CSE 473

Final Lecture:
A Smörgåsbord of Course Topics and Applications

Plan for Today

- Wrap up of boosting (AdaBoost)
- Course Review and Applications of AI
- Final Exam sneak preview
- Sayonara and Evals

2

Recall from Last Time: Boosting

training instance → correctly classified

This instance has large weight in this round

h_1 h_2 h_3 h_4

This DT has a strong vote (all correctly classified)

h
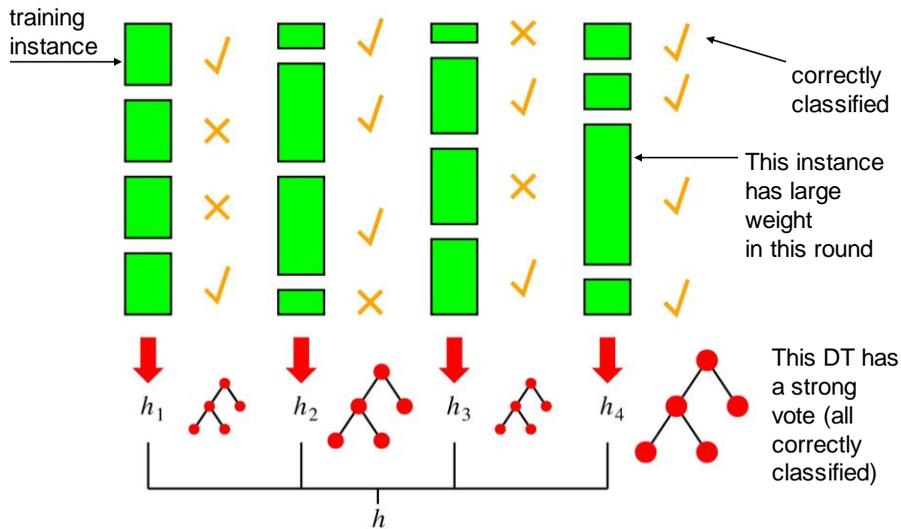
Output of h_{final} is weighted majority of outputs of $h_1, ..., h_4$

Adaptive Boosting (AdaBoost) Algorithm

- $w_j \leftarrow 1/N \quad \forall_j$

  w: vector of N instance weights
  z: vector of M hypoth. weights

- For m=1 to M do
  - $h_m \leftarrow$ learn(dataset,w)   Select classifier h_m with least
  - err $\leftarrow 0$ *weighted* classification error
  - For each (x_j, y_j) in dataset do
    - If $h_m(x_j) \neq y_j$ then err $\leftarrow$ err $+ w_j$ Compute total error
  - For each (x_j, y_j) in dataset do
    - If $h_m(x_j) = y_j$ then $w_j \leftarrow w_j$ err / (1-err)   Adjust all instance weights
  - $w \leftarrow$ normalize(w)
  - $z_m \leftarrow$ log [(1-err) / err]   Adjust weight for hypothesis m
- Return *weighted-majority(h,z)*

2

AdaBoost Example
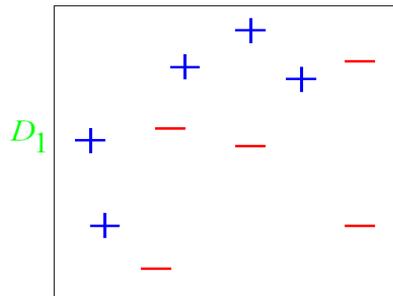
Original training set D_1 : Equal weights for all training inputs
Goal: In round t, learn classifier h_t that minimizes error with respect to weighted training set
h_t maps input to +1 or -1: $h_t : X \rightarrow \{-1, +1\}$

Taken from "A Tutorial on Boosting" by Yoav Freund and Rob Schapire

AdaBoost Example

ROUND 1

Misclassified   **Increase weights**

z_1 = 0.42

AdaBoost Example

ROUND 2

$z_2 = 0.65$

h_2

D_3

Now these are misclassified
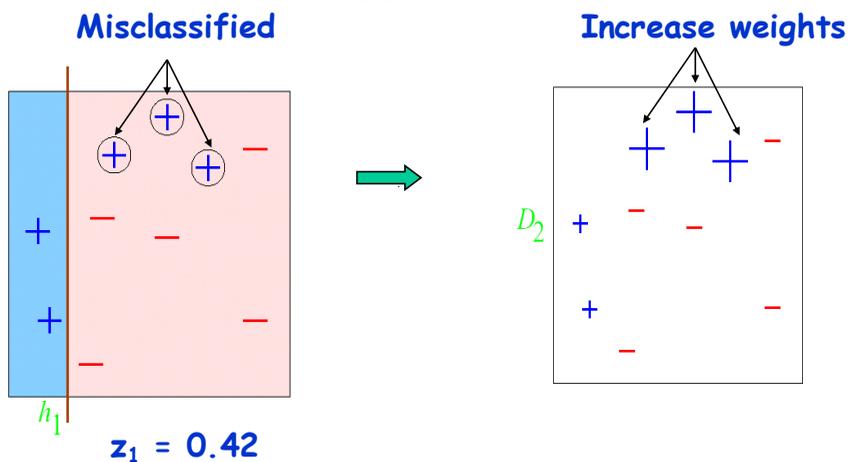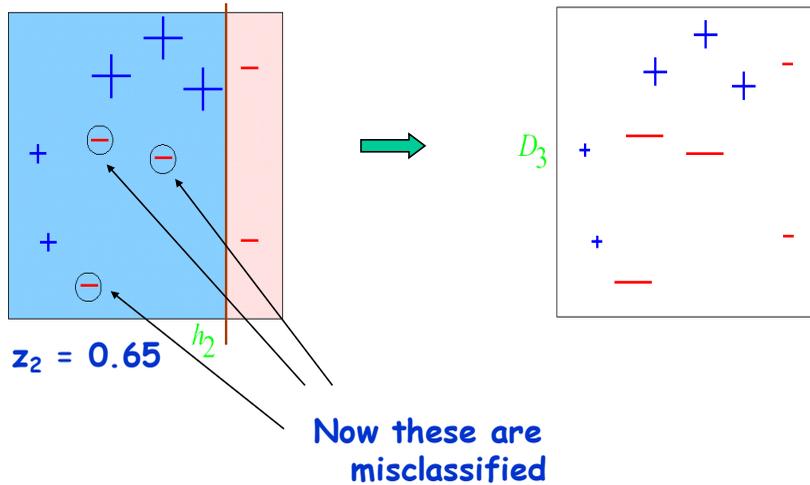
AdaBoost Example

ROUND 3

h_3

$z_3 = 0.92$

AdaBoost Example

h$_{final}$

$$= \text{sign}\left(0.42 \quad + 0.65 \quad + 0.92 \right)$$

sign(x) = +1 if x > 0 and -1 otherwise

Face Detection using AdaBoost

Classifiers = local feature detectors

1st 2nd
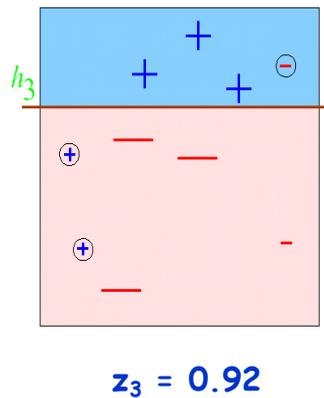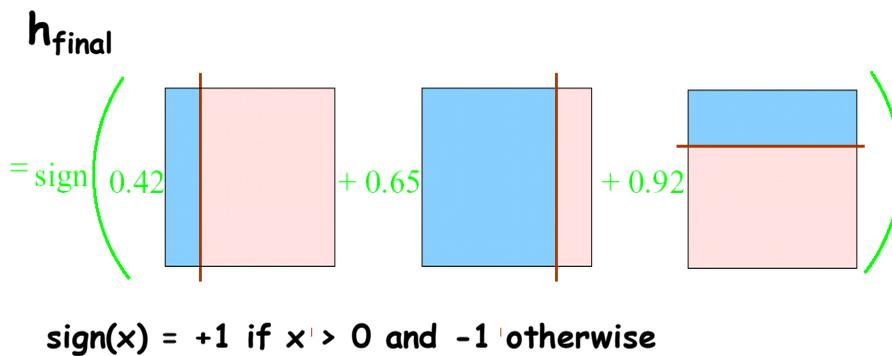
Training images
(non-face images not shown)
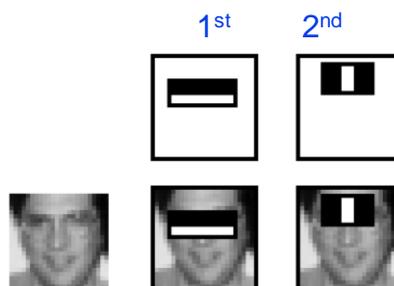
AdaBoost computes weighted majority of feature detectors

(Viola & Jones, 2001)

Face Detection using AdaBoost
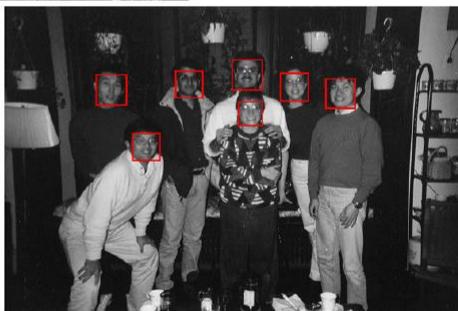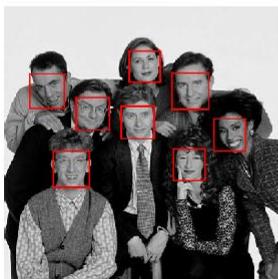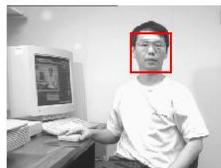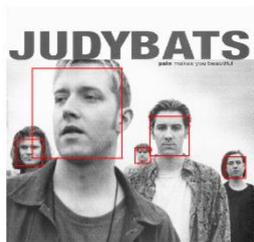
(Viola & Jones, 2001)

Let's look at some more applications of AI

Course Review and Applications of AI Concepts we studied

Recall: Heuristic Search
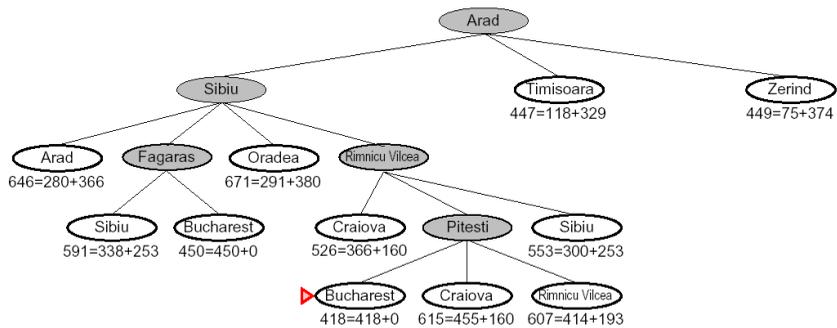
Best-first search, A* search, admissible heuristics

Application: Path Planning on Robots

Mars Rovers (2003-now)

AI concept: Heuristic search for path planning

Recall: Adversarial Search

MAX △ ≥3

MIN ▽ ≤2

3   12   8   2   X   X
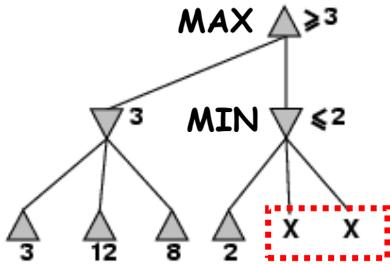
- Minimax Search
- Alpha Beta pruning
- Cut-off search
- Evaluation functions
- Pattern databases

Application: Game Playing

MONTY NEWBORN

KASPAROV
VERSUS
DEEP
BLUE
COMPUTER CHESS
COMES OF AGE
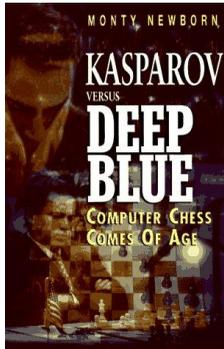
The New York Times

Once Again, Machine Beats Human Champion at Chess

Henning Kaiser/AFP -- Getty Images

AI concepts we studied
- Minimax Search
- Pattern databases
- Learning
  E.g., reinforcement
  learning
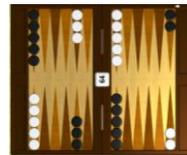
Recall: Probabilistic Reasoning
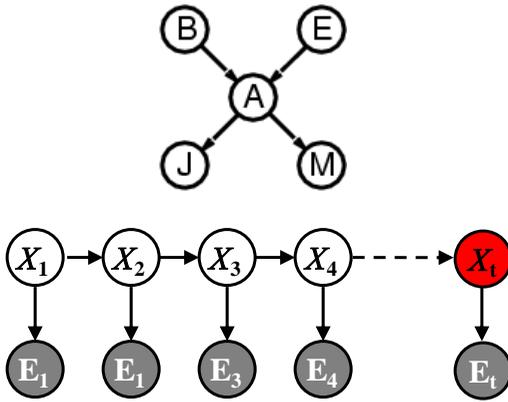
AI concepts we studied
- Bayesian networks
- Probabilistic inference
- Hidden Markov Models (HMMs)
- Forward algorithm
- Particle filtering

$$P(X_t \mid e_1,...,e_t) = \alpha \, P(e_t \mid X_t) \sum_{X_{t-1}} P(X_t \mid X_{t-1}) P(X_{t-1} \mid e_1,\ldots,e_{t-1})$$

Application: Speech Recognition
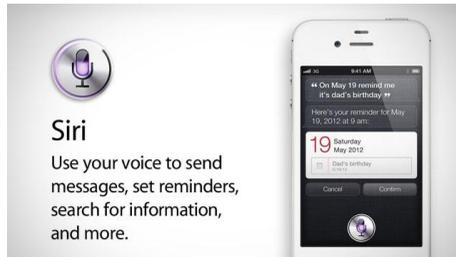
- Hidden Markov Models (HMMs)
- Forward algorithm
- Machine learning

Navigation Systems

Automated call centers

Application: Driverless Cars

Winners of the 2005 and 2007 DARPA Grand Challenges

Google's Driverless Car

- Probabilistic reasoning
- Filtering
- Markov models
- Machine learning

19

Application: Robot Localization and Mapping of Allen Center
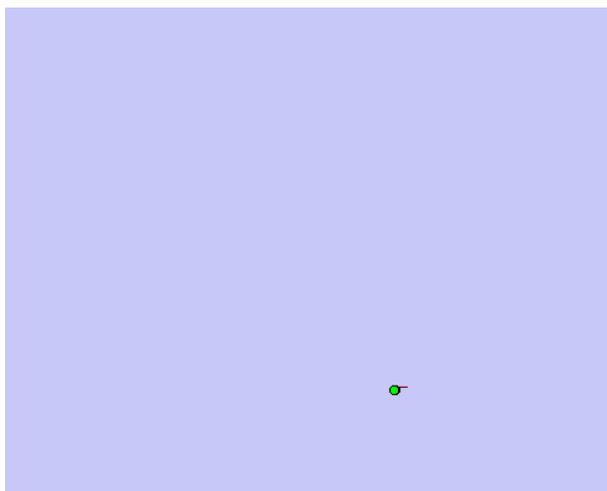
- Probabilistic reasoning
- Particle Filtering
- Machine learning

(Work of Prof. Dieter Fox and students)

20

Recall: Neural Networks

$$E(\mathbf{W}, \mathbf{w}) = \frac{1}{2} \sum_i (d_i - v_i)^2$$

$$v_i^m = g(\sum_j W_{ji} x_j)$$

W

w

u_k
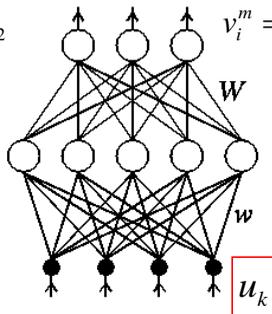
Backprop rule for input-hidden weights w:

$$w_{kj} \to w_{kj} - \varepsilon \frac{dE}{dw_{kj}} \quad \text{But}: \frac{dE}{dw_{kj}} = \frac{dE}{dx_j} \cdot \frac{dx_j}{dw_{kj}}$$

$$\frac{dE}{dw_{kj}} = \left[- \sum_i (d_i - v_i) g'(\sum_j W_{ji} x_j) W_{ji} \right] \cdot \left[g'(\sum_k w_{kj} u_k) u_k \right]$$

Application: Pattern Recognition

Artificial Neural Network Handwriting Recognizer

Written in Java™

by *Bob Mitchell*

CLICK HERE FOR DEMO

Recall: Classification Techniques

- Decision Trees
- Nearest Neighbors
- SVMs
- Etc.
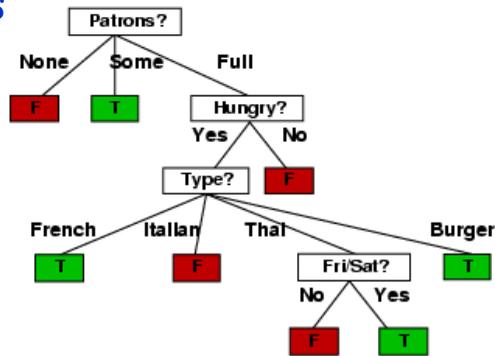
Application: Brain-Computer Interfaces

- Classifying brain signals recorded at the scalp
- Detect which object a person wants from a set of objects
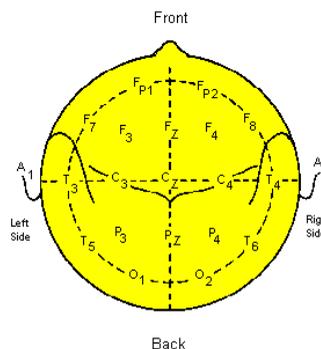
Brain-Controlled Robotic "Avatar"
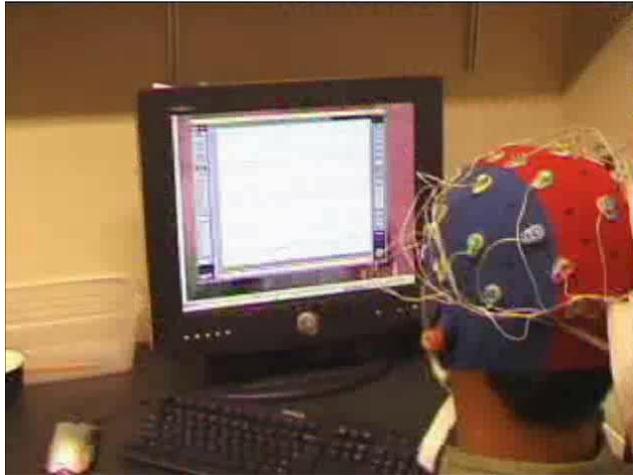
Interface uses SVM to classify brain signals

The Future of AI

Massive amounts of data

+

Sophisticated probabilistic reasoning
and machine learning algorithms

+

Massive computing power

= AI revolution?

AI in a Sensor-rich World

- Intelligent thermostats
- Intelligent refrigerators
- Intelligent houses
- Intelligent forests
- Intelligent oceans
- Intelligent bridges
- Etc.

Nest learning thermostat

Other future AI applications

- Smart power grids: electric power flows both ways and is distributed dynamically according to changing demand
- Security and military: Bomb diffusing robots, unmanned vehicles
- Robot firefighters
- AI Travel Agents
- AI Doctors
- AI Lawyers
- AI Football Coaches
- AI Football Players
- AI Rock Stars…

Take-Home Final: Details

- Will be posted on website later today
- Open book, open notes
- Focus mostly on post-midterm material
- Due Wednesday Dec 12 by 10:30AM
- Will involve a mix of problem solving and descriptive questions

     E.g., Computing probabilities in Bayesian networks, explaining important concepts in AI (A* search, alpha-beta pruning, etc.)

That concludes the course….

$$w_{kj} \to w_{kj}$$

$$\frac{dE}{dw_{kj}} = \left[- \sum_i \cdots \sum_k w_{kj} u_k \right) u_k \right]$$

$$w_{kj} \to w_{kj} - \varepsilon \frac{dE}{dw_{kj}} \quad \text{But:} \frac{dE}{dw_j}$$

$$\frac{dE}{dw_{kj}} = \left[- \sum_i (d_i - v_i) g'(\sum_j W \right.$$
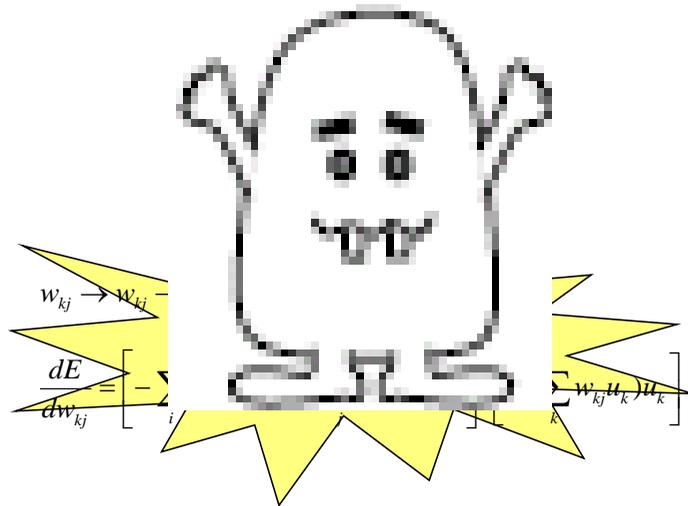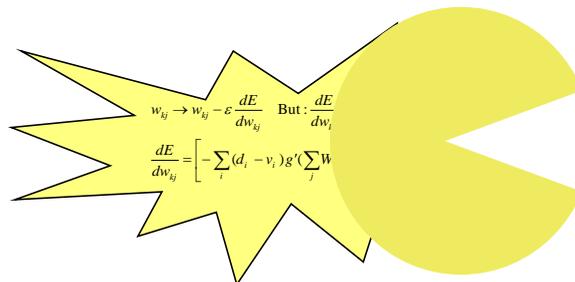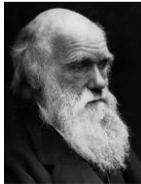
Have a great break!

Who glued my fingers?