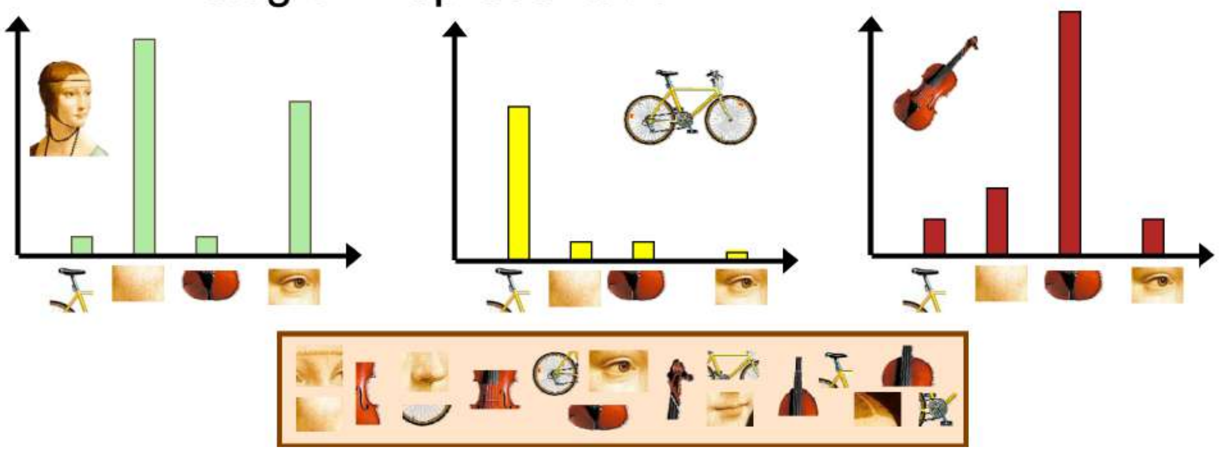
A clarification: definition of "BoW"

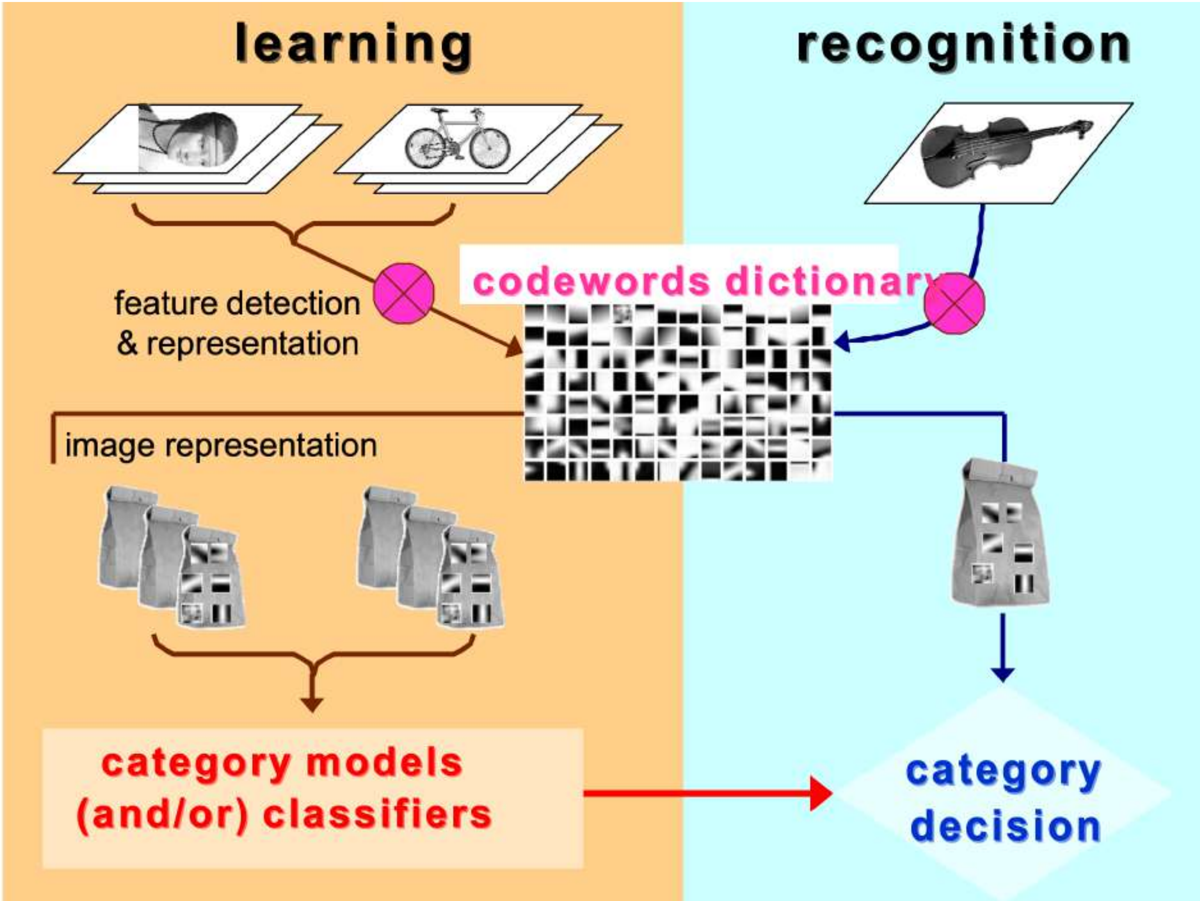
- Looser definition
  - Independent features



A clarification: definition of “BoW”

- Looser definition
  - Independent features
- Stricter definition
  - Independent features
  - histogram representation




learning
recognition
feature detection
& representation
codewords dictionary
image representation
category models
(and/or) classifiers
category
decision

Hints that DL … ***MUST*** WORK

Towards a regularity theory for ReLU networks – chain rule and global error estimates

Julius Berner*, Dennis Elbrächter*, Philipp Grohs‡, Arnulf Jentzen§

*Faculty of Mathematics, University of Vienna
Oskar-Morgenstern-Platz 1, 1090 Vienna, Austria

‡Faculty of Mathematics and Research Platform DataScience@UniVienna, University of Vienna
Oskar-Morgenstern-Platz 1, 1090 Vienna, Austria

§Department of Mathematics, ETH Zürich
Rämistrasse 101, 8092 Zürich, Switzerland

***Abstract*—Although for neural networks with locally Lipschitz continuous activation functions the classical derivative exists almost everywhere, the standard chain rule is in general not applicable. We will consider a way of introducing a derivative for neural networks that admits a chain rule, which is both rigorous and easy to work with. In addition we will present a method of converting approximation results on bounded domains to global (pointwise) estimates. This can be used to extend known neural network approximation theory to include the study of regularity properties. Of particular interest is the application to neural networks with ReLU activation function, where it contributes to the understanding of the success of deep learning methods for high-dimensional partial differential equations.**

a way that admits a chain rule which is both rigorous as well as easy to work with. Chain rules for functions which are not everywhere differentiable have been considered in a more general setting in e.g. [16], [17]. We employ the specific structure of neural networks to get stronger results using simpler arguments. In particular it allows for a stability result, i.e. Lemma III.3, the application of which will be discussed in Section V. We would also like to mention a very recent work [18] about approximation in Sobolev norms, where they deal with the issue by using a general bound for the Sobolev norm of the composition of functions from the Sobolev space $W^{1,\infty}$.

Thank you!

fabio.bonsignorio@gmail.com
fabio.bonsignorio@heronrobots.com

www.shanghailectures.org

The Shanghai Lectures 2020

HeronRobots Path-finder Lectures

Natural and Artificial Intelligence in Embodied Physical Agents

[redacted]