Introduction to statistical learning

Slides courtesy L. Lazebnik (Univ. of Illinois CS498) and others

Image source

Outline

- Statistical learning
- Two simple classification models: nearest neighbor, linear classifiers
- Learning beyond supervised classification: A brief taxonomy

Statistical learning

- We defined learning as “improving performance through experience”
- How do we operationalize the notion of experience?
  – Use *training data*

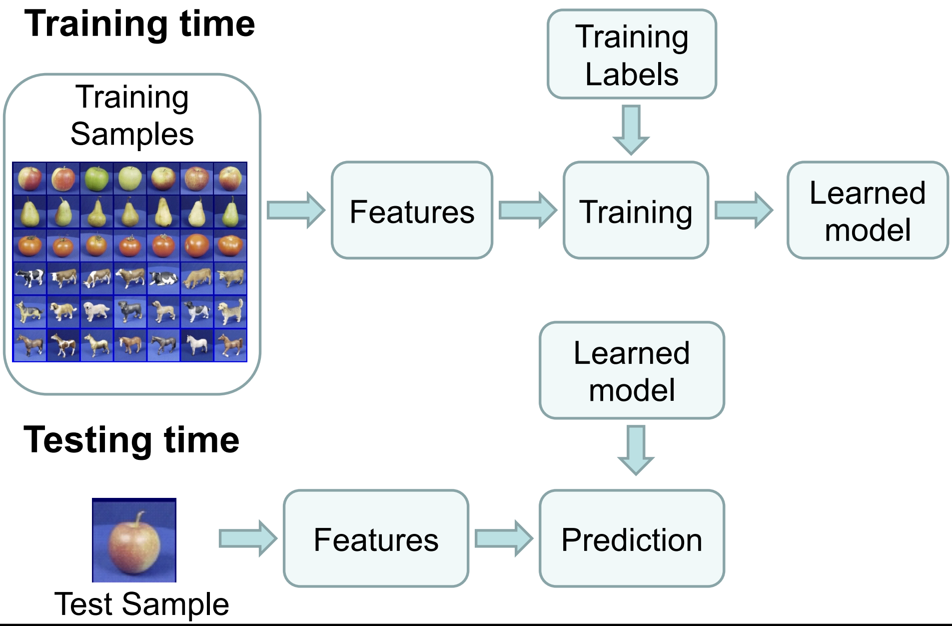
Example 1: Image classification

input	desired output
	apple pear tomato cow dog horse

Training data

apple
pear
tomato
cow
dog
horse

Training and testing

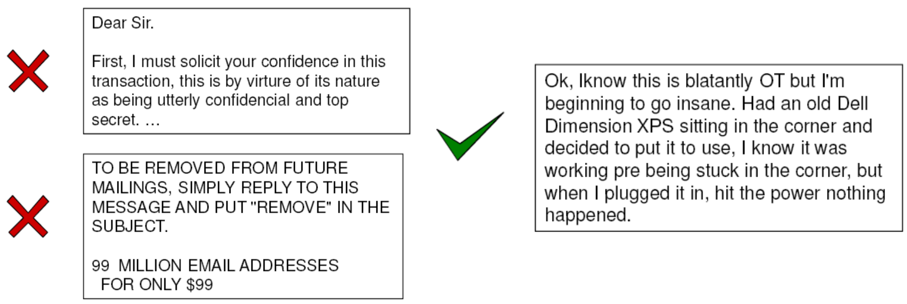


The basic *supervised learning* framework

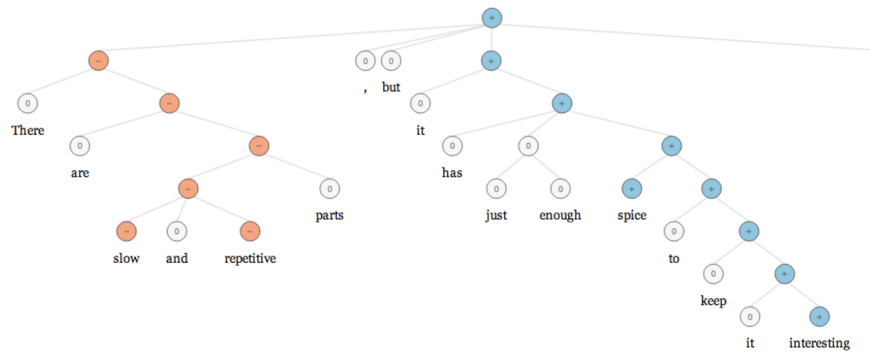
$$\underbrace{y}_{\text{output}} = \underbrace{f}_{\text{prediction function}}(\underbrace{\mathbf{x}}_{\text{input}})$$

- **Training** (or **learning**): given a *training set* of labeled examples $\{(\mathbf{x}_1,y_1), \ldots, (\mathbf{x}_N,y_N)\}$, instantiate a predictor f
- **Testing** (or **inference**): apply f to a new *test example* $\mathbf{x}$ and output the predicted value $y = f(\mathbf{x})$
- What is the connection between training and test data?

Example 2: Spam classification



Example 3: Sentiment classification

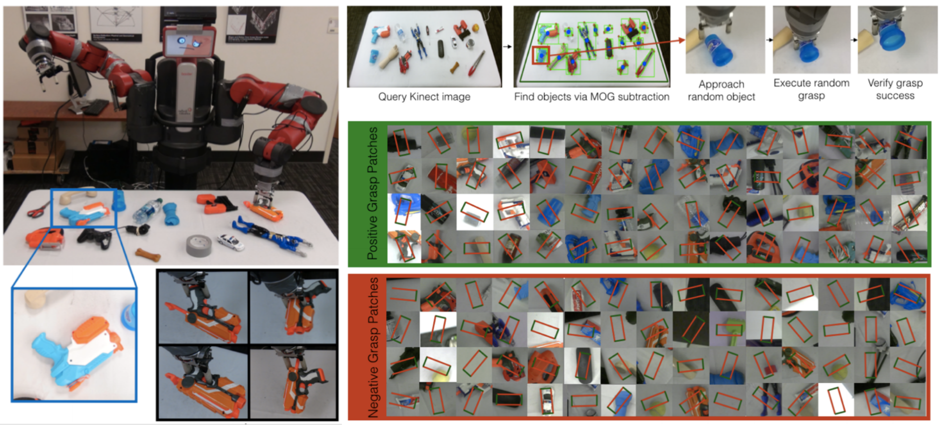


Recursive Deep Models for Semantic Compositionality Over a Sentiment Treebank R. Socher et al

http://gigaom.com/2013/10/03/stanford-researchers-to-open-source-model-they-say-has-nailed-sentiment-analysis/

http://nlp.stanford.edu:8080/sentiment/rntnDemo.html

Example 4: Grasp classification



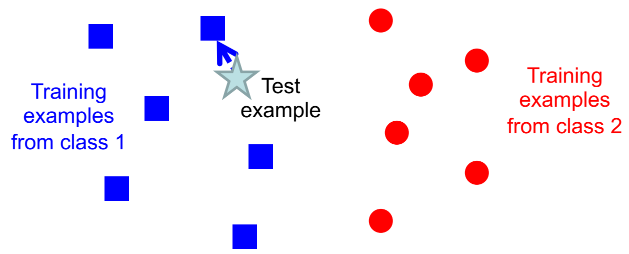
L. Pinto and A. Gupta, Supersizing self-supervision: Learning to grasp from 50K tries and 700 robot hours," arXiv.org/abs/1509.06825

YouTube video

Two simple models for supervised learning

- Nearest neighbor
- Linear classifiers

Nearest neighbor classifier

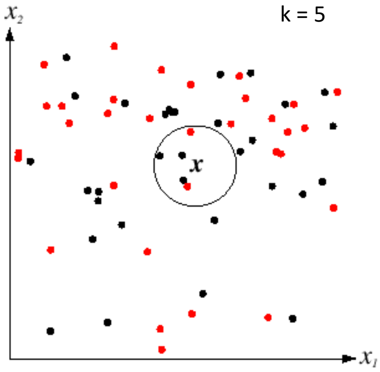


f(**x**) = label of the training example nearest to **x**

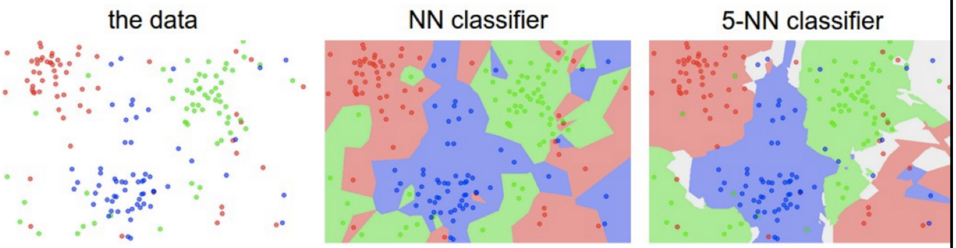
- All we need is a distance function for our inputs
- No training required!

K-nearest neighbor classifier

- For a new point, find the k closest points from training data
- Vote for class label with labels of the k points



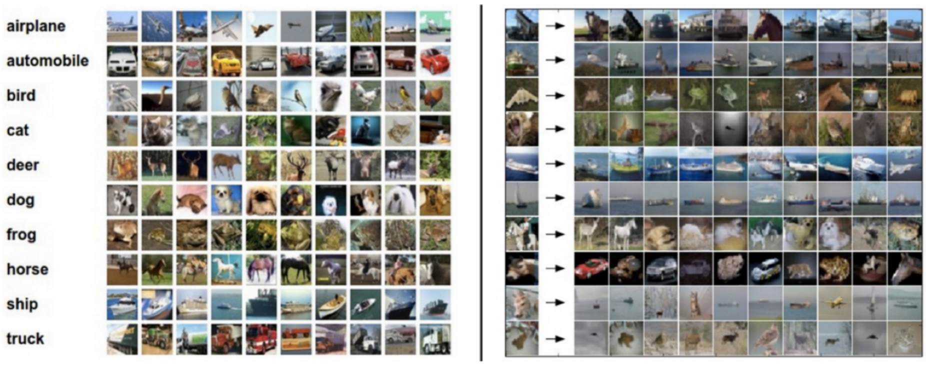
K-nearest neighbor classifier



- K-NN is more robust to *outliers*

Credit: Andrej Karpathy, http://cs231n.github.io/classification/

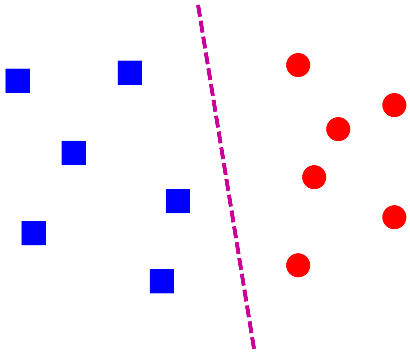
K-nearest neighbor classifier



Left: Example images from the CIFAR-10 dataset. Right: first column shows a few test images and next to each we show the top 10 nearest neighbors in the training set according to pixel-wise difference.

Credit: Andrej Karpathy, http://cs231n.github.io/classification/

Linear classifier

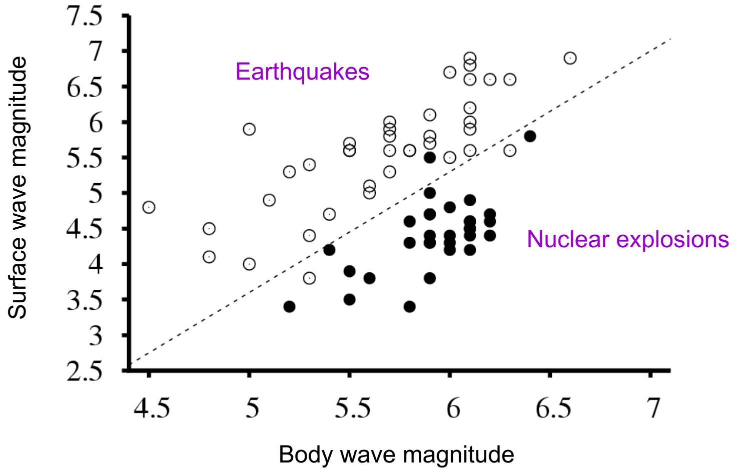


- Find a *linear function* to separate the classes

$f(\mathbf{x}) = \text{sgn}(w_1x_1 + w_2x_2 + \ldots + w_Dx_D + b) = \text{sgn}(\mathbf{w} \cdot \mathbf{x} + b)$

Visualizing linear classifiers

Seismic data classification



Visualizing linear classifiers

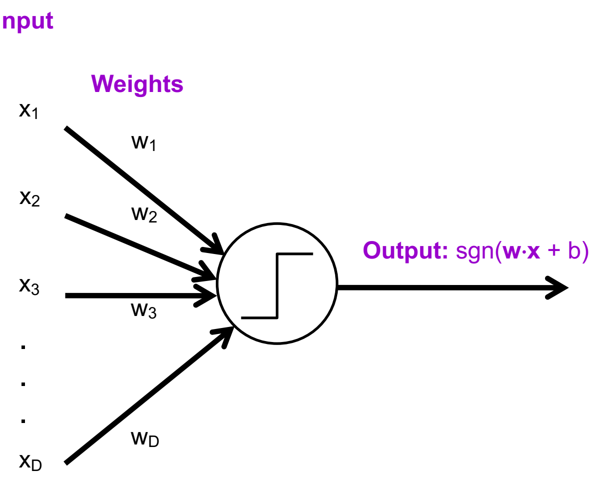
stretch pixels into single column

input image

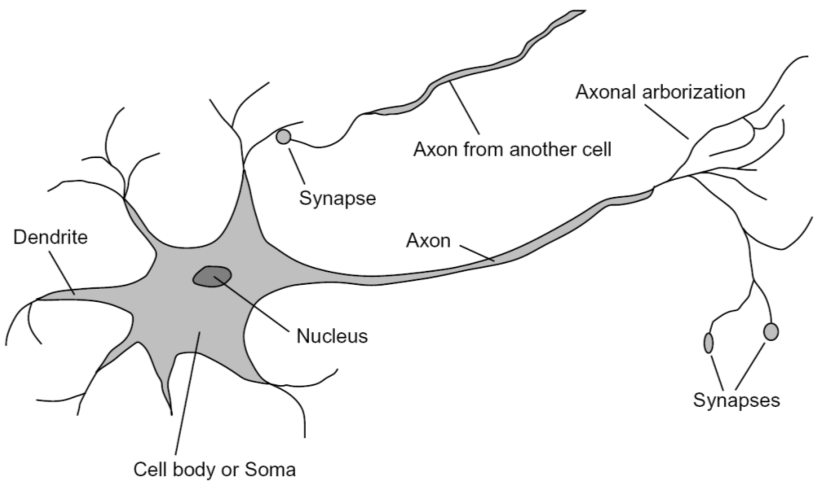
$$\underset{W}{\begin{bmatrix}0.2 & -0.5 & 0.1 & 2.0\\ 1.5 & 1.3 & 2.1 & 0.0\\ 0 & 0.25 & 0.2 & -0.3\end{bmatrix}} \underset{x_i}{\begin{bmatrix}56\\ 231\\ 24\\ 2\end{bmatrix}} + \underset{b}{\begin{bmatrix}1.1\\ 3.2\\ -1.2\end{bmatrix}} \longrightarrow \underset{f(x_i; W, b)}{\begin{bmatrix}-96.8\\ 437.9\\ 61.95\end{bmatrix}} \begin{matrix}\text{cat score}\\ \text{dog score}\\ \text{ship score}\end{matrix}$$

Source: Andrej Karpathy, http://cs231n.github.io/linear-classify/

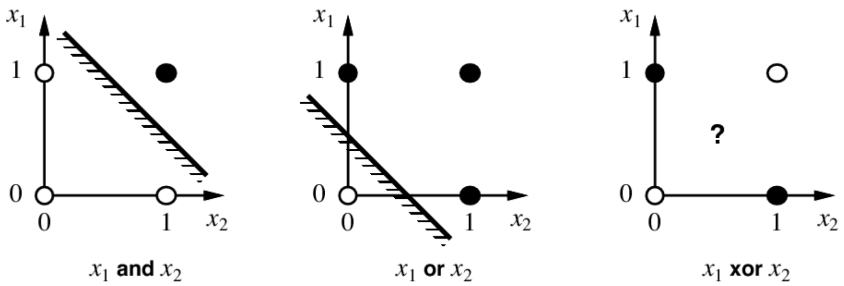
Linear classifier: Perceptron view



Loose inspiration: Human neurons



Perceptrons, linear separability and Boolean functions



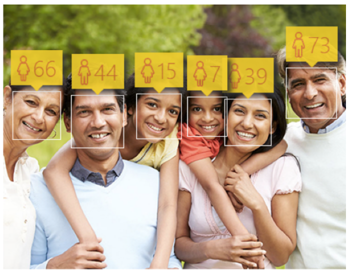
NN vs. linear classifiers

- NN pros:
  - \+ Simple to implement
  - \+ Decision boundaries not necessarily linear
  - \+ Works for any number of classes
  - \+ *Nonparametric* method
- NN cons:
  - – Need good distance function
  - – Slow at test time
- Linear pros:
  - \+ Low-dimensional *parametric* representation
  - \+ Very fast at test time
- Linear cons:
  - – Works for two classes
  - – How to train the linear function?
  - – What if data is not linearly separable?

Beyond supervised classification

- Other prediction scenarios
  - Regression
  - Structured prediction

Regression



Age estimation

When was that made?

IM2GPS

Structured Prediction

Image

brace

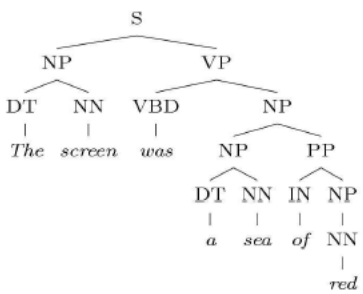
Word

Source: B. Taskar

Structured Prediction

The screen was
a sea of red

Sentence



Parse tree

Source: B. Taskar

Structured Prediction

RSCCPCYWGGCPW
GQNCYPEGCSGPKV

Amino-acid sequence

Bond structure

Source: B. Taskar

Structured Prediction

- Many scene understanding tasks can be thought of as "structured prediction" (but are not necessarily handled as such)

Keypoint prediction

K. He, G. Gkioxari, P. Dollar, and R. Girshick, Mask R-CNN, ICCV 2017

Beyond supervised classification

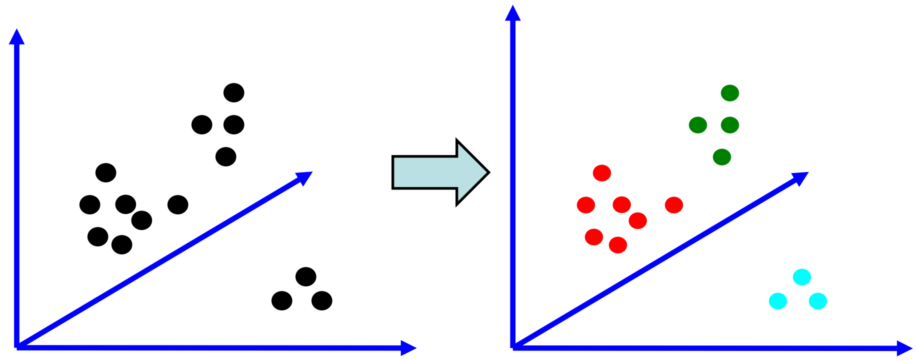
- Other prediction scenarios
  - Regression
  - Structured prediction
- Other supervision scenarios
  - Unsupervised learning
  - Self-supervised or predictive learning
  - Reinforcement learning
  - Active learning
  - Lifelong learning

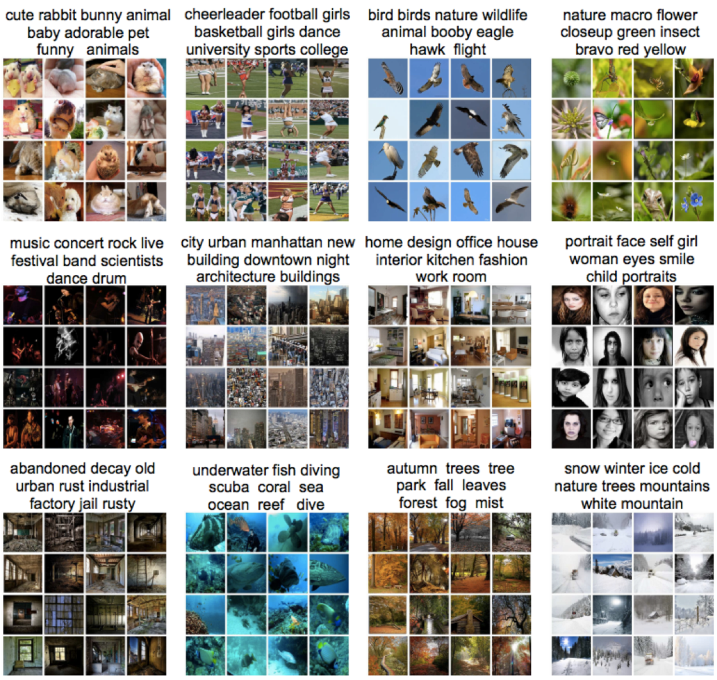
Unsupervised Learning

- **Idea:** Given only *unlabeled* data as input, learn some sort of structure
  - The goal is often more vague or subjective than in supervised learning
  - Also known as exploratory/descriptive data analysis

Unsupervised Learning

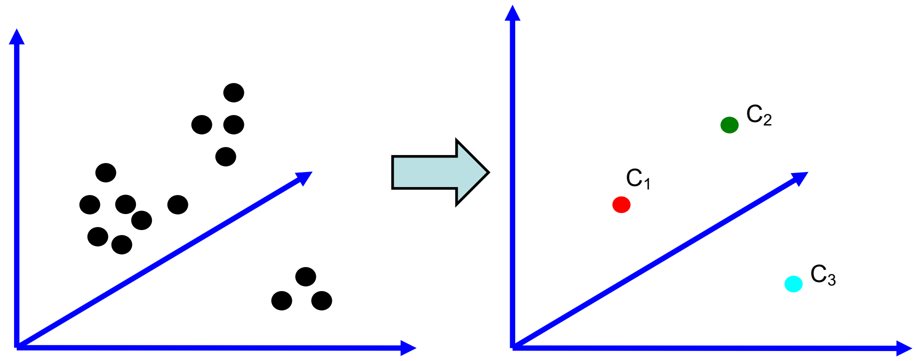
- **Clustering**
  - Discover groups of “similar” data points





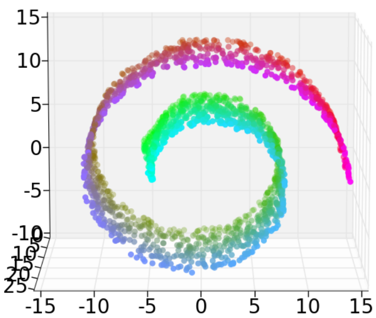
Unsupervised Learning

- **Quantization** or **data compression**
  - Encode the data into a more compact form



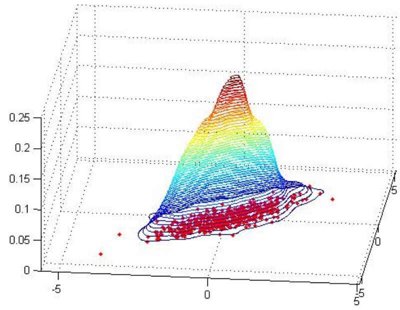
Unsupervised Learning

- **Dimensionality reduction, manifold learning**
  - Discover a lower-dimensional surface on which the data lives



Unsupervised Learning

- **Learning the data distribution**
  - Density estimation: Find a function that approximates the probability density of the data (i.e., value of the function is high for "typical" points and low for "atypical" points)
  - Can be used for anomaly detection

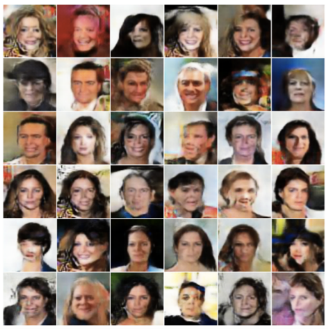


Unsupervised Learning

- **Learning the data distribution**
  - Learning to sample: Produce samples from a data distribution that mimics the training set

"Bedroom"

"Face"



[Generative adversarial networks]

Forms of incomplete supervision

Semi-supervised
(labels for a small portion of training data)

Unsupervised
(no labels)

Weakly supervised
(noisy labels, labels not exactly for the task of interest)

Supervised
(clean, complete training labels for the task of interest)

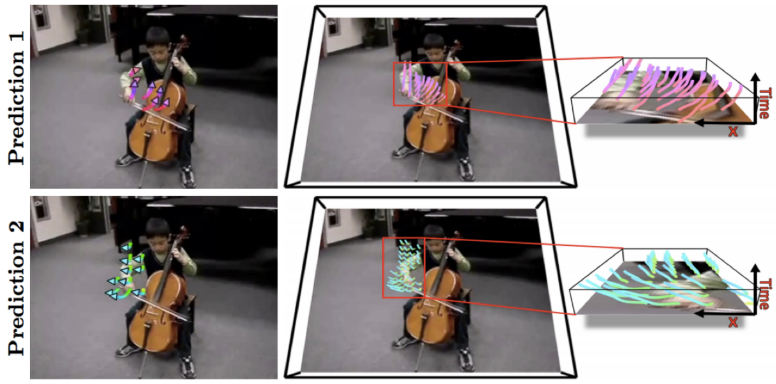
Self-supervised or predictive learning

- Use part of the data to predict other parts of the data
  - Example: Image colorization

R. Zhang et al., Colorful Image Colorization, ECCV 2016

Self-supervised or predictive learning

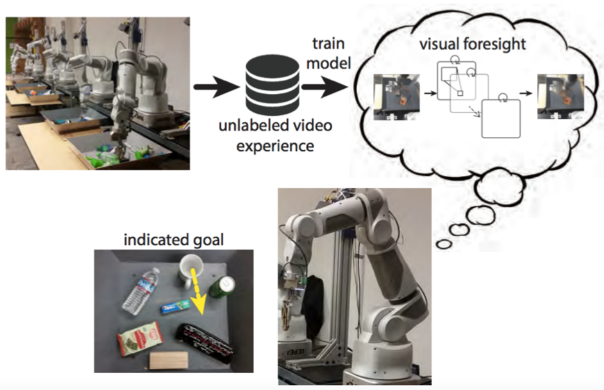
- Use part of the data to predict other parts of the data
  - Example: Future prediction



J. Walker et al. An Uncertain Future: Forecasting from Static Images Using Variational Autoencoders. ECCV 2016.

Self-supervised or predictive learning

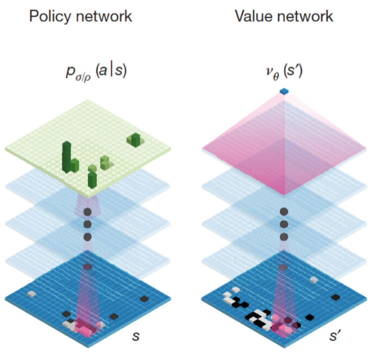
- Use part of the data to predict other parts of the data
  - Example: Future prediction



C. Finn and S. Levine. Deep Visual Foresight for Planning Robot Motion. ICRA 2017. YouTube video

Reinforcement learning

- Learn from rewards in a *sequential* environment



https://deepmind.com/research/alphago/

Reinforcement learning

- Learn from rewards in a *sequential* environment

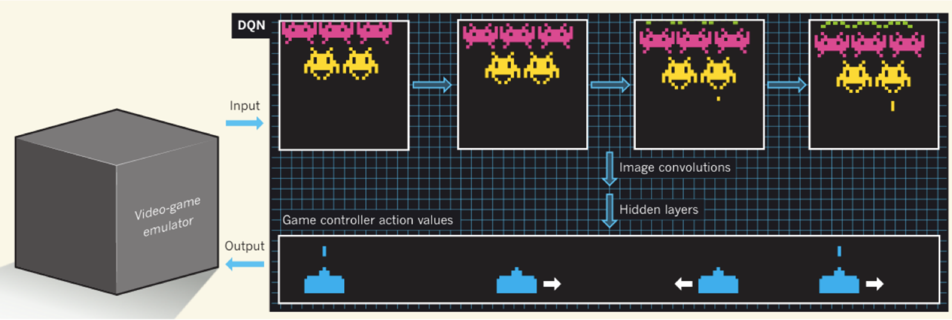
Initial gait

Learned gait

Policy Gradient Reinforcement Learning for Fast Quadrupedal Locomotion
Nate Kohl and Peter Stone.
IEEE International Conference on Robotics and Automation, 2004

Reinforcement learning

- [Playing Atari with deep reinforcement learning]

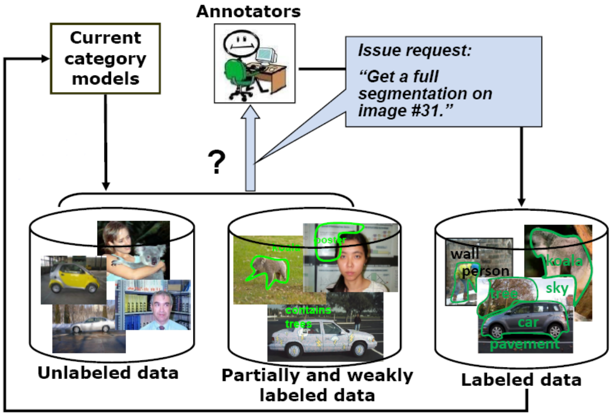


Breakout video

V. Mnih et al., *Nature*, February 2015

Active learning

- The learning algorithm can choose its own training examples, or ask a "teacher" for an answer on selected inputs



S. Vijayanarasimhan and K. Grauman, "Cost-Sensitive Active Visual Category Learning," 2009

Lifelong learning

Read the Web

Research Project at Carnegie Mellon University

Home | Project Overview | Resources & Data | Publications | People

NELL: Never-Ending Language Learning

Can computers learn to read? We think so. "Read the Web" is a research project that attempts to create a computer system that learns over time to read the web. Since January 2010, our computer system called NELL (Never-Ending Language Learner) has been running continuously, attempting to perform two tasks each day:

Browse the Knowledge Base!

- First, it attempts to "read," or extract facts from text found in hundreds of millions of web pages (e.g., `playsInstrument(George_Harrison, guitar)`).
- Second, it attempts to improve its reading competence, so that tomorrow it can extract more facts from the web, more accurately.

So far, NELL has accumulated over 50 million candidate beliefs by reading the web, and it is considering these at different levels of confidence. NELL has high confidence in 2,033,557 of these beliefs — these are displayed on this website. It is not perfect, but NELL is learning. You can track NELL's progress below or @cmunell on Twitter, browse and download its knowledge base, read more about our technical approach, or join the discussion group.

http://rtw.ml.cmu.edu/rtw/

Lifelong learning

Recently-Learned Facts twitter

Refresh

instance	iteration	date learned	confidence
goose_gossage is an athlete	787	16-nov-2013	100.0
fitchburg_state_college is a building	788	19-nov-2013	98.7
kirk_gibson is an actor	787	16-nov-2013	99.0
alex_turner ia a celebrity	787	16-nov-2013	97.5
anthony_r__birley is a criminal	788	19-nov-2013	92.2
the final score of the sports game semi_finals was 6-1	792	01-dec-2013	100.0
national_museum is a museum in the city tokyo	792	01-dec-2013	100.0
w__bush is a U.S. politician endorsed by the U.S. politician john_ashcroft	788	19-nov-2013	93.8
frank004 is a person who graduated from the university state_university	790	24-nov-2013	99.6
mississippi_state_university is a sports team also known as state_university	787	16-nov-2013	99.2

http://rtw.ml.cmu.edu/rtw/

NEIL: Never Ending Image Learner

I Crawl, I See, I Learn.

WHAT COMMON SENSE FACTS HAVE NEIL LEARNED?

Here are a few examples:

Airbus_330 can be a kind of / look similar to Airplane.

Deer can be a kind of / look similar to Antelope.

Car can have a part Wheel.

Airbus_330 can have a part Airplane_nose.

Leaning_tower can be found in Pisa.

Zebra can be found in Savanna.

Xinlei Chen, Abhinav Shrivastava and Abhinav Gupta. NEIL: Extracting Visual Knowledge from Web Data. In ICCV 2013