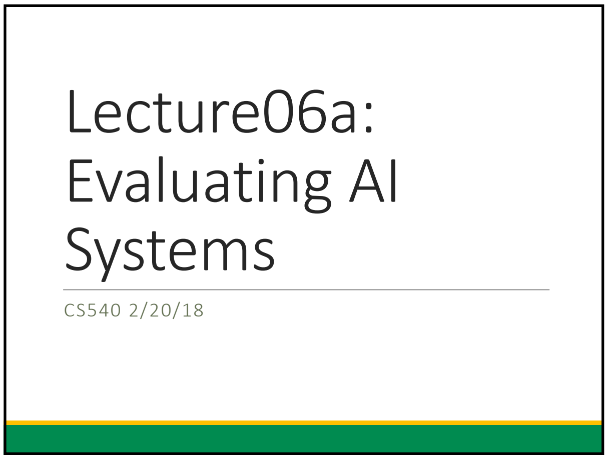
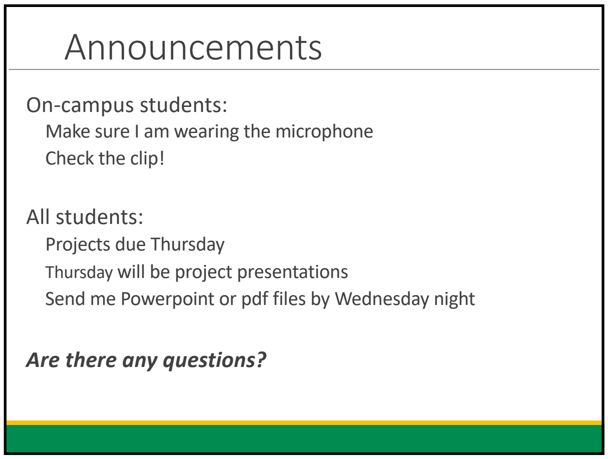


Lecture06a: Evaluating AI Systems

CS540 2/20/18

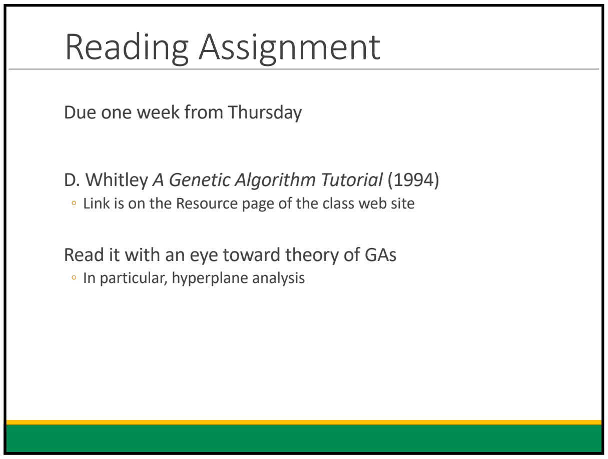


Announcements

On-campus students:
- Make sure I am wearing the microphone
- Check the clip!

All students:
- Projects due Thursday
- Thursday will be project presentations
- Send me Powerpoint or pdf files by Wednesday night

Are there any questions?

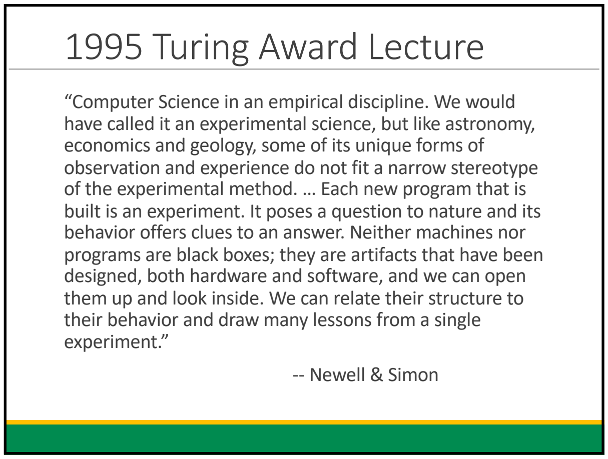


Reading Assignment

Due one week from Thursday

D. Whitley *A Genetic Algorithm Tutorial* (1994)
- Link is on the Resource page of the class web site

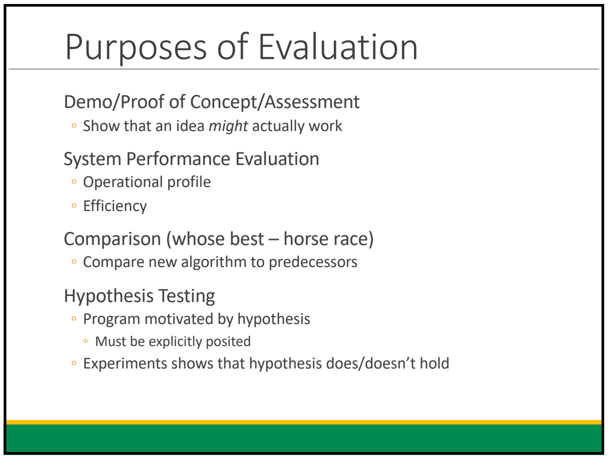
Read it with an eye toward theory of GAs
- In particular, hyperplane analysis



1995 Turing Award Lecture

"Computer Science in an empirical discipline. We would have called it an experimental science, but like astronomy, economics and geology, some of its unique forms of observation and experience do not fit a narrow stereotype of the experimental method. … Each new program that is built is an experiment. It poses a question to nature and its behavior offers clues to an answer. Neither machines nor programs are black boxes; they are artifacts that have been designed, both hardware and software, and we can open them up and look inside. We can relate their structure to their behavior and draw many lessons from a single experiment."

-- Newell & Simon



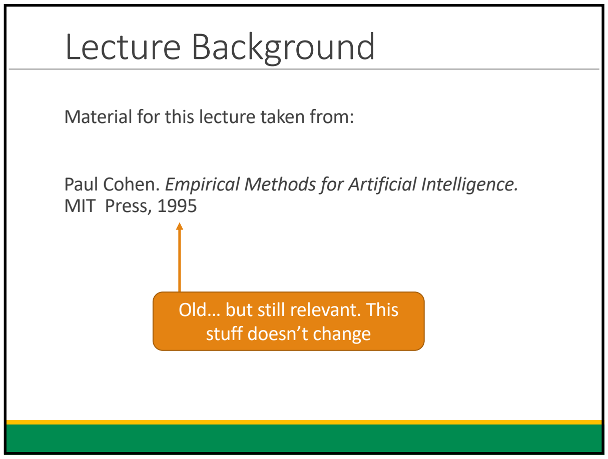
Purposes of Evaluation

Demo/Proof of Concept/Assessment
- Show that an idea *might* actually work

System Performance Evaluation
- Operational profile
- Efficiency

Comparison (whose best – horse race)
- Compare new algorithm to predecessors

Hypothesis Testing
- Program motivated by hypothesis
  - Must be explicitly posited
- Experiments shows that hypothesis does/doesn't hold



Lecture Background

Material for this lecture taken from:

Paul Cohen. *Empirical Methods for Artificial Intelligence.* MIT Press, 1995

Old… but still relevant. This stuff doesn't change

3 Research Questions

1. Description
   “What will happen when…” (Try it and see)
2. Prediction
   “Does this model predict what will happen when…” (Yes, No, Maybe)
3. Explanation
   ”Does this model explain what will happen when…” (Yes, No, Maybe)

How Programs are like Rats

1. Place rat (program) in a maze (problem, platform)
2. Vary training or reward or… (program parameters)
3. Measure effect on rat’s (program’s) time to traverse the maze (quality of solution, computation time…)
4. Analyze data and draw conclusions

5 Components of Empirical Studies

1. Subjects (rat, program, …)
2. Task to perform (maze, problem, …)
3. Environments in which to perform (lab, data…)
4. Metrics of performance
5. Protocol to follow

Empirical Study Types

1) Assessment Studies

- Characterize a program’s behavior. Determine what factors matter, and what measure best quantify the phenomena of interest
- “fishing expedition”
- Extensive visualization/summarization, less rigorous experiments
- Questions:
  - What problems are particularly difficult?
  - Which algorithms a re hopeless?
  - Does Quality of solution vary much?

Empirical Study Types (II)

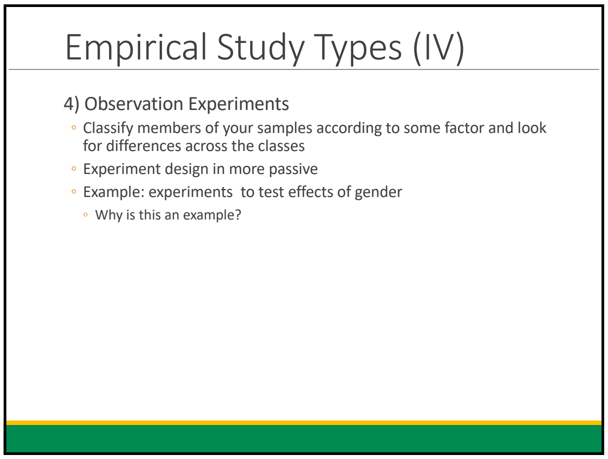
2) Exploratory Studies

- Identify patterns that suggest relationships between changes/parameters and performance results
- Pilot study: run small study to identify problems
- Experiment design with analysis
- Questions
  - What characteristics make a problem more/less difficult?
  - Which parameters matter, and how much?

Empirical Study Types (III)

3) Manipulation Experiments

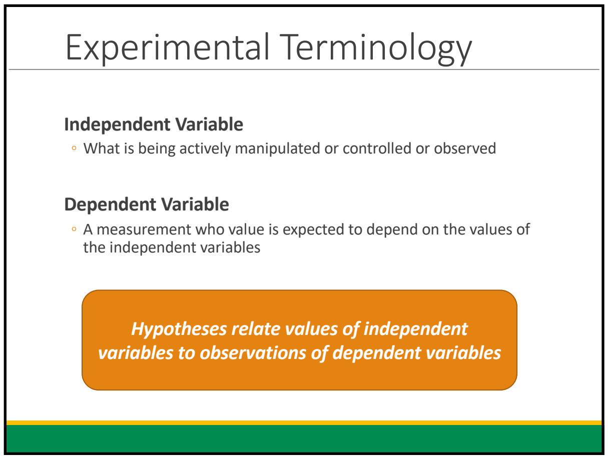
- Confirm/discredit hypothesis by actively manipulating factors
- Classical multi-factor experiments
- Visualization/summarization, experiment design, statistical analysis
- Questions
  - Phrased before-hand as hypotheses
  - Each hypothesis measured by a statistical analysis
  - Conclusions based on results of this analysis



Empirical Study Types (IV)

4) Observation Experiments

- Classify members of your samples according to some factor and look for differences across the classes
- Experiment design in more passive
- Example: experiments to test effects of gender
  - Why is this an example?



Experimental Terminology

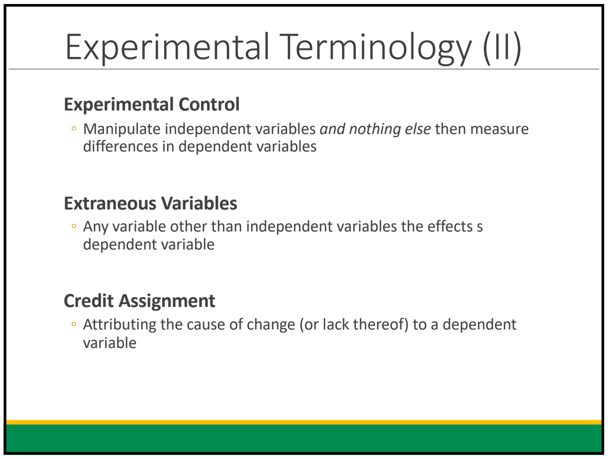
Independent Variable

- What is being actively manipulated or controlled or observed

Dependent Variable

- A measurement who value is expected to depend on the values of the independent variables

Hypotheses relate values of independent variables to observations of dependent variables



Experimental Terminology (II)

Experimental Control

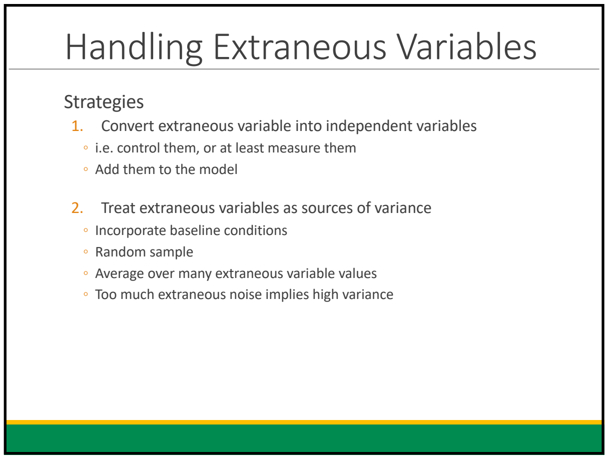
- Manipulate independent variables *and nothing else* then measure differences in dependent variables

Extraneous Variables

- Any variable other than independent variables the effects s dependent variable

Credit Assignment

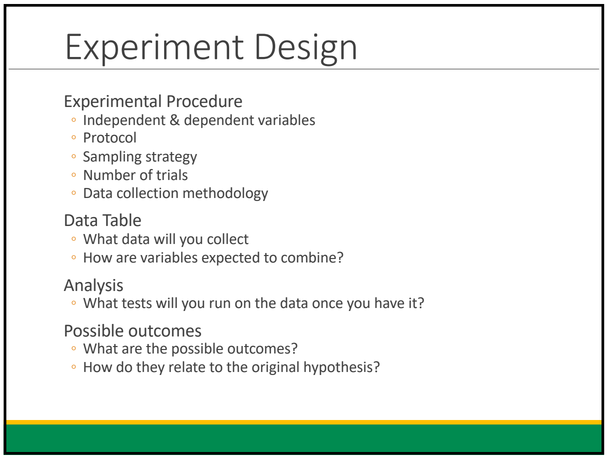
- Attributing the cause of change (or lack thereof) to a dependent variable



Handling Extraneous Variables

Strategies

1. Convert extraneous variable into independent variables
   - i.e. control them, or at least measure them
   - Add them to the model
2. Treat extraneous variables as sources of variance
   - Incorporate baseline conditions
   - Random sample
   - Average over many extraneous variable values
   - Too much extraneous noise implies high variance



Experiment Design

Experimental Procedure

- Independent & dependent variables
- Protocol
- Sampling strategy
- Number of trials
- Data collection methodology

Data Table

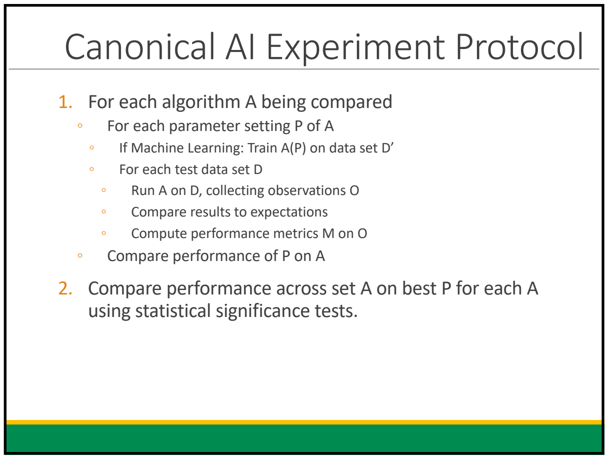
- What data will you collect
- How are variables expected to combine?

Analysis

- What tests will you run on the data once you have it?

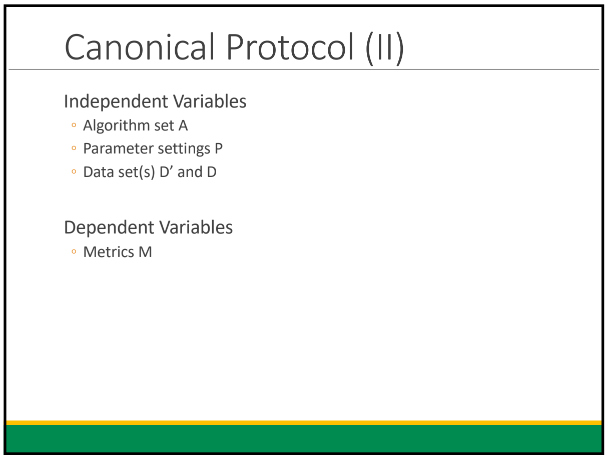
Possible outcomes

- What are the possible outcomes?
- How do they relate to the original hypothesis?



Canonical AI Experiment Protocol

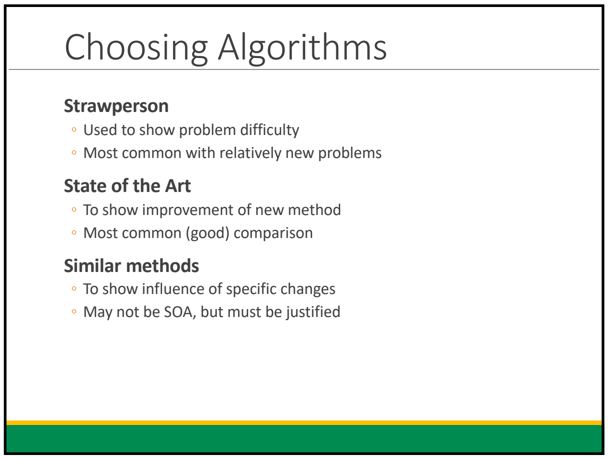
1. For each algorithm A being compared
   - For each parameter setting P of A
     - If Machine Learning: Train A(P) on data set D'
     - For each test data set D
       - Run A on D, collecting observations O
       - Compare results to expectations
       - Compute performance metrics M on O
   - Compare performance of P on A
2. Compare performance across set A on best P for each A using statistical significance tests.



Canonical Protocol (II)

Independent Variables
- Algorithm set A
- Parameter settings P
- Data set(s) D' and D

Dependent Variables
- Metrics M

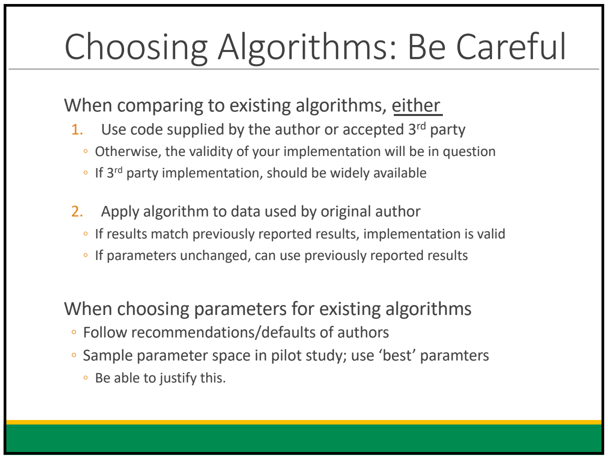


Choosing Algorithms

Strawperson
- Used to show problem difficulty
- Most common with relatively new problems

State of the Art
- To show improvement of new method
- Most common (good) comparison

Similar methods
- To show influence of specific changes
- May not be SOA, but must be justified

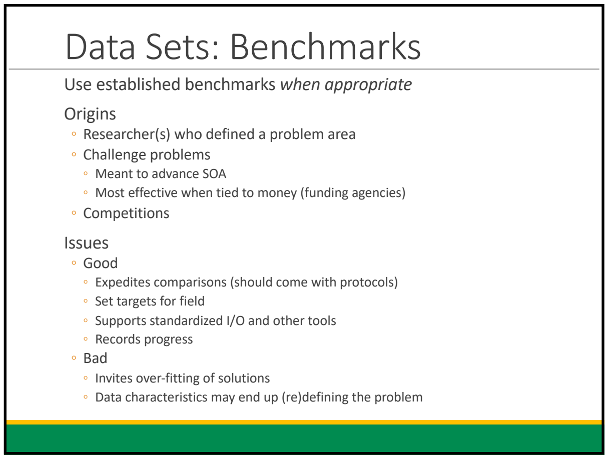


Choosing Algorithms: Be Careful

When comparing to existing algorithms, either

1. Use code supplied by the author or accepted 3rd party
   - Otherwise, the validity of your implementation will be in question
   - If 3rd party implementation, should be widely available
2. Apply algorithm to data used by original author
   - If results match previously reported results, implementation is valid
   - If parameters unchanged, can use previously reported results

When choosing parameters for existing algorithms
- Follow recommendations/defaults of authors
- Sample parameter space in pilot study; use 'best' paramters
  - Be able to justify this.

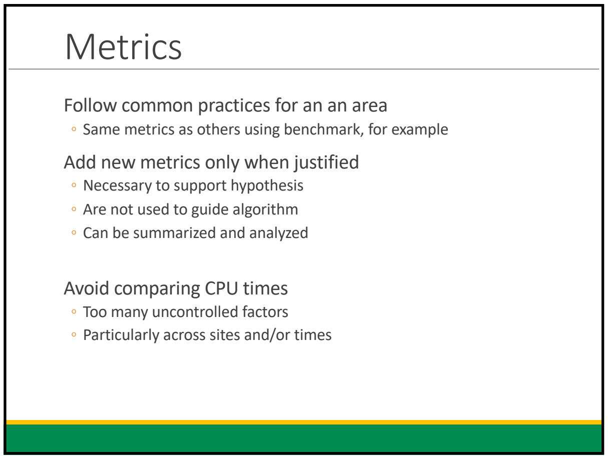


Data Sets: Benchmarks

Use established benchmarks *when appropriate*

Origins
- Researcher(s) who defined a problem area
- Challenge problems
  - Meant to advance SOA
  - Most effective when tied to money (funding agencies)
- Competitions

Issues
- Good
  - Expedites comparisons (should come with protocols)
  - Set targets for field
  - Supports standardized I/O and other tools
  - Records progress
- Bad
  - Invites over-fitting of solutions
  - Data characteristics may end up (re)defining the problem



Metrics

Follow common practices for an an area
- Same metrics as others using benchmark, for example

Add new metrics only when justified
- Necessary to support hypothesis
- Are not used to guide algorithm
- Can be summarized and analyzed

Avoid comparing CPU times
- Too many uncontrolled factors
- Particularly across sites and/or times