Tim Menzies
tim@menzies.us
LCSEE, WVU

Oct-8-10

Welcome to the Singularity?

Outline

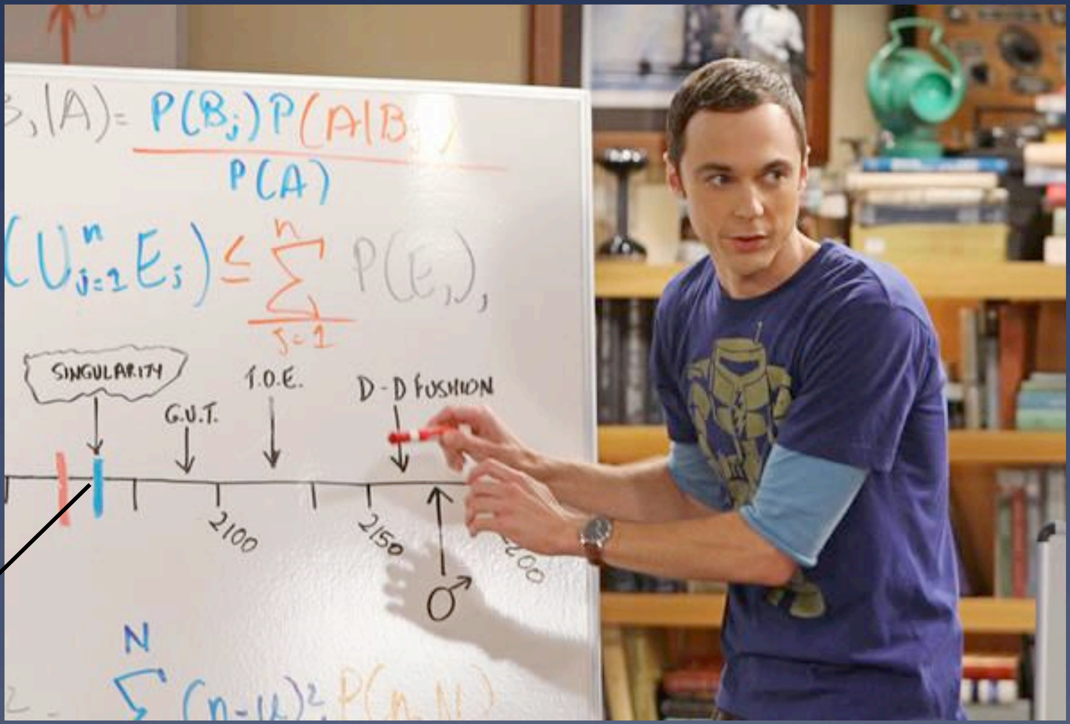
* There's this "thing"
  * Called the singularity
* That some people think will happen real soon
* That others think is a load of cr*p
* Which I think is already here (ish).

What is the singularity?

Leading experts predict…

* The singularity
  * An event, around 2060, where history will fundamentally change
  * After which, we lose our ability to make any further predictions.
* Which experts, you ask?

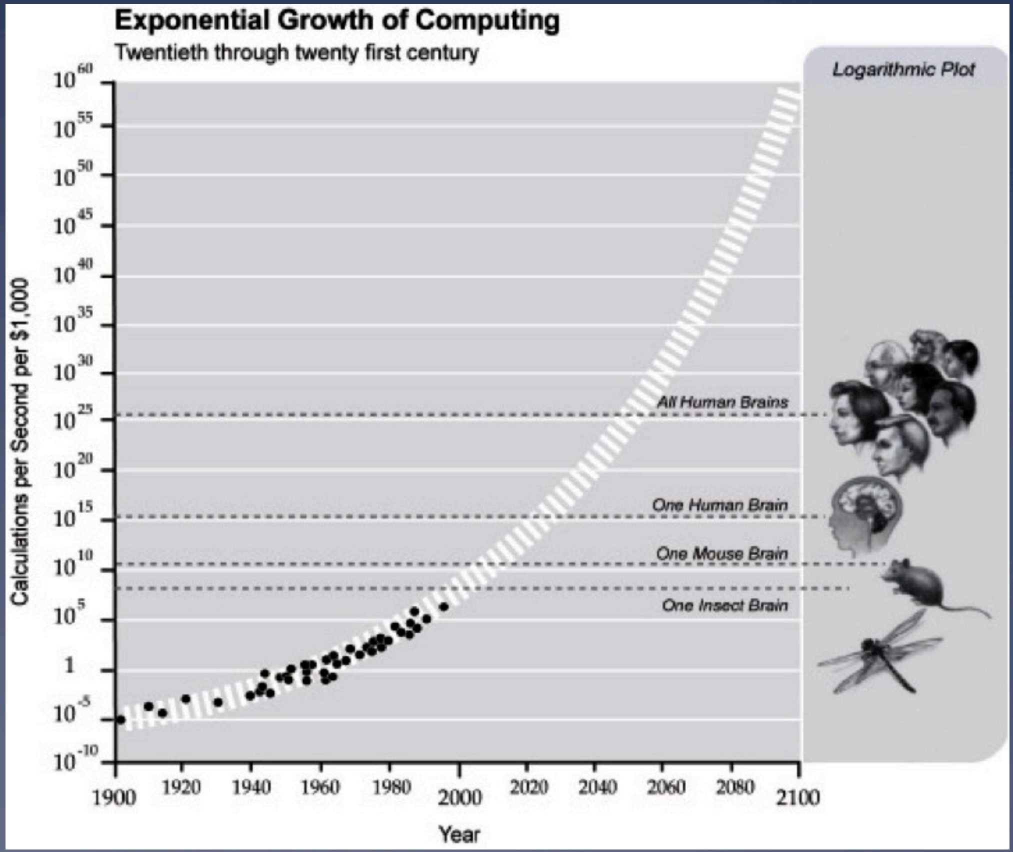
Sheldon Cooper (Ph.D.) CalTech



2060: "The earliest estimate of the Singularity, when man will be able to transfer his consciousness into machines and achieve immortality."
– S. Cooper

Dr. Cooper is not totally correct

- More precisely, the singularity is when
  - Technological progress becomes so extremely rapid, due to positive feedback
- Assumes extrapolation of current trends
  - Is that valid?
  - A point we will return too
- One common prediction a singularity event:
  - The creation of smarter-than-human intelligence
  - Human's would lose their ability to model/ predict/ control the future.
  - Since AIs can enhance their minds faster than humans.
  - And humans can't predict actions of more intelligent entities

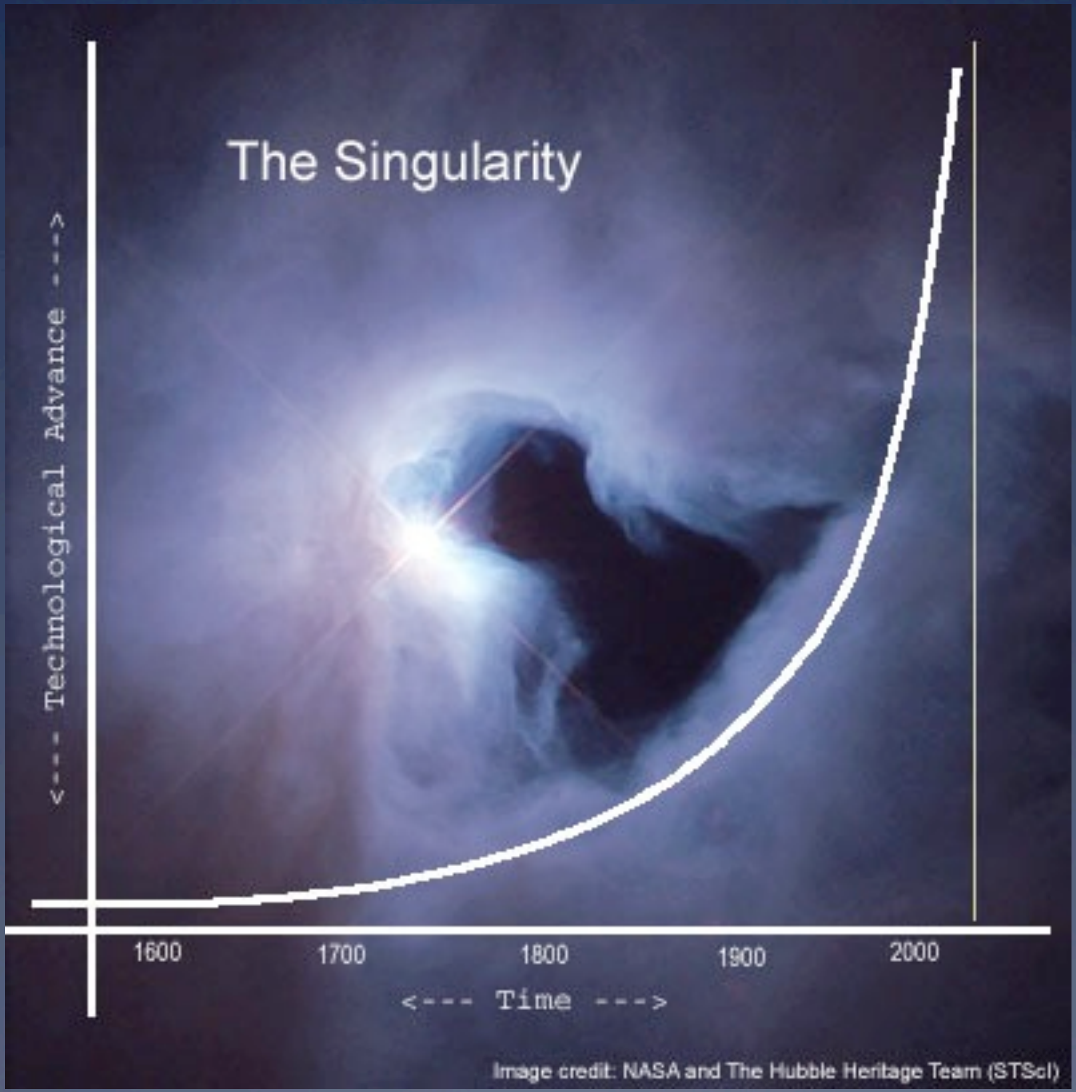


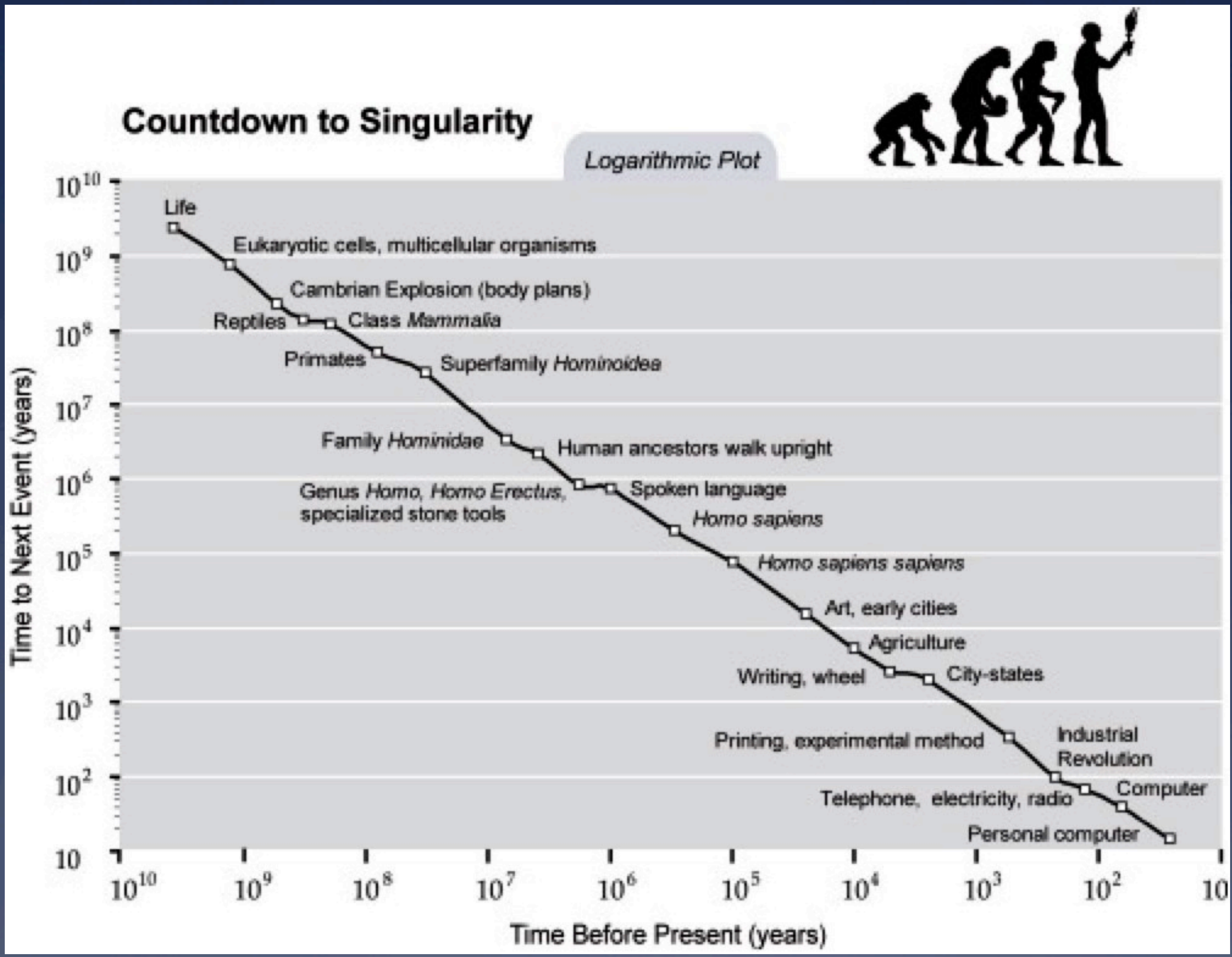
So, soon something may be thinking faster than you or me

- But it may not be Sheldon

Evidence for the singularity

Exponential trends

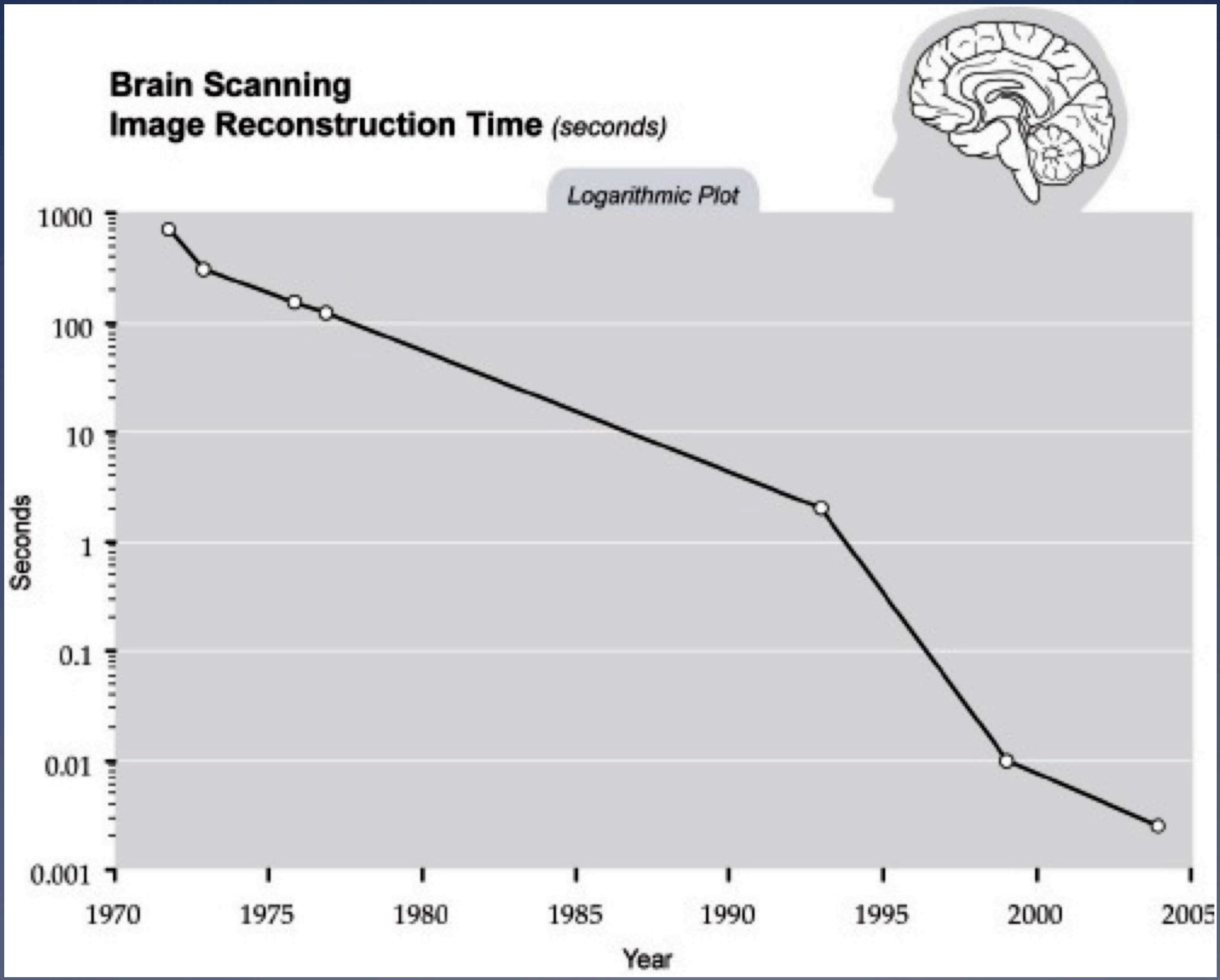




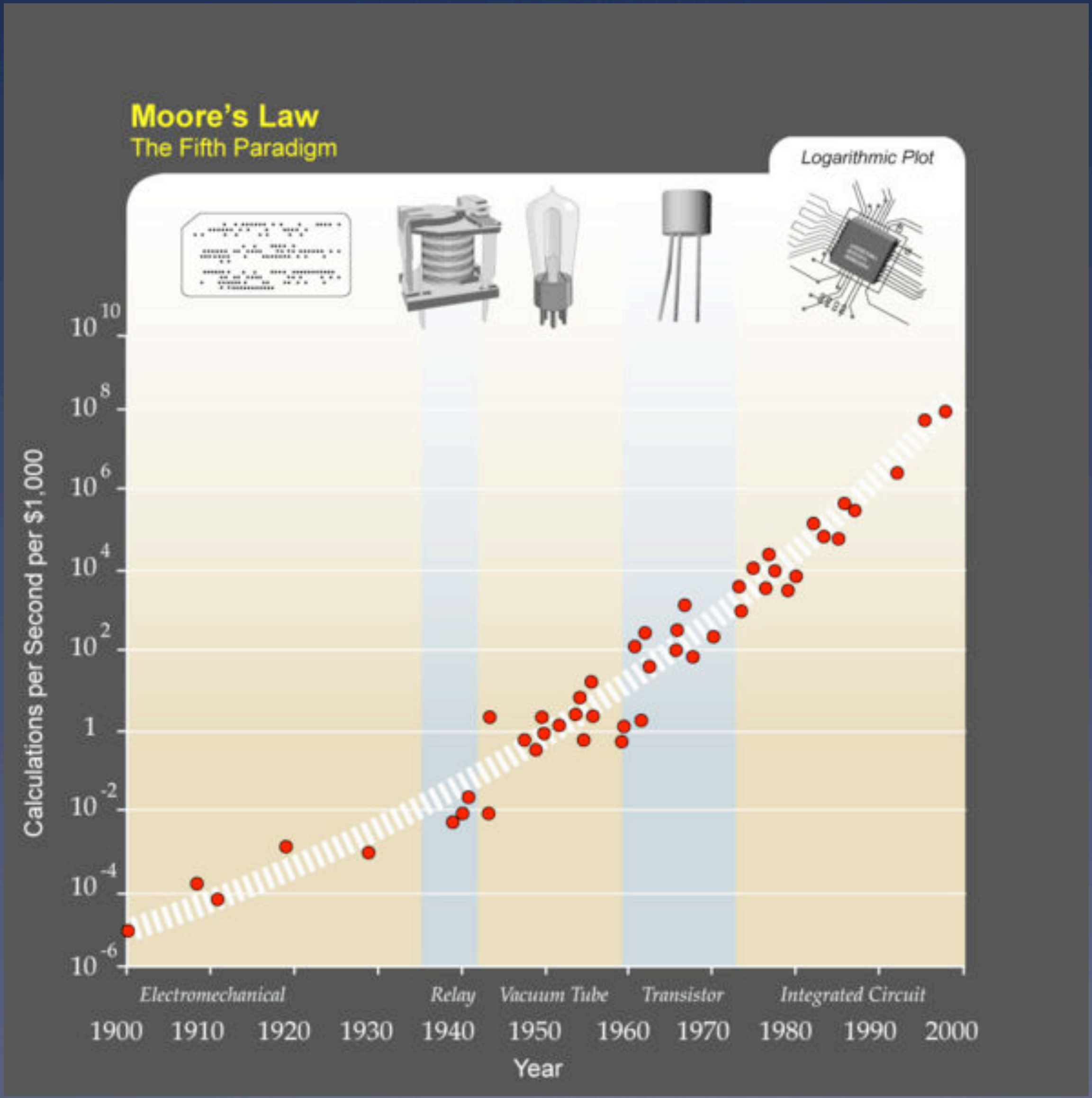
Complaint: selection bias in the points that Kurzweil chooses to use. For example, biologist PZ Myers points out that many of the early evolutionary "events" were picked arbitrarily

5 lists of paradigm shifts for key historic events shows an exponential trend. The lists' compilers include Carl Sagan, Paul D. Boyer, Encyclopædia Britannica, American Museum of Natural History, and University of Arizona.

Exponential trends

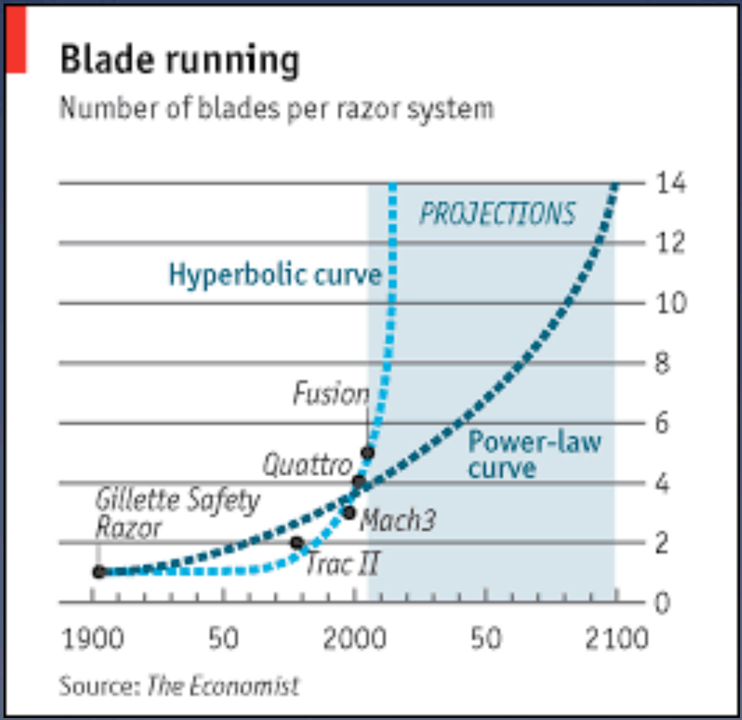


Exponential trends



Evidence against the singularity

Extrapolating is bogus? e.g. number of razor blades per razor



http://www.economist.com/node/5624861?story_id=5624861

Steven Pinker, 2008

- "(...) There is not the slightest reason to believe in a coming singularity. "
- "The fact that you can visualize a future in your imagination is not evidence that it is likely or even possible. "
- "Look at domed cities, jet-pack commuting, underwater cities, mile-high buildings, and nuclear-powered automobiles — all staples of futuristic fantasies when I was a child that have never arrived."
- "Sheer processing power is not a pixie dust that magically solves all your problems. (...)"

Some trends are nearly over: e.g. Moore's Law is dead, according to Gordon Moore, its inventor

* 13 April 2005
* "In terms of size [of transistor] you can see that we're approaching the size of atoms which is a fundamental barrier, "
* "We have another 10 to 20 years before we reach a fundamental limit. By then they'll be able to make bigger chips and have transistor budgets in the billions."

Ways through the limits?

* Quantum computing
  * More bits per bit
* Robert Yung (CTO) Tessera,
  * separating out processing to several cores
  * "scaling out" as opposed to scaling down.

* EUV extreme ultraviolet lithography
  * Ordinary lithography uses ultraviolet light, wavelength = 193 nm
  * EUV uses wavelengths of about 14 nanometers.
  * Open issue: will the manufacturing costs get so high, that the value of their lifetime productivity can never justify it.

But lets go back to the AI

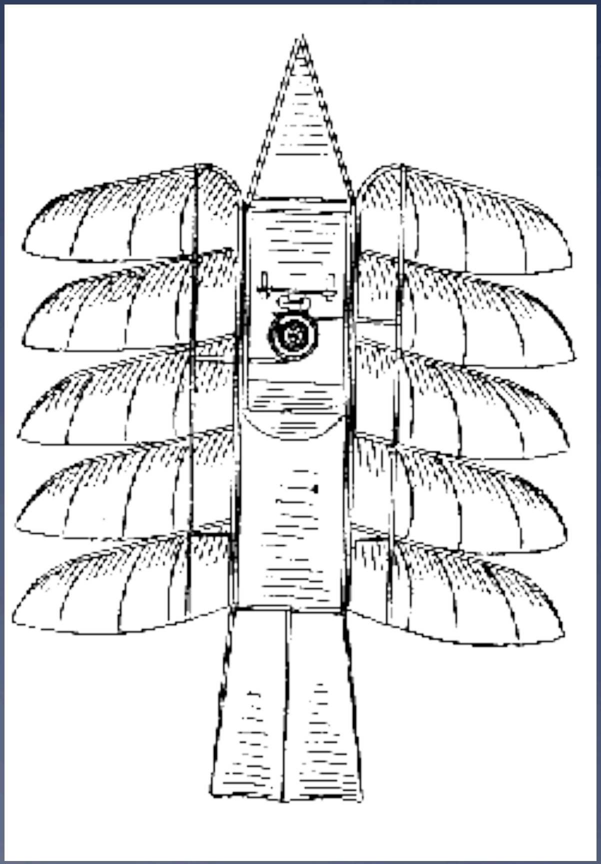
- Some critics go so far as to assert that no computer or machine will ever achieve human intelligence,
- While others hold that the definition of intelligence is irrelevant if the net result is the same
  - Who cares if it is "human intelligence"
  - Smart is smart

Machines don't think like we do

* There is a abstract notion of flying/thinking that is independent of birds/ humans.
  * "Intelligence does not require bulk, Mr. Scott" – Spock
* "Intelligence" could be coded any number of ways (biological, mechanical, a collection of wind-powered beer cans, whatever)

Do you object?

- "The only thing that can be rational like me is another person"?
  - That's like saying airplanes don't really fly (no wing flapping)
- Every computer scientist knows this to be true:
  - There are computational properties, independent of processor the algorithm runs on, or the implementation language.
  - The idea is different to the substrate
  - "computer science is no more about computers than astronomy is about telescopes"
    - Dijkstra (and he could have been talking about AI)

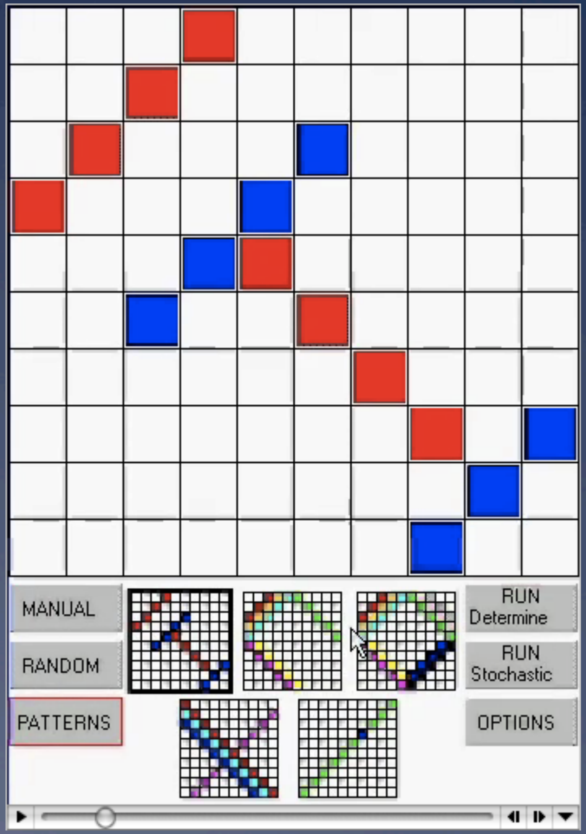


The Platonic Beast

- Thinking about thinking is hard
- Lets do something simpler- like walking
- Should a robot could/should walk like us? (see http://people.cs.ubc.ca/~pai/movies/beast.mpg)
  - walks by throwing a spare limb over its head
  - Such a move would tear us apart
  - but it's natural for that kind of walking thing.

Stochastic search

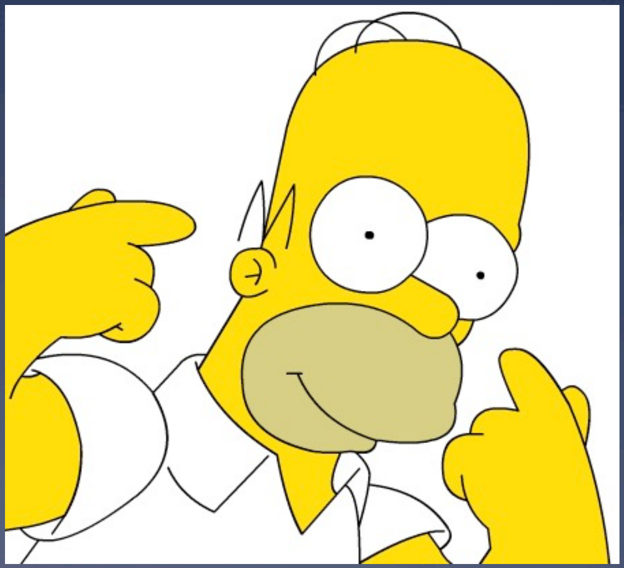
- "Latin's square": no 2 same colors on the same row or column
  - Method 1: deterministic exhaustive theorem proving
  - Method 2: stochastic: makes an initial guess, then refines that guess based on local feedback.
  - http://menzies.us/csx72/img/latin.mov
  - Stochastic kills deterministic



- You would not expect a human to think using stochastic search (too much CPU twiddling).
  - But for a computer, stochastics are useful since each local twiddle can be done very quickly
  - An AI may think wildly different to a human

Prediction

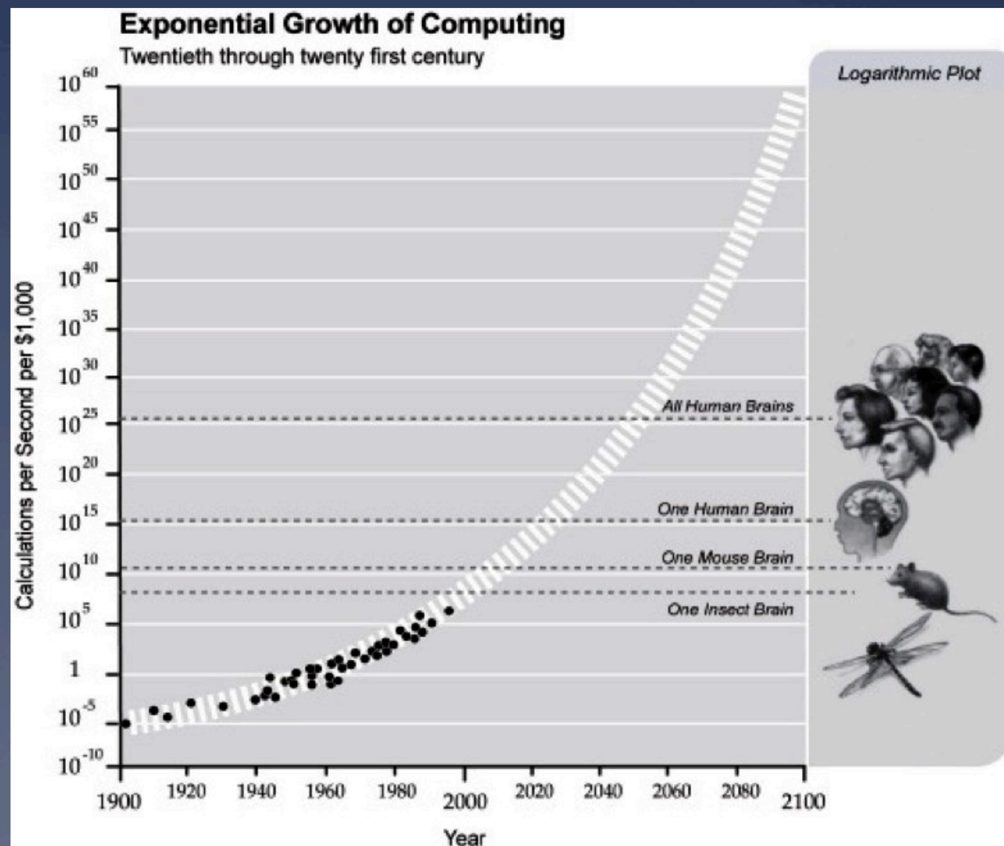
* The more we turn to computers,
  * The more we'll get answers which
    * Work
    * But which we don't understand



Is the AI singularity already here?

When is the AI singularity?

- Does the AI singularity:
  - Require decades of improvement in processor speed?
- Or have we already arrived at the AI singularity?



The Eureqa Machine

- Distilling Free-Form Natural Laws from Experimental Data
  - Michael Schmidt and Hod Lipson
  - SCIENCE VOL 324 3 APRIL 2009

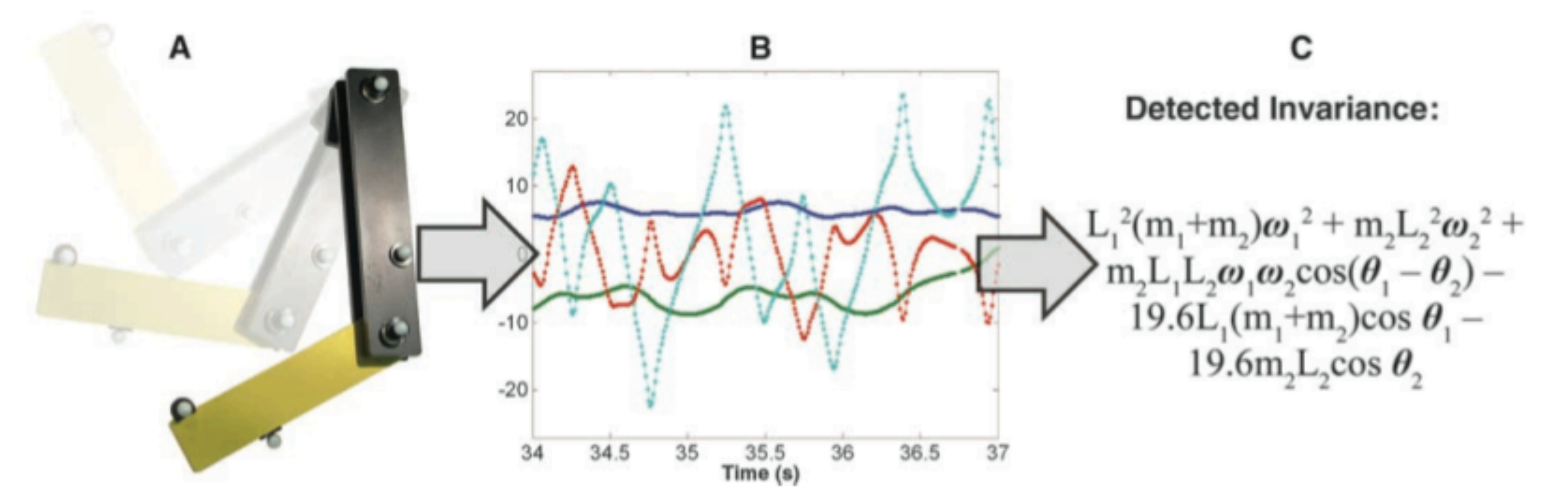
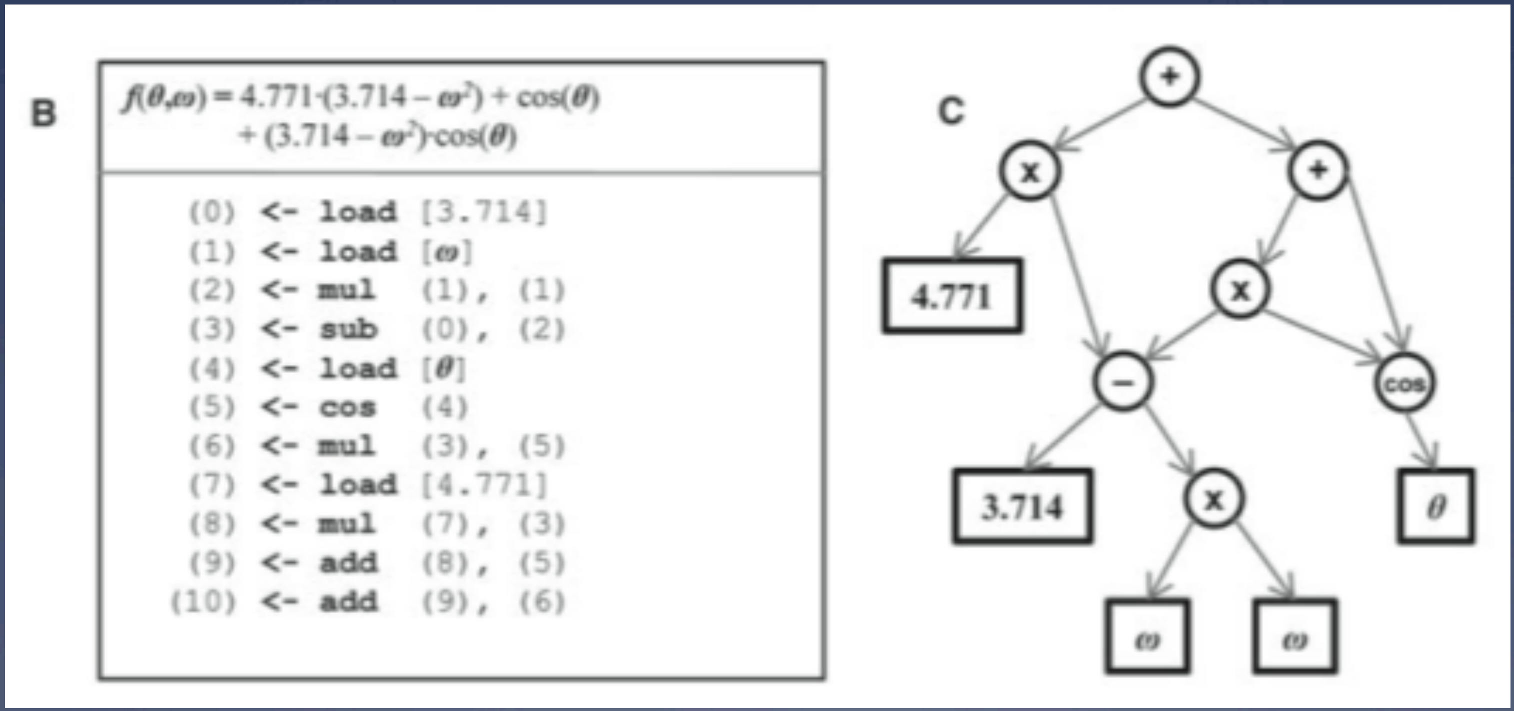
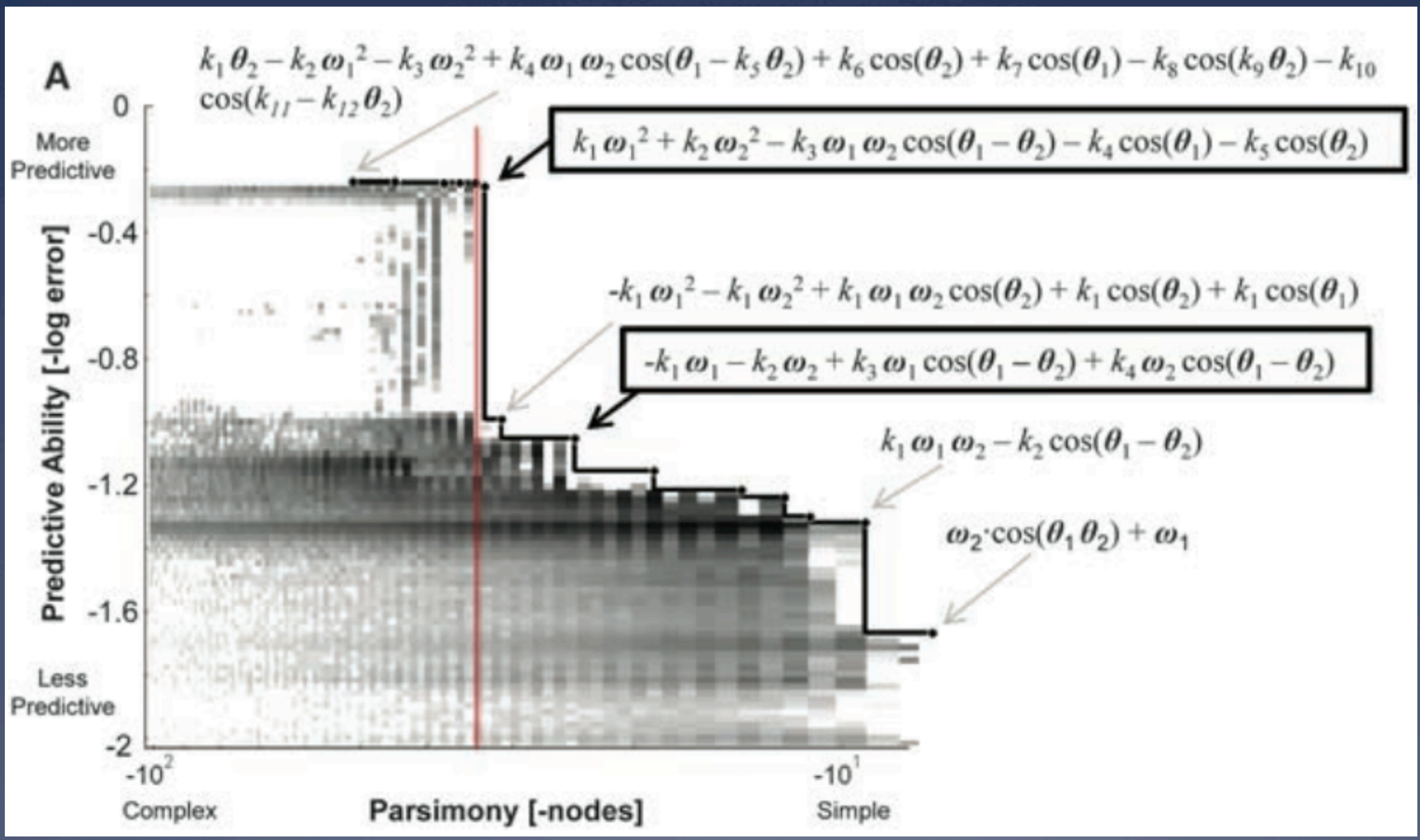


Fig. 1. Mining physical systems. We captured the angles and angular velocities of a chaotic double-pendulum (**A**) over time using motion tracking (**B**), then we automatically searched for equations that describe a single natural law relating these variables. Without any prior knowledge about physics or geometry, the algorithm found the conservation law (**C**), which turns out to be the double pendulum's Hamiltonian. Actual pendulum, data, and results are shown.

Searches the space of possible equations

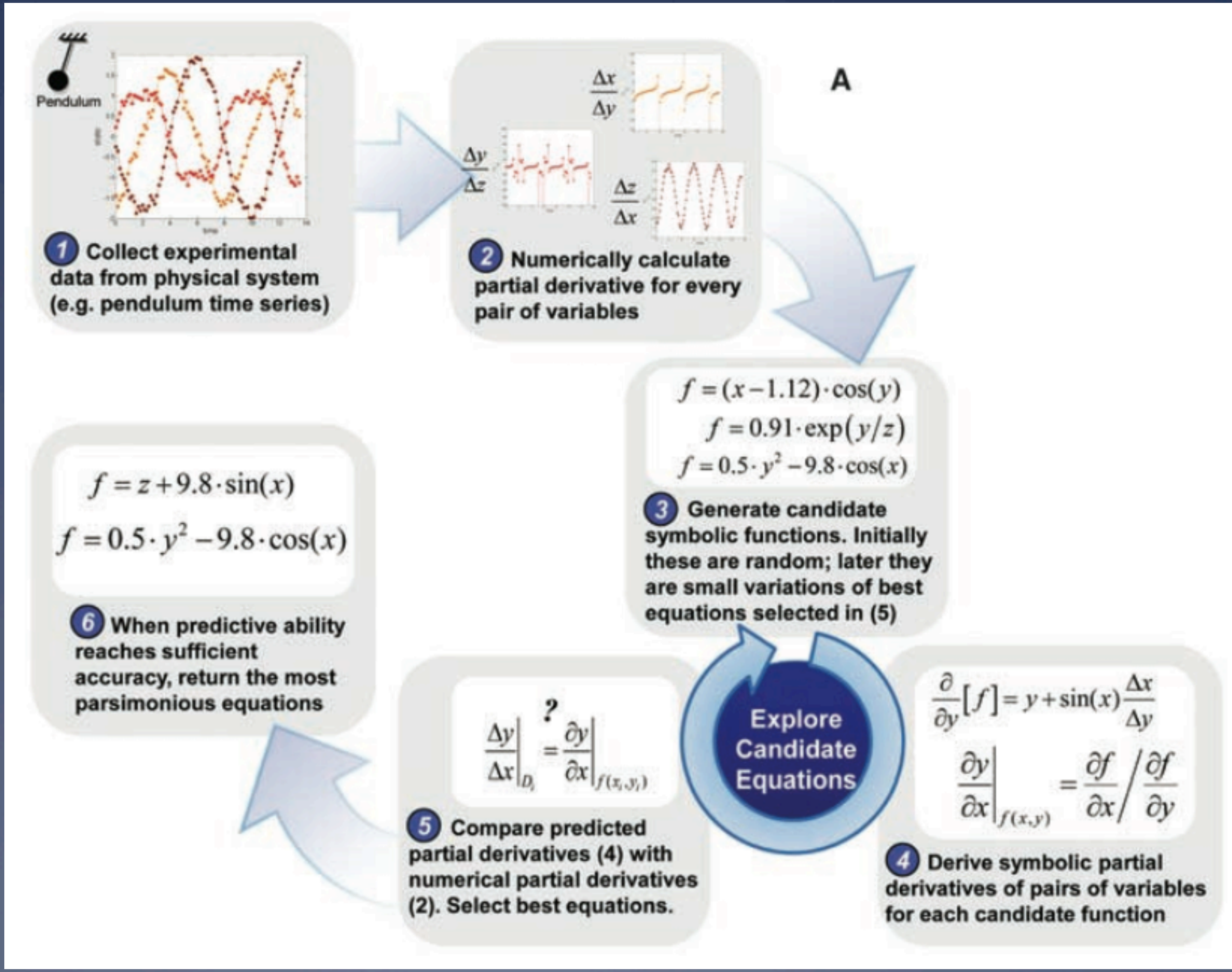


Trades off parsimony and predictive ability



Limits to the Eureqa machine

- Watches numeric data
- Infers error in partial differentials.
- Can't invent calculus.
- So Newton can't be replaced.
- Yet



Limits to belief

Eureqa and biology

* Single cell dynamics (bacteria):
  * how nutrients increase and decrease
* Crazy complex quilt of intra-cellular feedback
  * 1000s of effects
* Eureqa found two equations
  * Equations match data not used in training
  * Predicted results when tested on new data

Unpublishable

* Don't know what the equations mean
  * Got the answer , but not the insight
* The more we turn to computers,
  * The more we'll get answers we don't understand

Should we use laws that we can't explain?

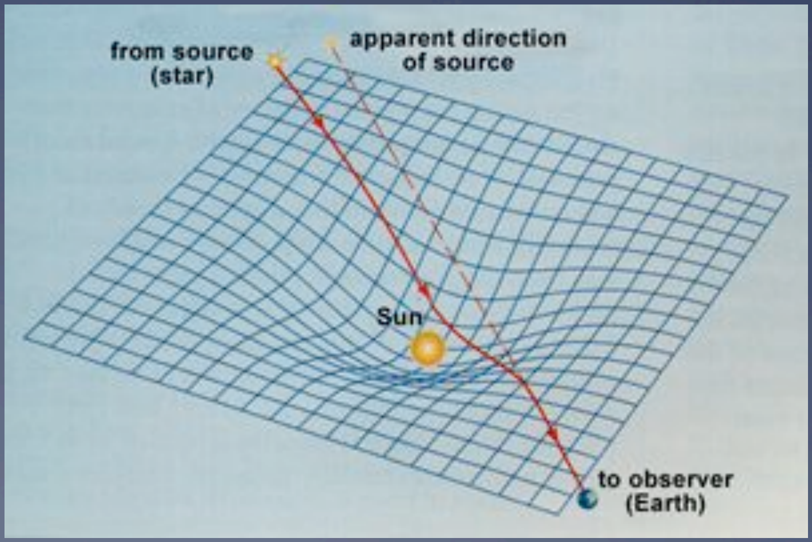
- Gravity $$F = G\frac{m_1 m_2}{r^2}$$

- 1846, rival astronomers John Adams (in England) and Urbain Leverrier (in France) raced to find a previously unseen planet that was disturbing the orbit of Uranus.

- Neptune was first sighted by Adams, then Leverrier, after both men pointed their telescopes at the precise point in the sky indicated by Newton's equations.

- But for 100s of years, there was no explanation for gravity

- Not till relativity and space-time curvature
  - 1600s till 1900s



Five types of theories (Gregor '06)

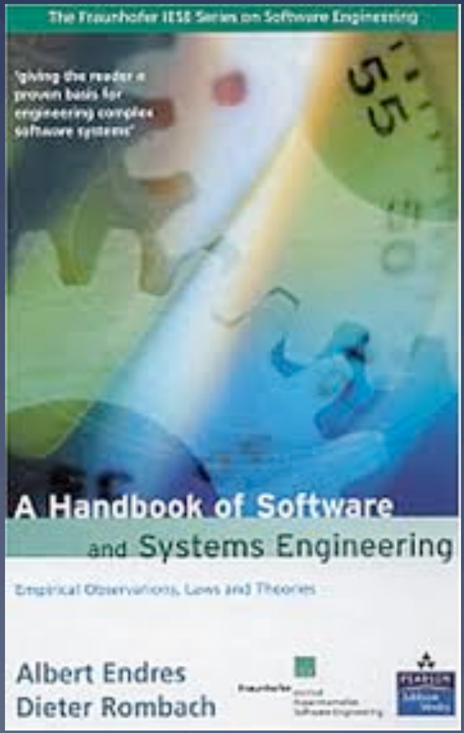
1. Lists of what is: taxonomies, ontologies
2. Explainers: e.g. case studies on historical data
   * No test on data not used in the analysis
3. Predictors
   * And may not don't explain why (Neural nets, Naïve Bayes)
4. Explain & predict
   * Learn a model: Decision tree learners, linear regression
   * Apply the model to predict
   * Browse the model to explain the predictions
5. Convincing models: on some domain-specific criteria
   * E.g. "goto considered harmful"

Post-singularity, expect these

Three kinds of statements (Endres & Rombach'03)

- Observations: what you see
- Laws:
  - predict observation B given observation A
- Theories:
  - explanations of laws

- Laws predict repeatable observations
- Theories explain laws
- Laws are either hypotheses (tentatively accepted) or conjectures (guesses)



- 100+ laws
  - No theories

Research into defect prediction

* Useful to (say) organize testing resources
  * Assumes that you don't have the resources to test everything 100% rigorously
* If something looks like its going to be bad
  * Allocate more resources to it
* Method: data mining on logs of prior defects
  * What predicts for defects?

Which of these most predict for software defects?

m = Mccabe			$v(g)$	cyclomatic_complexity
			$iv(G)$	design_complexity
			$ev(G)$	essential_complexity
locs	loc			loc_total (one line = one count
	loc(other)			loc_blank
				loc_code_and_comment
				loc_comments
				loc_executable
				number_of_lines (opening to closing brackets)
Halstead	h		N_1	num_operators
			N_2	num_operands
			μ_1	num_unique_operators
			μ_2	num_unique_operands
	H		N	length: $N = N_1 + N_2$
			V	volume: $V = N * log_2 \mu$
			L	level: $L = V^* / V$ where $V^* = (2 + \mu_2{}^*) log_2 (2 + \mu_2{}^*)$
			D	difficulty: $D = 1/L$
			I	content: $I = \hat{L} * V$ where $\hat{L} = \frac{2}{\mu_1} * \frac{\mu_2}{N_2}$
			E	effort: $E = V/\hat{L}$
			B	error_est
			T	prog_time: $T = E/18$ seconds

No common pattern

- pd, pf = prob.detection, prob.falseAlarm
- Data sets = pc1, mw1m kc3,cm1, pc2, kc4, pc4,pc4
- Learner = Naïve Bayes
- Attribute selection = WRAPPER

data	pd	pf	index of selected feature
pc1	48	17	3, 35, 37
mw1	52	15	23, 31, 35
kc3	69	28	16, 24, 26
cm1	71	27	5, 35, 36
pc2	72	14	5, 39
kc4	79	32	3, 13, 31
pc3	80	35	1, 20, 37
pc4	98	29	1, 4, 39
all	71	25	

ID	used in	what	type
1	2	loc_blanks	locs
3	2	call_pairs	misc
4	1	loc_code_and_command	locs
5	2	loc_comments	locs
13	1	edge_count	misc
16	1	loc_executable	locs
20	1	I	H'
23	1	B	H'
24	1	L	H'
26	1	T	H'
31	2	node_count	misc
35	3	μ_2	h
36	1	μ_1	h
37	2	number_of_lines	locs
39	2	percent_comments	misc

H' = derived Halstead
h = raw Halstead

Managing Software Development Projects

* Take a project described by
  * Task complexity
  * Skill of the analysts
  * Schedule pressure
  * Etc
* Assume partial partial control of some of these features
* Find changes that most
  * Reduce development time
  * Reduce defects
  * Reduce required staff
* Method: AI search algorithms
  * Model-based, instance-based

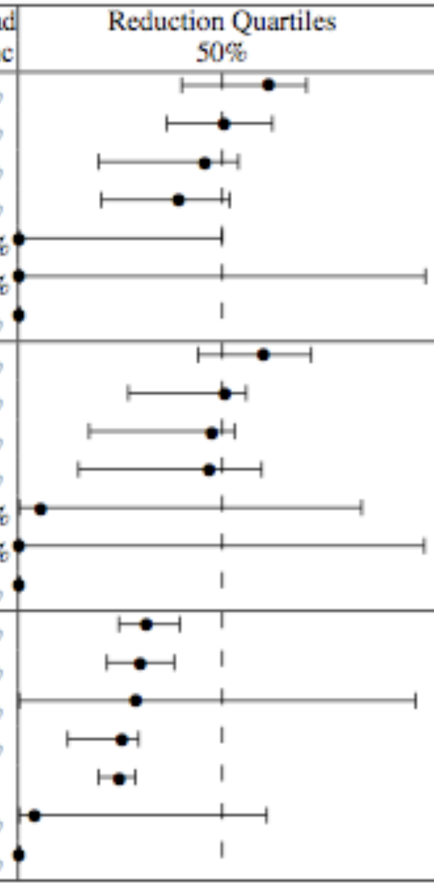
Which of these most predict for software development problems?

scale factors (exponentially decrease effort&cost)	prec: have we done this before?
	flex: development flexibility
	resl: any risk resolution activities?
	team: team cohesion
	pmat: process maturity
upper (linearly decrease effort&cost)	acap: analyst capability
	pcap: programmer capability
	pcon: programmer continuity
	aexp: analyst experience
	pexp: programmer experience
	ltex: language and tool experience
	tool: tool use
	site: multiple site development
	sced: length of schedule
lower (linearly increase effort&cost)	rely: required reliability
	data: secondary memory storage requirements
	cplx: program complexity
	ruse: software reuse
	docu: documentation requirements
	time: runtime pressure
	stor: main memory requirements
	pvol: platform volatility

Instance-based planner to reduce defects, effort, months

* NASA data
* W :
  * instance-based planner
* Other:
  * standard changes
* Median = 50^{th} %
* Spread = $(75\text{-}25)^{th}$ %
* Reduction = (init-final)/init

Rank	Goal	Change	Median Reduc	Spread Reduc
1	defects	W	61%	31%
2	defects	ProcMaturity	51%	26%
2	defects	ReduceFunct	46%	34%
2	defects	Tools&Tech	39%	32%
2	defects	ReduceQuality	0%	382%
2	defects	Personel	0%	100%
3	defects	RelaxScedule	-30%	78%
1	effort	W	60%	28%
2	effort	ProcMaturity	51%	29%
2	effort	ReduceFunct	48%	36%
2	effort	Tools&Tech	47%	45%
2	effort	ReduceQuality	5%	257%
2	effort	Personel	0%	100%
3	effort	RelaxScedule	-21%	64%
1	months	ProcMaturity	31%	15%
2	months	W	30%	17%
2	months	Personel	29%	98%
2	months	Tools&Tech	25%	17%
2	months	ReduceFunct	25%	9%
2	months	ReduceQuality	4%	61%
3	months	RelaxScedule	-7%	16%



But again, no common pattern

cases	*query*	*acap 3*	*apex 3*	*ltex 3*	*ltex 4*	*plex 3*	*pmat 3*	*pmat 4*	*sced 2*	*sced 3*	*stor 3*	*time 3*	*tool 3*	*# of Changes*
nasa93	**ground**					100%	55%				85%			3
nasa93	**flight**					95%	70%				100%			3
nasa93	**osp**	95%	90%										100%	3
nasa93	**osp2**				100%			80%	85%					3
coc81	**flight**						60%					65%		2
coc81	**osp2**			55%	55%		65%			100%				4
coc81	**ground**						80%					100%		2
coc81	**osp**						65%			65%				2
Overall:		12%	11%	7%	19%	24%	49%	10%	11%	21%	23%	21%	13%	

- 20 repeats over 2:1 split to train:test
- Learner = the "W" instance-based planner
- Only shows ranges found in > 50% of the repeats

In all these SE experiments

* Can generate useful predictions
  * Defect prediction
  * Software process planning
* But no general pattern
  * Performance, but no insight
  * These AI tools “just” work
  * And we don’t understand why

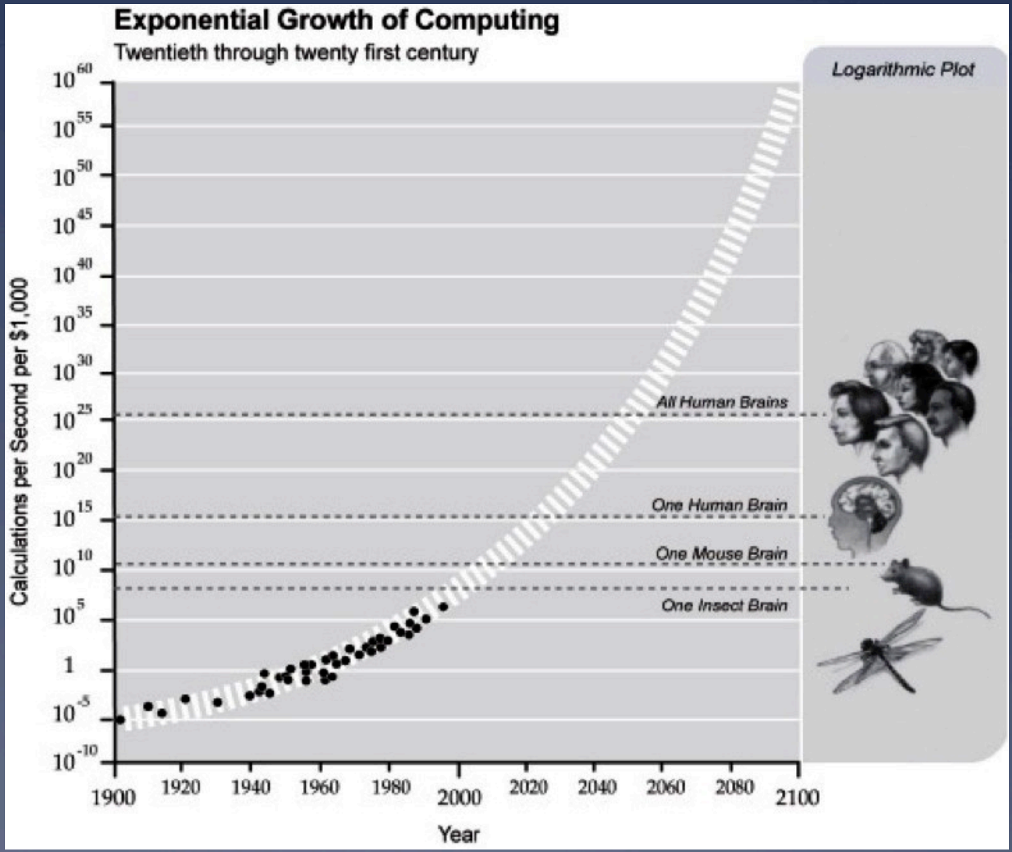
And so…

Outline

* There's this "thing"
  * Called the singularity
* That some people think will happen real soon
* That others think is a load of cr*p
* Which I think is already here (ish).

When is the AI singularity?

- Does the AI singularity:
  - Require decades of improvement in processor speed?
- Or have we already arrived at the AI singularity?



Michael Schmidt, Cornell U.

- Creator of Eureqa
- “We’ve have this window in human history when we could not just know things but actually understand them.
- “That is, we could you know why they were true and not just know but to know why.
- “And that’s a beautiful moment in human history.
- “But I feel like it may only be a moment.”

My results

* My AI tools are producing theories that:
  * Work
  * But offer no insight why they work
* Or is this the shape of things to come?
  * The more we turn to computers,
    * The more we'll get answers which
      * Work
      * But which we don't understand

Questions?
Comments?