ROBUST ARTIFICIAL INTELLIGENCE: WHY AND HOW

Tom Dietterich

Distinguished Professor (Emeritus)
Oregon State University

Past-President AAAI

Outline

- The Need for Robust AI
  - High Stakes Applications
  - Need to Act in the face of Unknown Unknowns
- Approaches toward Robust AI
  - Robustness to Known Unknowns
  - Robustness to Unknown Unknowns
- Concluding Remarks

Technical Progress is Encouraging the Development of High-Stakes Applications

Self-Driving Cars

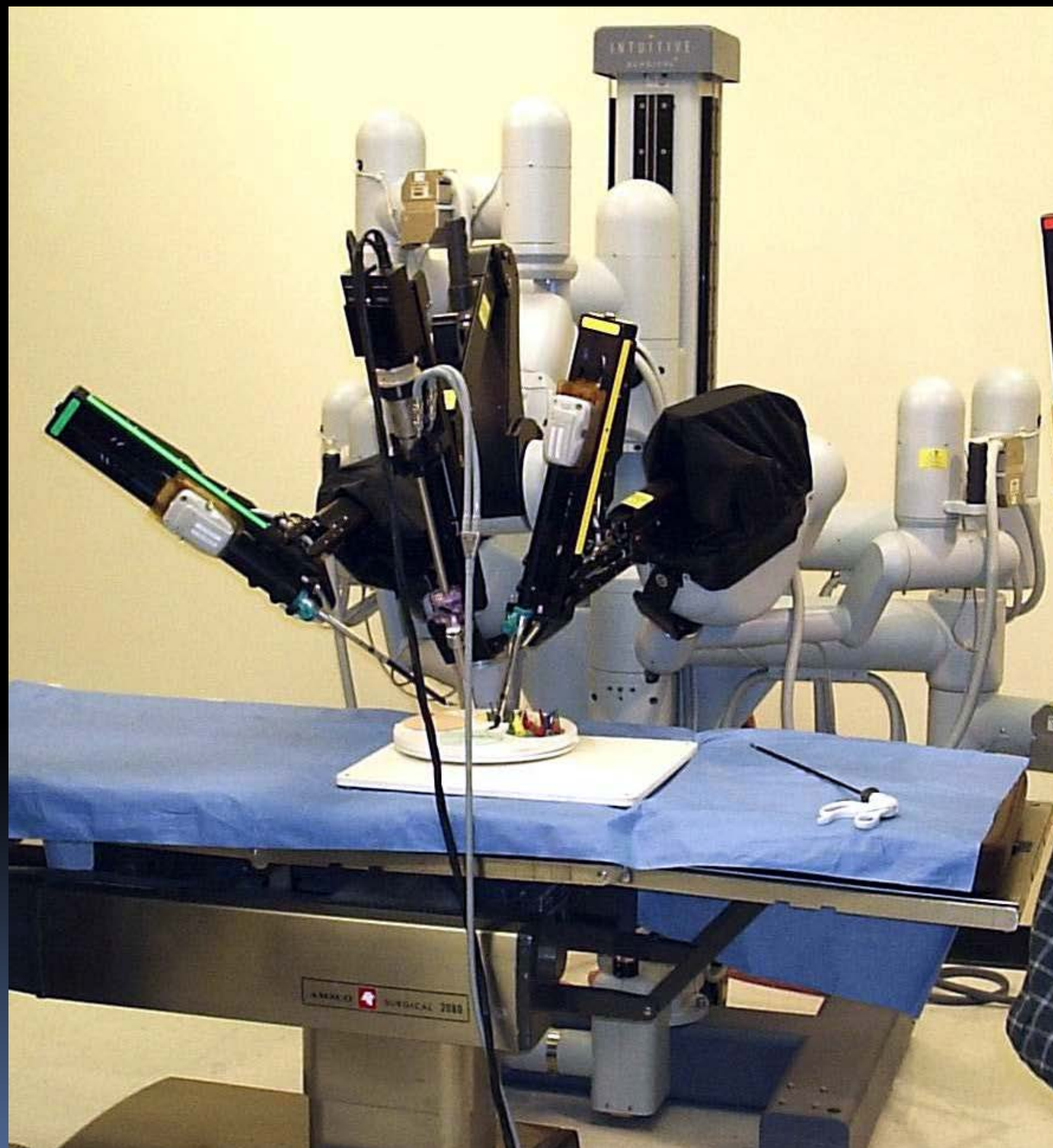
Credit: The Verge

Credit: delphi.com

Tesla AutoSteer

Credit: Tesla Motors

Automated Surgical Assistants



DaVinci

AI Hedge Funds

WIRED

CADE METZ BUSINESS 01.25.16 7:00 AM

THE RISE OF THE ARTIFICIALLY INTELLIGENT HEDGE FUND

AI Control of the Power Grid

CONTROLLING THE POWER GRID WITH ARTIFICIAL INTELLIGENCE

02.07.2015

Credit: EBM Netz AG

DARPA Exploring Ways to Protect Nation's Electrical Grid from Cyber Attack

Effort calls for creation of automated systems to restore power within seven days or less after attack

Credit: DARPA

Autonomous Weapons

Northroop Grumman X-47B

Credit: Wikipedia

Samsung SGR-1

Credit: AFP/Getty Images

UK Brimstone Anti-Armor Weapon

Credit: Duch.seb - Own work, CC BY-SA 3.0

High-Stakes Applications Require Robust AI

- Robustness to
  - Human user error
  - Cyberattack
  - Misspecified goals
  - Incorrect models
  - Unmodeled phenomena

Why Unmodeled Phenoma?

- It is impossible to model everything
- It is not desirable to model everything

It is impossible to model everything

- Qualification Problem:
  - It is impossible to enumerate all of the preconditions for an action

- Ramification Problem:
  - It is impossible to enumerate all of the implicit consequences of an action

It is important to not model everything

- Fundamental theorem of machine learning

$$\text{error rate} \propto \frac{\text{model complexity}}{\text{sample size}}$$

- Corollary:
  - If sample size is small, the model should be simple
  - We must deliberately oversimplify our models!

Conclusion:

An AI system must act without having a complete model of the world

Outline

- The Need for Robust AI
  - High Stakes Applications
  - Need to Act in the face of Unknown Unknowns
- Approaches toward Robust AI
  - Lessons from Biology
  - Robustness to Known Unknowns
  - Robustness to Unknown Unknowns
- Concluding Remarks

Robustness Lessons from Biology

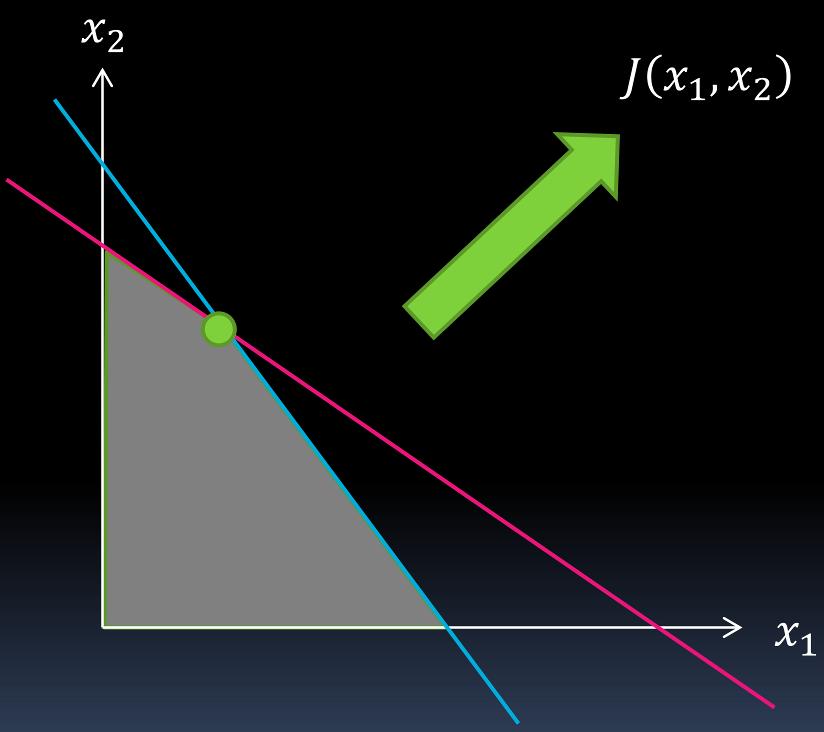
- Evolution is not optimization
  - You can't overfit if you don't optimize
- Competition against adversaries
  - "Survival of the Fittest"
- Populations of diverse individuals
  - A "portfolio" strategy
- Redundancy within individuals
  - diploidy/polyploidy = recessive alleles can be passed to future generations
  - alternative metabolic pathways
- Dispersal
  - Search for healthier environments

Approaches to Robust AI

- Robustness to Model Errors
  - Robust optimization
    - Regularize the model
    - Optimize a risk-sensitive objective
    - Employ robust inference algorithms
- Robustness to Unmodeled Phenomena
  - Detect model weaknesses
  - Expand the model
  - Learn a causal model
  - Employ a portfolio of models

Idea 1: Robust Optimization

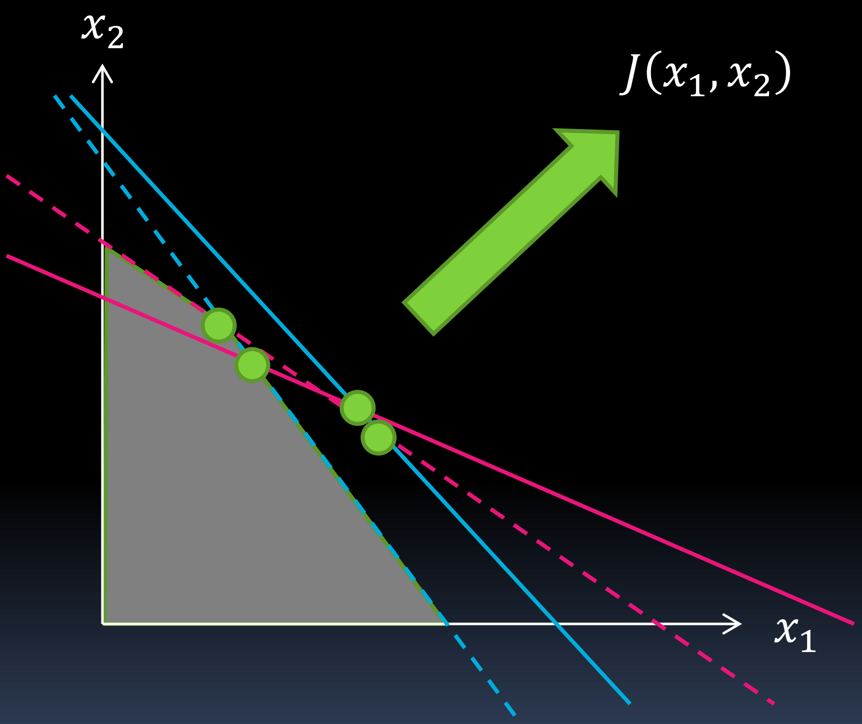
- Many AI reasoning problems can be formulated as optimization problems
- $\max_{x_1, x_2} J(x_1, x_2)$
- subject to
  - $ax_1 + bx_2 \leq r$
  - $cx_1 + dx_2 \leq s$



Uncertainty in the constraints

- $\max\limits_{x_1, x_2} \; J(x_1, x_2)$
- subject to
  - $ax_1 + bx_2 \leq r$
  - $cx_1 + dx_2 \leq s$

- Define uncertainty regions
  - $a \in U_a$
  - $b \in U_b$
  - …
  - $s \in U_s$



Minimax against the uncertainty

- $\max_{x_1,x_2} \min_{a,b,c,d,r,s} J(x_1, x_2; a, b, c, d, r, s)$
- subject to
  - $ax_1 + bx_2 \leq r$
  - $cx_1 + dx_2 \leq s$
  - $a \in U_a$
  - $b \in U_b$
  - …
  - $s \in U_s$

- Problem: Solutions can be too conservative

Impose a Budget on the Adversary

- $\max_{x_1, x_2} \min_{\delta_a, \ldots, \delta_s} J(x_1, x_2; \delta_a, \ldots, \delta_s)$
- subject to
  - $(a + \delta_a)x_1 + (b + \delta_b)x_2 \leq (r + \delta_r)$
  - $(c + \delta_c)x_1 + (d + \delta_d)x_2 \leq (s + \delta_s)$
  - $\delta_a \in U_a$
  - $\delta_b \in U_b$
  - …
  - $\delta_s \in U_s$
  - $\sum |\delta_i| \leq B$

Bertsimas, et al.

Existing AI Algorithms Implicitly Use Robust Optimization

- Given:
  - training examples (x_i, y_i) for an unknown function $y = f(x)$
  - a loss function $L(\hat{y}, y)$: how serious it is to output $\hat{y}$ when the right answer is y?
- Find:
  - the model h that minimizes

$$\sum_i L(h(x_i), y_i) + \lambda \|h\|$$

loss + complexity penalty

Regularization can be Equivalent to Robust Optimization

- Xu, Caramanis & Mannor (2009)
  - Suppose an adversary can move each training data point x_i by an amount δ_i
  - Optimizing the linear support vector objective

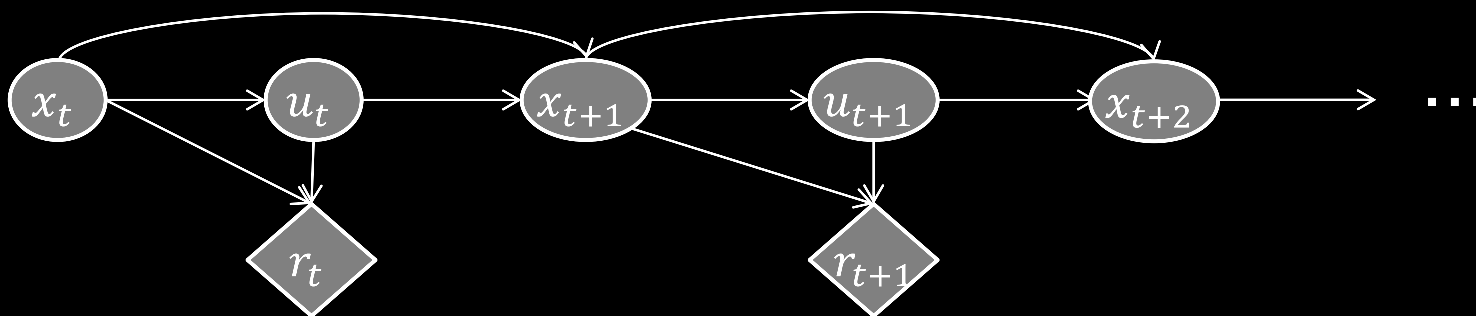
$$\sum_i L(\hat{y}_i, y_i) + \lambda \|w\|$$

  - is equivalent to minimaxing against this adversary who has a total budget

$$\sum_i \|\delta_i\| = \lambda$$

Idea 2: Optimize a Risk-Sensitive Objective

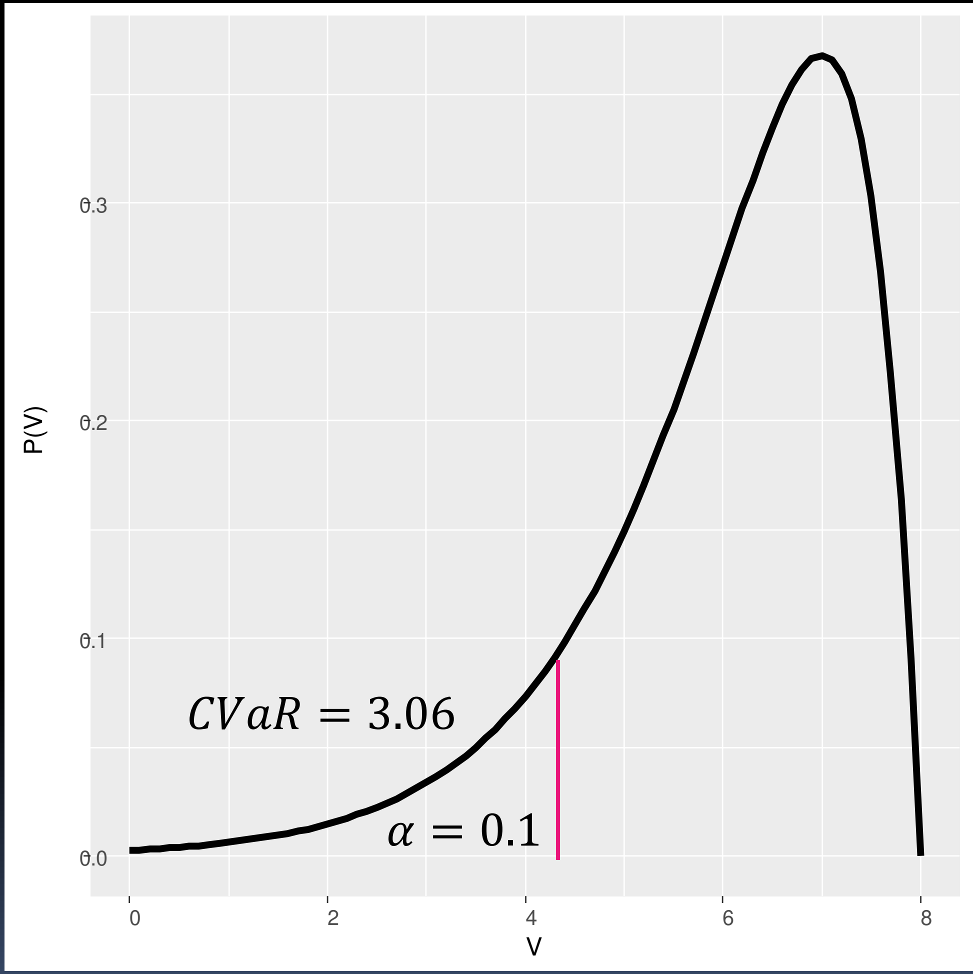
- Setting: Markov Decision Process



  - States: x_t, x_{t+1}, x_{t+2}
  - Actions: u_t, u_{t+1}
  - Control policy $u_t = \pi(x_t)$
  - Rewards: r_t, r_{t+1}
  - Total reward $\sum_t r_t$

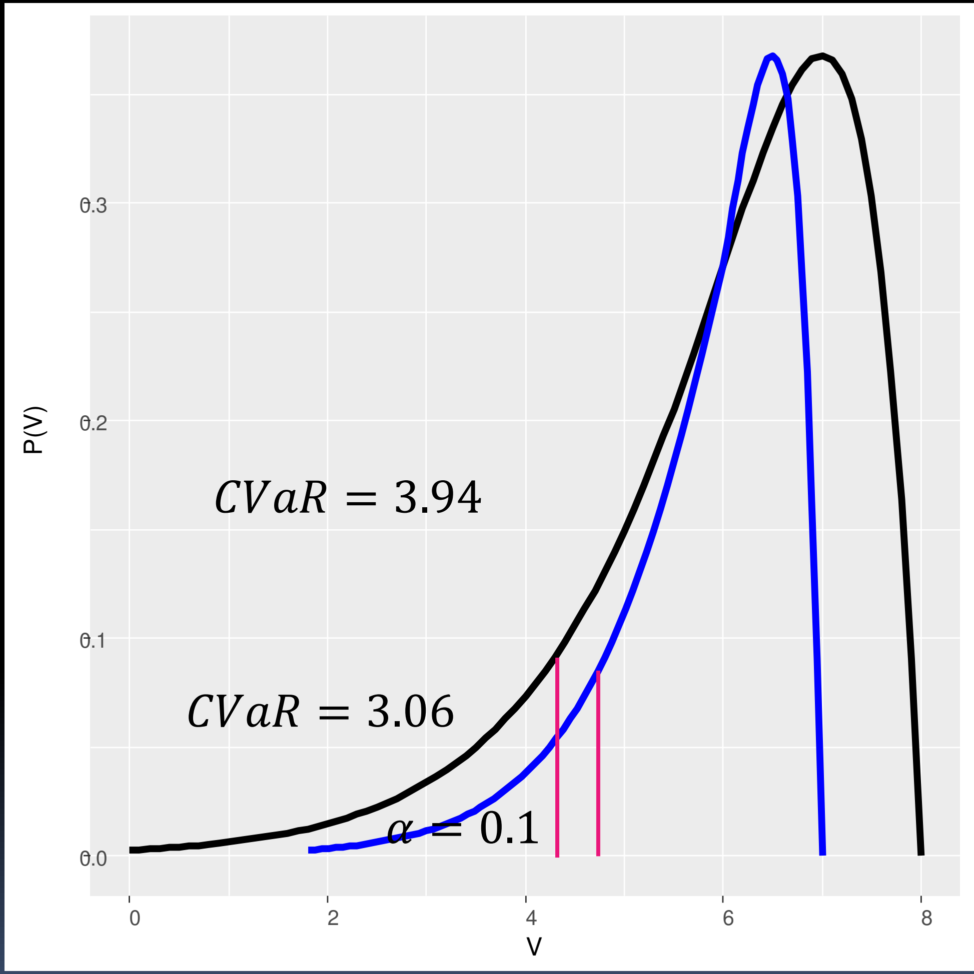
Idea 2: Optimize Conditional Value at Risk

- For any fixed policy π, the cumulative return $V^\pi = \sum_{t=1}^{T} r_t$ will have some distribution $P(V^\pi)$
- The Conditional Value at Risk at quantile α is the expected return of the bottom α quantile
- By changing π we can change the distribution $P(V^\pi)$, so we can try to push the probability to the right
- "Minimize downside risks"



Idea 2: Optimize Conditional Value at Risk

- For any fixed policy π, the cumulative return $V^\pi = \sum_{t=1}^{T} r_t$ will have some distribution $P(V^\pi)$
- The Conditional Value at Risk at quantile α is the expected return of the bottom α quantile
- By changing π we can change the distribution $P(V^\pi)$, so we can try to push the probability to the right
- "Minimize downside risks"



Optimizing CVaR gives robustness

- Suppose that for each time t, an adversary can choose a vector δ_t and define a new probability distribution

$$P(x_{t+1}|x_t, u_t) \cdot \delta_t(u_t)$$

- Optimizing CVaR at quantile α is equivalent to minimaxing against this adversary with a budget along each trajectory of

$$\prod_t \delta_t \leq \alpha$$

- Chow, Tamar, Mannor & Pavone (NIPS 2014)

- Conclusion: Acting Conservatively Gives Robustness to Model Errors

Many Other Examples

- Hierarchical Probabilistic Models
  - MCMC samples from the posterior distribution permit robust decision making
- Credal Bayesian Networks
  - Convex uncertainty sets over the probability distributions at nodes
  - Upper and lower probability models
  - (Cosman, 2000)
- Robust Classification
  - (Antonucci & Zaffalon, 2007)
- Robust Probabilistic Diagnosis (etc.)
  - (Chen, Choi, Darwiche, 2014, 2015)

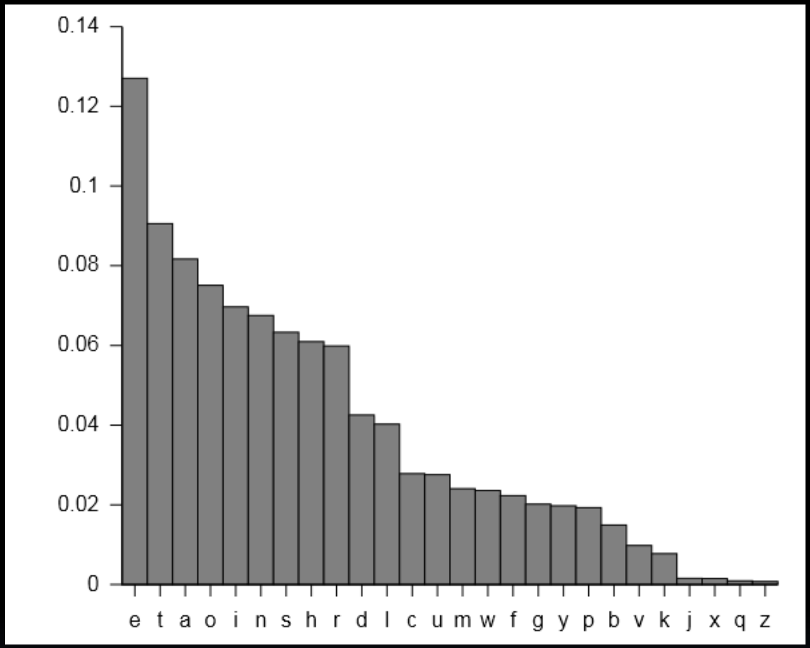
Approaches to Robust AI

- Robustness to Model Errors
  - Robust optimization
    - Regularize the model
    - Optimize a risk-sensitive objective
    - Employ robust inference algorithms
- Robustness to Unmodeled Phenomena
  - Detect model weaknesses
  - Expand the model
  - Learn a causal model
  - Employ a portfolio of models

Idea 3: Detect Surprises

- Supervised classification
  - On validation data, measure expected class frequencies
  - Detect departures from these on test data
- Mismatch can indicate a change in the class distribution or a failure in the classifier

Letter frequencies in English



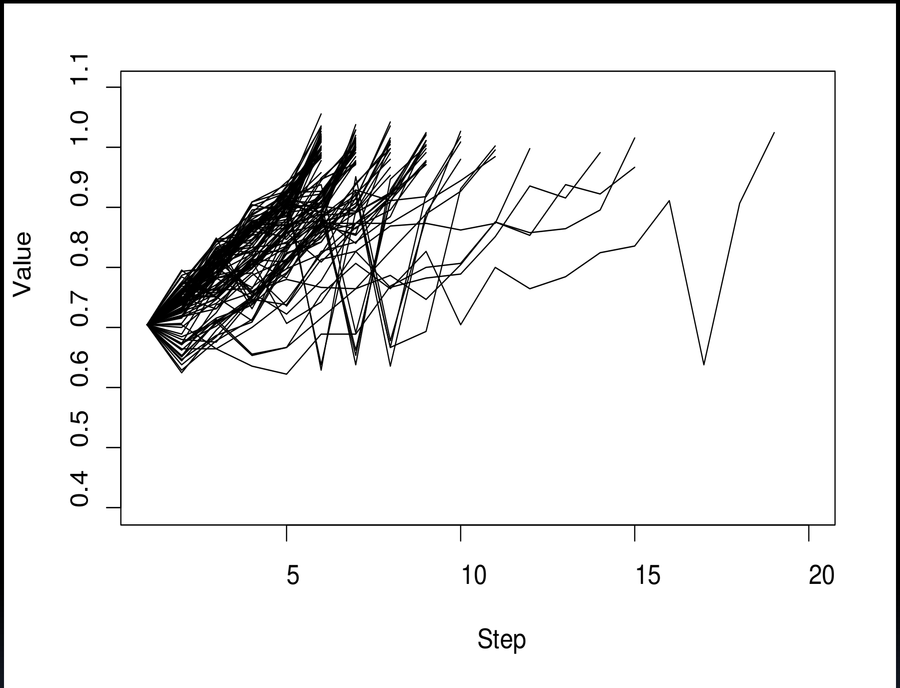
Credit: Nandhp, Wikipedia

Monitor Auxiliary Regularities

- Hermansky (2013): Each phoneme has characteristic inter-arrival time
- Monitor the inter-arrival times of recognized phonemes
- Apply to detect and suppress noisy frequency bands

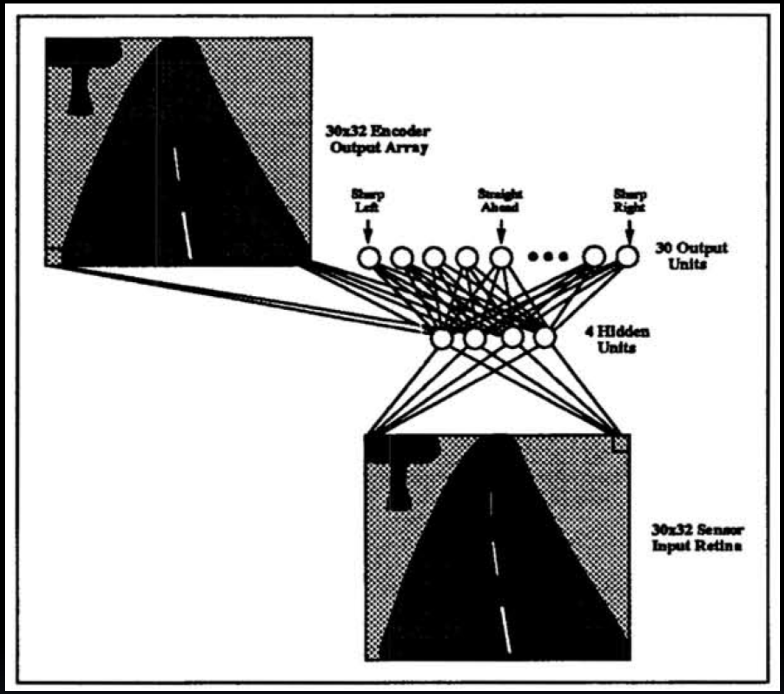
Look for Violated Expectations

- In search and reinforcement learning, we expect the estimated value to increase as we near the goal
- When false, this signals potential change in world, new obstacle, etc.



Monitor Auxiliary Tasks

- ALVINN auto-steer system
- Main task: Determine steering command
- Auxiliary task: Predict input image
- Perform both tasks with the same hidden layer information



Pomerleau, NIPS 1992

Watch for Anomalies

- Machine Learning
  - Training examples drawn from $P_{train}(x)$
  - Classifier $y = f(x)$ is learned
  - Test examples from $P_{test}(x)$
  - If $P_{test} = P_{train}$ then with high probability $f(x)$ will be correct for test queries
- What if $P_{test} \neq P_{train}$?

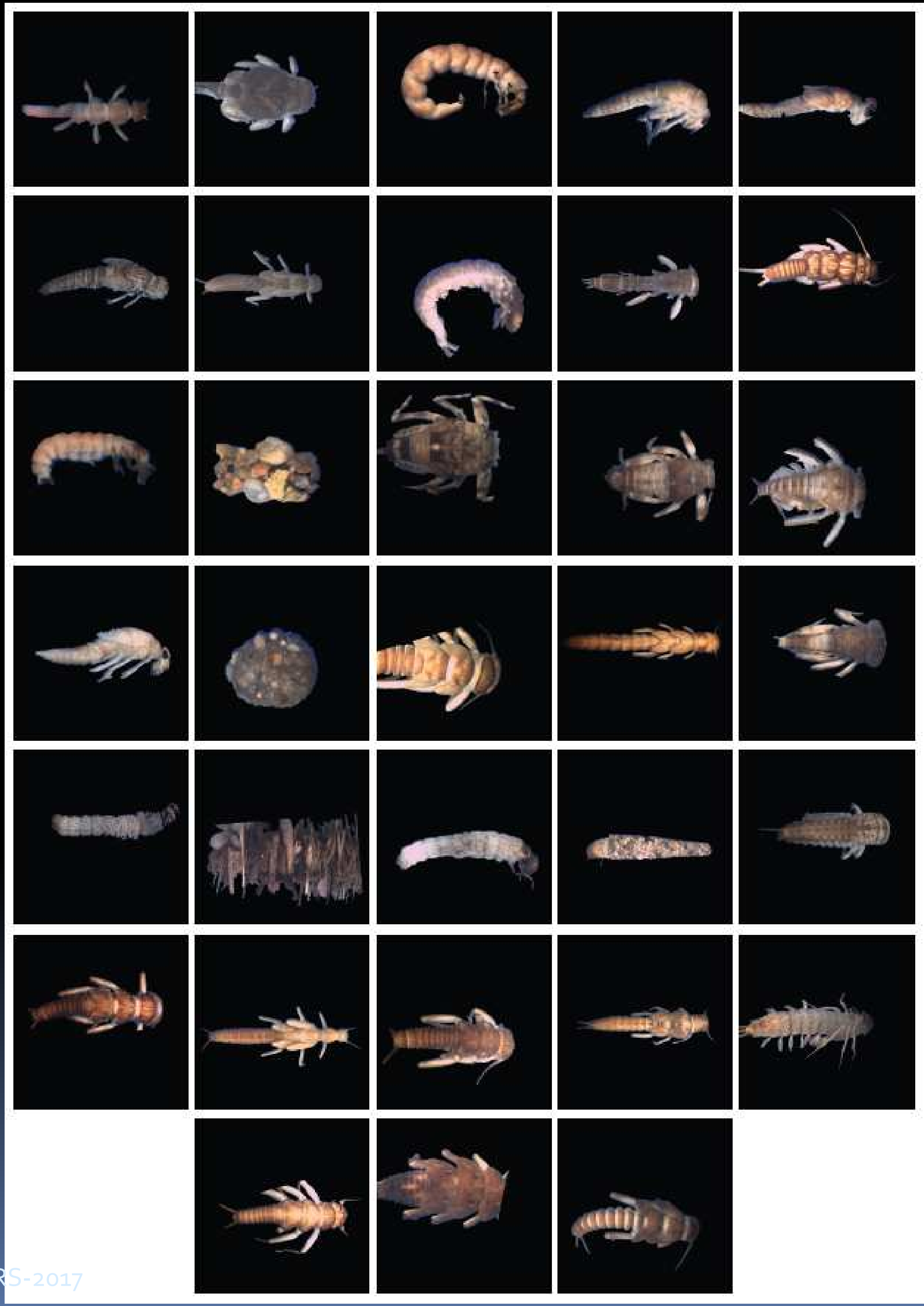
Automated Counting of Freshwater Macroinvertebrates

- Goal: Assess the health of freshwater streams
- Method:
  - Collect specimens via kicknet
  - Photograph in the lab
  - Classify to genus and species

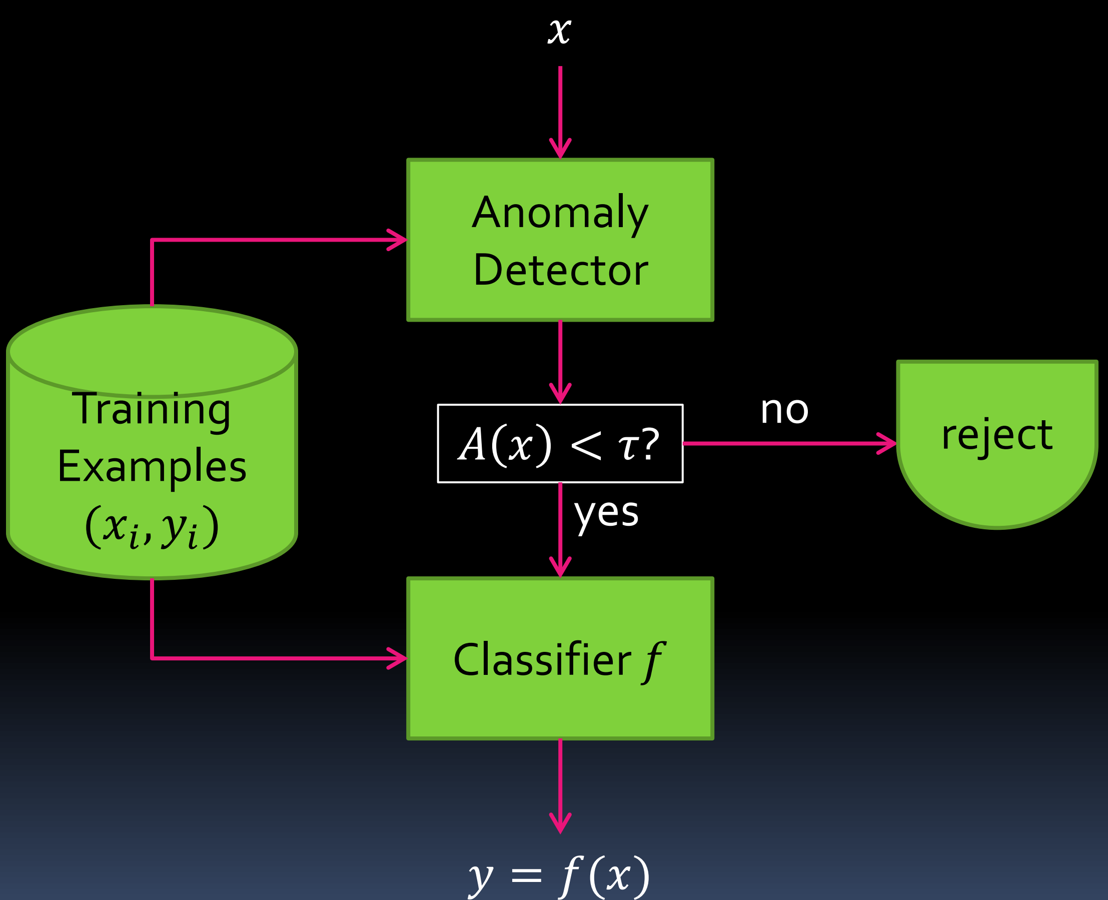
www.epa.gov

Open Category Object Recognition

- Train on 29 classes of insects
- Test set may contain additional species



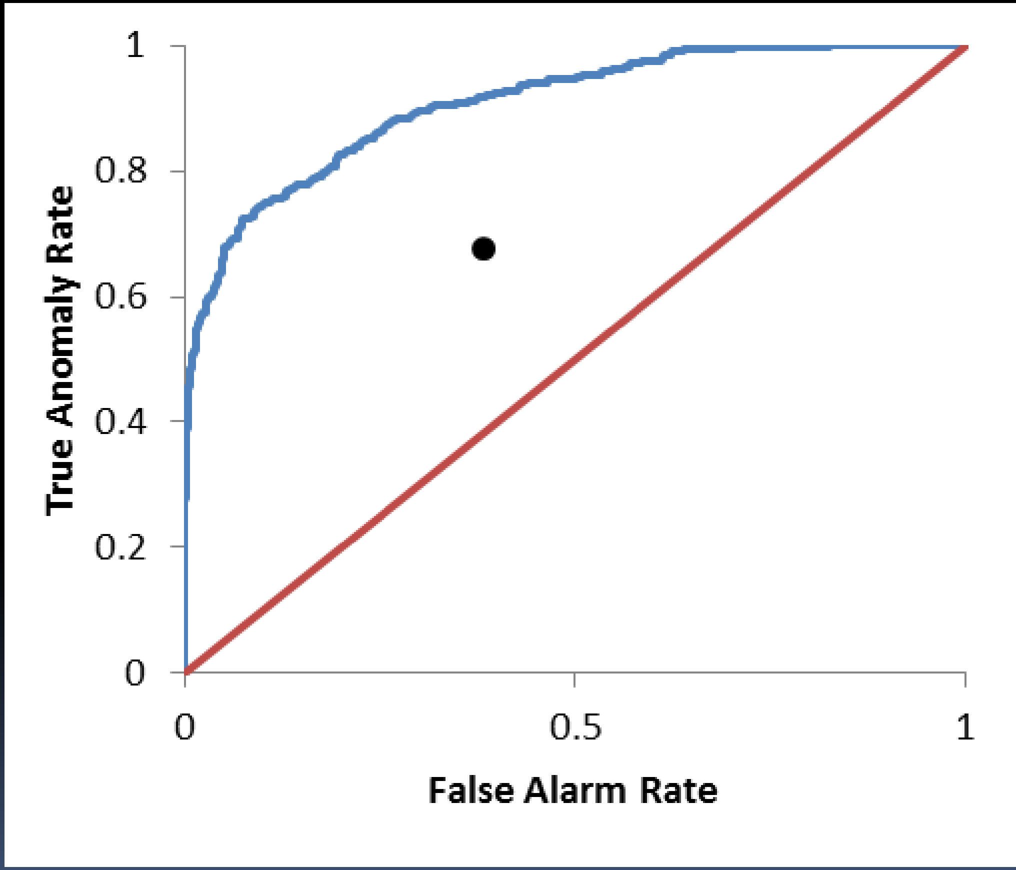
Prediction with Anomaly Detection



Source: Dietterich & Fern, unpublished

Novel Class Detection via Anomaly Detection

- Train a classifier on data from 2 classes
- Test on data from 26 classes
- Black dot: Best previous method



Anomaly Detection Notes

- We initially just used monochrome images
  - Feature selection studies showed this was sufficient
- But color is very useful for detecting novel classes
- Lesson: Use *all* of your features when looking for anomalies

Related Efforts

- Open Category Classification
  - (Salakhutdinov, Tenenbaum, & Torralba, 2012)
  - (Da, Yu & Zhou, AAAI 2014)
  - (Bendale & Boult, CVPR 2015)
- Change-Point Detection
  - (Page, 1955)
  - (Barry & Hartigan, 1993)
  - (Adams & MacKay, 2007)
- Covariate Shift Correction
  - (Sugiyama, Krauledat & Müller, 2007)
  - (Quinonero-Candela, Sugiyama, Schwaighofer & Lawrence, 2009)
- Domain Adaptation
  - (Blitzer, Dredze, Pereira, 2007)
  - (Daume & Marcu, 2006)

Idea 2: Repair or Expand the Model

- Learning Models of Actions in Planning and Reinforcement Learning
  - Gil (1994)
- Knowledge Base Construction
  - Cyc (Lenat & Guha, 1990)
- Information Extraction & Knowledge Base Population
  - Dankel (1980)
  - NELL (Mitchell, et al., AAAI 2015)
  - TAC-KBP (NIST)
  - Robust Logic (Valiant; AIJ 2001)

- Risk: Every new component added to a model may introduce an error

Idea 3: Use Causal Models

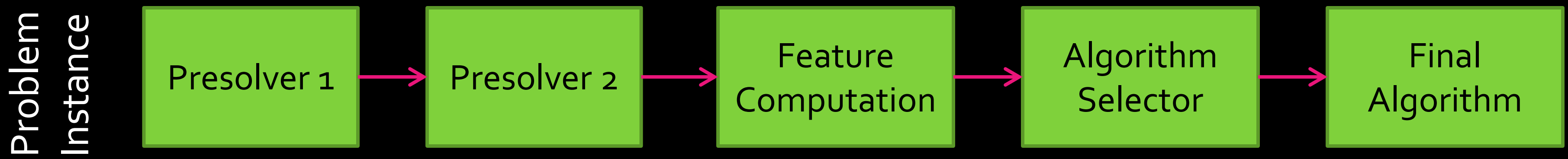
- Causal relations are more likely to be robust
  - Require less data to learn
    - (Heckerman & Breese, IEEE SMC 1997)
  - Can be transported to novel situations
    - (Pearl & Bareinboim, AAAI 2011)
    - (Schoelkopf, et al., ICML 2012)
    - (Lee & Honavar, AAAI 2013)

Idea 4: Employ a Portfolio of Models

- Ensemble machine learning methods regularly win Kaggle competitions
- Portfolios for SAT solving
- Portfolios for Question Answering and Search

Portfolio Methods in SAT & CSP

- SATzilla:

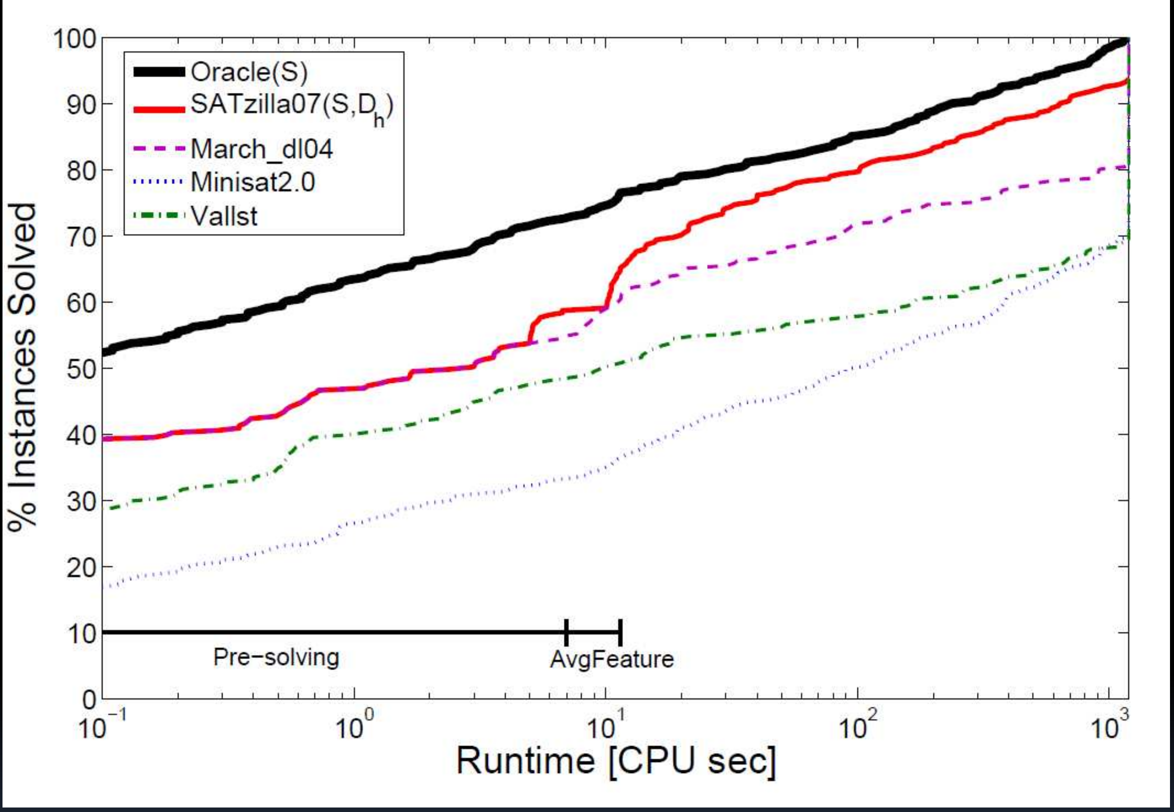


- Xu, Hoos, Hutter, Leyton-Brown (JAIR 2008)

SATzilla Results

- HANDMADE problem set
- Presolvers:
  - March_d104 (5 seconds)
  - SAPS (2 seconds)

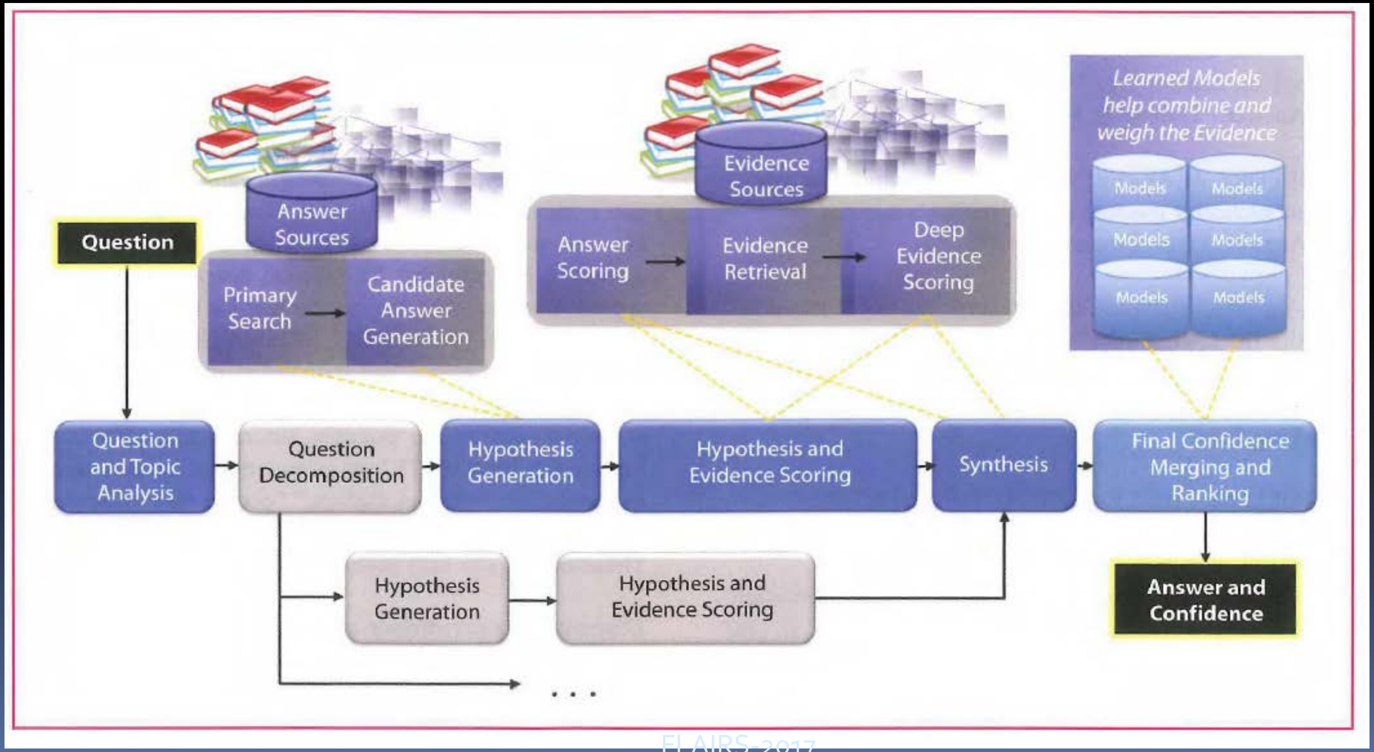
Cumulative Distribution



Xu, Hutter, Hoos, Leyton-Brown (JAI R2008)

IBM Watson / DeepQA

- Combines >100 different techniques for
  - analyzing natural language
  - identifying sources
  - finding and generating hypotheses
  - finding and scoring evidence
  - merging and ranking hypotheses



Ferrucci, IBM JRD 2012

Knowledge-Level Portfolios

- Minsky: “You don’t really understand something if you only understand it one way”
- Most AI systems only understand things one way:
  - Computer vision:
    - Object Appearance ➔ human labels
  - Natural Language:
    - Word Co-occurrence statistics ➔ human labels

“a black and white cat is sitting on a chair."

Credit: Jeff Donahue, Trevor Darrell

Multifaceted Understanding

- There is a person who is the cat's owner
- That person does not like the cat sitting on the chair
  - The cat is preventing a person from sitting on the chair
    - People often need to sit on chairs
  - The cat leaves hair on the chair
  - The cat is preventing the person from picking up the book
- The cat will soon not be on the chair
- The cat does this often

"a black and white cat is sitting on a chair."

Achieving Multifaceted Understanding

- We need to give our computers many different forms of experience
  - Performing tasks
  - Achieving goals through natural language dialogue
  - Interacting with other agents
  - Examples:
    - Minsky, "Learning Meaning" (1982 MIT TR)
    - Blum & Mitchell, "Multi-View Learning" (1998)
    - Lake, Salakhutdinov & Tenenbaum (Science 2016)

Outline

- The Need for Robust AI
  - High Stakes Applications
  - Need to Act in the face of Unknown Unknowns
- Approaches toward Robust AI
  - Lessons from Biology
  - Robustness to Known Unknowns
  - Robustness to Unknown Unknowns
- Concluding Remarks

Concluding Remarks

High Risk Emerging AI applications

…Require Robust AI Systems

AI systems can't model everything

…AI needs to be robust to "unknown unknowns"

Existing Approaches to Robust AI

- Robustness to Model Errors
  - Robust optimization
    - Regularize the model
    - Optimize a risk-sensitive objective
    - Employ robust inference algorithms
- Robustness to Unmodeled Phenomena
  - Detect model weaknesses
  - Expand the model
  - Learn a causal model
  - Employ a portfolio of models

We have many good ideas

We need many more!

Acknowledgments

- Juan Augusto
- Randall Davis
- Trevor Darrell
- Pedro Domingos
- Alan Fern
- Boi Faltings
- Stephanie Forrest
- Helen Gigley
- Barbara Grosz
- Vasant Honavar
- Holgar Hoos
- Eric Horvitz
- Michael Huhns
- Rebecca Hutchinson
- Pat Langley
- Sridhar Mahadevan
- Shie Mannor
- Melanie Mitchell
- Dana Nau
- Jeff Rosenschein
- Dan Roth
- Stuart Russell
- Tuomas Sandholm
- Rob Schapire
- Scott Sanner
- Prasad Tadepalli
- Milind Tambe
- Zhi-hua Zhou

Questions?