Applications of computer science in the life sciences

Robert D. Vincent

COMP102, McGill University

29 Nov 2011

McGill

What can computer science do for the life sciences?

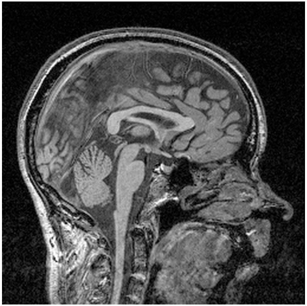
- Medical image analysis
- Genomics and phylogenetics
- Drug design and discovery
- Assistive robotics
- Brain-computer interfaces
- Simulation of biological systems
- Medical treatment optimization

Relevant techniques from computer science

- Searching and sorting
- Network analysis
- Robotics and artificial intelligence
- Machine learning and pattern classification
- **Reinforcement learning**

Medical image analysis

- The resolution and quality of medical images has exploded over the past two decades. Example applications from brain imaging are:



- **Automatic tissue classification**
- Image-guided neurosurgery
- Cortical thickness measurement
- **Decoding cognitive states**

Automatic tissue classification

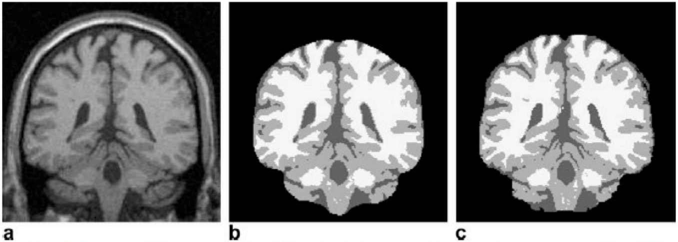


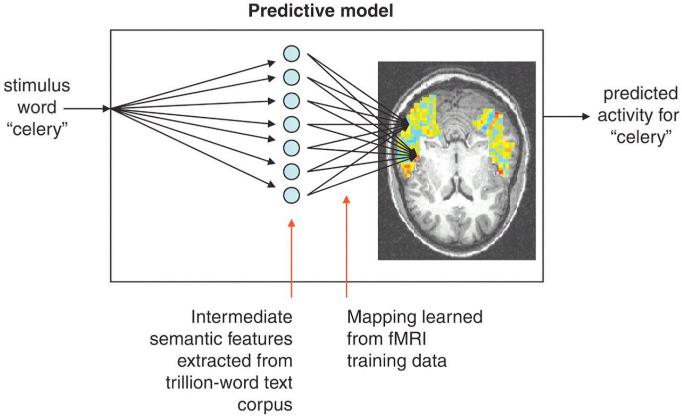
Fig. 1. Qualitative analysis of the proposed algorithm with BrainWeb data (Collins et al., 1998) with 5% noise and a 40% bias field: (a) a coronal slice of the data; (b) the classification produced by the proposed method and (c) the ground truth.

Awate et al. 2006

- Given a scalar intensity for each "voxel".
- Calculate contributions of white matter, grey matter, and cerebrospinal fluid.
- Central idea: Use knowledge about other images to calculate the "most likely" interpretation of a new image

Decoding cognitive states

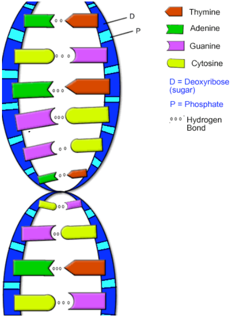
A model trained from fMRI images of a test subject can identify the noun the subject was thinking of with over 70% accuracy.



From Mitchell et al. 2008, *Science*

Genomics and phylogenetics

- Computer algorithms are increasingly critical to research in genomics and evolutionary relationships among organisms.
- Often lumped under “bioinformatics”

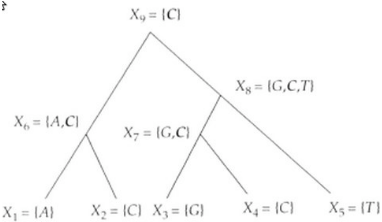


- Gene and protein sequencing
- Discovery of regulatory sites and relationships
- **Reconstruction of ancestral sequences**
- Analysis of regulatory networks

http://www.biologycorner.com/

Reconstruction of ancestral sequences

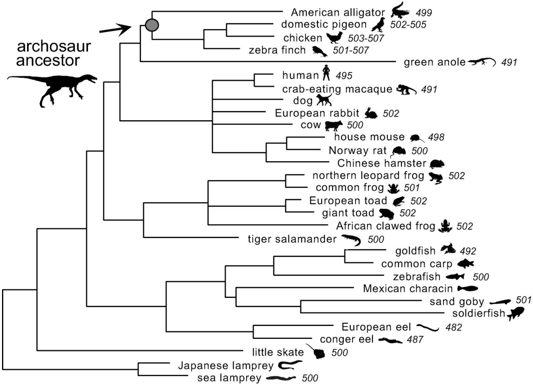
- First proposed in 1963 by Pauling and Zuckerkandl
- Begin with sequences of existing genes or proteins
- Assume constant rates of mutation (parsimony)
- Calculate most likely ancestral sequence
- Synthesize and evaluate ancestral sequence in laboratory



From Liberles (ed.) 2007

Reconstruction of ancestral sequences

- Chang et al. (2002) synthesized archosaur visual pigments (rhodopsin)
- Suggested that the wavelength of maximum sensitivity was consistent with nocturnal behavior



Drug design and discovery

- At least 500,000 proteins in the human proteome
- Roughly 2% are well studied (Young, 2009)
- Computational methods are applied to:
  - Virtual screening
  - Protein structure and folding

Assistive robotics

- Uses robotics to aid patients with impaired mobility or cognition.

SmartWheeler

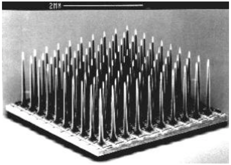
- Automatic obstacle avoidance
- Intelligent user interface
- Navigation and mapping

Brain-computer interfaces

- "A direct brain-computer interface is a device that provides the brain with a new, non-muscular communication and control channel." (Wolpaw et al. 2002)
- Electrical signals from surface or implanted electrodes can control assistive technologies
- Current research seeks methods to extract more information from signals
- Not all subjects perform equally well
- Of special interest in assistive robotics research

Brain-computer interfaces

- Surface electrodes (EEG) are inherently low bandwidth
- USF P300 System
- Surgically implanted cortical electrodes can improve bandwidth:



University of Utah

- Machine learning algorithms improve decoding of EEG and cortical signals

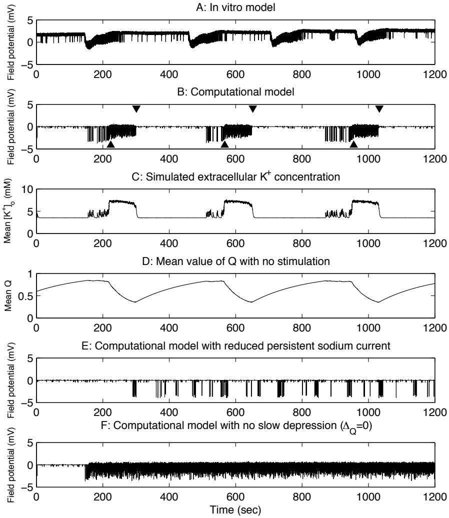
Simulation of biological systems

- Computer models can provide simulated data for a wide range of biological phenomena:
  - Cellular growth and development
  - **Nervous system activity**
  - Motor control (eye, arm, etc.)
  - Population dynamics (predator-prey relationships, e.g.)
  - Disease transmission and progression
- Such models may be used to make novel predictions for further research or to evaluate potential therapies.

Nervous system activity

- Builds on early quantitative models of the nervous system (e.g Hodgkin and Huxley, 1952)
- Relies on a simplified model of neurons and synapses
- Calculates timecourse of network behavior using numeric integration
- Can be used to predict effects of different connectivity patterns, drug effects, etc.
- Several major labs and publications emphasize these techniques

Nervous system activity



Vincent et al. 2011

Medical treatment optimization

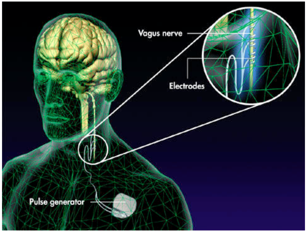
- Traditional approaches often rely on educated guesses
- Treatment and trials are simple and relatively static
- New emphasis on *adaptive treatment design* or *dynamic treatment regimes*.
- *Reinforcement learning* has been applied to optimize:
  - Antiretroviral drug treatments for HIV (Ernst et al. 2005)
  - Treatment for chronic depression (Pineau et al. 2007)
  - Lung cancer treatment (Zhao et al. 2009)
  - **Electrical stimulation for epilepsy** (Guez et al. 2008)
- These new methods suggest changes in clinical trial methodology (Collins et al. 2005)

About epilepsy

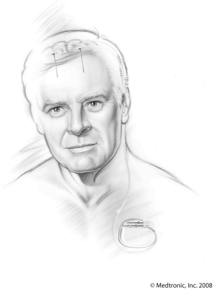
- A disorder characterized by abnormal periods of electrical activity in the brain, called seizures
  - Affects $\sim$1% of the population
  - Multiple causes - genetics, injury, tumors
  - Range of severities
  - Drugs have 60-70% success rate
  - Surgery required in extreme cases

Electrical stimulation for epilepsy treatment

- FDA-approved devices stimulate the vagus nerve
- Pending devices use deep brain stimulation
- No certain explanation for efficacy
- Existing devices are *open loop*
- Also used to treat Parkinson's disease, depression, etc.

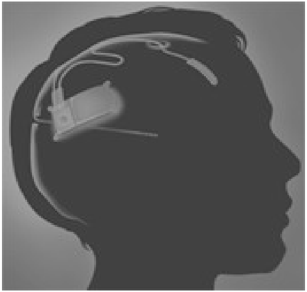


Cyberonics, Inc.



Responsive stimulation devices

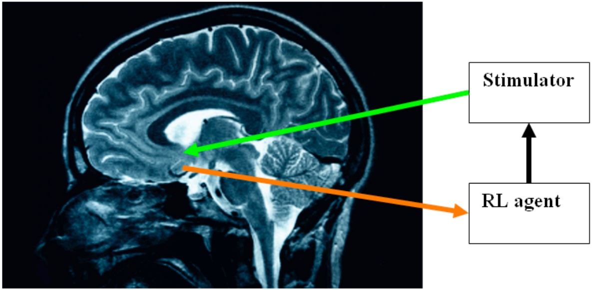
- "Responsive stimulation" (i.e. closed loop) devices are in preliminary trials
- At present, these implement a "detect and stimulate" policy
- Unclear whether *prediction* of seizures is possible



Neuropace, Inc.

Goal of our research

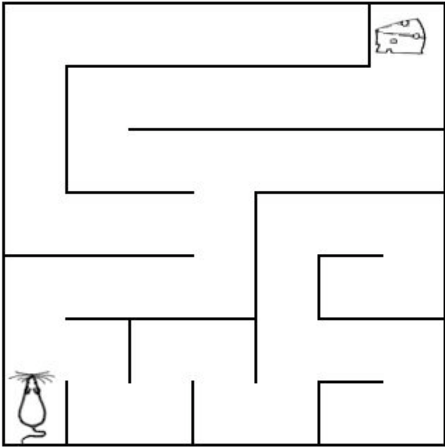
- An adaptive treatment algorithm using reinforcement learning
  - Improved efficacy
  - Reduced side effects
  - Increased battery life



What is reinforcement learning?

Agents that "learn by doing"

Inspired by ideas from psychology:



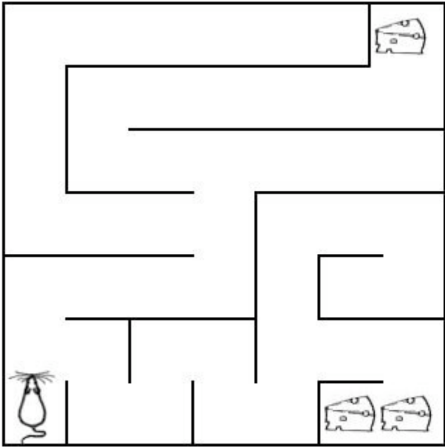
www.ratbehavior.org

What is reinforcement learning?

- Initially, the rat "explores" the environment, making many errors
- After many trials, the rat will move more quickly and accurately towards the goal, "exploiting" its knowledge

What is reinforcement learning?

Exploration vs. exploitation:



What is reinforcement learning?

- A “semi-supervised” machine learning method. An *agent* explores an *environment* consisting of:
  - A set of *states* describing the environment.
  - A set of *actions* that the agent may choose.
  - The next state depends only on the current state and action (the “Markov” assumption).
  - A *reward* or penalty is given for each action and state.
- The goal is to maximize the total discounted reward.
- The agent estimates the *value* of each state or state/action pair.
- The agent learns a *policy* which selects the best action for each state.

The mathematics of RL

- We treat the problem as a *Markov Decision Process*:
  - Sets of states $\mathcal{S}$ and actions $\mathcal{A}$.
  - A transition function which defines how states follow one another: $P(s, a, s') = Pr(s_{t+1} = s' | s_t = s, a_t = a)$.
  - A scalar reward function $r_t = R(s, a)$.
  - Some policy π, such that $a_t = \pi(s_t)$. The policy may be either deterministic or stochastic.
  - A discount factor, γ, which must be less than or equal to one. This causes the system to put a greater value on "immediate" rewards.
- The goal is to learn a policy to maximize the expected future rewards: $V = \sum_{t=0}^{T} \gamma^t R(s_t, \pi(s_t))$.

Q-functions

- Many of the simple RL algorithms work by estimating something we call the "Q-function": $q = Q(s, a)$.
- This is a scalar function which returns a value answering the question: "What is the value of taking action a in state s, under some assumed policy?"
- A deterministic policy can be derived from our $Q-$function: $\pi(s) = \arg\max_{a \in \mathcal{A}} Q(s, a)$.
- We can do this *without* knowing the state transition function or the reward function.

How does RL work?

- The agent begins in some initial state s
- For each time step:
  - The agent chooses an action a, based on its estimates of the values of possible future states, $Q(s, a)$
  - The agent performs the action, and observes the new state s', and receives a reward r
  - The agent improves its estimate of the current value of each state and action using a "temporal difference" rule:

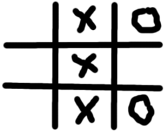
$$Q(s, a) = Q(s, a) + \alpha(r + \gamma \max_a Q(s', a) - Q(s, a)) \quad (1)$$

where α is a learning rate.

Successes of reinforcement learning

- Demonstrated ability to find good solutions in randomized or poorly-modeled problems:
  - Backgammon (Tesauro 1995)
  - Elevator scheduling (Crites and Barto 1996,1998)
  - Helicopter flight (Ng et al. 2004,2007)
  - Tetris (Farias and van Roy, 2006)
- Some serious challenges, however:
  - Tends to require lots of data for training
  - Hard to use in large or continuous state/action spaces

Easy RL: Tic Tac Toe



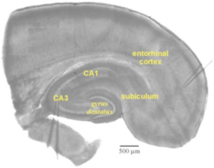
- A simple problem:
  - Finite: Countable states (board positions) and actions
  - Deterministic: Any legal action leads to a unique state
  - Simple actions: Limited set of legal moves
- Reward of $+1$ for a win, 0 for a loss (for example)
- As it visits states, an agent estimates the state's value using the temporal difference rule
- Agent must *exploit* knowledge and *explore* alternatives
- Given enough games, the agent is very likely to discover the best action for each state

Difficult RL: Adaptive stimulation

- A *much* more difficult problem:
  - Infinite: Many continuous state variables, derived from recordings of electrical activity
  - Non-deterministic: State changes aren't perfectly predictable
  - Complex actions: e.g. continuous stimulation frequency
- Reward may be, for example, 0 if normal, -1 for stimulation, -100 for seizure
- Preliminary algorithms have shown some promise using *in vitro* data

Current research

- RL algorithms trained *off line*
- We have begun testing the algorithms using *in vitro* experiments.



Hungarian Academy of Sciences

- Slices of rat brain tissue are treated with drugs that cause “seizure-like” activity, and we attach recording and stimulating electrodes.
- Preliminary results show that an RL agent can suppress seizures with less stimulation

Other medical applications of RL

- Use RL to optimize other chronic treatment:
  - Drug therapy for mental illness
  - Radiation therapy for cancer
  - Structured treatment interruptions for HIV
  - Substance abuse
- Design drug trials to allow for RL optimization of treatment (Murphy, 2005).

Conclusion

- Medical and biological research is now driving, and being driven by, developments in computer science.
- More and more data is being produced - the problem is often how to process it automatically.
- This requires something a bit deeper than off-the-shelf applications.
- Lots of opportunities for interdisciplinary collaboration.