Universal Artificial Intelligence

philosophical, mathematical, and computational foundations of inductive inference and intelligent agents the learn

Marcus Hutter

Australian National University

Canberra, ACT, 0200, Australia

`http://www.hutter1.net/`

ANU

Abstract: Motivation

The dream of creating artificial devices that reach or outperform human intelligence is an old one, however a computationally efficient theory of true intelligence has not been found yet, despite considerable efforts in the last 50 years. Nowadays most research is more modest, focussing on solving more narrow, specific problems, associated with only some aspects of intelligence, like playing chess or natural language translation, either as a goal in itself or as a bottom-up approach. The dual, top down approach, is to find a mathematical (not computational) definition of general intelligence. Note that the AI problem remains non-trivial even when ignoring computational aspects.

Abstract: Contents

In this course we will develop such an elegant mathematical parameter-free theory of an optimal reinforcement learning agent embedded in an arbitrary unknown environment that possesses essentially all aspects of rational intelligence. Most of the course is devoted to giving an introduction to the key ingredients of this theory, which are important subjects in their own right: Occam's razor; Turing machines; Kolmogorov complexity; probability theory; Solomonoff induction; Bayesian sequence prediction; minimum description length principle; agents; sequential decision theory; adaptive control theory; reinforcement learning; Levin search and extensions.

Background and Context

- Organizational

- Artificial General Intelligence

- Natural and Artificial Approaches

- On Elegant Theories of

- What is (Artificial) Intelligence?

- What is Universal Artificial Intelligence?

- Relevant Research Fields

- Relation between ML & RL & (U)AI

- Course Highlights

Organizational

- Suitable for a 1 or 2 semester course:
  with tutorials, assignments, exam, lab, group project, seminar, ...
  See e.g. [http://cs.anu.edu.au/courses/COMP4620/2010.html]

- Prescribed texts: Parts of: [Hut05] (theory), [Leg08] (philosophy),
  [VNH$^+$11] (implementation).

- Reference details: See end of each section.

- Main course sources: See end of all slides.

- For a shorter course: Sections 4.3, 4.4, 5, 6, 7, 10.1, 11
  might be dropped or shortened.

- For an even shorter course (4-8 hours):
  Use [http://www.hutter1.net/ai/suai.pdf]

Artificial General Intelligence

What is (not) the goal of AGI research?

- Is: Build general-purpose Super-Intelligences.
- Not: Create AI software solving specific problems.
- Might ignite a technological Singularity.

What is (Artificial) Intelligence?

What are we really doing and aiming at?

- Is it to build systems by trial&error, and if they do something we think is smarter than previous systems, call it success?
- Is it to try to mimic the behavior of biological organisms?

> We need (and have!) theories which
> can guide our search for intelligent algorithms.
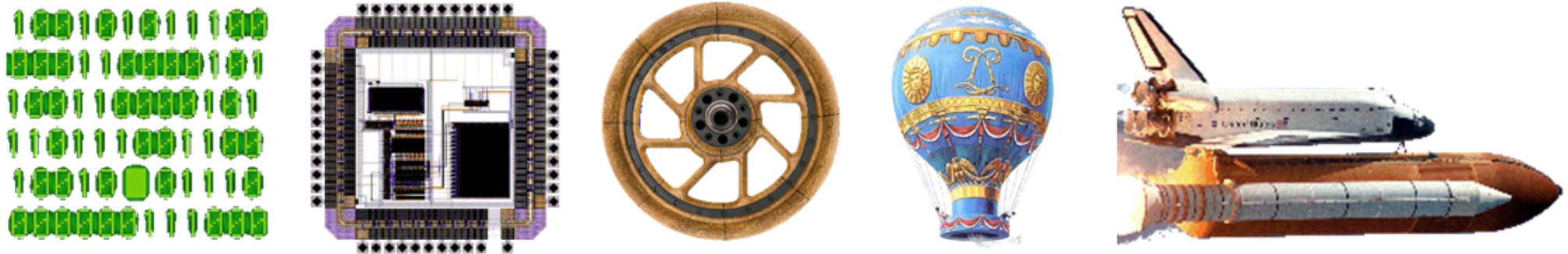
"Natural" Approaches

copy and improve (human) nature

Biological Approaches to Super-Intelligence

- Brain Scan & Simulation
- Genetic Enhancement
- Brain Augmentation

Not the topic of this course

"Artificial" Approaches

Design from first principles. At best inspired by nature.

Artificial Intelligent Systems:

- **Logic/language based:**   expert/reasoning/proving/cognitive systems.
- **Economics inspired:**   utility, sequential decisions, game theory.
- **Cybernetics:**   adaptive dynamic control.
- **Machine Learning:**   reinforcement learning.
- **Information processing:** data compression $\approx$ intelligence.

Separately too limited for AGI, but jointly very powerful.

Topic of this course: Foundations of "artificial" approaches to AGI

There is an Elegant Theory of ...

Cellular Automata	$\Rightarrow$	... Computing
Iterative maps	$\Rightarrow$	...Chaos and Order
QED	$\Rightarrow$	... Chemistry
Super-Strings	$\Rightarrow$	... the Universe
Universal AI	$\Rightarrow$	**... Super Intelligence**

What is (Artificial) Intelligence?

Intelligence can have many faces ⇒ formal definition difficult

- reasoning
- creativity
- association
- generalization
- pattern recognition
- problem solving
- memorization
- planning
- achieving goals
- learning
- optimization
- self-preservation
- vision
- language processing
- motor skills
- classification
- induction
- deduction
- ...

What is AI?	Thinking	Acting
humanly	Cognitive Science	Turing test, Behaviorism
rationally	Laws Thought	Doing the Right Thing

Collection of 70+ Defs of Intelligence
http://www.vetta.org/
       definitions-of-intelligence/

Real world is nasty: partially unobservable, uncertain, unknown, non-ergodic, reactive, vast, but luckily structured, ...

What is Universal Artificial Intelligence?

- Sequential Decision Theory solves the problem of rational agents in uncertain worlds if the environmental probability distribution is *known*.

- Solomonoff's theory of Universal Induction solves the problem of sequence prediction for *unknown* prior distribution.

- Combining both ideas one arrives at

A Unified View of Artificial Intelligence

$$=$$ $$=$$

Decision Theory $$=$$ Probability $$+$$ Utility Theory

$$+$$ $$+$$

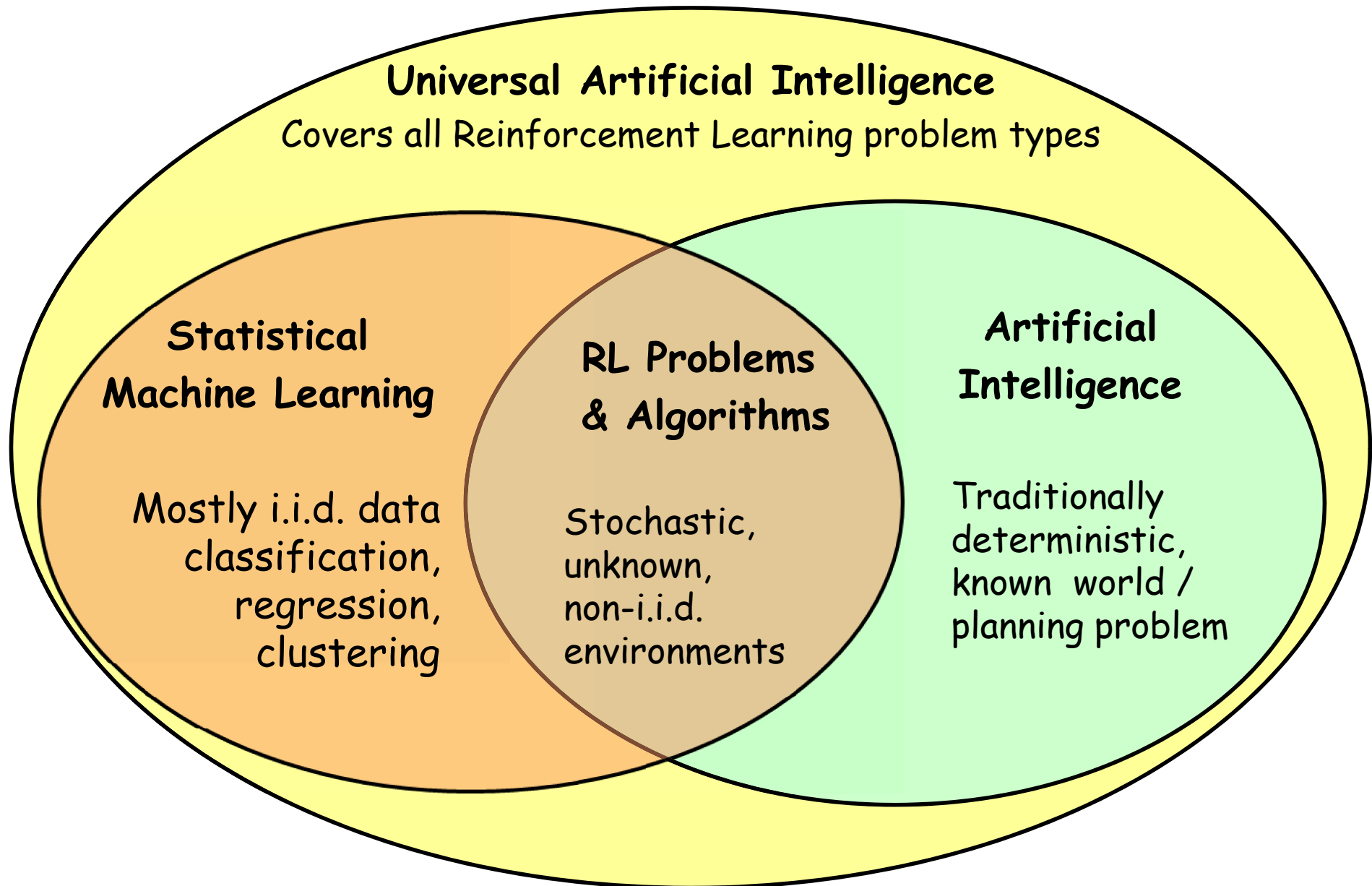Universal Induction $$=$$ Ockham $$+$$ Bayes $$+$$ Turing
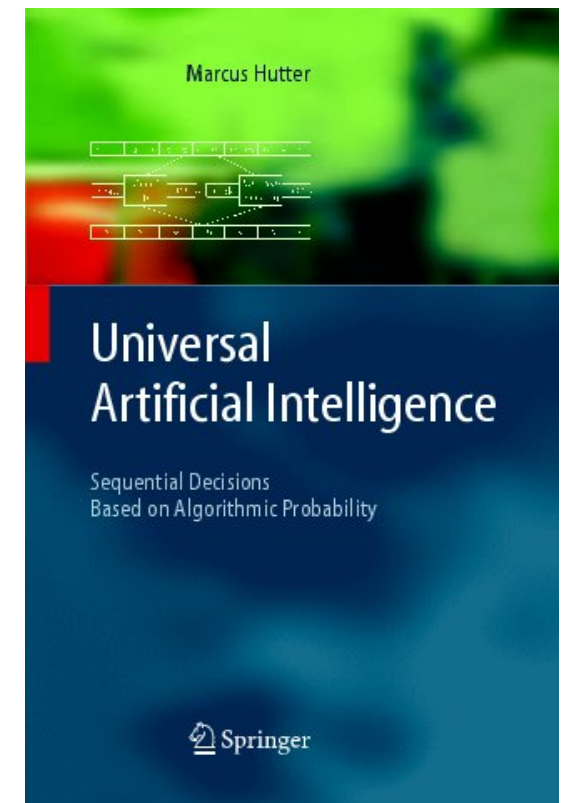
Approximation and Implementation: *Single* agent that *learns* to play TicTacToe/Pacman/Poker/... from scratch.

Relevant Research Fields

(Universal) Artificial Intelligence has interconnections with (draws from and contributes to) many research fields:

- computer science (artificial intelligence, machine learning),

- engineering (information theory, adaptive control),

- economics (rational agents, game theory),

- mathematics (statistics, probability),

- psychology (behaviorism, motivation, incentives),

- philosophy (reasoning, induction, knowledge).

Relation between ML & RL & (U)AI

Universal Artificial Intelligence
Covers all Reinforcement Learning problem types

**Statistical
Machine Learning**

Mostly i.i.d. data
classification,
regression,
clustering

**RL Problems
& Algorithms**

Stochastic,
unknown,
non-i.i.d.
environments

**Artificial
Intelligence**

Traditionally
deterministic,
known world /
planning problem

Course Highlights

- Formal definition of (general rational) Intelligence.

- Optimal rational agent for arbitrary problems.

- Philosophical, mathematical, and computational background.

- Some approximations, implementations, and applications.

  (learning TicTacToe, PacMan, simplified Poker from scratch)

- State-of-the-art artificial general intelligence.

Table of Contents

1. A SHORT TOUR THROUGH THE COURSE

2. INFORMATION THEORY & KOLMOGOROV COMPLEXITY

3. BAYESIAN PROBABILITY THEORY

4. ALGORITHMIC PROBABILITY & UNIVERSAL INDUCTION

5. MINIMUM DESCRIPTION LENGTH

6. THE UNIVERSAL SIMILARITY METRIC

7. BAYESIAN SEQUENCE PREDICTION

8. UNIVERSAL RATIONAL AGENTS

9. THEORY OF RATIONAL AGENTS

10. APPROXIMATIONS & APPLICATIONS

11. DISCUSSION

1 A SHORT TOUR THROUGH THE COURSE

Informal Definition of (Artificial) Intelligence

> Intelligence measures an agent's ability to achieve goals in a wide range of environments. [S. Legg and M. Hutter]

Emergent: Features such as the ability to learn and adapt, or to understand, are implicit in the above definition as these capacities enable an agent to succeed in a wide range of environments.

The science of Artificial Intelligence is concerned with the construction of intelligent systems/artifacts/agents and their analysis.

What next? Substantiate all terms above: agent, ability, utility, goal, success, learn, adapt, environment, ...

Never trust a ~~theory~~ experiment if it is not supported by an ~~experiment~~ theory

Induction→Prediction→Decision→Action

Having or acquiring or *learning* or inducing a model of the environment an agent interacts with allows the agent to make predictions and utilize them in its decision process of finding a good next action.

Induction infers general models from specific observations/facts/data, usually exhibiting regularities or properties or relations in the latter.

Example

Induction: Find a model of the world economy.

Prediction: Use the model for predicting the future stock market.

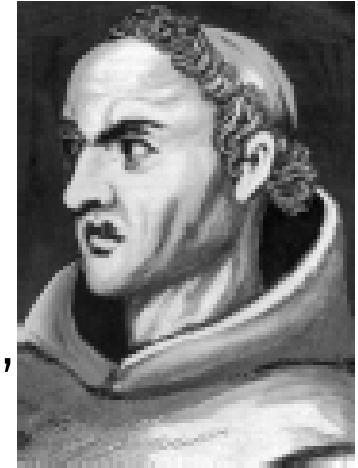Decision: Decide whether to invest assets in stocks or bonds.

Action: Trading large quantities of stocks influences the market.

Science $\approx$ Induction $\approx$ Occam's Razor

- Grue Emerald Paradox:

  Hypothesis 1: All emeralds are green.

  Hypothesis 2: All emeralds found till y2020 are green,
  thereafter all emeralds are blue.

- Which hypothesis is more plausible? H1! Justification?

- Occam's razor: take simplest hypothesis consistent with data.
  is the most important principle in machine learning and science.

- Problem: How to quantify "simplicity"? Beauty? Elegance?
  Description Length!

[The Grue problem goes much deeper. This is only half of the story]

Information Theory & Kolmogorov Complexity

- Quantification/interpretation of Occam's razor:

- Shortest description of object is best explanation.

- Shortest program for a string on a Turing machine T leads to best extrapolation=prediction.

$$K_T(x) = \min_p\{\ell(p) : T(p) = x\}$$

- Prediction is best for a universal Turing machine U.

Kolmogorov-complexity$(x) \;=\; K(x) \;=\; K_U(x) \qquad \leq K_T(x) + c_T$

Bayesian Probability Theory

Given (1): Models $P(D|H_i)$ for probability of observing data D, when H_i is true.

Given (2): Prior probability over hypotheses $P(H_i)$.

Goal: Posterior probability $P(H_i|D)$ of H_i, after having seen data D.

Solution:

Bayes' rule:
$$P(H_i|D) = \frac{P(D|H_i) \cdot P(H_i)}{\sum_i P(D|H_i) \cdot P(H_i)}$$

(1) Models $P(D|H_i)$ usually easy to describe (objective probabilities)

(2) But Bayesian prob. theory does not tell us how to choose the prior $P(H_i)$ (subjective probabilities)

Algorithmic Probability Theory

- Epicurus: If more than one theory is consistent with the observations, keep all theories.

- $\Rightarrow$ uniform prior over all H_i?

- Refinement with Occam's razor quantified in terms of Kolmogorov complexity:
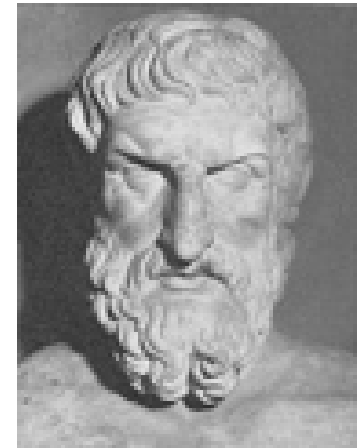
$$P(H_i) \;:=\; 2^{-K_{T/U}(H_i)}$$

- Fixing T we have a complete theory for prediction.
  Problem: How to choose T.

- Choosing U we have a universal theory for prediction.
  Observation: Particular choice of U does not matter much.
  Problem: Incomputable.

Inductive Inference & Universal Forecasting

- Solomonoff combined Occam, Epicurus, Bayes, and Turing into one formal theory of sequential prediction.

- $M(x) =$ probability that a universal Turing machine outputs x when provided with fair coin flips on the input tape.

- A posteriori probability of y given x is $M(y|x) = M(xy)/M(x)$.

- Given $\dot{x}_1, .., \dot{x}_{t-1}$, the probability of x_t is $M(x_t|\dot{x}_1...\dot{x}_{t-1})$.

- Immediate "applications":
  - Weather forecasting: $x_t \in \{\text{sun,rain}\}$.
  - Stock-market prediction: $x_t \in \{\text{bear,bull}\}$.
  - Continuing number sequences in an IQ test: $x_t \in \mathbb{N}$.

- Optimal universal inductive reasoning system!
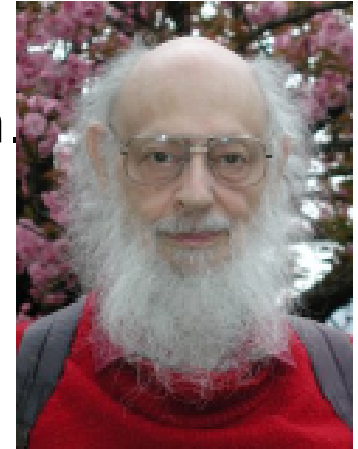
The Minimum Description Length Principle

- Approximation of Solomonoff,

  since M is incomputable:

- $M(x) \approx 2^{-K_U(x)}$ (quite good)

- $K_U(x) \approx K_T(x)$ (very crude)

- Predict y of highest $M(y|x)$ is approximately same as

- MDL: Predict y of smallest $K_T(xy)$.

Application: Universal Clustering

- Question: When is object x similar to object y?

- Universal solution: x similar to y
  $\Leftrightarrow x$ can be easily (re)constructed from y
  $\Leftrightarrow K(x|y) := \min\{\ell(p) : U(p, y) = x\}$ is small.

- Universal Similarity: Symmetrize&normalize $K(x|y)$.

- Normalized compression distance: Approximate $K \equiv K_U$ by K_T.

- Practice: For T choose (de)compressor like lzw or gzip or bzip(2).

- Multiple objects $\Rightarrow$ similarity matrix $\Rightarrow$ similarity tree.

- Applications: Completely automatic reconstruction (a) of the evolutionary tree of 24 mammals based on complete mtDNA, and (b) of the classification tree of 52 languages based on the declaration of human rights and (c) many others. [Cilibrasi&Vitanyi'05]

Sequential Decision Theory

Setup: For $t = 1, 2, 3, 4, ...$

Given sequence $x_1, x_2, ..., x_{t-1}$

(1) predict/make decision y_t,

(2) observe x_t,

(3) suffer loss $\text{Loss}(x_t, y_t)$,

(4) $t \to t + 1$, goto (1)

Goal: Minimize expected Loss.

Greedy minimization of expected loss is optimal if:

Important: Decision y_t does not influence env. (future observations).

Loss function is known.

Problem: Expectation w.r.t. what?

Solution: W.r.t. universal distribution M if true distr. is unknown.

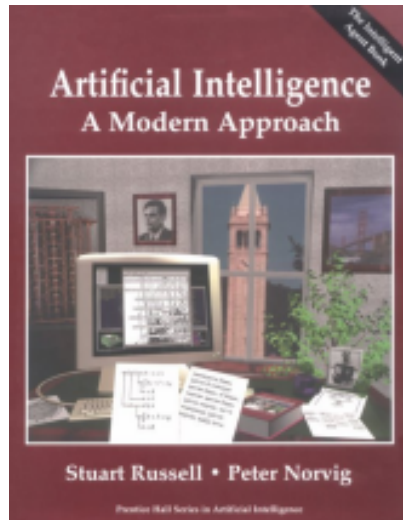# Example: Weather Forecasting

Observation $x_t \in \mathcal{X} = \{$sunny, rainy$\}$

Decision $y_t \in \mathcal{Y} = \{$umbrella, sunglasses$\}$

Loss	sunny	rainy
umbrella	0.1	0.3
sunglasses	0.0	1.0

Taking umbrella/sunglasses does not influence future weather

(ignoring butterfly effect)

Agent Model with Reward

if actions/decisions a

influence the environment q

$r_1 \mid o_1$	$r_2 \mid o_2$	$r_3 \mid o_3$	$r_4 \mid o_4$	$r_5 \mid o_5$	$r_6 \mid o_6$	...

work	**Agent** p	tape ...	work	**Environ- ment** q	tape ...

y_1	y_2	y_3	y_4	y_5	y_6	...
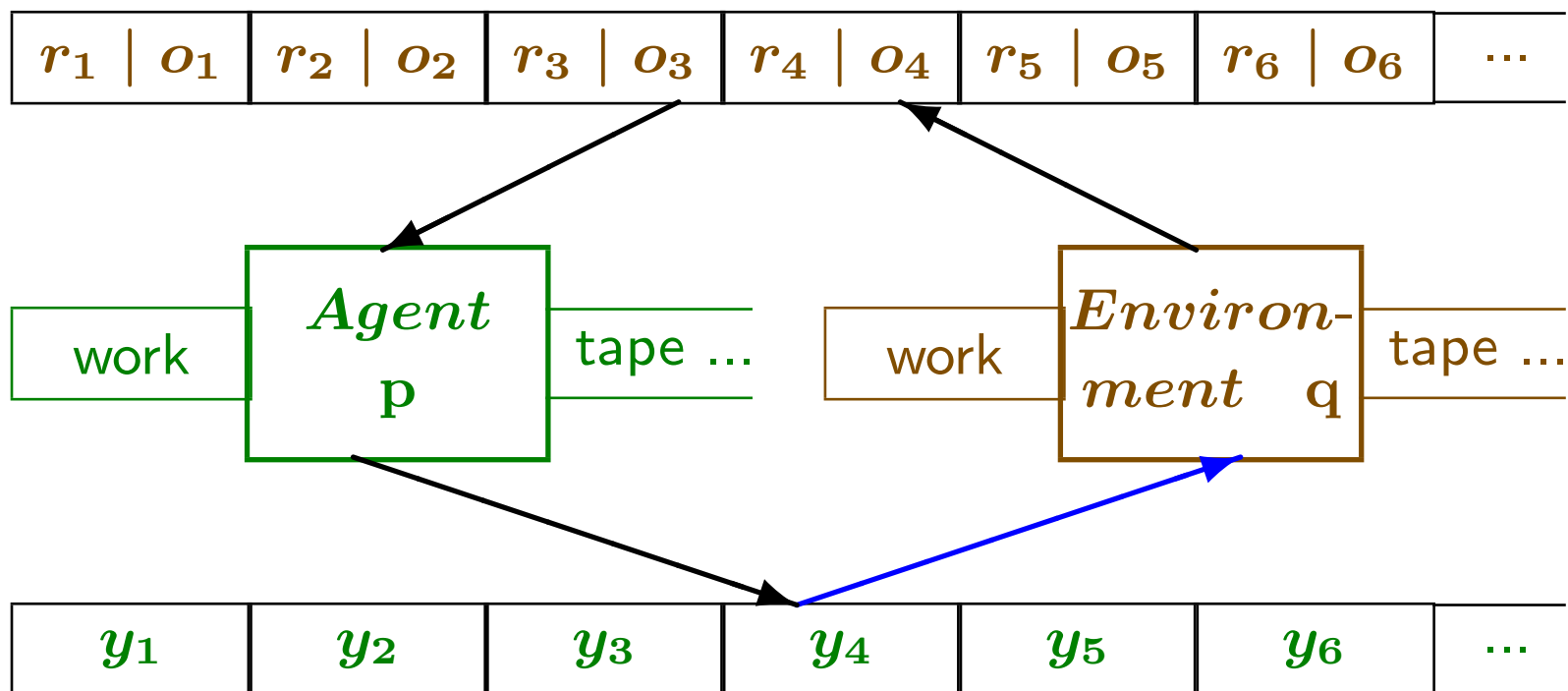
Rational Agents in Known Environment

- Setup: Known deterministic or probabilistic environment

- Fields: AI planning & sequential decision theory & control theory

- Greedy maximization of reward r ($=-$Loss) no longer optimal.
  Example: Chess

- Agent has to be farsighted.

- Optimal solution: Maximize future (expected) reward sum,
  called value.

- Problem: Things drastically change if environment is unknown

Rational Agents in Unknown Environment

Additional problem: (probabilistic) environment unknown.

Fields: reinforcement learning and adaptive control theory

Big problem: Exploration versus exploitation

Bayesian approach: Mixture distribution ξ.

1. What performance does Bayes-optimal policy imply?
   It does not necessarily imply self-optimization
   (Heaven&Hell example).

2. Computationally very hard problem.

3. Choice of horizon? Immortal agents are lazy.

Universal Solomonoff mixture $\Rightarrow$ universal agent AIXI.

Represents a formal (math., non-comp.) solution to the AI problem?

Most (all AI?) problems are easily phrased within AIXI.

Computational Issues: Universal Search

- **Levin search:** Fastest algorithm for inversion and optimization problems.

- **Theoretical application:**
  Assume somebody found a non-constructive proof of P=NP, then Levin-search is a polynomial time algorithm for every NP (complete) problem.

- **Practical (OOPS) applications** (J. Schmidhuber)
  Mazes, towers of hanoi, robotics, ...

- **FastPrg:** The asymptotically fastest and shortest algorithm for all well-defined problems.

- **Computable Approximations of AIXI:**
  AIXItl and AIξ and MC-AIXI-CTW and ΦMDP.

- **Human Knowledge Compression Prize:** (50'000€)

Monte-Carlo AIXI Applications

without providing any domain knowledge, the same agent is
able to self-adapt to a diverse range of interactive environments.
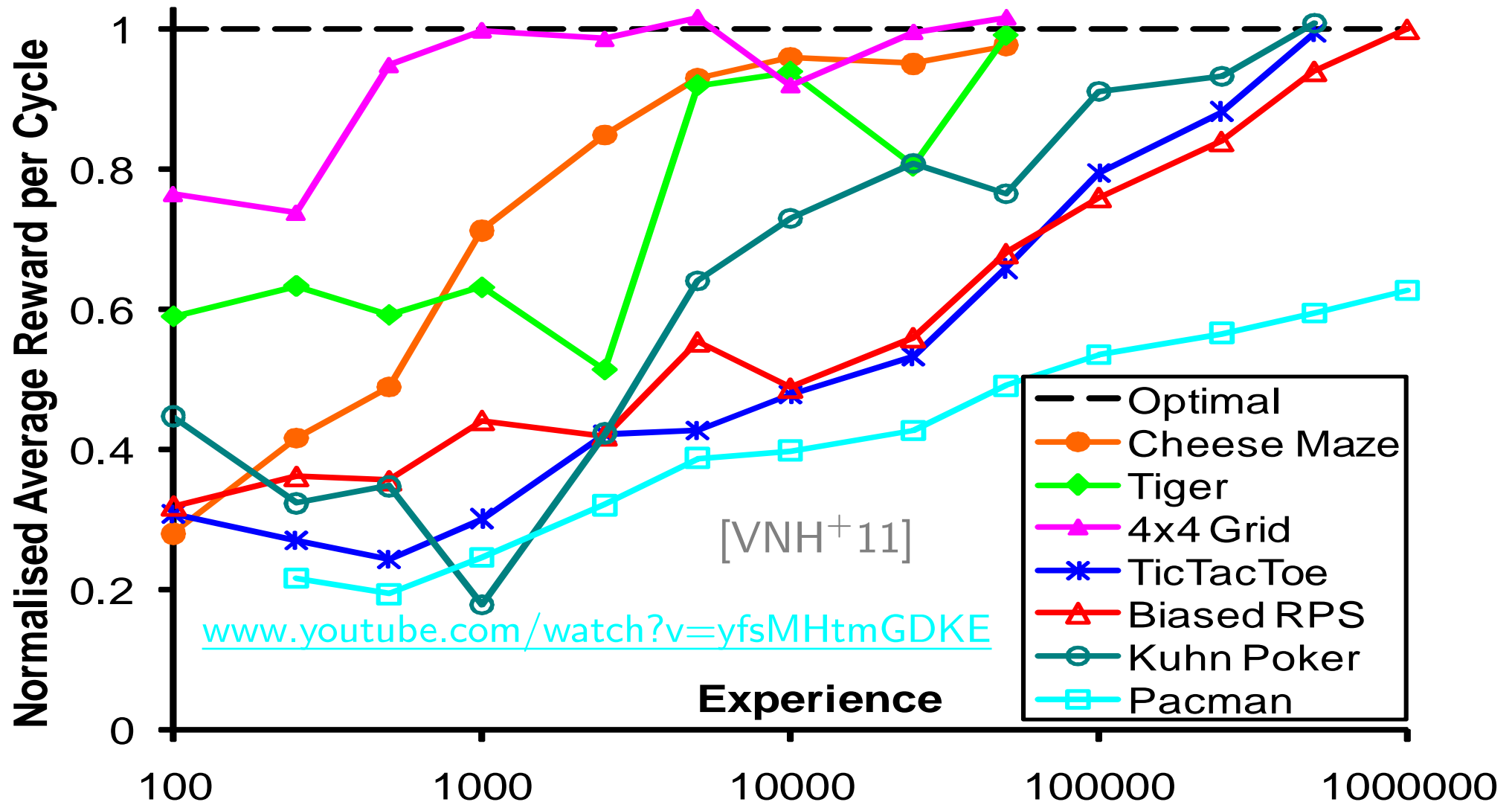
Discussion at End of Course

- What has been achieved?

- Made assumptions.

- General and personal remarks.

- Open problems.

- Philosophical issues.

Exercises

1. [C10] What could the probability p that the sun will rise tomorrow be? What might a philosopher, statistician, physicist, etc. say?

2. [C15] Justify Laplace' rule ($p = \frac{n+1}{n+2}$, where $n=$ #days sun rose in past)

3. [C05] Predict sequences:
   2,3,5,7,11,13,17,19,23,29,31,37,41,43,47,53,59,?
   3,1,4,1,5,9,2,6,5,3,?,
   1,2,3,4,?

4. [C10] Argue in (1) and (3) for different continuations.

Introductory Literature

[HMU06] J. E. Hopcroft, R. Motwani, and J. D. Ullman. *Introduction to Automata Theory, Language, and Computation*. Addison-Wesley, 3rd edition, 2006.

[RN10]   S. J. Russell and P. Norvig. *Artificial Intelligence. A Modern Approach*. Prentice-Hall, Englewood Cliffs, NJ, 3rd edition, 2010.

[LV08]   M. Li and P. M. B. Vitányi. *An Introduction to Kolmogorov Complexity and its Applications*. Springer, Berlin, 3rd edition, 2008.

[SB98]   R. S. Sutton and A. G. Barto. *Reinforcement Learning: An Introduction*. MIT Press, Cambridge, MA, 1998.

[Leg08] S. Legg. *Machine Super Intelligence*. PhD Thesis, Lugano, 2008.

[Hut05] M. Hutter. *Universal Artificial Intelligence: Sequential Decisions based on Algorithmic Probability*. Springer, Berlin, 2005.

See   http://www.hutter1.net/ai/introref.htm   for more.

2 INFORMATION THEORY & KOLMOGOROV COMPLEXITY

- Philosophical Issues

- Definitions & Notation

- Turing Machines

- Kolmogorov Complexity

- Computability Concepts

- Discussion & Exercises

2.1   PHILOSOPHICAL ISSUES: CONTENTS

- Induction/Prediction Examples

- The Need for a Unified Theory

- On the Foundations of AI and ML

- Example 1: Probability of Sunrise Tomorrow

- Example 2: Digits of a Computable Number

- Example 3: Number Sequences

- Occam's Razor to the Rescue

- Foundations of Induction

- Sequential/Online Prediction – Setup

- Dichotomies in AI and ML

- Induction versus Deduction

Philosophical Issues: Abstract

I start by considering the philosophical problems concerning machine learning in general and induction in particular. I illustrate the problems and their intuitive solution on various (classical) induction examples. The common principle to their solution is Occam's simplicity principle. Based on Occam's and Epicurus' principle, Bayesian probability theory, and Turing's universal machine, Solomonoff developed a formal theory of induction. I describe the sequential/online setup considered in this lecture series and place it into the wider machine learning context.

Induction/Prediction Examples

Hypothesis testing/identification: Does treatment X cure cancer?
Do observations of white swans confirm that all ravens are black?

Model selection: Are planetary orbits circles or ellipses?
How many wavelets do I need to describe my picture well?
Which genes can predict cancer?

Parameter estimation: Bias of my coin. Eccentricity of earth's orbit.

Sequence prediction: Predict weather/stock-quote/... tomorrow,
based on past sequence. Continue IQ test sequence like 1,4,9,16,?

Classification can be reduced to sequence prediction:
Predict whether email is spam.

Question: Is there a general & formal & complete & consistent theory
for induction & prediction?

Beyond induction: active/reward learning, fct. optimization, game theory.

The Need for a Unified Theory

Why do we need or should want a unified theory of induction?

- Finding new rules for every particular (new) problem is cumbersome.

- A plurality of theories is prone to disagreement or contradiction.

- Axiomatization boosted mathematics&logic&deduction and so (should) induction.

- Provides a convincing story and conceptual tools for outsiders.

- Automatize induction&science (that's what machine learning does)

- By relating it to existing narrow/heuristic/practical approaches we deepen our understanding of and can improve them.

- Necessary for resolving philosophical problems.

- Unified/universal theories are often beautiful gems.

- There is no convincing argument that the goal is unattainable.

On the Foundations of Artificial Intelligence

- Example: Algorithm/complexity theory: The goal is to find fast algorithms solving problems and to show lower bounds on their computation time. Everything is rigorously defined: algorithm, Turing machine, problem classes, computation time, ...

- Most disciplines start with an informal way of attacking a subject. With time they get more and more formalized often to a point where they are completely rigorous. Examples: set theory, logical reasoning, proof theory, probability theory, infinitesimal calculus, energy, temperature, quantum field theory, ...

- Artificial Intelligence: Tries to build and understand systems that learn from past data, make good prediction, are able to generalize, act intelligently, ... Many terms are only vaguely defined or there are many alternate definitions.

Example 1: Probability of Sunrise Tomorrow

What is the probability $p(1|1^d)$ that the sun will rise tomorrow?
($d =$ past # days sun rose, $1 =$ sun rises. $0 =$ sun will not rise)

- p is undefined, because there has never been an experiment that tested the existence of the sun *tomorrow* (ref. class problem).

- The $p = 1$, because the sun rose in all past experiments.

- $p = 1 - \epsilon$, where ϵ is the proportion of stars that explode per day.

- $p = \frac{d+1}{d+2}$, which is Laplace rule derived from Bayes rule.

- Derive p from the type, age, size and temperature of the sun, even though we never observed another star with those exact properties.

Conclusion: We predict that the sun will rise tomorrow with high probability independent of the justification.

Example 2: Digits of a Computable Number

- Extend 14159265358979323846264338327950288419716939937?

- Looks random?!

- Frequency estimate: $n =$ length of sequence. $k_i =$ number of occured $i \implies$ Probability of next digit being i is $\frac{k_i}{n}$. Asymptotically $\frac{k_i}{n} \to \frac{1}{10}$ (seems to be) true.

- But we have the strong feeling that (i.e. with high probability) the next digit will be 5 because the previous digits were the expansion of π.

- Conclusion: We prefer answer 5, since we see more structure in the sequence than just random digits.

Example 3: Number Sequences

Sequence:
$$x_1, \quad x_2, \quad x_3, \quad x_4, \quad x_5, \quad \ldots$$
$$1, \quad 2, \quad 3, \quad 4, \quad ?, \quad \ldots$$

- $x_5 = 5$, since $x_i = i$ for $i = 1..4$.
- $x_5 = 29$, since $x_i = i^4 - 10i^3 + 35i^2 - 49i + 24$.

Conclusion: We prefer 5, since linear relation involves less arbitrary parameters than 4th-order polynomial.

Sequence: 2,3,5,7,11,13,17,19,23,29,31,37,41,43,47,53,59,?

- 61, since this is the next prime
- 60, since this is the order of the next simple group

Conclusion: We prefer answer 61, since primes are a more familiar concept than simple groups.

On-Line Encyclopedia of Integer Sequences:
http://www.research.att.com/~njas/sequences/

Occam's Razor to the Rescue

- Is there a unique principle which allows us to formally arrive at a
  prediction which
  - coincides (always?) with our intuitive guess -or- even better,
  - which is (in some sense) most likely the best or correct answer?

- Yes! Occam's razor: Use the simplest explanation consistent with
  past data (and use it for prediction).

- Works! For examples presented and for many more.

- Actually Occam's razor can serve as a foundation of machine
  learning in general, and is even a fundamental principle (or maybe
  even the mere definition) of science.

- Problem: Not a formal/mathematical objective principle.
  What is simple for one may be complicated for another.

Dichotomies in Artificial Intelligence

scope of this course	⇔	scope of other lectures
(machine) learning / statistical	⇔	logic/knowledge-based (GOFAI)
online learning	⇔	offline/batch learning
passive prediction	⇔	(re)active learning
Bayes ⇔ MDL	⇔	Expert ⇔ Frequentist
uninformed / universal	⇔	informed / problem-specific
conceptual/mathematical issues	⇔	computational issues
exact/principled	⇔	heuristic
supervised learning	⇔	unsupervised ⇔ RL learning
exploitation	⇔	exploration
action ⇔ decision	⇔	prediction ⇔ induction

Induction ⇔ Deduction

Approximate correspondence between
the most important concepts in induction and deduction.

	Induction	⇔	**Deduction**
Type of inference:	generalization/prediction	⇔	specialization/derivation
Framework:	probability axioms	≙	logical axioms
Assumptions:	prior	≙	non-logical axioms
Inference rule:	Bayes rule	≙	modus ponens
Results:	posterior	≙	theorems
Universal scheme:	Solomonoff probability	≙	Zermelo-Fraenkel set theory
Universal inference:	universal induction	≙	universal theorem prover
Limitation:	incomputable	≙	incomplete (Gödel)
In practice:	approximations	≙	semi-formal proofs
Operation:	computation	≙	proof

The foundations of induction are as solid as those for deduction.

2.2   Definitions & Notation: Contents

- Strings and Natural Numbers

- Identification of Strings & Natural Numbers

- Prefix Sets & Codes / Kraft Inequality

- Pairing Strings

- Asymptotic Notation

Strings and Natural Numbers

- $i, k, n, t \in \mathbb{N} = \{1, 2, 3, ...\}$ natural numbers,

- $\mathbb{B} = \{0, 1\}$ binary alphabet,

- $x, y, z \in \mathbb{B}^*$ finite binary strings,

- $\omega \in \mathbb{B}^\infty$ infinite binary sequences,

- ϵ for the empty string,

- 1^n the string of n ones,

- $\ell(x)$ for the length of string x,

- $xy = x \circ y$ for the concatenation of string x with y.

- $\widehat{=}$ means 'corresponds to'. No formal meaning.

Identification of Strings & Natural Numbers

- Every countable set is $\cong \mathbb{N}$ (by means of a bijection).

- Interpret a string as a binary representation of a natural number.

- Problem: Not unique: $00101 \cong 5 \cong 101$.

- Use some bijection between natural numbers $\mathbb{N}$ and strings $\mathbb{B}^*$.

- Problem: Not unique when concatenated, e.g.
  $5 \circ 2 \cong 10 \circ 1 = 101 = 1 \circ 01 \cong 2 \circ 4$.

- First-order prefix coding $\bar{x} := 1^{\ell(x)} 0 x$.

- Second-order prefix coding $x^{\prime} := \overline{\ell(x)} x$.   [Elias Delta coding]

Identification of Strings & Natural Numbers

$x \in \mathbb{N}_0$	0	1	2	3	4	5	6	7	$\cdots$
$x \in \mathbb{B}^*$	ϵ	0	1	00	01	10	11	000	$\cdots$
$\ell(x)$	0	1	1	2	2	2	2	3	$\cdots$
$\bar{x} = 1^{\ell(x)}0x$	0	100	101	11000	11001	11010	11011	1110000	$\cdots$
$x` = \overline{\ell(x)}x$	0	100 0	100 1	101 00	101 01	101 10	101 11	11000 000	$\cdots$

$x`$ is longer than $\bar{x}$ only for $x < 15$, but shorter for all $x > 30$.

With this identification

$$\log(x+1) - 1 \;<\; \ell(x) \;\leq\; \log(x+1).$$

$$\ell(\bar{x}) \;=\; 2\ell(x) + 1 \leq 2\log(x+1) + 1 \;\sim\; 2\log x$$

$$\ell(x`) \;\leq\; \log(x+1) + 2\log(\log(x+1)+1) + 1 \;\sim\; \log x + 2\log\log x$$

[Higher order code: Recursively define $\epsilon' := 0$ and $x' := 1[\ell(x)-1]'x$

Prefix Sets & Codes

String x is (proper) prefix of y $\quad :\Longleftrightarrow \quad \exists\, z(\neq \epsilon)$ such that $xz = y$.

Set $\mathcal{P}$ is prefix-free or a prefix code $\quad :\Longleftrightarrow \quad$ no element is a proper prefix of another.

Example: A self-delimiting code (e.g. $\mathcal{P} = \{0, 10, 11\}$) is prefix-free.

Kraft Inequality

Theorem 2.1 (Kraft Inequality)

For a prefix code $\mathcal{P}$ we have $\sum_{x \in \mathcal{P}} 2^{-\ell(x)} \leq 1$.

Conversely, let $\ell_1, \ell_2, \dots$ be a countable sequence of natural numbers such that Kraft's inequality $\sum_k 2^{-\ell_k} \leq 1$ is satisfied. Then there exists a prefix code $\mathcal{P}$ with these lengths of its binary code.

Proof of the Kraft-Inequality

Proof $\Rightarrow$: Assign to each $x \in \mathcal{P}$ the interval $\Gamma_x := [0.x, 0.x + 2^{-\ell(x)})$.

Length of interval Γ_x is $2^{-\ell(x)}$.

Intervals are disjoint, since $\mathcal{P}$ is prefix free, hence

$$\sum_{x \in \mathcal{P}} 2^{-\ell(x)} = \sum_{x \in \mathcal{P}} \mathsf{Length}(\Gamma_x) \leq \mathsf{Length}([0,1]) = 1$$

Proof idea $\Leftarrow$:

Choose l_1, l_2, ... in increasing order.

Successively chop off intervals of lengths 2^{-l_1}, 2^{-l_2}, ...

from left to right from $[0,1)$ and

define left interval boundary as code.

Pairing Strings

- $\mathcal{P} = \{\bar{x} : x \in \mathbb{B}^*\}$ is a prefix code with $\ell(\bar{x}) = 2\ell(x) + 1$.

- $\mathcal{P} = \{x\hspace{0.5pt}\text{\textquoteleft} : x \in \mathbb{B}^*\}$ forms an asymptotically shorter prefix code with $\ell(x\hspace{0.5pt}\text{\textquoteleft}) = \ell(x) + 2\ell(\ell(x)) + 1$.

- We pair strings x and y (and z) by $\langle x, y \rangle := x\hspace{0.5pt}\text{\textquoteleft}y$ (and $\langle x, y, z \rangle := x\hspace{0.5pt}\text{\textquoteleft}y\hspace{0.5pt}\text{\textquoteleft}z$) which are uniquely decodable, since $x\hspace{0.5pt}\text{\textquoteleft}$ and $y\hspace{0.5pt}\text{\textquoteleft}$ are prefix.

- Since ' serves as a separator we also write $f(x, y)$ instead of $f(x\hspace{0.5pt}\text{\textquoteleft}y)$ for functions f.

Asymptotic Notation

- $f(n) \overset{n\to\infty}{\longrightarrow} g(n)$ means $\lim_{n\to\infty}[f(n) - g(n)] = 0$.
  Say: f converges to g, w/o implying that $\lim_{n\to\infty} g(n)$ itself exists.

- $f(n) \sim g(n)$ means $\exists 0 < c < \infty : \lim_{n\to\infty} f(n)/g(n) = c$.
  Say: f is asymptotically proportional to g.

- $a \lesssim b$ means a is not much larger than b (precision unspecified).

- $f(x) = O(g(x))$ means $\exists c \forall x : |f(x)| \le c|g(x)| + c$ for some c.
  $f(x) = o(g(x))$ means $\lim_{x\to\infty} f(x)/g(x) = 0$.

- $f(x) \overset{\times}{<} g(x)$ means $f(x) = O(g(x))$,
  $f(x) \overset{+}{<} g(x)$ means $f(x) \le g(x) + O(1)$,
  $f(x) \overset{\log}{<} g(x)$ means $f(x) \le g(x) + O(\log g(x))$.

- $f \overset{*}{>} g :\Leftrightarrow g \overset{*}{<} f$, $f \overset{*}{=} g :\Leftrightarrow f \overset{*}{<} g \wedge f \overset{*}{>} g$, $* \in \{+, \times, \log, ...\}$

2.3   Turing Machines: Contents

- Turing Machines & Effective Enumeration

- Church-Turing Theses

- Short Compiler Assumption

- (Universal) Prefix & Monotone Turing Machine

- Halting Problem

Turing Machines & Effective Enumeration

- Turing machine (TM) = (mathematical model for an) idealized computer.

- See e.g. textbook [HMU06]

  Show Turing Machine in Action: TuringBeispielAnimated.gif

- Instruction i: If symbol on tape under head is 0/1, write 0/1/- and move head left/right/not and goto instruction=state j.

- {partial recursive functions } $\equiv$ {functions computable with a TM}.

- A set of objects $S = \{o_1, o_2, o_3, ...\}$ can be (effectively) enumerated $:\Longleftrightarrow \exists$ TM machine mapping i to $\langle o_i \rangle$, where $\langle \rangle$ is some (often omitted) default coding of elements in S.
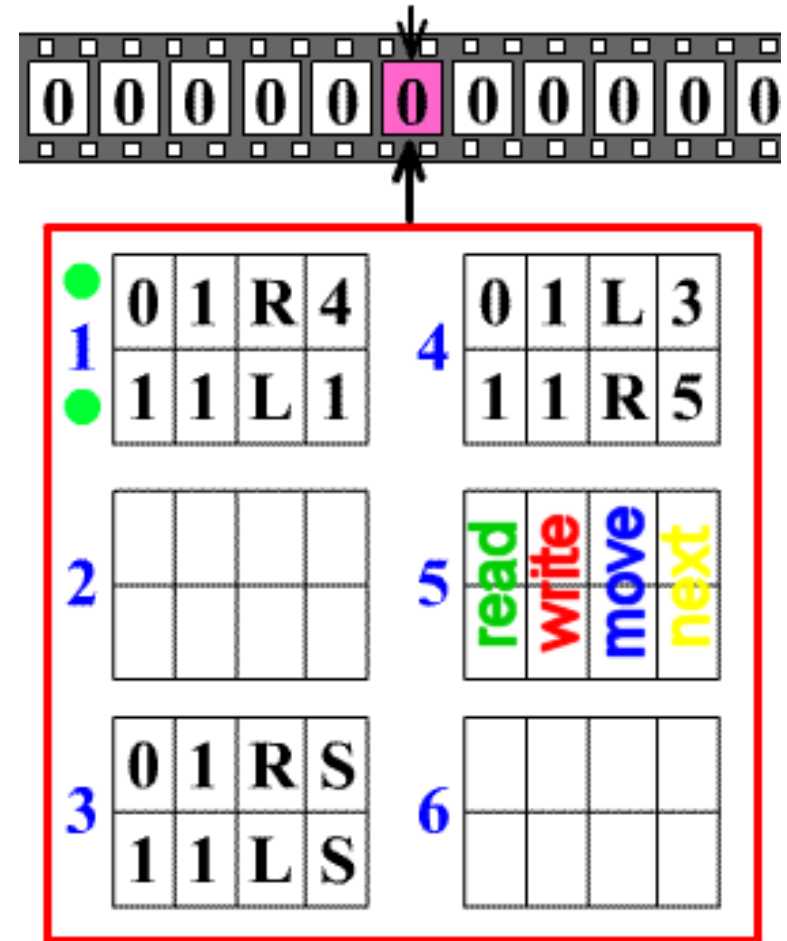
Church-Turing Theses

The importance of partial recursive functions and Turing machines stems from the following theses:

Thesis 2.2 (Turing) Everything that can be reasonably said to be computable by a human using a fixed procedure can also be computed by a Turing machine.

Thesis 2.3 (Church) The class of algorithmically computable numerical functions (in the intuitive sense) coincides with the class of partial recursive functions.

Short Compiler Assumption

Assumption 2.4 (Short compiler)

Given two natural Turing-equivalent formal systems $F1$ and $F2$, then there always exists a single short program on $F2$ which is capable of interpreting all $F1$-programs.

Lisp, Forth, C, Universal TM, ... have mutually short interpreters.

$\Rightarrow$ equivalence is effective

$\Rightarrow$ size of shortest descriptions essentially the same.

Conversion: Interpreter $\rightsquigarrow$ compiler, by attaching the interpreter to the program to be interpreted and by "selling" the result as a compiled version.

Informality of the Theses & Assumption

- The theses are not provable or falsifiable theorems, since human, reasonable, intuitive, and natural have not been defined rigorously.

- One may **define** *intuitively computable* as Turing computable and a *natural Turing-equivalent system* as one which has a small (say $< 10^5$ bits) interpreter/compiler on a once and for all agreed-upon fixed reference universal Turing machine.

- The theses would then be that these definitions are reasonable.

Prefix Turing Machine

For technical reasons we need the following variants of a Turing machine

Definition 2.5 (Prefix Turing machine T (pTM))

- one unidirectional read-only input tape,

- one unidirectional write-only output tape,

- some bidirectional work tapes, initially filled with zeros.

- all tapes are binary (no blank symbol!),

- T halts on input p with output $x :\Longleftrightarrow T(p) = x$

  $:\Longleftrightarrow$ exactly p is to the left of the input head

  and x is to the left of the output head after T halts.

- $\{p : \exists x : T(p) = x\}$ forms a prefix code.

- We call such codes p self-delimiting programs.

Monotone Turing Machine

For technical reasons we need the following variants of a Turing machine

Definition 2.6 (Monotone Turing machine T (mTM))

- one unidirectional read-only input tape,

- one unidirectional write-only output tape,

- some bidirectional work tapes, initially filled with zeros.

- all tapes are binary (no blank symbol!),

- T outputs/computes a string starting with x (or a sequence ω) on input $p :\Longleftrightarrow T(p) = x*$ (or $T(p) = \omega$) $:\Longleftrightarrow p$ is to the left of the input head when the last bit of x is output.

- T may continue operation and need not to halt.

- For given x, $\{p : T(p) = x*\}$ forms a prefix code.

- We call such codes p minimal programs.

Universal Prefix/Monotone Turing Machine

$\langle T \rangle :=$ some canonical binary coding of (table of rules) of TM T

$\Rightarrow$ set of Turing-machines $\{T_1, T_2, ...\}$ can be effectively enumerated.

$\Rightarrow \exists$

Theorem 2.7 (Universal prefix/monotone Turing machine U)
which simulates (any) pTM/mTM T_i with input $y`q$ if fed with input $y`i`q$, i.e.

$$U(y`i`q) = T_i(y`q) \quad \forall y, i, q$$

For $p \neq y`i`q$, $U(p)$ outputs nothing. y is side information.

Proof: See [HMU06] for normal Turing machines. ∎

Illustration

U = some Personal Computer

T_i = Lisp machine,

q = Lisp program.

y = input to Lisp program.

$\Rightarrow T_i(y\text{'}q)$ = execution of Lisp program q with input y on Lisp machine T_i.

$\Rightarrow U(y\text{'}i\text{'}q)$ = running on Personal computer U the Lisp interpreter i with program q and input y.

Call one particular prefix/monotone U the reference UTM.

Halting Problem

We have to pay a big price for the existence of universal TM U:

Namely the undecidability of the halting problem [Turing 1936]:

> **Theorem 2.8 (Halting Problem)**
> There is no TM T: $T(i'p) = 1 \iff T_i(p)$ does not halt.

Proof: Diagonal argument:

Assume such a TM T exists

$\Rightarrow \quad R(i) := T(i'i)$ is computable.

$\Rightarrow \quad \exists j : T_j \equiv R$

$\Rightarrow \quad R(j) = T(j'j) = 1 \iff T_j(j) = R(j)$ does not halt. $\dagger$ ∎

2.4   KOLMOGOROV COMPLEXITY: CONTENTS

- Formalization of Simplicity & Complexity

- Prefix Kolmogorov Complexity K

- Properties of K

- General Proof Ideas

- Monotone Kolmogorov Complexity Km

Formalization of Simplicity & Complexity

- Intuition: A string is simple if it can be described in a few words, like "the string of one million ones",

- and is complex if there is no such short description, like for a random string whose shortest description is specifying it bit by bit.

- Effective descriptions or codes $\Rightarrow$ Turing machines as decoders.

- p is description/code of x on pTM $T :\Longleftrightarrow T(p) = x$.

- Length of shortest description: $K_T(x) := \min_p\{\ell(p) : T(p) = x\}$.

- This complexity measure depends on T :-(

Universality/Minimality of K_U

Is there a TM which leads to shortest codes among all TMs for all x?

Remarkably, there exists a Turing machine (the universal one) which "nearly" has this property:

Theorem 2.9 (Universality/Minimality of K_U)
$$K_U(x) \leq K_T(x) + c_{TU},$$
where $c_{TU} \overset{+}{<} K_U(T) < \infty$ is independent of x

Pair of UTMs U' and U'': $|K_{U'}(x) - K_{U''}(x)| \leq c_{U'U''}$.

Assumption 2.4 holds $\iff$ $c_{U'U''}$ small for natural UTMs U' and U''.

Henceforth we write $O(1)$ for terms like $c_{U'U''}$.

Proof of Universality of K_U

Proof idea: If p is the shortest description of x under $T = T_i$, then $i`p$ is a description of x under U.

Formal proof:

Let p be shortest description of x under T, i.e. $\ell(p) = K_T(x)$.

$\exists i : T = T_i$

$\Rightarrow U(i`p) = x$

$\Rightarrow K_U(x) \leq \ell(i`p) = \ell(p) + c_{TU}$ with $c_{TU} := \ell(i`)$.

Refined proof:

$p := \arg\min_p \{\ell(p) : T(p) = x\} =$ shortest description of x under T

$r := \arg\min_p \{\ell(p) : U(p) = \langle T \rangle\} =$ shortest description of T under U

$q :=$ decode r and simulate T on p.

$\Rightarrow U(qrp) = T(p) = x \Rightarrow$

$K_U(x) \leq \ell(qrp) \stackrel{+}{=} \ell(p) + \ell(r) = K_T(x) + K_U(\langle T \rangle)$.

(Conditional) Prefix Kolmogorov Complexity

Definition 2.10 ((conditional) prefix Kolmogorov complexity)

= shortest program p, for which reference U outputs x (given y):

$$K(x) \; := \; \min_p \{\ell(p) : U(p) = x\},$$

$$K(x|y) \; := \; \min_p \{\ell(p) : U(y`p) = x\}$$

For (non-string) objects: $K(\text{object}) := K(\langle\text{object}\rangle)$,

e.g. $K(x, y) = K(\langle x, y\rangle) = K(x`y)$.

Upper Bound on K

Theorem 2.11 (Upper Bound on K)

$$K(x) \;\stackrel{+}{<}\; \ell(x) + 2\log\ell(x), \qquad K(n) \;\stackrel{+}{<}\; \log n + 2\log\log n$$

Proof:

There exists a TM T_{i_0} with $i_0 = O(1)$ and $T_{i_0}(\epsilon`x`) = x$,

then $U(\epsilon`i_0`x`) = x$,

hence $K(x) \leq \ell(\epsilon`i_0`x`) \stackrel{\pm}{=} \ell(x`) \stackrel{+}{<} \ell(x) + 2\log\ell(x).$ ∎

Lower Bound on K / Kraft Inequality

Theorem 2.12 (lower bound for most n, Kraft inequality)

$$\sum_{x \in \mathbb{B}^*} 2^{-K(x)} \leq 1, \quad K(x) \geq l(x) \quad \text{for 'most'} \quad x$$

$$K(n) \to \infty \quad \text{for} \quad n \to \infty.$$

This is just Kraft's inequality which implies a lower bound on K valid for 'most' n.

'most' means that there are only $o(N)$ exceptions for $n \in \{1, ..., N\}$.

Extra Information & Subadditivity

Theorem 2.13 (Extra Information)

$$K(x|y) \overset{+}{<} K(x) \overset{+}{<} K(x,y)$$

Providing side information y can never increase code length,

Requiring extra information y can never decrease code length.

Proof: Similarly to Theorem 2.11

Theorem 2.14 (Subadditivity)

$$K(xy) \overset{+}{<} K(x,y) \overset{+}{<} K(x) + K(y|x) \overset{+}{<} K(x) + K(y)$$

Coding x and y separately never helps.

Proof: Similarly to Theorem 2.13

Symmetry of Information

> **Theorem 2.15 (Symmetry of Information)**
>
> $$K(x|y, K(y)) + K(y) \;\overset{+}{=}\; K(x, y) \;\overset{+}{=}\; K(y, x) \;\overset{+}{=}\; K(y|x, K(x)) + K(x)$$

Is the analogue of the logarithm of the multiplication rule for conditional probabilities (see later).

Proof: $\geq\,=\,\leq$ similarly to Theorem 2.14.

For $\leq\,=\,\geq$, deep result: see [LV08, Th.3.9.1]. ∎

Proof Sketch of $K(y|x) + K(x) \leq K(x,y) + O(\log)$

all $+O(\log)$ terms will be suppressed and ignored. Counting argument:

(1) Assume $K(y|x) > K(x,y) - K(x)$.

(2) $(x,y) \in A := \{\langle u, z \rangle : K(u,z) \leq k\}$, $k := K(x,y)$, $K(k) = O(\log)$

(3) $y \in A_x := \{z : K(x,z) \leq k\}$

(4) Use index of y in A_x to describe y: $K(y|x) \leq \log |A_x|$

(5) $\log |A_x| > K(x,y) - K(x) =: l$ by (1) and (4), $K(l) = O(\log)$

(6) $x \in U := \{u : \log |A_u| > l\}$ by (5)

(7) $\{\langle u, z \rangle : u \in U, z \in A_u\} \subseteq A$

(8) $\log |A| \leq k$ by (2), since at most 2^k codes of length $\leq k$

(9) $2^l |U| < \min\{|A_u| : u \in U\} |U| \leq |A| \leq 2^k$ by (6),(7),(8), resp.

(10) $K(x) \leq \log |U| < k - l = K(x)$ by (6) and (9). Contradiction! ∎

Information Non-Increase

> **Theorem 2.16 (Information Non-Increase)**
>
> $$K(f(x)) \overset{+}{<} K(x) + K(f) \quad \text{for recursive} \quad f : \mathbb{B}^* \to \mathbb{B}^*$$

Definition: The Kolmogorov complexity $K(f)$ of a function f is defined as the length of the shortest self-delimiting program on a prefix TM computing this function.

Interpretation: Transforming x does not increase its information content.

Hence: Switching from one coding scheme to another by means of a recursive bijection leaves K unchanged within additive $O(1)$ terms.

Proof: Similarly to Theorem 2.14

Coding Relative to Probability Distribution, Minimal Description Length (MDL) Bound

Theorem 2.17 (Probability coding / MDL)

$$K(x) \overset{+}{<} -\log P(x) + K(P)$$

$$\text{if} \quad P : \mathbb{B}^* \to [0,1] \text{ is enumerable and } \sum_{x \in \mathbb{B}^*} P(x) \leq 1$$

This is at the heart of the MDL principle [Ris89],

which approximates $K(x)$ by $-\log P(x) + K(P)$.

Proof of MDL Bound

Proof for $\sum_x P(x) = 1$:   [see [LV08, Sec.4.3] for general P]

Idea: Use the Shannon-Fano code based on probability distribution P.

Let $s_x := \lceil -\log_2 P(x)\rceil \in \mathbb{N}$

$\Rightarrow \sum_x 2^{-s_x} \leq \sum_x P(x) \leq 1$.

$\Rightarrow$: $\exists$ prefix code p for x with $\ell(p) = s_x$ (by Kraft inequality)

Since the proof of Kraft inequality for known $\sum_x P(x)$ is (can be made) constructive, there exists an effective prefix code in the sense that $\exists$ pTM T : $\forall x \exists p : T(p) = x$ and $\ell(p) = s_x$.

$\Rightarrow K(x) \overset{+}{<} K_T(x) + K(T) \leq s_x + K(T) \overset{+}{<} -\mathrm{log} P(x) + K(P)$

where we used Theorem 2.9.   ∎

General Proof Ideas

- All upper bounds on $K(z)$ are easily proven by devising **some** (effective) code for z of the length of the right-hand side of the inequality and by noting that $K(z)$ is the length of the shortest code among all possible effective codes.

- Lower bounds are usually proven by counting arguments (Easy for Thm.2.12 by using Thm.2.1 and hard for Thm.2.15)

- The number of short codes is limited.
  More precisely: The number of prefix codes of length $\leq \ell$ is bounded by 2^ℓ.

Remarks on Theorems 2.11-2.17

All (in)equalities remain valid if K is (further) conditioned under some z, i.e. $K(...) \rightsquigarrow K(...|z)$ and $K(...|y) \rightsquigarrow K(...|y, z)$.
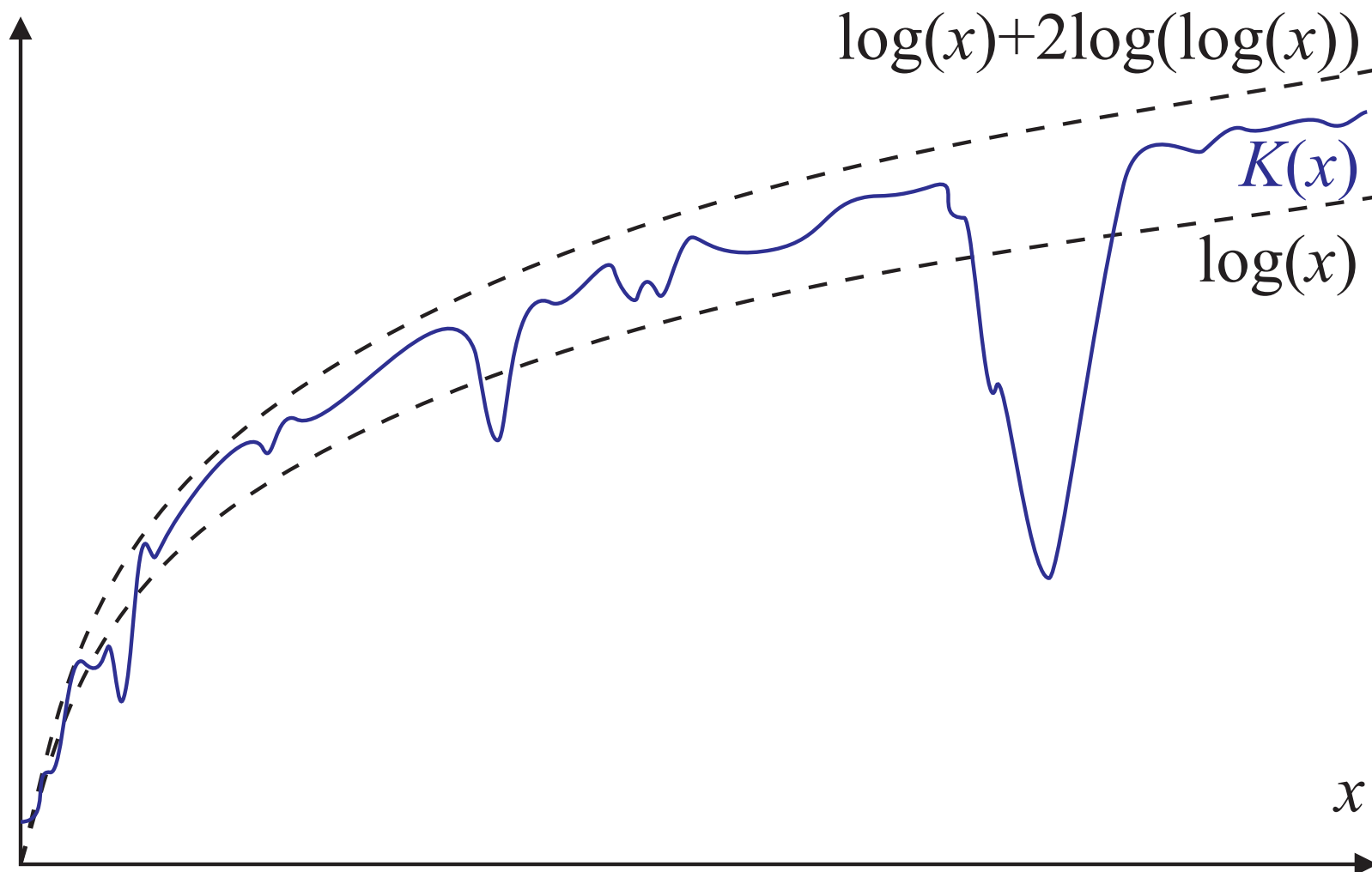
Relation to Shannon Entropy

Let $X, Y \in \mathcal{X}$ be discrete random variables with distribution $P(X, Y)$.

Definition 2.18 (Definition of Shannon entropy)

$\mathrm{Entropy}(X) \quad \equiv H(X) \quad := -\sum_{x \in \mathcal{X}} P(x) \log P(x)$

$\mathrm{Entropy}(X|Y) \equiv H(X|Y) := -\sum_{y \in \mathcal{Y}} P(y) \sum_{x \in \mathcal{X}} P(x|y) \log P(x|y)$

Theorem 2.19 (Properties of Shannon entropy)

- Upper bound: $\qquad\qquad\qquad H(X) \leq \log|\mathcal{X}| \quad = n$ for $\mathcal{X} = \mathbb{B}^n$

- Extra information: $\qquad\qquad\quad H(X|Y) \leq H(X) \leq H(X, Y)$

- Subadditivity: $\qquad\qquad\qquad\quad H(X, Y) \leq H(X) + H(Y)$

- Symmetry: $\qquad H(X|Y) + H(Y) = H(X, Y) = H(Y, X)$

- Information non-increase: $\qquad\quad H(f(X)) \leq H(X)$ for any f

---

Relations for H are essentially expected versions of relations for K.

Monotone Kolmogorov Complexity Km

A variant of K is the monotone complexity $Km(x)$ defined as the shortest program on a monotone TM computing a string starting with x:

Theorem 2.20 (Monotone Kolmogorov Complexity Km)

$$Km(x) := \min_p\{\ell(p) : U(p) = x*\}$$

has the following properties:

- $Km(x) \overset{+}{<} \ell(x),$

- $Km(xy) \geq Km(x) \in \mathbb{N}_0,$

- $Km(x) \overset{+}{<} -\log \mu(x) + K(\mu)$ if μ comp. measure (defined later).

It is natural to call an infinite sequence ω computable if $Km(\omega) < \infty$.

2.5   COMPUTABILITY CONCEPTS: CONTENTS

- Computability Concepts

- Computability: Discussion

- (Non)Computability of K and Km

Computable Functions

Definition 2.21 (Computable functions) We consider functions $f : \mathbb{N} \to \mathbb{R}$:

f is finitely computable or recursive *iff* there are Turing machines $T_{1/2}$ with output interpreted as natural numbers and $f(x) = \frac{T_1(x)}{T_2(x)}$,

$$\Downarrow$$

f is estimable *iff* $\exists$ recursive $\phi(\cdot, \cdot)$ $\forall$ $\varepsilon > 0$: $|\phi(x, \lfloor \frac{1}{\varepsilon} \rfloor) - f(x)| < \varepsilon$ $\forall x$.

$$\Downarrow$$

f is lower semicomputable or enumerable *iff* $\phi(\cdot, \cdot)$ is recursive and $\lim_{t \to \infty} \phi(x, t) = f(x)$ and $\phi(x, t) \le \phi(x, t+1)$.

$$\Downarrow$$

f is approximable *iff* $\phi(\cdot, \cdot)$ is recursive and $\lim_{t \to \infty} \phi(x, t) = f(x)$.

Computability: Discussion

- What we call **estimable** is often just called **computable**.

- If f is estimable we can determine an interval estimate
  $f(x) \in [\hat{y} - \varepsilon, \hat{y} + \varepsilon]$.

- If f is only approximable or semicomputable we can still come
  arbitrarily close to $f(x)$ but we cannot devise a terminating
  algorithm which produces an ε-approximation.

- f is upper semicomputable or co-enumerable
  $:\Leftrightarrow -f$ is lower semicomputable or enumerable.

- In the case of lower/upper semicomputability we can at least finitely
  compute lower/upper bounds to $f(x)$.

- In case of approximability, the weakest computability form, even this
  capability is lost.

(Non)Computability of K and Km complexity

> **Theorem 2.22 ((Non)computability of K and Km Complexity)**
> The prefix complexity $K : \mathbb{B}^* \to \mathbb{N}$ and the monotone complexity $Km : \mathbb{B}^* \to \mathbb{N}$ are co-enumerable, but not finitely computable.

Proof: Assume K is computable.

$\Rightarrow f(m) := \min\{n : K(n) \geq m\}$ exists by Theorem 2.12 and is computable (and unbounded).

$K(f(m)) \geq m$ by definition of f.

$K(f(m)) \leq K(m) + K(f) \stackrel{+}{<} 2\log m$ by Theorem 2.16 and 2.11.

$\Rightarrow m \leq 2\log m + c$ for some c, but this is false for sufficiently large m.

Co-enumerability of K as exercise. ∎

2.6   DISCUSSION: CONTENTS

- Applications of KC/AIT

- Outlook

- Summary

- Exercises

- Literature

KC/AIT is a Useful Tool in/for

- quantifying simplicity/complexity and Ockham's razor,
- quantification of Gödel's incompleteness result,
- computational learning theory,
- combinatorics,
- time and space complexity of computations,
- average case analysis of algorithms,
- formal language and automata theory,
- lower bound proof techniques,
- probability theory,
- string matching,
- clustering by compression,
- physics and thermodynamics of computing,
- statistical thermodynamics / Boltzmann entropy / Maxwell daemon

General Applications of AIT/KC

- (Martin-Löf) randomness of individual strings/sequences/object,

- information theory and statistics of individual objects,

- universal probability,

- general inductive reasoning and inference,

- universal sequence prediction,

- the incompressibility proof method,

- Turing machine complexity,

- structural complexity theory,

- oracles,

- logical depth,

- universal optimal search,

- dissipationless reversible computing,

- information distance,

- algorithmic rate-distortion theory.

Industrial Applications of KC/AIT

• language recognition, linguistics,

• picture similarity,

• bioinformatics,

• phylogeny tree reconstruction,

• cognitive psychology,

• optical / handwritten character recognitions.

Outlook

- Many more KC variants beyond K, Km, and KM.

- Resource (time/space) bounded (computable!) KC.

- See the excellent textbook [LV08].

Summary

- A quantitative theory of information has been developed.

- Occam's razor serves as the philosophical foundation of induction and scientific reasoning.

- All enumerable objects are coded=identified as strings.

- Codes need to be prefix free, satisfying Kraft's inequality.

- Augment Church-Turing thesis with the short compiler assumption.

- Kolmogorov complexity quantifies Occam's razor, and is **the** complexity measure.

- Major drawback: K is only semicomputable.

Exercises 1–6

1. [C05] Formulate a sequence prediction task as a classification task (Hint: add time tags).

2. [C15] Complete the table identifying natural numbers with (prefix) strings to numbers up to 16. For which x is x' longer/shorter than $\bar{x}$, and how much?

3. [C10] Show that $\log(x+1) - 1 < \ell(x) \leq \log(x+1)$ and $\ell(x') \lesssim \log x + 2\log\log x$.

4. [C15] Prove $\Leftarrow$ of Theorem 2.1

5. [C05] Show that for every string x there exists a universal Turing machine U' such that $K_{U'}(x) = 1$. Argue that U' is not a natural Turing machine if x is complex.

6. [C10] Show $K(0^n) \stackrel{+}{=} K(1^n) \stackrel{+}{=} K(n \text{ digits of } \pi) \stackrel{+}{=} K(n) \leq \log n + O(\log\log n)$.

Exercises 7–12

7. [C15] The halting sequence $h_{1:\infty}$ is defined as $h_i = 1 \iff T_i(\varepsilon)$ halts, otherwise $h_i = 0$. Show $K(h_1...h_n) \leq 2\log n + O(\log\log n)$ and $Km(h_1...h_n) \leq \log n + O(\log\log n)$.

8. [C25] Show that the Kolmogorov complexity K, the halting sequence h, and the halting probability $\Omega := \sum_{p:U(p)\ halts} 2^{-\ell(p)}$ are Turing-reducible to each other.

9. [C10–40] Complete the proofs of the properties of K.

10. [C15] Show that a function is estimable if and only if it is upper and lower semi-computable.

11. [C10] Prove Theorem 2.20 items 1-2.

12. [C15] Prove the implications in Definition 2.21

Literature

[HMU01] J. E. Hopcroft, R. Motwani, and J. D. Ullman. *Introduction to Automata Theory, Language, and Computation*. Addison-Wesley, 3rd edition, 2006.

[LV08]   M. Li and P. M. B. Vitányi. *An Introduction to Kolmogorov Complexity and its Applications*. Springer, Berlin, 3rd edition, 2008.

[Cal02] C. S. Calude. *Information and Randomness: An Algorithmic Perspective*. Springer, Berlin, 2nd edition, 2002.

[Hut05] M. Hutter. *Universal Artificial Intelligence: Sequential Decisions based on Algorithmic Probability*. Springer, Berlin, 2005. http://www.hutter1.net/ai/uaibook.htm

3 BAYESIAN PROBABILITY THEORY

- Uncertainty and Probability

- Frequency Interpretation: Counting

- Objective Interpretation: Uncertain Events

- Subjective Interpretation: Degrees of Belief

- Kolmogorov's Axioms of Probability Theory

- Bayes and Laplace Rule

- How to Determine Priors

- Discussion

Bayesian Probability Theory: Abstract

The aim of probability theory is to describe uncertainty. There are various sources and interpretations of uncertainty. I compare the frequency, objective, and subjective probabilities, and show that they all respect the same rules. I derive Bayes' and Laplace's famous and fundamental rules, discuss the indifference, the maximum entropy, and Ockham's razor principle for choosing priors, and finally present two brain-teasing paradoxes.

Uncertainty and Probability

The aim of probability theory is to describe uncertainty.

Sources/interpretations for uncertainty:

- Frequentist: probabilities are relative frequencies.
  (e.g. the relative frequency of tossing head.)

- Objectivist: probabilities are real aspects of the world.
  (e.g. the probability that some atom decays in the next hour)

- Subjectivist: probabilities describe an agent's degree of belief.
  (e.g. it is (im)plausible that extraterrestrians exist)

3.1 Frequency Interpretation: Counting: Contents

- Frequency Interpretation: Counting

- Problem 1: What does Probability Mean?

- Problem 2: Reference Class Problem

- Problem 3: Limited to I.I.D

Frequency Interpretation: Counting

- The frequentist interprets probabilities as relative frequencies.

- If in a sequence of n independent identically distributed (i.i.d.) experiments (trials) an event occurs $k(n)$ times, the relative frequency of the event is $k(n)/n$.

- The limit $\lim_{n\to\infty} k(n)/n$ is defined as the probability of the event.

- For instance, the probability of the event head in a sequence of repeatedly tossing a fair coin is $\frac{1}{2}$.

- The frequentist position is the easiest to grasp, but it has several shortcomings:

What does Probability Mean?

- What does it **mean** that a property holds with a certain probability?

- The frequentist obtains probabilities from physical processes.

- To scientifically reason about probabilities one needs a math theory. Problem: how to define random sequences?

- This is much more intricate than one might think, and has only been solved in the 1960_s by Kolmogorov and Martin-Löf.

Problem 1: Frequency Interpretation is Circular

- Probability of event E is $p := \lim_{n \to \infty} \frac{k_n(E)}{n}$,
  $n = \#$ i.i.d. trials, $k_n(E) = \#$ occurrences of event E in n trials.

- Problem: Limit may be anything (or nothing):
  e.g. a fair coin can give: Head, Head, Head, Head, ... $\Rightarrow p = 1$.

- Of course, for a fair coin this sequence is "unlikely".
  For fair coin, $p = 1/2$ with "high probability".

- But to make this statement rigorous we need to formally know what
  "high probability" means.   Circularity!

Problem 2: Reference Class Problem

- Philosophically and also often in real experiments it is hard to justify the choice of the so-called reference class.

- For instance, a doctor who wants to determine the chances that a patient has a particular disease by counting the frequency of the disease in "similar" patients.

- But if the doctor considered everything he knows about the patient (symptoms, weight, age, ancestry, ...) there would be no other comparable patients left.

Problem 3: Limited to I.I.D

- The frequency approach is limited to a (sufficiently large) sample of i.i.d. data.

- In complex domains typical for AI, data is often non-i.i.d. and (hence) sample size is often 1.

- For instance, a single non-i.i.d. historic weather data sequences is given. We want to know whether certain properties hold for this **particular** sequence.

- Classical probability non-constructively tells us that the set of sequences possessing these properties has measure near 1, but cannot tell which objects have these properties, in particular whether the single observed sequence of interest has these properties.

3.2   Objective Interpretation: Uncertain Events: Contents

- Objective Interpretation: Uncertain Events

- Kolmogorov's Axioms of Probability Theory

- Conditional Probability

- Example: Fair Six-Sided Die

- Bayes' Rule 1

Objective Interpretation: Uncertain Events

- For the objectivist probabilities are real aspects of the world.

- The outcome of an observation or an experiment is not deterministic, but involves physical random processes.

- The set Ω of all possible outcomes is called the sample space.

- It is said that an event $E \subset \Omega$ occurred if the outcome is in E.

- In the case of i.i.d. experiments the probabilities p assigned to events E should be interpretable as limiting frequencies, but the application is not limited to this case.

- The Kolmogorov axioms formalize the properties which probabilities should have.

Kolmogorov's Axioms of Probability Theory

Axioms 3.1 (Kolmogorov's axioms of probability theory)
Let Ω be the sample space. Events are subsets of Ω.

- If A and B are events, then also the intersection $A \cap B$, the union $A \cup B$, and the difference $A \setminus B$ are events.

- The sample space Ω and the empty set $\{\}$ are events.

- There is a function p which assigns nonnegative reals, called probabilities, to each event.

- $p(\Omega) = 1$, $p(\{\}) = 0$.

- $p(A \cup B) = p(A) + p(B) - p(A \cap B)$.

- For a decreasing sequence $A_1 \supset A_2 \supset A_3...$ of events with $\bigcap_n A_n = \{\}$ we have $\lim_{n \to \infty} p(A_n) = 0$.

The function p is called a probability mass function, or, probability measure, or, more loosely probability distribution (function).

Conditional Probability

Definition 3.2 (Conditional probability) If A and B are events with $p(A) > 0$, then the probability that event B will occur under the condition that event A has occured is defined as

$$p(B|A) := \frac{p(A \cap B)}{p(A)}$$

- $p(\cdot|A)$ (as a function of the first argument) is also a probability measure, if $p(\cdot)$ satisfies the Kolmogorov axioms.

- One can "verify the correctness" of the Kolmogorov axioms and the definition of conditional probabilities in the case where probabilities are identified with limiting frequencies.

- But the idea is to take the axioms as a starting point to avoid some of the frequentist's problems.

Example: Fair Six-Sided Die

- Sample space: $\Omega = \{1, 2, 3, 4, 5, 6\}$

- Events: Even$= \{2, 4, 6\}$, Odd$= \{1, 3, 5\}$ $\subseteq \Omega$

- Probability: $p(6) = \frac{1}{6}$, $p(\text{Even}) = p(\text{Odd}) = \frac{1}{2}$

- Outcome: $6 \in E$.

- Conditional probability: $p(6|\text{Even}) = \dfrac{p(6 \, and \, \text{Even})}{p(\text{Even})} = \dfrac{1/6}{1/2} = \dfrac{1}{3}$

Bayes' Rule 1

Theorem 3.3 (Bayes' rule 1) If A and B are events with $p(A) > 0$ and $p(B) > 0$,   then $p(B|A) = \dfrac{p(A|B)p(B)}{p(A)}$

Bayes' theorem is easily proven by applying Definition 3.2 twice.

3.3 Subjective Interpretation: Degrees of Belief: Contents

- Subjective Interpretation: Degrees of Belief

- Cox's Axioms for Beliefs

- Cox's Theorem

- Bayes' Famous Rule

Subjective Interpretation: Degrees of Belief

- The subjectivist uses probabilities to characterize an agent's degree of belief in something, rather than to characterize physical random processes.

- This is the most relevant interpretation of probabilities in AI.

- We define the plausibility of an event as the degree of belief in the event, or the subjective probability of the event.

- It is natural to assume that plausibilities/beliefs $\mathrm{Bel}(\cdot|\cdot)$ can be repr. by real numbers, that the rules qualitatively correspond to common sense, and that the rules are mathematically consistent. $\Rightarrow$

Cox's Axioms for Beliefs

Axioms 3.4 (Cox's (1946) axioms for beliefs)

- The degree of belief in event B (plausibility of event B), given that event A occurred can be characterized by a real-valued function $\mathrm{Bel}(B|A)$.

- $\mathrm{Bel}(\Omega \setminus B|A)$ is a twice differentiable function of $\mathrm{Bel}(B|A)$ for $A \neq \{\}$.

- $\mathrm{Bel}(B \cap C|A)$ is a twice continuously differentiable function of $\mathrm{Bel}(C|B \cap A)$ and $\mathrm{Bel}(B|A)$ for $B \cap A \neq \{\}$.

One can motivate the functional relationship in Cox's axioms by analyzing all other possibilities and showing that they violate common sense [Tribus 1969].

The somewhat strong differentiability assumptions can be weakened to more natural continuity and monotonicity assumptions [Aczel 1966].

Cox's Theorem

Theorem 3.5 (Cox's theorem) Under Axioms 3.4 and some additional denseness conditions, $\mathrm{Bel}(\cdot|A)$ is isomorphic to a probability function in the sense that there is a continuous one–to-one onto function $g : \mathbb{R} \to [0,1]$ such that $p := g \circ \mathrm{Bel}$ satisfies Kolmogorov's Axioms 3.1 and is consistent with Definition 3.2.

Only recently, a loophole in Cox's and other's derivations have been exhibited [Paris 1995] and fixed by making the mentioned "additional denseness assumptions".

Conclusion: Plausibilities follow the same rules as limiting frequencies.

Other justifications: Gambling / Dutch Book / Utility theory

Bayes' Famous Rule

Let D be some possible data (i.e. D is event with $p(D) > 0$) and $\{H_i\}_{i \in I}$ be a countable complete class of mutually exclusive hypotheses (i.e. H_i are events with $H_i \cap H_j = \{\} \; \forall i \neq j$ and $\bigcup_{i \in I} H_i = \Omega$).

Given: $p(H_i) =$ a priori plausibility of hypotheses H_i (subj. prob.)

Given: $p(D|H_i) =$ likelihood of data D under hypothesis H_i (obj. prob.)

Goal: $p(H_i|D) =$ a posteriori plausibility of hypothesis H_i (subj. prob.)

Theorem 3.6 (Bayes' rule) $\qquad p(H_i|D) = \dfrac{p(D|H_i)p(H_i)}{\sum_{i \in I} p(D|H_i)p(H_i)}$

Proof sketch: From the definition of conditional probability and

$$\sum_{i \in I} p(H_i|...) = 1 \quad \Rightarrow \quad \sum_{i \in I} p(D|H_i)p(H_i) = \sum_{i \in I} p(H_i|D)p(D) = p(D)$$

Proof of Bayes Rule

$p(A \cup B) = p(A) + p(B)$ if $A \cap B = \{\}$, since $p(\{\}) = 0$.

$\Rightarrow$ for finite I by induction: $\sum_{i \in I} p(H_i) = p(\bigcup_i H_i) = p(\Omega) = 1$.

$\Rightarrow$ for countably infinite $I = \{1, 2, 3, ...\}$ with $S_n := \bigcup_{i=n}^{\infty} H_i$:

$$\sum_{i=1}^{n-1} p(H_i) + p(S_n) = p(\bigcup_{i=1}^{n-1} H_i \cup \bigcup_{i=n}^{\infty} H_i) = p(\Omega) = 1$$

$S_1 \supset S_2 \supset S_3....$

Further, $\omega \in \Omega \Rightarrow \exists n : \omega \in H_n \Rightarrow \omega \notin H_i \forall i > n \Rightarrow \omega \notin S_i \ \forall i > n$

$\Rightarrow \omega \notin \bigcap_n S_n \Rightarrow \bigcap_n S_n = \{\}$ (since ω was arbitrary).

$$\Rightarrow \quad 1 = \lim_{n \to \infty} \sum_{i=1}^{n-1} p(H_i) + p(S_n) = \sum_{i=1}^{\infty} p(H_i) = \sum_{i \in I} p(H_i)$$

Proof of Bayes Rule (ctnd)

By Definition 3.2 of conditional probability we have

$$p(H_i|D)p(D) \;=\; p(H_i \cap D) \;=\; p(D|H_i)p(H_i)$$

Summing over all hypotheses H_i gives

$$\sum_{i \in I} p(D|H_i)p(H_i) \;=\; \sum_{i \in I} p(H_i|D) \cdot p(D) \;=\; 1 \cdot p(D)$$

$$\Rightarrow \quad p(H_i|D) \;=\; \frac{p(D|H_i)p(H_i)}{p(D)} \;=\; \frac{p(D|H_i)p(H_i)}{\sum_{i \in I} p(D|H_i)p(H_i)}$$

3.4   Determining Priors: Contents

- How to Choose the Prior?

- Indifference or Symmetry Principle

- Example: Bayes' and Laplace's Rule

- The Maximum Entropy Principle ...

- Occam's Razor — The Simplicity Principle

How to Choose the Prior?

The probability axioms allow relating probabilities and plausibilities of different events, but they do not uniquely fix a numerical value for each event, except for the sure event Ω and the empty event $\{\}$.

We need new principles for determining values for at least some basis events from which others can then be computed.

There seem to be only 3 general principles:

- The principle of indifference — the symmetry principle

- The maximum entropy principle

- Occam's razor — the simplicity principle

Concrete: How shall we choose the hypothesis space $\{H_i\}$ and their prior $p(H_i)$.

Indifference or Symmetry Principle

Assign same probability to all hypotheses:

$p(H_i) = \frac{1}{|I|}$ for finite I

$p(H_\theta) = [\text{Vol}(\Theta)]^{-1}$ for compact and measurable Θ.

$\Rightarrow p(H_i|D) \propto p(D|H_i) \stackrel{\wedge}{=}$ classical Hypothesis testing (Max.Likelihood).

Example: $H_\theta =$Bernoulli(θ) with $p(\theta) = 1$ for $\theta \in \Theta := [0,1]$.

Problems: Does not work for "large" hypothesis spaces:

(a) Uniform distr. on infinite $I = \mathbb{N}$ or noncompact Θ not possible!

(b) Reparametrization: $\theta \rightsquigarrow f(\theta)$. Uniform in θ is not uniform in $f(\theta)$.

Example: "Uniform" distr. on space of all (binary) sequences $\{0,1\}^\infty$:

$p(x_1...x_n) = (\frac{1}{2})^n \,\forall n\forall x_1...x_n \Rightarrow p(x_{n+1} = 1|x_1...x_n) = \frac{1}{2}$ always!

Inference so not possible (No-Free-Lunch myth).

Predictive setting: All we need is $p(x)$.

Example: Bayes' and Laplace's Rule

Assume data is generated by a biased coin with head probability θ, i.e. $H_\theta :=$ Bernoulli(θ) with $\theta \in \Theta := [0,1]$.

Finite sequence: $x = x_1 x_2 ... x_n$ with n_1 ones and n_0 zeros.

Sample infinite sequence: $\omega \in \Omega = \{0,1\}^\infty$

Basic event: $\Gamma_x = \{\omega : \omega_1 = x_1, ..., \omega_n = x_n\} =$ set of all sequences starting with x.

Data likelihood: $p_\theta(x) := p(\Gamma_x | H_\theta) = \theta^{n_1}(1-\theta)^{n_0}$.
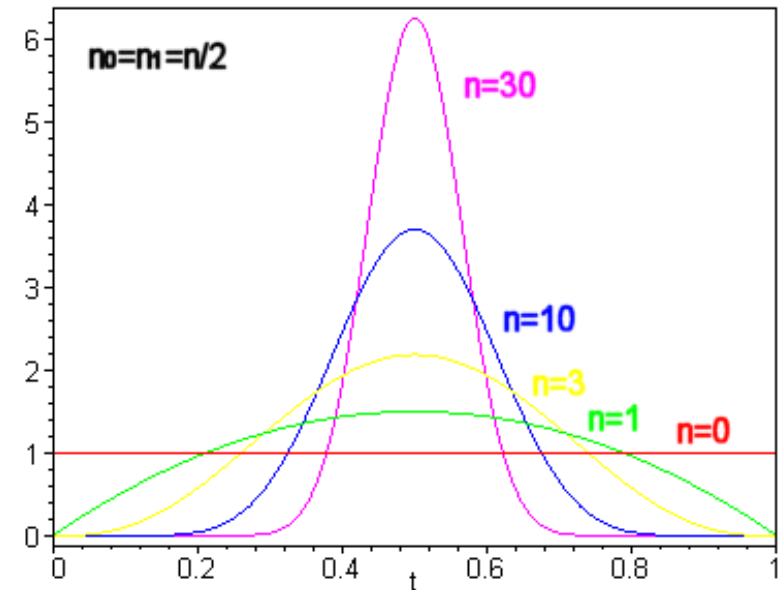
Bayes (1763): Uniform prior plausibility: $p(\theta) := p(H_\theta) = 1$

$\quad\quad (\int_0^1 p(\theta)\, d\theta = 1$ instead $\sum_{i \in I} p(H_i) = 1)$

Evidence: $p(x) = \int_0^1 p_\theta(x) p(\theta)\, d\theta = \int_0^1 \theta^{n_1}(1-\theta)^{n_0}\, d\theta = \frac{n_1! n_0!}{(n_0 + n_1 + 1)!}$

Example: Bayes' and Laplace's Rule

Bayes: Posterior plausibility of θ
after seeing x is:

$$p(\theta|x) = \frac{p(x|\theta)p(\theta)}{p(x)} = \frac{(n+1)!}{n_1!n_0!}\theta^{n_1}(1-\theta)^{n_0}$$

.

Laplace: What is the probability of seeing 1 after having observed x?

$$p(x_{n+1} = 1|x_1...x_n) = \frac{p(x1)}{p(x)} = \frac{n_1+1}{n+2}$$

Laplace believed that the sun had risen for 5000 years $= 1'826'213$ days, so he concluded that the probability of doomsday tomorrow is $\frac{1}{1826215}$.

The Maximum Entropy Principle ...

... is based on the foundations of statistical physics.

... chooses among a class of distributions the one which has maximal entropy.

The class is usually characterized by constraining the class of all distributions.

... generalizes the symmetry principle.

... reduces to the symmetry principle in the special case of no constraint.

... has same limitations as the symmetry principle.

Occam's Razor — The Simplicity Principle

- Only Occam's razor (in combination with Epicurus' principle) is general enough to assign prior probabilities in *every* situation.

- The idea is to assign high (subjective) probability to simple events, and low probability to complex events.

- Simple events (strings) are more plausible a priori than complex ones.

- This gives (approximately) justice to both Occam's razor and Epicurus' principle.

    this prior will be quantified and discussed later

3.5 Discussion: Contents

- Probability Jargon

- Applications

- Outlook

- Summary

- Exercises

- Literature

Probability Jargon

Example: (Un)fair coin: $\Omega = \{\text{Tail,Head}\} \simeq \{0,1\}$. $p(1) = \theta \in [0,1]$:

Likelihood: $p(1101|\theta) = \theta \times \theta \times (1-\theta) \times \theta$

Maximum Likelihood (ML) estimate: $\hat{\theta} = \arg\max_\theta p(1101|\theta) = \frac{3}{4}$

Prior: If we are indifferent, then $p(\theta) = \text{const.}$

Evidence: $p(1101) = \sum_\theta p(1101|\theta)p(\theta) = \frac{1}{20}$ (actually $\int$)

Posterior: $p(\theta|1101) = \frac{p(1101|\theta)p(\theta)}{p(1101)} \propto \theta^3(1-\theta)$ (BAYES RULE!).

Maximum a Posterior (MAP) estimate: $\hat{\theta} = \arg\max_\theta p(\theta|1101) = \frac{3}{4}$

Predictive distribution: $p(1|1101) = \frac{p(11011)}{p(1101)} = \frac{2}{3}$

Expectation: $\mathbb{E}[f|...] = \sum_\theta f(\theta)p(\theta|...)$, e.g. $\mathbb{E}[\theta|1101] = \frac{2}{3}$

Variance: $\text{Var}(\theta) = \mathbb{E}[(\theta - \mathbb{E}\theta)^2|1101] = \frac{2}{63}$

Probability density: $p(\theta) = \frac{1}{\varepsilon}p([\theta, \theta+\varepsilon])$ for $\varepsilon \to 0$

Applications

- Bayesian dependency networks

- (Naive) Bayes classification

- Bayesian regression

- Model parameter estimation

- Probabilistic reasoning systems

- Pattern recognition

- ...

Outlook

- Likelihood functions from the exponential family
  (Gauss, Multinomial, Poisson, Dirichlet)

- Conjugate priors

- Approximations: Gaussian, Laplace, Gradient Descent, ...

- Monte Carlo simulations: Gibbs sampling, Metropolis-Hastings,

- Bayesian model comparison

- Consistency of Bayesian estimators

Summary

- The aim of probability theory is to describe uncertainty.

- Frequency interpretation of probabilities is simple,
  but is circular and limited to i.i.d.

- Distinguish between subjective and objective probabilities.

- Both kinds of probabilities satisfy Kolmogorov's axioms.

- Use Bayes rule for getting posterior from prior probabilities.

- But where do the priors come from?

- Occam's razor: Choose a simplicity biased prior.

- Still: What do objective probabilities really mean?

Exercise 1 [C25] Envelope Paradox

- I offer you two closed envelopes, one of them contains twice the amount of money than the other. You are allowed to pick one and open it. Now you have two options. Keep the money or decide for the other envelope (which could double or half your gain).

- Symmetry argument: It doesn't matter whether you switch, the expected gain is the same.

- Refutation: With probability $p = 1/2$, the other envelope contains twice/half the amount, i.e. if you switch your expected gain increases by a factor 1.25=1/2*2+1/2*1/2.

- Present a Bayesian solution.

Exercise 2 [C15-45] Confirmation Paradox

(i) $R \to B$ is confirmed by an R-instance with property B

(ii) $\neg B \to \neg R$ is confirmed by a $\neg B$-instance with property $\neg R$.

(iii) Since $R \to B$ and $\neg B \to \neg R$ are logically equivalent, $R \to B$ is also confirmed by a $\neg B$-instance with property $\neg R$.

Example: Hypothesis (o): All ravens are black (R=Raven, B=Black).

(i) observing a Black Raven confirms Hypothesis (o).

(iii) observing a White Sock also confirms that all Ravens are Black, since a White Sock is a non-Raven which is non-Black.

This conclusion sounds absurd.

Present a Bayesian solution.

More Exercises

3. [C15] Conditional probabilities: Show that $p(\cdot|A)$ (as a function of the first argument) also satisfies the Kolmogorov axioms, if $p(\cdot)$ does.

4. [C20] Prove Bayes rule (Theorem 3.6).

5. [C05] Assume the prevalence of a certain disease in the general population is 1%. Assume some test on a diseased/healthy person is positive/negative with 99% probability. If the test is positive, what is the chance of having the disease?

6. [C20] Compute $\int_0^1 \theta^n (1-\theta)^m \, d\theta$ (without looking it up)

Literature (from easy to hard)

[Jay03]   E. T. Jaynes. *Probability Theory: The Logic of Science*. Cambridge University Press, Cambridge, MA, 2003.

[Bis06]   C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.

[Pre02]   S. J. Press. *Subjective and Objective Bayesian Statistics: Principles, Models, and Applications*. Wiley, 2nd edition, 2002.

[GCSR95] A. Gelman, J. B. Carlin, H. S. Stern, and D. B. Rubin. *Bayesian Data Analysis.* Chapman & Hall / CRC, 1995.

[Fel68]   W. Feller. *An Introduction to Probability Theory and its Applications*. Wiley, New York, 3rd edition, 1968.

[Szé86]   G. J. Székely. *Paradoxes in Probability Theory and Mathematical Statistics*. Reidel, Dordrecht, 1986.

4   ALGORITHMIC PROBABILITY & UNIVERSAL INDUCTION

- The Universal a Priori Probability M

- Universal Sequence Prediction

- Universal Inductive Inference

- Martin-Löf Randomness

- Discussion

Algorithmic Probability & Universal Induction: Abstract

Solomonoff completed the Bayesian framework by providing a rigorous, unique, formal, and universal choice for the model class and the prior. I will discuss in breadth how and in which sense universal (non-i.i.d.) sequence prediction solves various (philosophical) problems of traditional Bayesian sequence prediction. I show that Solomonoff's model possesses many desirable properties: Strong total and weak instantaneous bounds , and in contrast to most classical continuous prior densities has no zero p(oste)rior problem, i.e. can confirm universal hypotheses, is reparametrization and regrouping invariant, and avoids the old-evidence and updating problem. It even performs well (actually better) in non-computable environments.

Problem Setup

- Since our primary purpose for doing induction is to forecast (time-series), we will concentrate on sequence prediction tasks.

- Classification is a special case of sequence prediction.
  (With some tricks the other direction is also true)

- This Course focusses on maximizing profit (minimizing loss).
  We're not (primarily) interested in finding a (true/predictive/causal) model.

- Separating noise from data is *not* necessary in this setting!

Philosophy & Notation

Occam's razor: take simplest hypothesis consistent with data.

Epicurus' principle of multiple explanations: Keep all theories consistent with the data.

$$\Downarrow \qquad \qquad \Downarrow$$

We now combine both principles:

Take all consistent explanations into account,
but weight the simpler ones higher.

Formalization with Turing machines and Kolmogorov complexity

Additional notation: We denote binary strings of length $\ell(x) = n$ by $x = x_{1:n} = x_1 x_2 ... x_n$ with $x_t \in \mathbb{B}$ and further abbreviate $x_{<n} := x_1 ... x_{n-1}$.

4.1 The Universal a Priori Probability M: Contents

- The Universal a Priori Probability M

- Relations between Complexities

- (Semi)Measures

- Sample Space / σ-Algebra / Cylinder Sets

- M is a SemiMeasure

- Properties of Enumerable Semimeasures

- Fundamental Universality Property of M

The Universal a Priori Probability M

Solomonoff defined the universal probability distribution $M(x)$ as the probability that the output of a universal monotone Turing machine starts with x when provided with fair coin flips on the input tape.

Definition 4.1 (Solomonoff distribution) Formally,

$$M(x) \; := \sum_{p \, : \, U(p)=x*} 2^{-\ell(p)}$$

The sum is over minimal programs p for which U outputs a string starting with x (see Definition 2.6).

Since the shortest programs p dominate the sum, $M(x)$ is roughly $2^{-Km(x)}$.   More precisely ...

Relations between Complexities

> **Theorem 4.2 (Relations between Complexities)**
>
> $KM := -\log M$, Km, and K are ordered in the following way:
>
> $$0 \;\leq\; K(x|\ell(x)) \;\overset{+}{<}\; KM(x) \;\leq\; Km(x) \;\leq\; K(x) \;\overset{+}{<}\; \ell(x) + 2\log\ell(x)$$

Proof sketch:

The second inequality follows from the fact that,

given n and Kraft's inequality $\sum_{x \in \mathcal{X}^n} M(x) \leq 1$,

there exists for $x \in \mathcal{X}^n$ a Shannon-Fano code of length $-\log M(x)$,

which is effective since M is enumerable.

Now use Theorem 2.17 conditioned to n.

The other inequalities are obvious from the definitions. ■

(Semi)Measures

Before we can discuss the stochastic properties of M we need the concept of (semi)measures for strings.

> **Definition 4.3 ((Semi)measures)** $\rho(x)$ denotes the probability that a binary sequence starts with string x. We call $\rho \geq 0$ a semimeasure if $\rho(\epsilon) \leq 1$ and $\rho(x) \geq \rho(x0) + \rho(x1)$, and a probability measure if equality holds.

The reason for calling ρ with the above property a probability measure is that it satisfies Kolmogorov's Axioms Definition 3.1 of probability in the following sense ...

Sample Space / Events / Cylinder Sets

- The The sample space is $\Omega = \mathbb{B}^\infty$ with elements $\omega = \omega_1 \omega_2 \omega_3 ... \in \mathbb{B}^\infty$ being infinite binary sequences.

- The set of events (the σ-algebra) is defined as the set generated from the cylinder sets $\Gamma_{x_{1:n}} := \{\omega : \omega_{1:n} = x_{1:n}\}$ by countable union and complement.

- A probability measure ρ is uniquely defined by giving its values $\rho(\Gamma_{x_{1:n}})$ on the cylinder sets, which we abbreviate by $\rho(x_{1:n})$.

- We will also call ρ a measure, or even more loosely a probability distribution.

M is a SemiMeasure

- The reason for extending the definition to semimeasures is that M itself is unfortunately **not** a probability measure.

- We have $M(x0) + M(x1) < M(x)$ because there are programs p, which output x, neither followed by 0 nor 1.

- They just stop after printing x -or- continue forever without any further output.

- Since $M(\epsilon) = 1$, M **is** at least a semimeasure.

Properties of (Semi)Measure ρ

- Properties of ρ: $\displaystyle\sum_{x_{1:n} \in \mathcal{X}^n} \rho(x_{1:n}) \overset{(\le)}{=} 1,$

$$\rho(x_t | x_{<t}) \quad := \quad \rho(x_{1:t}) / \rho(x_{<t}),$$
$$\rho(x_1 ... x_n) \quad = \quad \rho(x_1) \cdot \rho(x_2 | x_1) \cdot ... \cdot \rho(x_n | x_1 ... x_{n-1}).$$

- One can show that ρ is an enumerable semimeasure

$$\Longleftrightarrow \; \exists \; \text{mTM} \; T \; : \; \rho(x) = \sum_{p \,:\, T(p) = x*} 2^{-\ell(p)} \quad \text{and} \quad \ell(T) \overset{+}{=} K(\rho)$$

- Intuition: Fair coin flips are sufficient to create any probability distribution.

- Definition: $K(\rho) :=$ length of shortest self-delimiting code of a Turing machine computing function ρ in the sense of Def. 2.21.

Fundamental Universality Property of M

> **Theorem 4.4 (Universality of M)**
>
> M is a universal semimeasure in the sense that
> $M(x) \overset{\times}{>} 2^{-K(\rho)} \cdot \rho(x)$ for all enumerable semimeasures ρ.
> M is enumerable, but not estimable.

Up to a multiplicative constant, M assigns higher probability to all x
than any other computable probability distribution.

Proof sketch:

$$M(x) = \sum_{p\,:\,U(p)=x*} 2^{-\ell(p)} \geq \sum_{q\,:\,U(Tq)=x*} 2^{-\ell(Tq)} = 2^{-\ell(T)} \sum_{q\,:\,T(q)=x*} 2^{-\ell(q)} \overset{\times}{=} 2^{-K(\rho)}\rho(x)$$

4.2 Universal Sequence Prediction: Contents

- Solomonoff, Occam, Epicurus

- Prediction

- Simple Deterministic Bound

- Solomonoff's Major Result

- Implications of Solomonoff's Result

- Entropy Inequality

- Proof of the Entropy Bound

Solomonoff, Occam, Epicurus

- In which sense does M incorporate Occam's razor and Epicurus' principle of multiple explanations?

- From $M(x) \approx 2^{-K(x)}$ we see that M assigns high probability to simple strings (Occam).

- More useful is to think of x as being the observed history.

- We see from Definition 4.1 that every program p consistent with history x is allowed to contribute to M (Epicurus).

- On the other hand, shorter programs give significantly larger contribution (Occam).

Prediction

How does all this affect prediction?

If $M(x)$ correctly describes our (subjective) prior belief in x, then

$$M(y|x) := M(xy)/M(x)$$

must be our posterior belief in y.

From the symmetry of algorithmic information
$K(x,y) \stackrel{+}{=} K(y|x, K(x)) + K(x)$ (Theorem 2.15), and assuming
$K(x,y) \approx K(xy)$, and approximating $K(y|x, K(x)) \approx K(y|x)$,
$M(x) \approx 2^{-K(x)}$, and $M(xy) \approx 2^{-K(xy)}$ we get:
$$M(y|x) \approx 2^{-K(y|x)}$$

This tells us that M predicts y with high probability iff y has an easy
explanation, given x (Occam & Epicurus).

Simple Deterministic Bound

Sequence prediction algorithms try to predict the continuation $x_t \in \mathbb{B}$ of
a given sequence $x_1...x_{t-1}$. Simple deterministic bound:

$$\sum_{t=1}^{\infty} |1{-}M(x_t|x_{<t})| \overset{a}{\leq} -\sum_{t=1}^{\infty} \ln M(x_t|x_{<t}) \overset{b}{=} -\ln M(x_{1:\infty}) \overset{c}{\leq} Km(x_{1:\infty})\ln 2$$

(a) use $|1 - a| \leq -\ln a$ for $0 \leq a \leq 1$.

(b) exchange sum with logarithm and eliminate product by chain rule.

(c) used Theorem 4.2.

If $x_{1:\infty}$ is a computable sequence, then $Km(x_{1:\infty})$ is finite,
which implies $M(x_t|x_{<t}) \to 1$ $(\sum_{t=1}^{\infty} |1 - a_t| < \infty \Rightarrow a_t \to 1)$.

$\Rightarrow$ if the environment is a computable sequence (digits of π or e or ...),
after having seen the first few digits, M correctly predicts the next digit
with high probability, i.e. it recognizes the structure of the sequence.

Solomonoff's Major Result

Assume sequence $x_{1:\infty}$ is sampled from the unknown distribution μ, i.e. the true objective probability of $x_{1:n}$ is $\mu(x_{1:n})$.

The probability of x_t given $x_{<t}$ hence is $\mu(x_t|x_{<t}) = \mu(x_{1:t})/\mu(x_{<t})$.

Solomonoff's central result [Hut05] is that M converges to μ.

More precisely, he showed that

Theorem 4.5 (Predictive Convergence of M)

$$\sum_{t=1}^{\infty} \sum_{x_{<t} \in \mathbb{B}^{t-1}} \mu(x_{<t})\Big(M(0|x_{<t}) - \mu(0|x_{<t})\Big)^2 \overset{+}{<} \tfrac{1}{2}\ln 2 \cdot K(\mu) < \infty$$

Implications of Solomonoff's Result

- The infinite sum can only be finite if the difference $M(0|x_{<t}) - \mu(0|x_{<t})$ tends to zero for $t \to \infty$ with μ-probability 1.

- Convergence is rapid: The expected number of times t in which $|M(0|x_{<t}) - \mu(0|x_{<t})| > \varepsilon$ is finite and bounded by c/ε^2 and the probability that the number of ε-deviations exceeds $\frac{c}{\varepsilon^2\delta}$ is smaller than δ, where $c \stackrel{+}{=} \ln 2 \cdot K(\mu)$.

- No statement is possible for which t these deviations occur.

- This holds for **any** computable probability distribution μ.

- How does M know to which μ? The set of μ-random sequences differ for different μ.

- Intuition: Past data $x_{<t}$ are exploited to get a (with $t \to \infty$) improving estimate $M(x_t|x_{<t})$ of $\mu(x_t|x_{<t})$.

- Fazit: M is universal predictor. The only assumption made is that data are generated from a computable distribution.

Entropy Inequality

Proof of Solomonoff's bound: We need (proof as exercise)

Lemma 4.6 (Entropy Inequality)

$2(z-y)^2 \leq y \ln \frac{y}{z} + (1-y) \ln \frac{1-y}{1-z}$ for $0 < z < 1$ and $0 \leq y \leq 1$.

$$\leq y \ln \frac{y}{z} + (1-y) \ln \frac{1-y}{c-z} \quad \text{for} \quad 0 < z < c \leq 1 \quad \text{and} \quad 0 \leq y \leq 1.$$

The latter inequality holds, since the r.h.s. is decreasing in c. Inserting

$$0 \leq y := \mu(0|x_{<t}) = 1 - \mu(1|x_{<t}) \leq 1 \quad \text{and}$$

$$0 < z := M(0|x_{<t}) < c := M(0|x_{<t}) + M(1|x_{<t}) < 1 \quad \text{we get}$$

$$2(M(0|x_{<t}) - \mu(0|x_{<t}))^2 \leq \sum_{x_t \in \mathbb{B}} \mu(x_t|x_{<t}) \ln \frac{\mu(x_t|x_{<t})}{M(x_t|x_{<t})} =: d_t(x_{<t})$$

The r.h.s. is the relative entropy between μ and M.

Proof of the Entropy Bound

$$D_n(\mu\|M) \equiv \sum_{t=1}^{n} \sum_{x_{<t}} \mu(x_{<t})d_t(x_{<t}) \stackrel{(a)}{=} \sum_{t=1}^{n}\sum_{x_{1:t}} \mu(x_{1:t}) \ln \frac{\mu(x_t|x_{<t})}{M(x_t|x_{<t})} =$$

$$\stackrel{(b)}{=} \sum_{x_{1:n}} \mu(x_{1:n}) \ln \prod_{t=1}^{n} \frac{\mu(x_t|x_{<t})}{M(x_t|x_{<t})} \stackrel{(c)}{=} \sum_{x_{1:n}} \mu(x_{1:n}) \ln \frac{\mu(x_{1:n})}{M(x_{1:n})} \stackrel{(d)}{\underset{\times}{<}} K(\mu) \ln 2$$

(a) Insert def. of d_t and use product rule $\mu(x_{<t}) \cdot \mu(x_t|x_{<t}) = \mu(x_{1:t})$.

(b) $\sum_{x_{1:t}} \mu(x_{1:t}) = \sum_{x_{1:n}} \mu(x_{1:n})$ and argument of log is independent of $x_{t+1:n}$. The t sum can now be exchanged with the $x_{1:n}$ sum and transforms to a product inside the logarithm.

(c) Use chain rule again for μ and M.

(d) Use dominance $M(x) \stackrel{\times}{>} 2^{-K(\mu)}\mu(x)$.

Inserting d_t into D_n yields Solomonoff's Theorem 4.5. ■

4.3 Universal Inductive Inference: Contents

- Bayesian Sequence Prediction and Confirmation

- The Universal Prior

- The Problem of Zero Prior

- Reparametrization and Regrouping Invariance

- Universal Choice of Class $\mathcal{M}$

- The Problem of Old Evidence / New Theories

- Universal is Better than Continuous $\mathcal{M}$

- More Bounds / Critique / Problems

Bayesian Sequence Prediction and Confirmation

- Assumption: Sequence $\omega \in \mathcal{X}^\infty$ is sampled from the "true" probability measure μ, i.e. $\mu(x) := \mathbf{P}[x|\mu]$ is the μ-probability that ω starts with $x \in \mathcal{X}^n$.

- Model class: We assume that μ is unknown but known to belong to a countable class of environments=models=measures $\mathcal{M} = \{\nu_1, \nu_2, ...\}$.   [no i.i.d./ergodic/stationary assumption]

- Hypothesis class: $\{H_\nu : \nu \in \mathcal{M}\}$ forms a mutually exclusive and complete class of hypotheses.

- Prior: $w_\nu := \mathbf{P}[H_\nu]$ is our prior belief in H_ν

$\Rightarrow$ Evidence: $\xi(x) := \mathbf{P}[x] = \sum_{\nu \in \mathcal{M}} \mathbf{P}[x|H_\nu]\mathbf{P}[H_\nu] = \sum_\nu w_\nu \nu(x)$ must be our (prior) belief in x.

$\Rightarrow$ Posterior: $w_\nu(x) := \mathbf{P}[H_\nu|x] = \frac{\mathbf{P}[x|H_\nu]\mathbf{P}[H_\nu]}{\mathbf{P}[x]}$ is our posterior belief in ν (Bayes' rule).

The Universal Prior

- Quantify the complexity of an environment ν or hypothesis H_ν by its Kolmogorov complexity $K(\nu)$.

- Universal prior: $w_\nu = \boxed{w_\nu^U := 2^{-K(\nu)}}$ is a decreasing function in the model's complexity, and sums to (less than) one.

$\Rightarrow$ $D_n(\mu||\xi) \leq K(\mu) \ln 2$, i.e. the number of ε-deviations of ξ from μ is proportional to the complexity of the environment.

- No other semi-computable prior leads to better prediction (bounds).

- For continuous $\mathcal{M}$, we can assign a (proper) universal prior (not density) $w_\theta^U = 2^{-K(\theta)} > 0$ for computable θ, and 0 for uncomp. θ.

- This effectively reduces $\mathcal{M}$ to a discrete class $\{\nu_\theta \in \mathcal{M} : w_\theta^U > 0\}$ which is typically dense in $\mathcal{M}$.

- This prior has many advantages over the classical prior (densities).

The Problem of Zero Prior

= the problem of confirmation of universal hypotheses

Problem: If the prior is zero, then the posterior is necessarily also zero.

Example: Consider the hypothesis $H = H_1$ that all balls in some urn or all ravens are black $(=1)$ or that the sun rises every day.

Starting with a prior density as $w(\theta) = 1$ implies that prior $\mathbf{P}[H_\theta] = 0$ for all θ, hence posterior $P[H_\theta|1..1] = 0$, hence H never gets confirmed.

3 non-solutions: define $H = \{\omega = 1^\infty\}$ | use finite population | abandon strict/logical/all-quantified/universal hypotheses in favor of soft hyp.

Solution: Assign non-zero prior to $\theta = 1 \quad \Rightarrow \quad \mathbf{P}[H|1^n] \to 1$.

Generalization: Assign non-zero prior to all "special" θ, like $\frac{1}{2}$ and $\frac{1}{6}$, which may naturally appear in a hypothesis, like "is the coin or die fair".

Universal solution: Assign non-zero prior to all comp. θ, e.g. $w_\theta^U = 2^{-K(\theta)}$

Reparametrization Invariance

- New parametrization e.g. $\psi = \sqrt{\theta}$, then the ψ-density
  $\tilde{w}(\psi) = 2\sqrt{\theta}\,w(\theta)$ is no longer uniform if $w(\theta) = 1$ is uniform
  $\Rightarrow$ indifference principle is not reparametrization invariant (RIP).

- Jeffrey's and Bernardo's principle satisfy RIP w.r.t. differentiable
  bijective transformations $\psi = f^{-1}(\theta)$.

- The universal prior $w_\theta^U = 2^{-K(\theta)}$ also satisfies RIP w.r.t. simple
  computable f.   (within a multiplicative constant)

Regrouping Invariance

- Non-bijective transformations:
  E.g. grouping ball colors into categories black/non-black.

- No classical principle is regrouping invariant.

- Regrouping invariance is regarded as a very important and desirable property. [Walley's (1996) solution: sets of priors]

- The universal prior $w_\theta^U = 2^{-K(\theta)}$ is invariant under regrouping, and more generally under all simple [computable with complexity $O(1)$] even non-bijective transformations. (within a multiplicative constant)

- Note: Reparametrization and regrouping invariance hold for arbitrary classes and are not limited to the i.i.d. case.

Universal Choice of Class $\mathcal{M}$

- The larger $\mathcal{M}$ the less restrictive is the assumption $\mu \in \mathcal{M}$.

- The class $\mathcal{M}_U$ of all (semi)computable (semi)measures, although only countable, is pretty large, since it includes all valid physics theories. Further, ξ_U is itself semi-computable [ZL70].

- Solomonoff's universal prior $M(x) :=$ probability that the output of a universal TM U with random input starts with x.

- Formally: $\boxed{M(x) := \sum_{p \, : \, U(p) = x*} 2^{-\ell(p)}}$ where the sum is over all (minimal) programs p for which U outputs a string starting with x.

- M may be regarded as a $2^{-\ell(p)}$-weighted mixture over all deterministic environments ν_p. ($\nu_p(x) = 1$ if $U(p) = x*$ and 0 else)

- $M(x)$ coincides with $\xi_U(x)$ within an irrelevant multiplicative constant.

The Problem of Old Evidence / New Theories

- What if some evidence $E \,\widehat{=}\, x$ (e.g. Mercury's perihelion advance) is known well-before the correct hypothesis/theory/model $H \,\widehat{=}\, \mu$ (Einstein's general relativity theory) is found?

- How shall H be added to the Bayesian machinery a posteriori?

- What should the "prior" of H be?

- Should it be the belief in H in a hypothetical counterfactual world in which E is not known?

- Can old evidence E confirm H?

- After all, H could simply be constructed/biased/fitted towards "explaining" E.

Solution of the Old-Evidence Problem

- The universal class $\mathcal{M}_U$ and universal prior w_ν^U formally solves this problem.

- The universal prior of H is $2^{-K(H)}$ independent of $\mathcal{M}$ and of whether E is known or not.

- Updating $\mathcal{M}$ is unproblematic, and even not necessary when starting with $\mathcal{M}_U$, since it includes all hypothesis (including yet unknown or unnamed ones) a priori.

Universal is Better than Continuous $\mathcal{M}$

- Although $\nu_\theta()$ and w_θ are incomp. for cont. classes $\mathcal{M}$ for most θ, $\xi()$ is typically computable. (exactly as for Laplace or numerically)

$$\Rightarrow \boxed{D_n(\mu||M) \stackrel{+}{<} D_n(\mu||\xi) + K(\xi)\ln 2 \text{ for all } \mu}$$

- That is, M is superior to all computable mixture predictors ξ based on any (continuous or discrete) model class $\mathcal{M}$ and weight $w(\theta)$, save an additive constant $K(\xi)\ln 2 = O(1)$, even if environment μ is not computable.

- While $D_n(\mu||\xi) \sim \frac{d}{2}\ln n$ for all $\mu \in \mathcal{M}$, $D_n(\mu||M) \le K(\mu)\ln 2$ is even finite for computable μ.

Fazit: $\boxed{\text{Solomonoff prediction works also in non-computable environments}}$

Convergence and Bounds

- Total (loss) bounds: $\sum_{n=1}^{\infty} \mathbb{E}[h_n] \overset{+}{<} K(\mu)\ln 2$, where
  $h_t(\omega_{<t}) := \sum_{a \in \mathcal{X}} (\sqrt{\xi(a|\omega_{<t})} - \sqrt{\mu(a|\omega_{<t})})^2$.

- Instantaneous i.i.d. bounds: For i.i.d. $\mathcal{M}$ with continuous, discrete, and universal prior, respectively:
  $\mathbb{E}[h_n] \overset{\times}{<} \frac{1}{n}\ln w(\mu)^{-1}$ and $\mathbb{E}[h_n] \overset{\times}{<} \frac{1}{n}\ln w_\mu^{-1} = \frac{1}{n}K(\mu)\ln 2$.

- Bounds for computable environments: Rapidly $M(x_t|x_{<t}) \to 1$ on every computable sequence $x_{1:\infty}$ (whichsoever, e.g. 1^∞ or the digits of π or e), i.e. M quickly recognizes the structure of the sequence.

- Weak instantaneous bounds: valid for all n and $x_{1:n}$ and $\bar{x}_n \neq x_n$:
  $2^{-K(n)} \overset{\times}{<} M(\bar{x}_n|x_{<n}) \overset{\times}{<} 2^{2Km(x_{1:n})-K(n)}$

- Magic instance numbers: e.g. $M(0|1^n) \overset{\times}{=} 2^{-K(n)} \to 0$, but spikes up for simple n. M is cautious at magic instance numbers n.

- Future bounds / errors to come: If our past observations $\omega_{1:n}$ contain a lot of information about μ, we make few errors in future:
  $\sum_{t=n+1}^{\infty} \mathbb{E}[h_t|\omega_{1:n}] \overset{+}{<} [K(\mu|\omega_{1:n})+K(n)]\ln 2$

More Stuff / Critique / Problems

- Prior knowledge y can be incorporated by using "subjective" prior $w_{\nu|y}^U = 2^{-K(\nu|y)}$ or by prefixing observation x by y.

- Additive/multiplicative constant fudges and U-dependence is often (but not always) harmless.

- Incomputability: K and M can serve as "gold standards" which practitioners should aim at, but have to be (crudely) approximated in practice (MDL [Ris89], MML [Wal05], LZW [LZ76], CTW [WST95], NCD [CV05]).

4.4   MARTIN-LÖF RANDOMNESS: CONTENTS

- When is a Sequence Random? If it is incompressible!

- Motivation: For a fair coin 00000000 is as likely as 01100101, but we "feel" that 00000000 is less random than 01100101.

- Martin-Löf randomness captures the important concept of randomness of **individual** sequences.

- Martin-Löf random sequences pass all effective randomness tests.

When is a Sequence Random?

a) Is 0110010100101101101001111011 generated by a fair coin flip?

b) Is 1111111111111111111111111111 generated by a fair coin flip?

c) Is 1100100100001111110110101010 generated by a fair coin flip?

d) Is 0101010101010101010101010101 generated by a fair coin flip?

- Intuitively: (a) and (c) look random, but (b) and (d) look unlikely.

- Problem: Formally (a-d) have equal probability $(\frac{1}{2})^{length}$.

- Classical solution: Consider hypothesis class $H := \{\text{Bernoulli}(p) : p \in \Theta \subseteq [0,1]\}$ and determine p for which sequence has maximum likelihood $\Longrightarrow$ (a,c,d) are fair Bernoulli($\frac{1}{2}$) coins, (b) not.

- Problem: (d) is non-random, also (c) is binary expansion of π.

- Solution: Choose H larger, but how large? Overfitting? MDL?

- AIT Solution: A sequence is **random** iff it is **incompressible**.

Martin-Löf Random Sequences

Characterization equivalent to Martin-Löf's original definition:

Theorem 4.7 (Martin-Löf random sequences)

A sequence $x_{1:\infty}$ is μ-random (in the sense of Martin-Löf)
$\iff$ there is a constant c such that $M(x_{1:n}) \leq c \cdot \mu(x_{1:n})$ for all n.

Equivalent formulation for computable μ:

$$x_{1:\infty} \text{ is } \mu.\text{M.L.-random} \iff Km(x_{1:n}) \overset{+}{=} -\log\mu(x_{1:n}) \; \forall n, \quad (4.8)$$

Theorem 4.7 follows from (4.8) by exponentiation, "using $2^{-Km} \approx M$"
and noting that $M \overset{\times}{>} \mu$ follows from universality of M.

Properties of ML-Random Sequences

- Special case of μ being a fair coin, i.e. $\mu(x_{1:n}) = 2^{-n}$, then

  $x_{1:\infty}$ is random $\Longleftrightarrow Km(x_{1:n}) \overset{+}{=} n$, i.e. iff $x_{1:n}$ is incompressible.

- For general μ, $-\log\mu(x_{1:n})$ is the length of the Arithmetic code of $x_{1:n}$, hence $x_{1:\infty}$ is μ-random $\Longleftrightarrow$ the Arithmetic code is optimal.

- One can show that a μ-random sequence $x_{1:\infty}$ passes **all** thinkable effective randomness tests, e.g. the law of large numbers, the law of the iterated logarithm, etc.

- In particular, the set of all μ-random sequences has μ-measure 1.

4.5   DISCUSSION: CONTENTS

- Limitations of Other Approaches

- Summary

- Exercises

- Literature

Limitations of Other Approaches 1

- **Popper's philosophy of science** is seriously flawed:
  - falsificationism is too limited,
  - corroboration $\equiv$ confirmation or meaningless,
  - simple $\neq$ easy-to-refute.

- **No free lunch myth** relies on unrealistic uniform sampling. Universal sampling permits free lunch.

- **Frequentism:** definition circular, limited to i.i.d. data, reference class problem.

- **Statistical Learning Theory:** Predominantly considers i.i.d. data: Empirical Risk Minimization, PAC bounds, VC-dimension, Rademacher complexity, Cross-Validation.

Limitations of Other Approaches 2

- Subjective Bayes: No formal procedure/theory to get prior.

- Objective Bayes: Right in spirit, but limited to small classes unless community embraces information theory.

- MDL/MML: practical approximations of universal induction.

- Pluralism is globally inconsistent.

- Deductive Logic: Not strong enough to allow for induction.

- Non-monotonic reasoning, inductive logic, default reasoning do not properly take uncertainty into account.

- Carnap's confirmation theory: Only for exchangeable data. Cannot confirm universal hypotheses.

- Data paradigm: Data may be more important than algorithms for "simple" problems, but a "lookup-table" AGI will not work.

- Eliminative induction ignores uncertainty and information theory.

Summary

- Solomonoff's universal a priori probability $M(x)$

  $=$ Occam $+$ Epicurus $+$ Turing $+$ Bayes $+$ Kolmogorov

  $=$ output probability of a universal TM with random input

  $=$ enum. semimeasure that dominates all enum. semimeasures

  $\approx 2^{-\text{Kolmogorov complexity}(x)}$

- $M(x_t|x_{<t}) \to \mu(x_t|x_{<t})$ rapid w.p.1 $\forall$ computable μ.

- M solves/avoids/meliorates many if not all philosophical and statistical problems around induction.

- Fazit: M is universal predictor.

- Matin-Löf /Kolmogorov define randomness of individual sequences: A sequence is random iff it is incompressible.

Exercises

1. [C10] Show that Definition 4.1 of M and the one given above it are equivalent.

2. [C30] Prove that ρ is an enumerable semimeasure if and only if there exists a TM T with $\rho(x) = \sum_{p:T(p)=x*} 2^{-\ell(p)} \; \forall x$.

3. [C10] Prove the bounds of Theorem 4.2

4. [C15] Prove the entropy inequality Lemma 4.6.
   Hint: Differentiate w.r.t. z and consider $y < z$ and $y > z$ separately.

5. [C10] Prove the claim about (rapid) convergence after Theorem 4.5 (Hint: Markov-Inequality).

6. [C20] Prove the instantaneous bound $M(1|0^n) \stackrel{\times}{=} 2^{-K(n)}$.

Literature

[Sol64]   R. J. Solomonoff. *A formal theory of inductive inference: Parts 1 and 2.* Information and Control, 7:1–22 and 224–254, 1964.

[LV08]   M. Li and P. M. B. Vitányi. *An Introduction to Kolmogorov Complexity and its Applications*. Springer, Berlin, 3rd edition, 2008.

[Hut05]   M. Hutter. *Universal Artificial Intelligence: Sequential Decisions based on Algorithmic Probability*. Springer, Berlin, 2005. http://www.hutter1.net/ai/uaibook.htm

[Hut07]   M. Hutter. On universal prediction and Bayesian confirmation. *Theoretical Computer Science*, 384(1):33–48, 2007. http://arxiv.org/abs/0709.1516

[RH11]   S. Rathmanner and M. Hutter. A philosophical treatise of universal induction. *Entropy*, 16(6):1076–1136, 2011. http://dx.doi.org/10.3390/e13061076

5 MINIMUM DESCRIPTION LENGTH

- MDL as Approximation of Solomonoff's M

- The Minimum Description Length Principle

- Application: Sequence Prediction

- Application: Regression / Polynomial Fitting

- Summary

Minimum Description Length: Abstract

The Minimum Description/Message Length principle is one of the most important concepts in Machine Learning, and serves as a scientific guide, in general. The motivation is as follows: To make predictions involves finding regularities in past data, regularities in data allows for compression, hence short descriptions of data should help in making predictions. In this lecture series we approach MDL from a Bayesian perspective and relate it to a MAP (maximum a posteriori) model choice. The Bayesian prior is chosen in accordance with Occam and Epicurus and the posterior is approximated by the MAP solution. We reconsider (un)fair coin flips and compare the M(D)L to Bayes-Laplace's solution, and similarly for general sequence prediction tasks. Finally I present an application to regression / polynomial fitting.

From Compression to Prediction

The better you can compress, the better you can predict.

Being able to predict (the env.) well is key for being able to act well.

Simple Example: Consider "14159...[990 more digits]...01989".

- If it looks random to you, you can neither compress it

  nor can you predict the 1001st digit.

- If you realize that they are the first 1000 digits of π,

  you can compress the sequence and predict the next digit.

Practical Example: The quality of natural language models is typically judged by its perplexity, which is essentially a compression ratio.

Later: Sequential decision theory tells you how to exploit such models for optimal rational actions.

MDL as Approximation of Solomonoff's M

- Approximation of Solomonoff, since M incomputable:

- $M(x) \approx 2^{-Km(x)}$ (excellent approximation)

- $Km(x) \equiv Km_U(x) \approx Km_T(x)$
  (approximation quality depends on T and x)

- Predict y of highest $M(y|x)$ is approximately same as

- MDL: Predict y of smallest complexity $Km_T(xy)$.

- Examples for x: Daily weather or stock market data.

- Example for T: Lempel-Ziv decompressor.

- Prediction $\widehat{=}$ finding regularities $\widehat{=}$ compression $\widehat{=}$ MDL.

- Improved compressors lead to improved predictors.

Human Knowledge Compression Contest

- compression $=$ finding regularities $\Rightarrow$ prediction $\approx$ intelligence
  [hard file size numbers] [slippery concept]

- Many researchers analyze data and find compact models.

- Compressors beating the current compressors need to be smart(er).

- "universal" corpus of data $\Rightarrow$ "universally" smart compressors.

- Wikipedia seems a good snapshot of the Human World Knowledge.

- The ultimate compressor of Wikipedia will "understand"
  all human knowledge, i.e. be really smart.

- Contest: Compress Wikipedia
  better than the current record.

- Prize: 50'000 Euro $\times$ the relative
  improvement to previous record.

[http://prize.hutter1.net]

The Minimum Description Length Principle

Identification of probabilistic model "best" describing data:

Probabilistic model(=hypothesis) H_ν with $\nu \in \mathcal{M}$ and data D.

Most probable model is $\nu^{\mathrm{MDL}} = \arg\max_{\nu \in \mathcal{M}} p(H_\nu|D)$.

Bayes' rule: $p(H_\nu|D) = p(D|H_\nu) \cdot p(H_\nu)/p(D)$.

Occam's razor: $p(H_\nu) = 2^{-Kw(\nu)}$.

By definition: $p(D|H_\nu) = \nu(x), \quad D = x =$data-seq., $\quad p(D) =$const.

Take logarithm:

> **Definition 5.1 (MDL)** $\qquad \nu^{\mathrm{MDL}} = \arg\min_{\nu \in \mathcal{M}} \{K\nu(x) + Kw(\nu)\}$

$K\nu(x) := -\log \nu(x) =$ length of Shannon-Fano code of x given H_ν.

$Kw(\nu) =$ length of model H_ν.

Names: Two-part MDL or MAP or MML ($\exists$ "slight" differences)

Predict with Best Model

- Use best model from class of models $\mathcal{M}$ for prediction:

- Predict y with probability $\nu^{\mathsf{MDL}}(y|x) = \frac{\nu^{\mathsf{MDL}}(xy)}{\nu^{\mathsf{MDL}}(x)}$ (3 variants)

- $y^{\mathsf{MDL}} = \arg\max\limits_{y}\{\nu^{\mathsf{MDL}}(y|x)\}$ is most likely continuation of x

- Special case: $Kw(\nu) =$const.
  $\implies$ MDL $\rightsquigarrow$ ML:=Maximum likelihood principle.

- Example: $H_\theta =$Bernoulli(θ) with $\theta \in [0,1]$ and $Kw(\theta) :=$const. and
  $\nu(x_{1:n}) = \theta^{n_1}(1-\theta)^{n_0}$ with $n_1 = x_1 + ... + x_n = n - n_0$.
  $\Rightarrow \theta^{\mathsf{MDL}} = \arg\min\limits_{\theta}\{-\log\theta^{n_1}(1-\theta)^{n_0}+Kw(\theta)\} = \frac{n_1}{n} = \nu^{\mathsf{MDL}}(1|x)$
       $=$ ML frequency estimate.   (overconfident, e.g. $n_1 = 0$)

- Compare with Laplace' rule based on Bayes' rule: $\theta^{Laplace} = \frac{n_1+1}{n+2}$.

Application: Sequence Prediction

Instead of Bayes mixture $\xi(x) = \sum_\nu w_\nu \nu(x)$, consider MAP/MDL

$$\nu^{\mathrm{MDL}}(x) = \max\{w_\nu \nu(x) : \nu \in \mathcal{M}\} = \arg\min_{\nu \in \mathcal{M}}\{K\nu(x) + Kw(\nu)\}.$$

Theorem 5.2 (MDL bound)

$$\sum_{t=1}^{\infty} \mathbb{E}\left[\sum_{x_t}(\mu(x_t|x_{<t}) - \nu^{\mathrm{MDL}}(x_t|x_{<t}))^2\right] \leq 8w_\mu^{-1}$$

No log as for ξ $\qquad\qquad w_\mu \widehat{=} 2^{-K(\mu)}$ $\qquad\qquad$ Proof: [PH05]

$\Rightarrow$ MDL converges, but speed can be exp. worse than Bayes&Solomonoff

$\Rightarrow$ be careful (bound is tight).

For continuous smooth model class $\mathcal{M}$ and prior w_ν,

MDL is as good as Bayes.

Application: Regression / Polynomial Fitting

- Data $D = \{(x_1, y_1), ..., (x_n, y_n)\}$

- Fit polynomial $f_d(x) := a_0 + a_1 x + a_2 x^2 + ... + a_d x^d$ of degree d through points D

- Measure of error: $SQ(a_0...a_d) = \sum_{i=1}^{n} (y_i - f_d(x_i))^2$

- Given d, minimize $SQ(a_{0:d})$ w.r.t. parameters $a_0...a_d$.

- This classical approach does not tell us how to choose d? ($d \geq n - 1$ gives perfect fit)
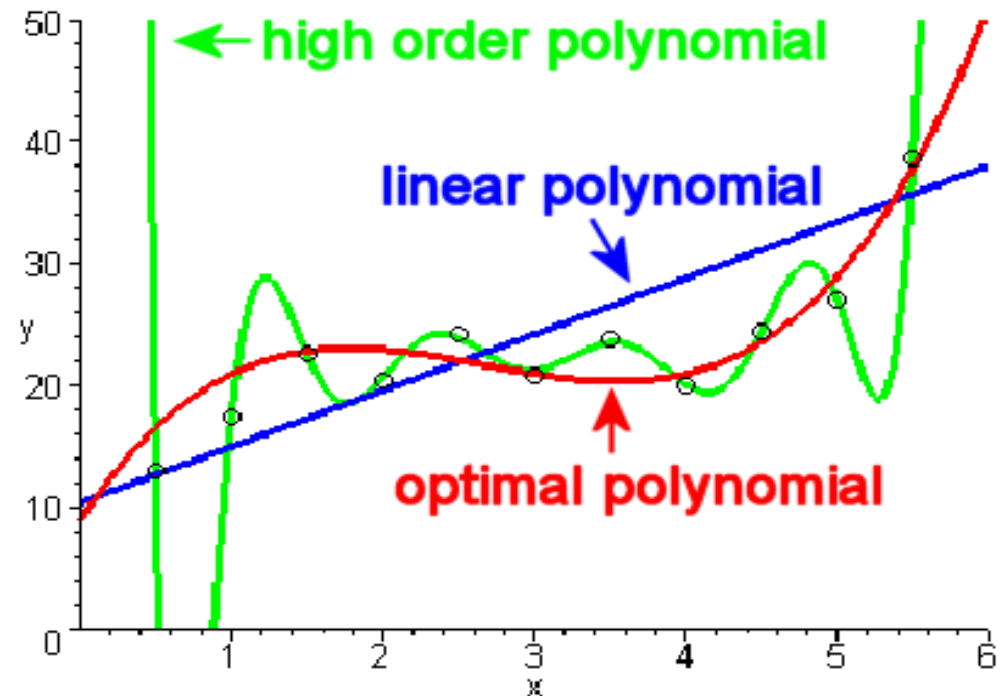
MDL Solution to Polynomial Fitting

Assume y given x is Gaussian with variance σ^2 and mean $f_d(x)$, i.e.

$$P((x,y)|f_d) := P(y|x,f_d) = \frac{1}{\sqrt{2\pi}\sigma}\exp\left(-\frac{(y-f_d(x))^2}{2\sigma^2}\right)$$

$$\implies \quad P(D|f_d) = \prod_{i=1}^{d} P((x_i,y_i)|f_d) = \frac{e^{-SQ(a_{0:d})/2\sigma^2}}{(2\pi\sigma^2)^{n/2}}$$

The larger the error SQ, the less likely the data.

Occam: $P(f_d) = 2^{-Kw(f_d)}$. Simple coding: $Kw(f_d) \approx (d+1)\cdot C$, where C is the description length=accuracy of each coefficient a_k in bits $\implies$

$$f^{\mathsf{MDL}} = \underset{f}{\operatorname{argmin}}\{-\log P(D|f) + Kw(f)\} = \underset{d,a_{0:d}}{\operatorname{argmin}}\left\{\frac{SQ(a_{0:d})}{2\sigma^2\ln 2} + (d+1)C\right\}$$

Fixed d $\quad\Rightarrow\quad a_{0:d}^{\mathsf{ML}} = \underset{a_{0:d}}{\arg\min} SQ(a_{0:d}) =$ classical solution

(by linear invariance of $\arg\min$)

MDL Polynomial Fitting: Determine Degree d

Determine d ($\min_f = \min_d \min_{f_d}$):

$$d = \arg\min_d\{\frac{1}{2\sigma^2 \ln 2} \underbrace{SQ(a_{0:d}^{\mathsf{ML}})}_{\text{least square fit}} + \underbrace{\frac{n}{2}\log(2\pi\sigma^2)}_{\text{"constant"}} + \underbrace{(d+1)C}_{\text{complexity penalty}} \}$$

Interpretation: Tradeoff between SQuare error and complexity penalty

Minimization w.r.t. σ leads to $n\sigma^2 = SQ(d) := SQ(a_{0:d}^{\mathsf{ML}})$, hence

$$d = \arg\min_d\{\tfrac{n}{2}\ln SQ(d) + (d+1)C\}.$$

With subtle arguments one can derive $C \stackrel{+}{=} \tfrac{1}{2}\ln n$.

Numerically find minimum of r.h.s.

Minimum Description Length: Summary

- Probability axioms give no guidance of how to choose the prior.

- Occam's razor is the only general (always applicable) principle for determining priors, especially in complex domains typical for AI.

- Prior $= 2^{-\text{descr.length}}$ — Universal prior $= 2^{-\text{Kolmogorov complexity}}$.

- Prediction $\widehat{=}$ finding regularities $\widehat{=}$ compression $\widehat{=}$ MDL.

- MDL principle: from a model class, a model is chosen that: minimizes the joint description length of the model and the data observed so far given the model.

- Similar to (Bayesian) Maximum a Posteriori (MAP) principle.

- MDL often as good as Bayes but not always.

Exercises

1. [C15] Determine an explicit expression for the $a_{0:d}^{\mathsf{ML}}$ estimates.

2. [C25] Use some artificial data by sampling from a polynomial with Gaussian or other noise. Use the MDL estimator to fit polynomials through the data points. Is the poly-degree correctly estimated?

3. [C20] Derive similar M(D)L estimators for other function classes like fourier decompositions. Use $C = \frac{1}{2}\ln n$ also for them.

4. [C25] Search for some real data. If other regression curves are available, compare them with your MDL results.

Literature

[Ris89]   J. J. Rissanen. *Stochastic Complexity in Statistical Inquiry*. World Scientific, Singapore, 1989.

[Wal05]   C. S. Wallace. *Statistical and Inductive Inference by Minimum Message Length*. Springer, Berlin, 2005.

[Gru05]   P. D. Grünwald. *Introduction and Tutorial.* In Advances in MDL, Chapters 1 and 2. MIT Press, 2005.
          http://www.cwi.nl/~pdg/ftp/mdlintro.pdf

[PH05]   J. Poland and M. Hutter. *Asymptotics of discrete MDL for online prediction.* IEEE Transactions on Information Theory, 51(11):3780–3795, 2005.

6 THE UNIVERSAL SIMILARITY METRIC

- Kolmogorov Complexity

- The Universal Similarity Metric

- Tree-Based Clustering

- Genomics & Phylogeny: Mammals, SARS Virus & Others

- Classification of Different File Types

- Language Tree (Re)construction

- Classify Music w.r.t. Composer

- Further Applications

- Summary

The Similarity Metric: Abstract

The MDL method has been studied from very concrete and highly tuned practical applications to general theoretical assertions. Sequence prediction is just one application of MDL. The MDL idea has also been used to define the so called information distance or universal similarity metric, measuring the similarity between two individual objects. I will present some very impressive recent clustering applications based on standard Lempel-Ziv or bzip2 compression, including a completely automatic reconstruction (a) of the evolutionary tree of 24 mammals based on complete mtDNA, and (b) of the classification tree of 52 languages based on the declaration of human rights and (c) others.

Based on [Cilibrasi&Vitanyi'05]

Kolmogorov Complexity

Question: When is object=string x similar to object=string y?

Universal solution: x similar $y \Leftrightarrow x$ can be easily (re)constructed from y
$\Leftrightarrow$ Kolmogorov complexity $K(x|y) := \min\{\ell(p) : U(p,y) = x\}$ is small

Examples:

1) x is very similar to itself ($K(x|x) \overset{+}{=} 0$)

2) A processed x is similar to x ($K(f(x)|x) \overset{+}{=} 0$ if $K(f) = O(1)$).
   e.g. doubling, reverting, inverting, encrypting, partially deleting x.

3) A random string is with high probability not similar to any other
   string ($K(\text{random}|y) = \text{length(random)}$).

The problem with $K(x|y)$ as similarity=distance measure is that it is
neither symmetric nor normalized nor computable.

The Universal Similarity Metric

- Symmetrization and normalization leads to a/the universal metric d:

$$0 \leq d(x,y) := \frac{\max\{K(x|y), K(y|x)\} - K(x|x)}{\max\{K(x), K(y)\}} \leq 1$$

- Every effective similarity between x and y is detected by d

- Use $K(x|y) \approx K(xy) - K(y)$ (coding T) and $K(x) \equiv K_U(x) \approx K_T(x)$
  $\implies$ computable approximation: Normalized compression distance:

$$d(x,y) \approx \frac{K_T(xy) - \min\{K_T(x), K_T(y)\}}{\max\{K_T(x), K_T(y)\}} \lesssim 1$$

- For T choose Lempel-Ziv or gzip or bzip(2) (de)compressor in the applications below.

- Theory: Lempel-Ziv compresses asymptotically better than any probabilistic finite state automaton predictor/compressor.

Tree-Based Clustering

- If many objects $x_1, ..., x_n$ need to be compared, determine the

$$\text{similarity matrix} \qquad M_{ij} = d(x_i, x_j) \quad \text{for} \quad 1 \leq i, j \leq n$$

- Now cluster similar objects.

- There are various clustering techniques.

- Tree-based clustering: Create a tree connecting similar objects,

- e.g. quartet method (for clustering)
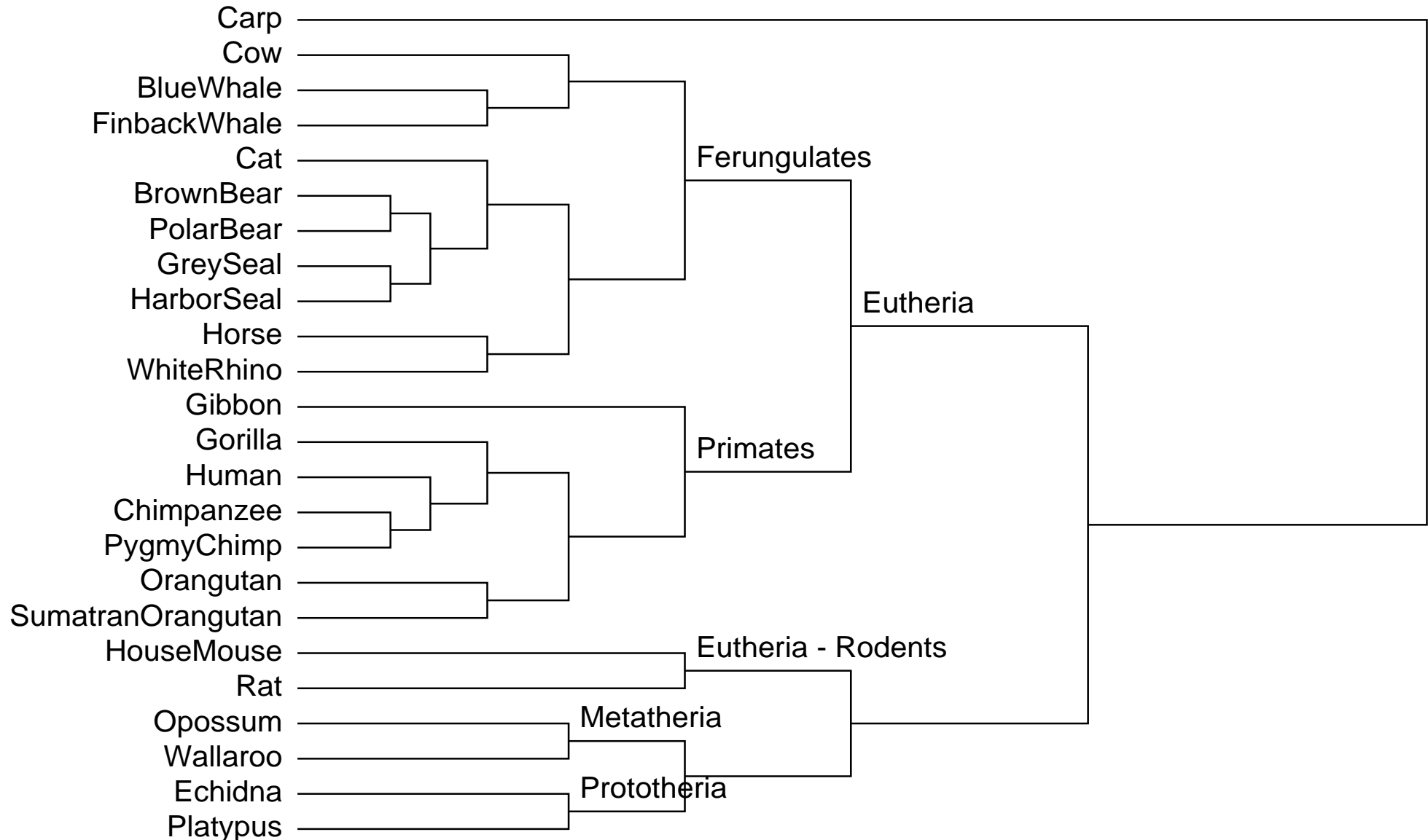
Genomics & Phylogeny: Mammals

Let $x_1, ..., x_n$ be mitochondrial genome sequences of different mammals:

Partial distance matrix M_{ij} using bzip2(?)

|	BrownBear	Carp	Cat	Chimpanzee	Cow	Echidna	FinWhale	Gibbon	Gorilla	HouseMouse	Human	...
BrownBear	0.002	0.943	0.887	0.935	0.906	0.944	0.915	0.939	0.940	0.934	0.930	...
Carp	0.943	0.006	0.946	0.954	0.947	0.955	0.952	0.951	0.957	0.956	0.946	...
Cat	0.887	0.946	0.003	0.926	0.897	0.942	0.905	0.928	0.931	0.919	0.922	...
Chimpanzee	0.935	0.954	0.926	0.006	0.926	0.948	0.926	0.849	0.731	0.943	0.667	...
Cow	0.906	0.947	0.897	0.926	0.006	0.936	0.885	0.931	0.927	0.925	0.920	...
Echidna	0.944	0.955	0.942	0.948	0.936	0.005	0.936	0.947	0.947	0.941	0.939	...
FinbackWhale	0.915	0.952	0.905	0.926	0.885	0.936	0.005	0.930	0.931	0.933	0.922	...
Gibbon	0.939	0.951	0.928	0.849	0.931	0.947	0.930	0.005	0.859	0.948	0.844	...
Gorilla	0.940	0.957	0.931	0.731	0.927	0.947	0.931	0.859	0.006	0.944	0.737	...
HouseMouse	0.934	0.956	0.919	0.943	0.925	0.941	0.933	0.948	0.944	0.006	0.932	...
Human	0.930	0.946	0.922	0.667	0.920	0.939	0.922	0.844	0.737	0.932	0.005	...
...	...	...	...	...	...	...	...	...	...	...	...	...

Genomics & Phylogeny: Mammals

Evolutionary tree built from complete mammalian mtDNA of 24 species:

Genomics & Phylogeny: SARS Virus and Others

- Clustering of SARS virus in relation to potential similar virii based on complete sequenced genome(s) using bzip2:

- The relations are very similar to the definitive tree based on medical-macrobio-genomics analysis from biologists.

Genomics & Phylogeny: SARS Virus and Others
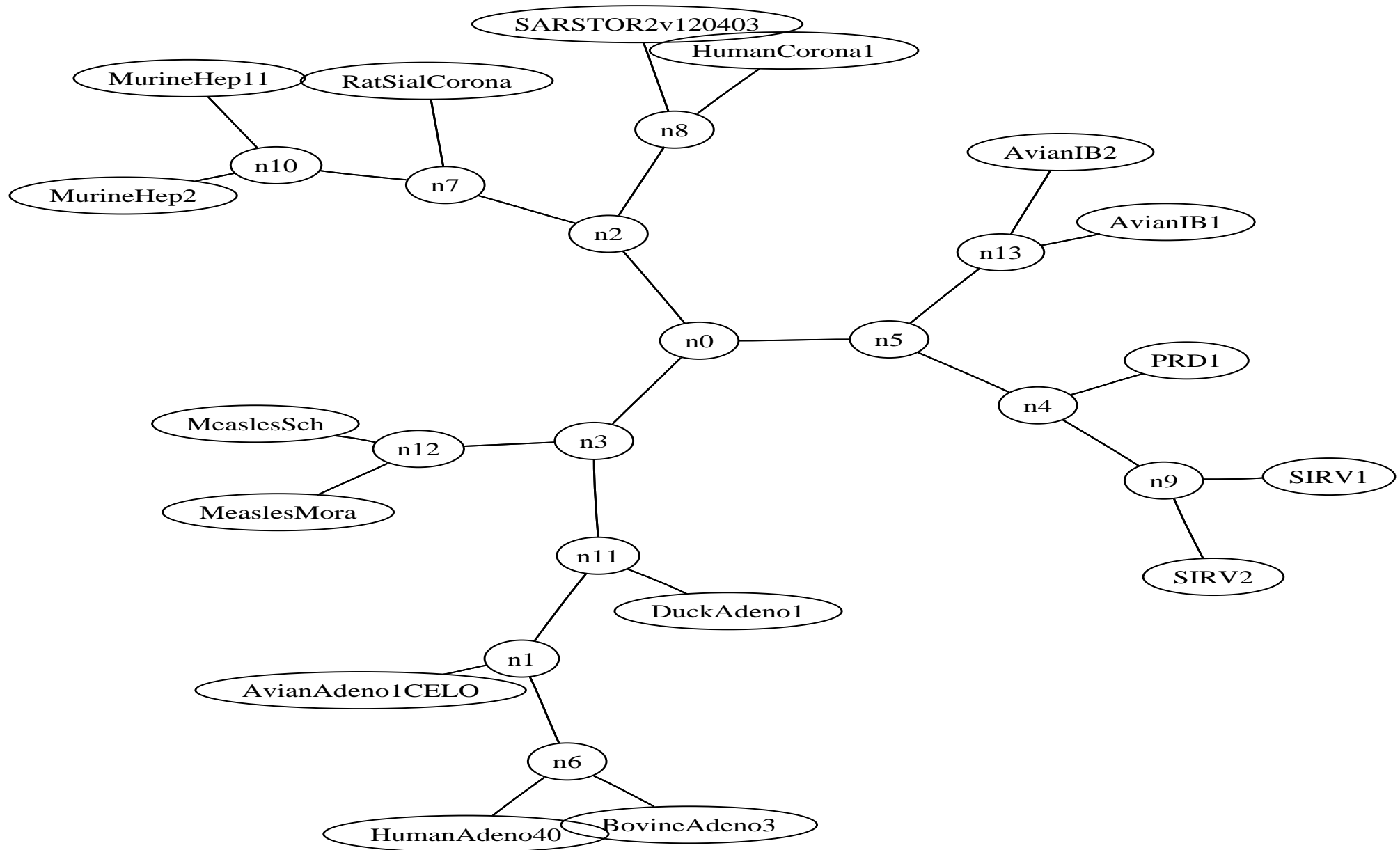
Classification of Different File Types

Classification of files based on markedly different file types using bzip2

- Four mitochondrial gene sequences

- Four excerpts from the novel "The Zeppelin's Passenger"

- Four MIDI files without further processing

- Two Linux x86 ELF executables (the cp and rm commands)

- Two compiled Java class files

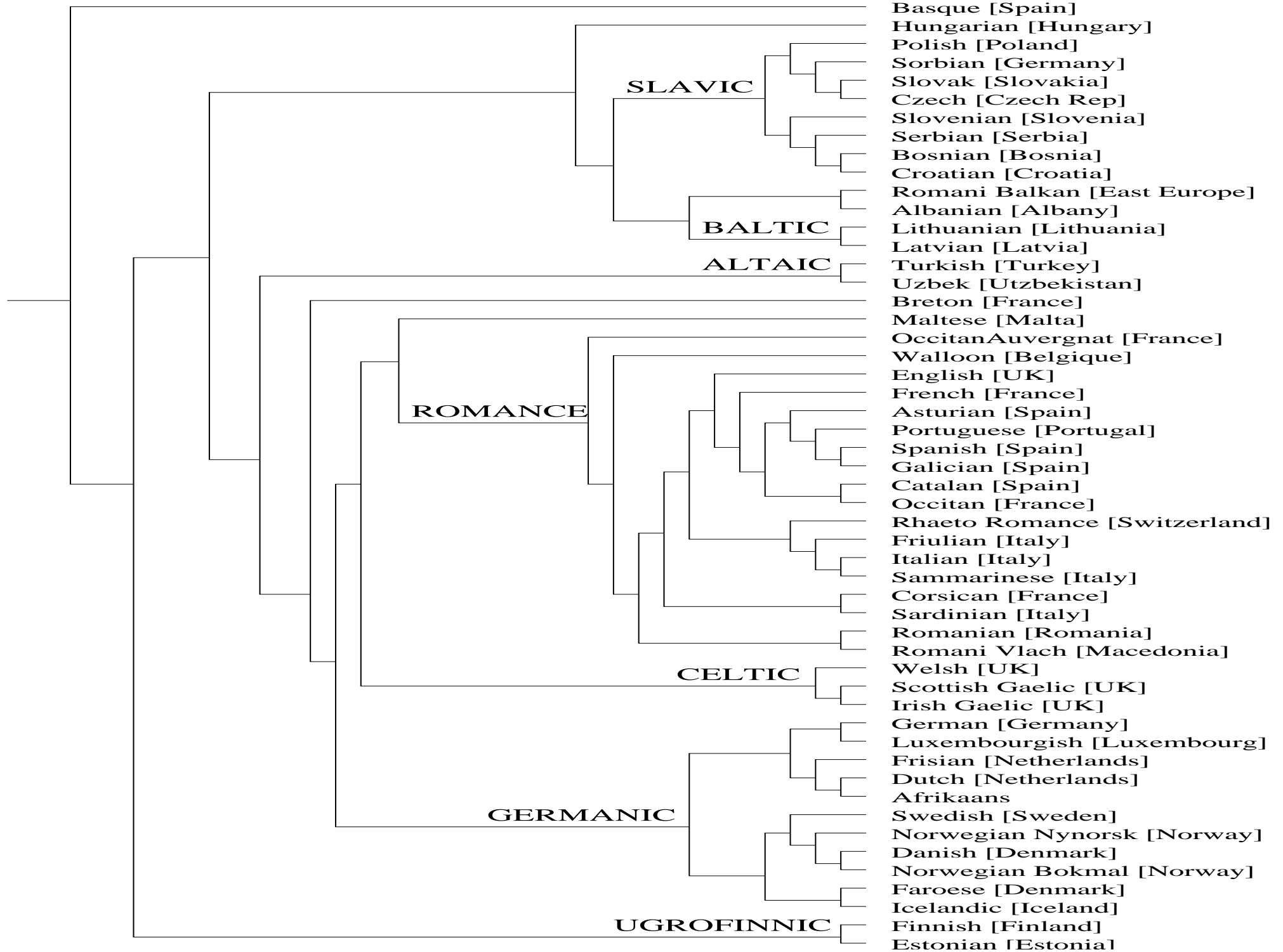No features of any specific domain of application are used!

Classification of Different File Types
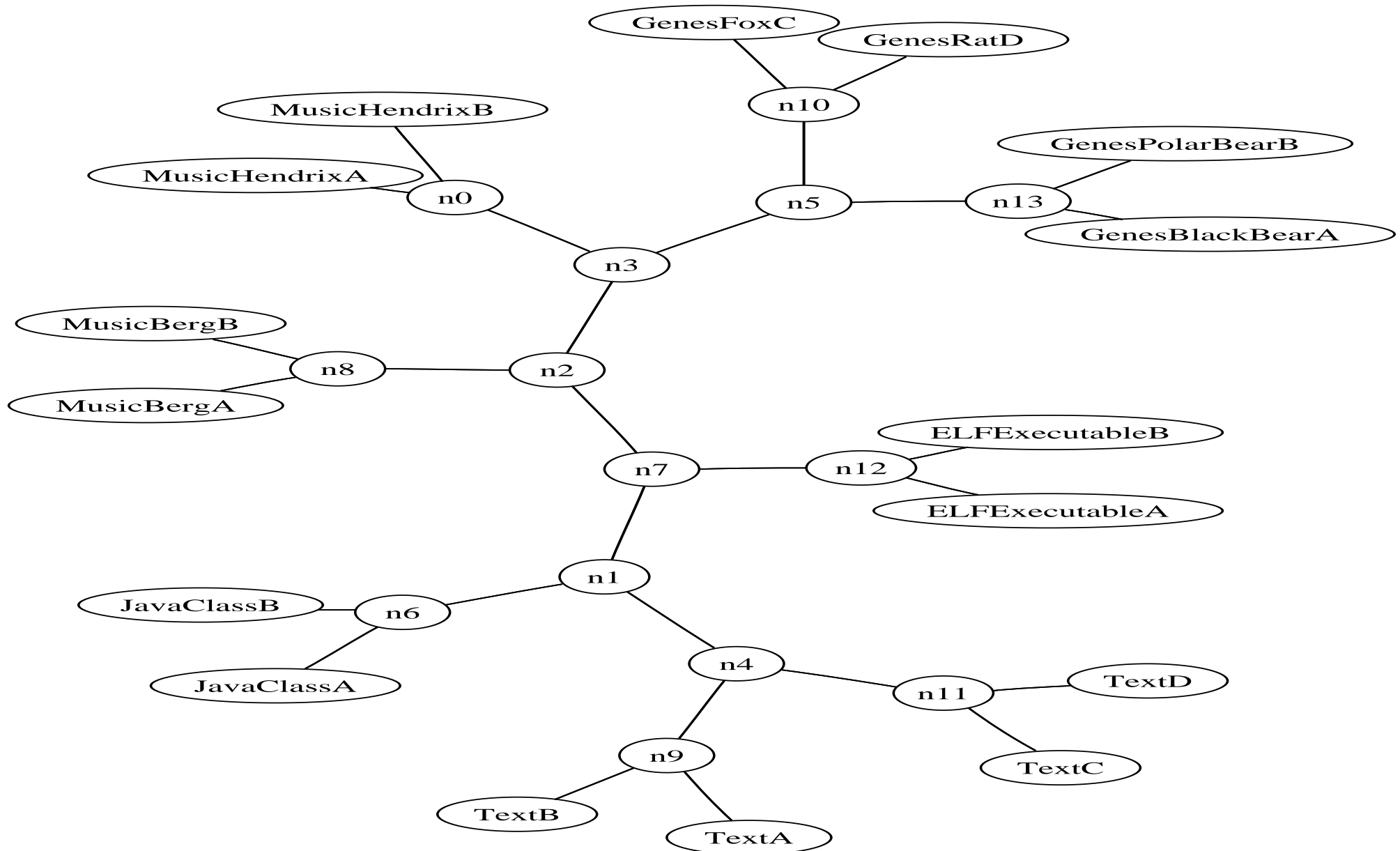
Perfect classification!

Language Tree (Re)construction

- Let $x_1, ..., x_n$ be the "The Universal Declaration of Human Rights" in various languages $1, ..., n$.

- Distance matrix M_{ij} based on gzip. Language tree constructed from M_{ij} by the Fitch-Margoliash method.   [Li 03]

- All main linguistic groups can be recognized (next slide)

Basque [Spain]
Hungarian [Hungary]
Polish [Poland]
Sorbian [Germany]
Slovak [Slovakia]
Czech [Czech Rep]
Slovenian [Slovenia]
Serbian [Serbia]
Bosnian [Bosnia]
Croatian [Croatia]
Romani Balkan [East Europe]
Albanian [Albany]
Lithuanian [Lithuania]
Latvian [Latvia]
Turkish [Turkey]
Uzbek [Utzbekistan]
Breton [France]
Maltese [Malta]
OccitanAuvergnat [France]
Walloon [Belgique]
English [UK]
French [France]
Asturian [Spain]
Portuguese [Portugal]
Spanish [Spain]
Galician [Spain]
Catalan [Spain]
Occitan [France]
Rhaeto Romance [Switzerland]
Friulian [Italy]
Italian [Italy]
Sammarinese [Italy]
Corsican [France]
Sardinian [Italy]
Romanian [Romania]
Romani Vlach [Macedonia]
Welsh [UK]
Scottish Gaelic [UK]
Irish Gaelic [UK]
German [Germany]
Luxembourgish [Luxembourg]
Frisian [Netherlands]
Dutch [Netherlands]
Afrikaans
Swedish [Sweden]
Norwegian Nynorsk [Norway]
Danish [Denmark]
Norwegian Bokmal [Norway]
Faroese [Denmark]
Icelandic [Iceland]
Finnish [Finland]
Estonian [Estonia]

SLAVIC

BALTIC

ALTAIC

ROMANCE

CELTIC

GERMANIC

UGROFINNIC

Classify Music w.r.t. Composer

Let $m_1, ..., m_n$ be pieces of music in MIDI format.

Preprocessing the MIDI files:

- Delete identifying information (composer, title, ...), instrument indicators, MIDI control signals, tempo variations, ...

- Keep only note-on and note-off information.

- A note, $k \in \mathbb{Z}$ half-tones above the average note is coded as a signed byte with value k.

- The whole piece is quantized in 0.05 second intervals.

- Tracks are sorted according to decreasing average volume, and then output in succession.

Processed files $x_1, ..., x_n$ still sounded like the original.

Classify Music w.r.t. Composer

12 pieces of music: $4\times$Bach $+$ $4\times$Chopin $+$ $4\times$Debussy. Class. by bzip2
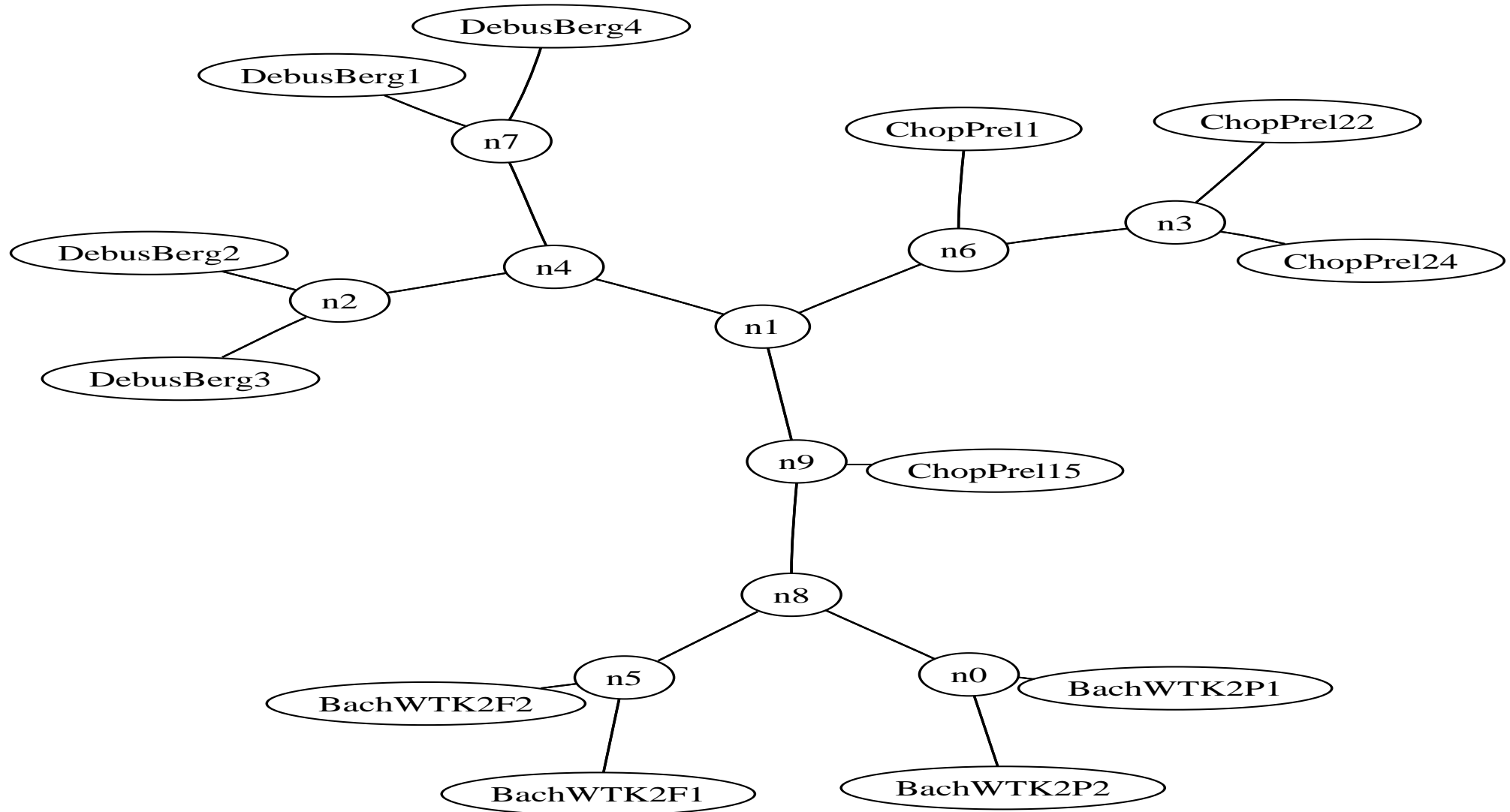
Perfect grouping of processed MIDI files w.r.t. composers.

Further Applications

- Classification of Fungi

- Optical character recognition

- Classification of Galaxies

- Clustering of novels w.r.t. authors

- Larger data sets

See [Cilibrasi&Vitanyi'05]

The Clustering Method: Summary

- based on the universal similarity metric,

- based on Kolmogorov complexity,

- approximated by bzip2,

- with the similarity matrix represented by tree,

- approximated by the quartet method

- **leads to excellent classification in many domains.**

Exercises

1. [C20] Prove that $d(x,y) := \frac{\max\{K(x|y), K(y|x)\} - K(x|x)}{\max\{K(x), K(y)\}}$ is a metric.

2. [C25] Reproduce the phylogenetic tree of mammals and the language tree using the CompLearn Toolkit available from http://www.complearn.org/.

Literature

[Ben98] C. H. Bennett et al. *Information distance.* IEEE Transactions on Information Theory, 44(4):1407–1423, 1998.

[Li 04] M. Li et al. *The similarity metric.* IEEE Transactions on Information Theory, 50(12):3250–3264, 2004.

[CVW04] R. Cilibrasi, P. M. B. Vitányi, and R. de Wolf. *Algorithmic clustering of music based on string compression.* Computer Music Journal, 28(4):49–67, 2004. http://arXiv.org/abs/cs/0303025.

[CV05] R. Cilibrasi and P. M. B. Vitányi. *Clustering by compression.* IEEE Trans. Information Theory, 51(4):1523–1545, 2005.

[CV06] R. Cilibrasi and P. M. B. Vitányi. *Similarity of objects and the meaning of words.* In Proc. 3rd Annual Conferene on Theory and Applications of Models of Computation (TAMC'06), LNCS. Springer, 2006.

7 BAYESIAN SEQUENCE PREDICTION

- The Bayes-Mixture Distribution

- Relative Entropy and Bound

- Predictive Convergence

- Sequential Decisions and Loss Bounds

- Generalization: Continuous Probability Classes

- Summary

Bayesian Sequence Prediction: Abstract

We define the Bayes mixture distribution and show that the posterior converges rapidly to the true posterior by exploiting some bounds on the relative entropy. Finally we show that the mixture predictor is also optimal in a decision-theoretic sense w.r.t. any bounded loss function.

Notation: Strings & Probabilities

Strings: $x = x_{1:n} := x_1 x_2 ... x_n$ with $x_t \in \mathcal{X}$ and $x_{<n} := x_1 ... x_{n-1}$.

Probabilities: $\rho(x_1 ... x_n)$ is the probability that an (infinite) sequence starts with $x_1 ... x_n$.

Conditional probability:

$$\rho_n := \rho(x_n | x_{<n}) = \rho(x_{1:n}) / \rho(x_{<n}),$$

$$\rho(x_1 ... x_n) = \rho(x_1) \cdot \rho(x_2 | x_1) \cdot ... \cdot \rho(x_n | x_1 ... x_{n-1}).$$

True data generating distribution: μ

The Bayes-Mixture Distribution ξ

- Assumption: The true (objective) environment μ is unknown.

- Bayesian approach: Replace true probability distribution μ by a Bayes-mixture ξ.

- Assumption: We know that the true environment μ is contained in some known countable (in)finite set $\mathcal{M}$ of environments.

Definition 7.1 (Bayes-mixture ξ)

$$\xi(x_{1:m}) := \sum_{\nu \in \mathcal{M}} w_\nu \nu(x_{1:m}) \quad \text{with} \quad \sum_{\nu \in \mathcal{M}} w_\nu = 1, \quad w_\nu > 0 \; \forall \nu$$

- The weights w_ν may be interpreted as the prior degree of belief that the true environment is ν, or $k^\nu = \ln w_\nu^{-1}$ as a complexity penalty (prefix code length) of environment ν.

- Then $\xi(x_{1:m})$ could be interpreted as the prior subjective belief probability in observing $x_{1:m}$.

A Universal Choice of ξ and $\mathcal{M}$

- We have to assume the existence of some structure on the environment to avoid the No-Free-Lunch Theorems [Wolpert 96].

- We can only unravel effective structures which are describable by (semi)computable probability distributions.

- So we may include *all* (semi)computable (semi)distributions in $\mathcal{M}$.

- Occam's razor and Epicurus' principle of multiple explanations tell us to assign high prior belief to simple environments.

- Using Kolmogorov's universal complexity measure $K(\nu)$ for environments ν one should set $w_\nu = 2^{-K(\nu)}$, where $K(\nu)$ is the length of the shortest program on a universal TM computing ν.

- The resulting mixture ξ is Solomonoff's (1964) universal prior.

- In the following we consider generic $\mathcal{M}$ and w_ν.

Relative Entropy

Relative entropy: $D(\mathbf{p}||\mathbf{q}) := \sum_i p_i \ln \frac{p_i}{q_i}$

Properties: $D(\mathbf{p}||\mathbf{q}) \geq 0$ and $D(\mathbf{p}||\mathbf{q}) = 0 \Leftrightarrow \mathbf{p} = \mathbf{q}$

Instantaneous relative entropy: $d_t(x_{<t}) := \sum_{x_t \in \mathcal{X}} \mu(x_t|x_{<t}) \ln \frac{\mu(x_t|x_{<t})}{\xi(x_t|x_{<t})}$

Theorem 7.2 (Total relative entropy) $D_n := \sum_{t=1}^{n} \mathbb{E}[d_t] \leq \ln w_\mu^{-1}$

$\mathbb{E}[f] =$ Expectation of f w.r.t. the *true* distribution μ, e.g.

If $f : \mathcal{X}^n \to \mathbb{R}$, then $\mathbb{E}[f] := \sum_{x_{1:n}} \mu(x_{1:n}) f(x_{1:n})$.

Proof based on dominance or universality: $\xi(x) \geq w_\mu \mu(x)$.

Proof of the Entropy Bound

$$D_n \equiv \sum_{t=1}^{n} \sum_{x_{<t}} \mu(x_{<t}) \cdot d_t(x_{<t}) \stackrel{(a)}{=} \sum_{t=1}^{n} \sum_{x_{1:t}} \mu(x_{1:t}) \ln \frac{\mu(x_t|x_{<t})}{\xi(x_t|x_{<t})} =$$

$$\stackrel{(b)}{=} \sum_{x_{1:n}} \mu(x_{1:n}) \ln \prod_{t=1}^{n} \frac{\mu(x_t|x_{<t})}{\xi(x_t|x_{<t})} \stackrel{(c)}{=} \sum_{x_{1:n}} \mu(x_{1:n}) \ln \frac{\mu(x_{1:n})}{\xi(x_{1:n})} \stackrel{(d)}{\leq} \ln w_\mu^{-1}$$

(a) Insert def. of d_t and used chain rule $\mu(x_{<t}) \cdot \mu(x_t|x_{<t}) = \mu(x_{1:t})$.

(b) $\sum_{x_{1:t}} \mu(x_{1:t}) = \sum_{x_{1:n}} \mu(x_{1:n})$ and argument of log is independent of $x_{t+1:n}$. The t sum can now be exchanged with the $x_{1:n}$ sum and transforms to a product inside the logarithm.

(c) Use chain rule again for μ and ξ.

(d) Use dominance $\xi(x) \geq w_\mu \mu(x)$.

Predictive Convergence

Theorem 7.3 (Predictive convergence)
$$\xi(x_t|x_{<t}) \to \mu(x_t|x_{<t}) \text{ rapid w.p.1 for } t \to \infty$$

Proof: $D_\infty \equiv \sum_{t=1}^{\infty} \mathbb{E}[d_t] \leq \ln w_\mu^{-1}$ and $d_t \geq 0$

$$\Longrightarrow \quad d_t \overset{t\to\infty}{\longrightarrow} 0 \quad \Longleftrightarrow \quad \xi_t \to \mu_t.$$

Fazit: ξ is excellent universal predictor if unknown μ belongs to $\mathcal{M}$.

How to choose $\mathcal{M}$ and w_μ? Both as large as possible?! More later.

Sequential Decisions

A prediction is very often the basis for some decision. The decision results in an action, which itself leads to some reward or loss.

Let $\mathsf{Loss}(x_t, y_t) \in [0, 1]$ be the received loss when taking action $y_t \in \mathcal{Y}$ and $x_t \in \mathcal{X}$ is the tth symbol of the sequence.

For instance, decision $\mathcal{Y} = \{\text{umbrella, sunglasses}\}$ based on weather forecasts $\mathcal{X} = \{\text{sunny, rainy}\}$.

Loss	sunny	rainy
umbrella	0.1	0.3
sunglasses	0.0	1.0

The goal is to minimize the μ-expected loss. More generally we define the Λ_ρ prediction scheme, which minimizes the ρ-expected loss:

$$y_t^{\Lambda_\rho} := \arg\min_{y_t \in \mathcal{Y}} \sum_{x_t} \rho(x_t | x_{<t}) \mathsf{Loss}(x_t, y_t)$$

Loss Bounds

- Definition: μ-expected loss when Λ_ρ predicts the tth symbol:
$$\mathsf{Loss}_t(\Lambda_\rho)(x_{<t}) := \sum_{x_t} \mu(x_t|x_{<t})\mathsf{Loss}(x_t, y_t^{\Lambda_\rho})$$

- $\mathsf{Loss}_t(\Lambda_{\mu/\xi})$ made by the informed/universal scheme $\Lambda_{\mu/\xi}$.
$\mathsf{Loss}_t(\Lambda_\mu) \le \mathsf{Loss}_t(\Lambda) \; \forall t, \Lambda$.

- Theorem: $0 \le \mathsf{Loss}_t(\Lambda_\xi) - \mathsf{Loss}_t(\Lambda_\mu) \le \sum_{x_t}|\xi_t - \mu_t| \le \sqrt{2d_t} \xrightarrow{w.p.1} 0$

- Total $\mathsf{Loss}_{1:n}(\Lambda_\rho) := \sum_{t=1}^n \mathbb{E}[\mathsf{Loss}_t(\Lambda_\rho)]$.

- Theorem: $\sqrt{\mathsf{Loss}_{1:n}(\Lambda_\xi)} - \sqrt{\mathsf{Loss}_{1:n}(\Lambda_\mu)} \le \sqrt{2D_n} \le \sqrt{2\ln w_\mu^{-1}}$

- Corollary: If $\mathsf{Loss}_{1:\infty}(\Lambda_\mu)$ is finite, then $\mathsf{Loss}_{1:\infty}(\Lambda_\xi)$ is finite, and $\mathsf{Loss}_{1:n}(\Lambda_\xi)/\mathsf{Loss}_{1:\infty}(\Lambda_\mu) \to 1$ if $\mathsf{Loss}_{1:\infty}(\Lambda_\mu) \to \infty$.

- Remark: Holds for any loss function $\in [0, 1]$ with no assumptions (like i.i.d., Markovian, stationary, ergodic, ...) on $\mu \in \mathcal{M}$.

Proof of Instantaneous Loss Bounds

Abbreviations: $\mathcal{X} = \{1, ..., N\}, \quad N = |\mathcal{X}|, \quad i = x_t, \quad y_i = \mu(x_t|x_{<t})$,

$z_i = \xi(x_t|x_{<t}), \quad m = y_t^{\Lambda_\mu}, \quad s = y_t^{\Lambda_\xi}, \quad \ell_{xy} = \mathsf{Loss}(x, y)$.

This and definition of $y_t^{\Lambda_\mu}$ and $y_t^{\Lambda_\xi}$ and $\sum_i z_i \ell_{is} \leq \sum_i z_i \ell_{ij} \; \forall j$ implies

$$\mathsf{Loss}_t(\Lambda_\xi) - \mathsf{Loss}_t(\Lambda_\mu) \equiv \sum_i y_i \ell_{is} - \sum_i y_i \ell_{im} \overset{(a)}{\leq} \sum_i (y_i - z_i)(\ell_{is} - \ell_{im})$$

$$\leq \sum_i |y_i - z_i| \cdot |\ell_{is} - \ell_{im}| \overset{(b)}{\leq} \sum_i |y_i - z_i| \overset{(c)}{\leq} \sqrt{\sum_i y_i \ln \frac{y_i}{z_i}} \equiv \sqrt{2 d_t(x_{<t})}$$

(a) We added $\sum_i z_i (\ell_{im} - \ell_{is}) \geq 0$.

(b) $|\ell_{is} - \ell_{im}| \leq 1$ since $\ell \in [0, 1]$.

(c) Pinsker's inequality (elementary, but not trivial)

Optimality of the Universal Predictor

- There are $\mathcal{M}$ and $\mu \in \mathcal{M}$ and weights w_μ for which the loss bounds are tight.

- The universal prior ξ is pareto-optimal, in the sense that there is no ρ with $\mathcal{F}(\nu, \rho) \leq \mathcal{F}(\nu, \xi)$ for all $\nu \in \mathcal{M}$ and strict inequality for at least one ν, where $\mathcal{F}$ is the instantaneous or total squared distance s_t, S_n, or entropy distance d_t, D_n, or general Loss_t, $\text{Loss}_{1:n}$.

- ξ is balanced pareto-optimal in the sense that by accepting a slight performance decrease in some environments one can only achieve a slight performance increase in other environments.

- Within the set of enumerable weight functions with short program, the universal weights $w_\nu = 2^{-K(\nu)}$ lead to the smallest performance bounds within an additive (to $\ln w_\mu^{-1}$) constant in all enumerable environments.

Continuous Probability Classes $\mathcal{M}$

In statistical parameter estimation one often has a continuous

hypothesis class (e.g. a Bernoulli(θ) process with unknown $\theta \in [0,1]$).

$$\mathcal{M} := \{\mu_\theta : \theta \in I\!\!R^d\}, \quad \xi(x_{1:n}) := \int_{I\!\!R^d} d\theta \, w(\theta) \, \mu_\theta(x_{1:n}), \quad \int_{I\!\!R^d} d\theta \, w(\theta) = 1$$

We only used $\xi(x_{1:n}) \geq w_\mu \cdot \mu(x_{1:n})$

which was obtained by dropping the sum over μ.

Here, restrict integral over $I\!\!R^d$ to a small vicinity N_δ of θ.

For sufficiently smooth μ_θ and $w(\theta)$ we expect

$$\xi(x_{1:n}) \gtrsim |N_{\delta_n}| \cdot w(\theta) \cdot \mu_\theta(x_{1:n}) \implies D_n \lesssim \ln w_\mu^{-1} + \ln |N_{\delta_n}|^{-1}$$

Continuous Probability Classes $\mathcal{M}$

Average Fisher information $\bar{\jmath}_n$ measures curvature (parametric complexity) of $\ln \mu_\theta$.

$$\bar{\jmath}_n := \frac{1}{n} \sum_{x_{1:n}} \mu(x_{1:n}) \nabla_\theta \ln \mu_\theta(x_{1:n}) \nabla_\theta^T \ln \mu_\theta(x_{1:n})_{|\theta=\theta_0}$$

Under weak regularity conditions on $\bar{\jmath}_n$ on can prove:

> **Theorem 7.4 (Continuous entropy bound)**
> $$D_n \;\leq\; \ln w_\mu^{-1} + \tfrac{d}{2} \ln \tfrac{n}{2\pi} + \tfrac{1}{2} \ln \det \bar{\jmath}_n + o(1)$$

i.e. D_n grows only logarithmically with n.

E.g. $\bar{\jmath}_n = O(1)$ for the practically very important class of stationary (k^{th}-order) finite-state Markov processes ($k = 0$ is i.i.d.).

Bayesian Sequence Prediction: Summary

- General sequence prediction: Use known (subj.) Bayes mixture $\xi = \sum_{\nu \in \mathcal{M}} w_\nu \nu$ in place of unknown (obj.) true distribution μ.

- Bound on the relative entropy between ξ and μ.

$\Rightarrow$ posterior of ξ converges rapidly to the true posterior μ.

- ξ is also optimal in a decision-theoretic sense w.r.t. any bounded loss function.

- No structural assumptions on $\mathcal{M}$ and $\nu \in \mathcal{M}$.

Literature

[Hut05]   M. Hutter. *Universal Artificial Intelligence: Sequential Decisions based on Algorithmic Probability*. Springer, Berlin, 2005. http://www.hutter1.net/ai/uaibook.htm.

[Jef83]   R. C. Jeffrey. *The Logic of Decision*. University of Chicago Press, Chicago, IL, 2nd edition, 1983.

[Fer67]   T. S. Ferguson. *Mathematical Statistics: A Decision Theoretic Approach*. Academic Press, New York, 3rd edition, 1967.

[DeG70]   M. H. DeGroot. *Optimal Statistical Decisions*. McGraw-Hill, New York, 1970.

8   UNIVERSAL RATIONAL AGENTS

- Agents in Known (Probabilistic) Environments

- The Universal Algorithmic Agent AIXI

- Important Environmental Classes

- Discussion

Universal Rational Agents: Abstract

Sequential decision theory formally solves the problem of rational agents in uncertain worlds if the true environmental prior probability distribution is known. Solomonoff's theory of universal induction formally solves the problem of sequence prediction for unknown prior distribution.

Here we combine both ideas and develop an elegant parameter-free theory of an optimal reinforcement learning agent embedded in an arbitrary unknown environment that possesses essentially all aspects of rational intelligence. The theory reduces all conceptual AI problems to pure computational ones. The resulting AIXI model is the most intelligent unbiased agent possible.

Other discussed topics are optimality notions, asymptotic consistency, and some particularly interesting environment classes.

Overview

- Decision Theory solves the problem of rational agents in uncertain worlds if the environmental probability distribution is known.

- Solomonoff's theory of Universal Induction solves the problem of sequence prediction for unknown prior distribution.

- We combine both ideas and get a parameterless model of

$$
\begin{array}{ccc}
\text{Universal Artificial Intelligence without Parameters} \\
= & & = \\
\text{Decision Theory} & = & \text{Probability} + \text{Utility Theory} \\
+ & & + \\
\text{Universal Induction} & = & \text{Ockham} + \text{Epicurus} + \text{Bayes}
\end{array}
$$

Preliminary Remarks

- The goal is to mathematically define a unique model superior to any other model in any environment.

- The AIXI agent is unique in the sense that it has no parameters which could be adjusted to the actual environment in which it is used.

- In this first step toward a universal theory of AI we are not interested in computational aspects.

- Nevertheless, we are interested in maximizing a utility function, which means to learn in as minimal number of cycles as possible. The interaction cycle is the basic unit, not the computation time per unit.

8.1 AGENTS IN KNOWN (PROBABILISTIC) ENVIRONMENTS: CONTENTS

- The Agent-Environment Model & Interaction Cycle

- Rational Agents in Deterministic Environments

- Utility Theory for Deterministic Environments

- Emphasis in AI/ML/RL $\Leftrightarrow$ Control Theory

- Probabilistic Environment / Perceptions

- Functional$\equiv$Recursive$\equiv$Iterative AIμ Model

- Limits we are Interested in

- Relation to Bellman Equations

- (Un)Known environment μ
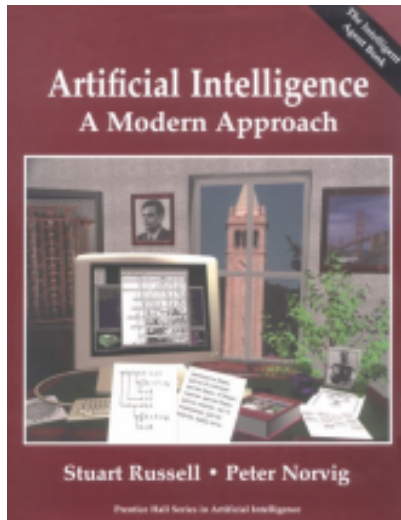
The Agent Model

Most if not all AI problems can be formulated within the agent framework

$r_1 \mid o_1$	$r_2 \mid o_2$	$r_3 \mid o_3$	$r_4 \mid o_4$	$r_5 \mid o_5$	$r_6 \mid o_6$	...

work	*Agent* p	tape ...	work	*Environ-ment* q	tape ...

y_1	y_2	y_3	y_4	y_5	y_6	...
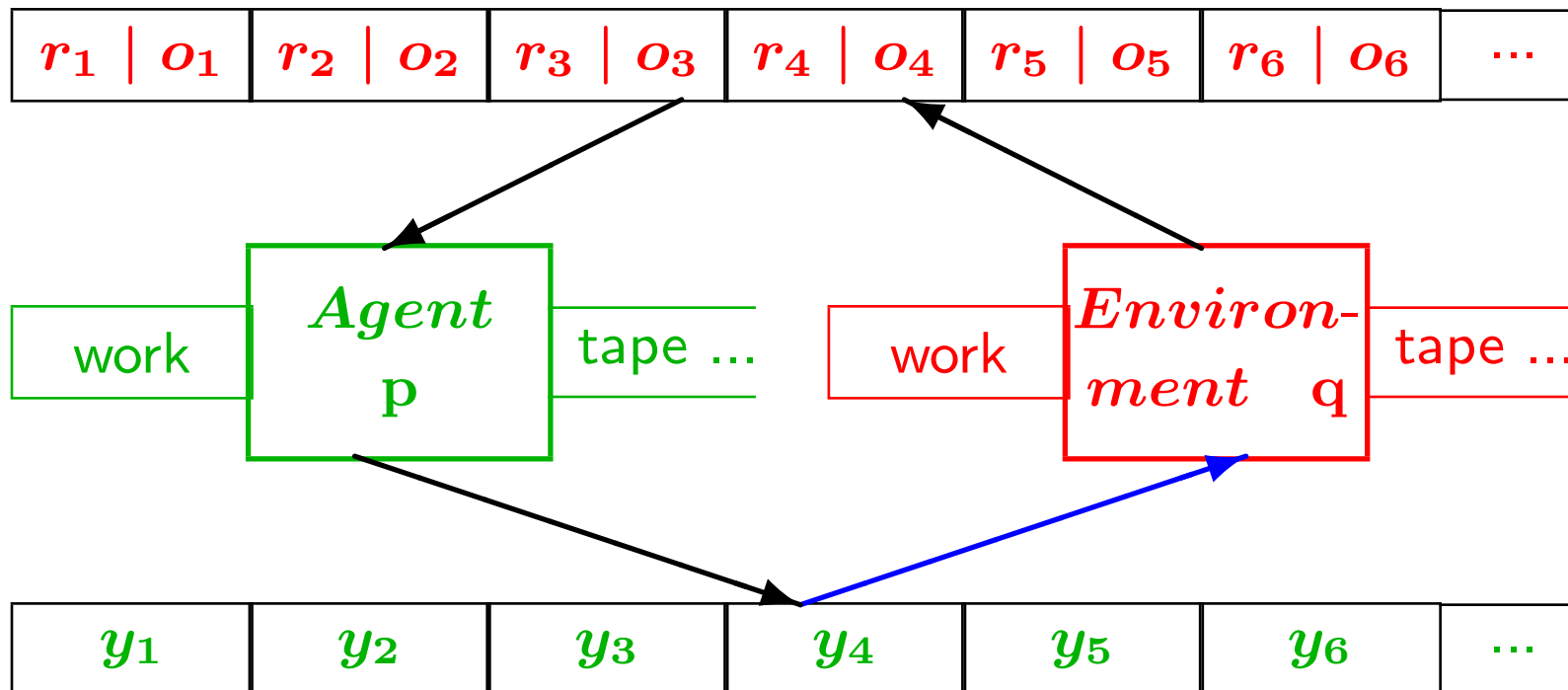
The Agent-Environment Interaction Cycle

for k:=1 to m do

- p thinks/computes/modifies internal state $=$ work tape.

- p writes output $y_k \in \mathcal{Y}$.

- q reads output y_k.

- q computes/modifies internal state.

- q writes reward input $r_k \in \mathcal{R} \subset \mathbb{R}$.

- q writes regular input $o_k \in \mathcal{O}$.

- p reads input $x_k := r_k o_k \in \mathcal{X}$.

endfor

- m is lifetime of system (total number of cycles).
- Often $\mathcal{R} = \{0, 1\} = \{bad, good\} = \{error, correct\}$.

Agents in Deterministic Environments

- $p : \mathcal{X}^* \to \mathcal{Y}^*$ is deterministic policy of the agent,

  $p(x_{<k}) = y_{1:k}$ with $x_{<k} \equiv x_1 ... x_{k-1}$.

- $q : \mathcal{Y}^* \to \mathcal{X}^*$ is deterministic environment,

  $q(y_{1:k}) = x_{1:k}$ with $y_{1:k} \equiv y_1 ... y_k$.

- Input $x_k \equiv r_k o_k$ consists of a regular informative part o_k
  and reward $r(x_k) := r_k \in [0 .. r_{max}]$.

Utility Theory for Deterministic Environments

The ($agent$,$environment$) pair (p,q) produces the unique I/O sequence

$$\omega^{pq} := y_1^{pq} x_1^{pq} y_2^{pq} x_2^{pq} y_3^{pq} x_3^{pq} ...$$

Total reward (value) in cycles k to m is defined as

$$V_{km}^{pq} := r(x_k^{pq}) + ... + r(x_m^{pq})$$

Optimal agent is policy that maximizes total reward

$$p^* := \arg\max_p V_{1m}^{pq}$$

$$\Downarrow$$

$$V_{km}^{p^*q} \geq V_{km}^{pq} \quad \forall p$$

Emphasis in AI/ML/RL ⇔ Control Theory

Both fields start from Bellman-equations and aim at agents/controllers that behave optimally and are adaptive, but differ in terminology and emphasis:

agent	$\widehat{=}$	controller
environment	$\widehat{=}$	system
(instantaneous) reward	$\widehat{=}$	(immediate) cost
model learning	$\widehat{=}$	system identification
reinforcement learning	$\widehat{=}$	adaptive control
exploration↔exploitation problem	$\widehat{=}$	estimation↔control problem
qualitative solution	⇔	high precision
complex environment	⇔	simple (linear) machine
temporal difference	⇔	Kalman filtering / Ricatti eq.

AIξ is the first non-heuristic formal approach that is general enough to cover both fields.   [Hut05]

Probabilistic Environment / Functional AIμ

Replace q by a prior probability distribution $\mu(q)$ over environments.

The total expected reward in cycles k to m is

$$V_{km}^{p\mu}(\ddot{y}\ddot{x}_{<k}) := \frac{1}{\mathcal{N}} \sum_{q:q(\dot{y}_{<k})=\dot{x}_{<k}} \mu(q) \cdot V_{km}^{pq}$$

The history is no longer uniquely determined.

$\ddot{y}\ddot{x}_{<k} := \dot{y}_1\dot{x}_1...\dot{y}_{k-1}\dot{x}_{k-1} :=$actual history.

AIμ maximizes expected future reward by looking $h_k \equiv m_k - k + 1$ cycles ahead (horizon). For $m_k = m$, AIμ is optimal.

$$\dot{y}_k := \arg\max_{y_k} \max_{p:p(\dot{x}_{<k})=\dot{y}_{<k}y_k} V_{km_k}^{p\mu}(\ddot{y}\ddot{x}_{<k})$$

Environment responds with $\dot{x}_k$ with probability determined by μ.

This functional form of AIμ is suitable for theoretical considerations. The iterative form (next slides) is more suitable for 'practical' purpose.

Probabilistic Perceptions

The probability that the environment produces input x_k in cycle k under the condition that the history h is $y_1x_1...y_{k-1}x_{k-1}y_k$ is abbreviated by

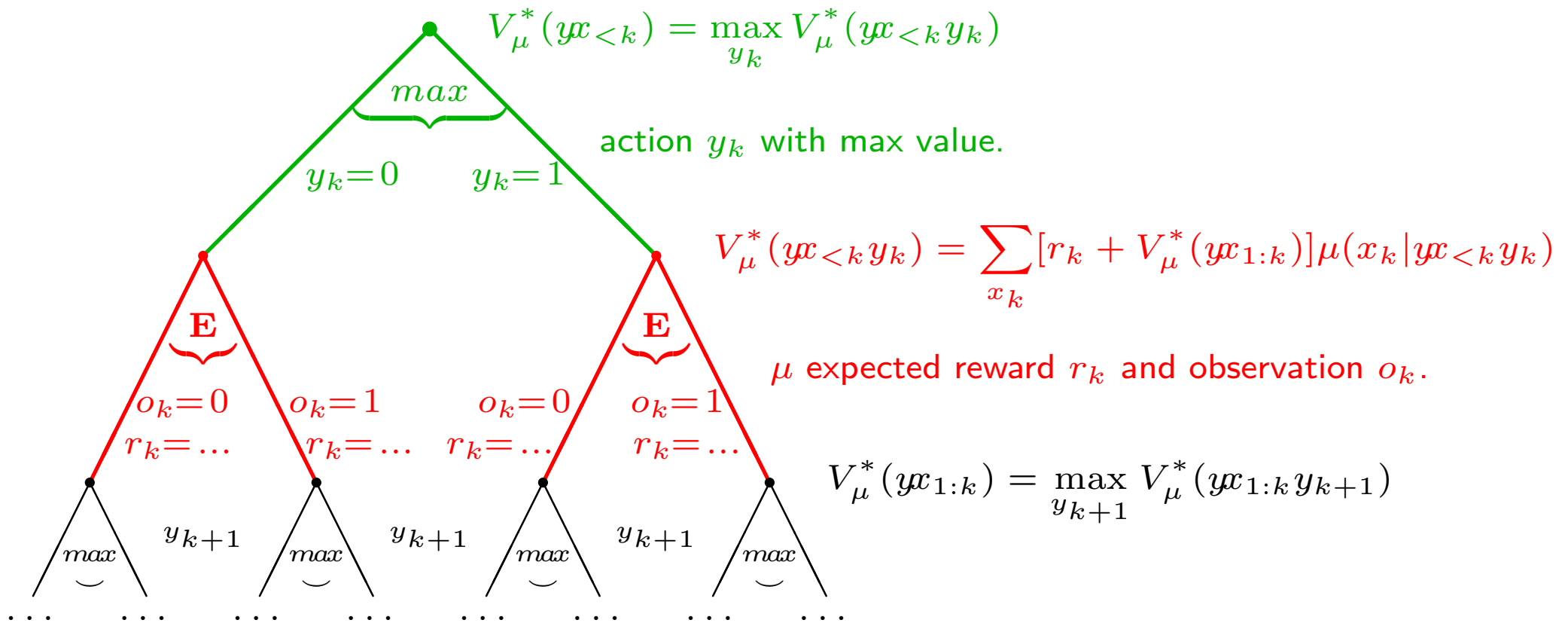$$\mu(x_k|y\!x_{<k}y_k) \;\equiv\; \mu(x_k|y_1x_1...y_{k-1}x_{k-1}y_k)$$

With the chain rule, the probability of input $x_1...x_k$ if system outputs $y_1...y_k$ is

$$\mu(x_1...x_k|y_1...y_k) \;=\; \mu(x_1|y_1)\cdot\mu(x_2|y\!x_1y_2)\cdot\,...\,\cdot\mu(x_k|y\!x_{<k}y_k)$$

A μ of this form is called a chronological probability distribution.

Expectimax Tree – Recursive AIμ Model

$V_\mu^*(h) \equiv V_{km}^{*\mu}(h)$ is the value (future expected reward sum) of the optimal informed agent AIμ in environment μ in cycle k given history h.

$$V_\mu^*(y\!x_{<k}) = \max_{y_k} V_\mu^*(y\!x_{<k}\,y_k)$$

max

action y_k with max value.

$y_k{=}0$ $y_k{=}1$

$$V_\mu^*(y\!x_{<k}\,y_k) = \sum_{x_k}[r_k + V_\mu^*(y\!x_{1:k})]\mu(x_k|y\!x_{<k}\,y_k)$$

E   **E**

μ expected reward r_k and observation o_k.

$o_k{=}0$ $o_k{=}1$ $o_k{=}0$ $o_k{=}1$
$r_k{=}...$ $r_k{=}...$ $r_k{=}...$ $r_k{=}...$

$$V_\mu^*(y\!x_{1:k}) = \max_{y_{k+1}} V_\mu^*(y\!x_{1:k}\,y_{k+1})$$

max y_{k+1} max y_{k+1} max y_{k+1} max

$\ldots$ $\ldots$ $\ldots$ $\ldots$ $\ldots$ $\ldots$ $\ldots$ $\ldots$

Iterative AIμ Model

The Expectimax sequence/algorithm: Take reward expectation over the x_i and maximum over the y_i in chronological order to incorporate correct dependency of x_i and y_i on the history.

$$V_{km}^{*\mu}(\dot{y}\dot{x}_{<k}) = \max_{y_k}\sum_{x_k}...\max_{y_m}\sum_{x_m}(r(x_k)+...+r(x_m))\cdot\mu(x_{k:m}|\dot{y}\dot{x}_{<k}y_{k:m})$$

$$\dot{y}_k = \arg\max_{y_k}\sum_{x_k}...\max_{y_{m_k}}\sum_{x_{m_k}}(r(x_k)+...+r(x_{m_k}))\cdot\mu(x_{k:m_k}|\dot{y}\dot{x}_{<k}y_{k:m_k})$$

This is the essence of Sequential Decision Theory.

Decision Theory = Probability + Utility Theory

Functional=Recursive=Iterative AIμ Model

The functional and recursive/iterative AIμ models behave identically with the natural identification

$$\mu(x_{1:k}|y_{1:k}) \;=\; \sum_{q:q(y_{1:k})=x_{1:k}} \mu(q)$$

Remaining Problems:

- Computational aspects.

- The true prior probability is usually not (even approximately not) known.

Limits we are Interested in

$$1 \quad \ll \quad \langle l(y_k x_k) \rangle \quad \ll \quad k \quad \ll \quad m \quad \ll \quad |\mathcal{Y} \times \mathcal{X}| \quad < \quad \infty$$

$$1 \quad \overset{a}{\ll} \quad 2^{16} \quad \overset{b}{\ll} \quad 2^{24} \quad \overset{c}{\ll} \quad 2^{32} \quad \overset{d}{\ll} \quad 2^{65536} \quad \overset{e}{\underset{<}{}} \quad \infty$$

(a) The agents interface is wide.

(b) The interface is sufficiently explored.

(c) The death is far away.

(d) Most input/outputs do not occur.

(e) All spaces are finite.

These limits are never used in proofs but ...

... we are only interested in theorems which do not degenerate under the above limits.

Relation to Bellman Equations

- If μ^{AI} is a completely observable Markov decision process, then AIμ reduces to the recursive Bellman equations [BT96].

- Recursive AIμ may in general be regarded as (pseudo-recursive) Bellman equation with complete history $y\!x_{<k}$ as environmental state.

- The AIμ model assumes neither stationarity, nor Markov property, nor complete observability of the environment.

$\Rightarrow$ every "state" occurs at most once in the lifetime of the agent. Every moment in the universe is unique!

- There is no obvious universal similarity relation on $(\mathcal{X}\times\mathcal{Y})^*$ allowing an effective reduction of the size of the state space.

Known environment μ

- Assumption: μ is the true environment in which the agent operates

- Then, policy p^μ is optimal in the sense that no other policy for an agent leads to higher μ^{AI}-expected reward.

- Special choices of μ: deterministic or adversarial environments, Markov decision processes (MDPs).

- There is no principle problem in computing the optimal action y_k as long as μ^{AI} is known and computable and $\mathcal{X}$, $\mathcal{Y}$ and m are finite.

- Things drastically change if μ^{AI} is unknown ...

Unknown environment μ

- Reinforcement learning algorithms [SB98] are commonly used in this case to learn the unknown μ or directly its value.

- They succeed if the state space is either small or has effectively been made small by so-called generalization techniques.

- Solutions are either ad hoc, or work in restricted domains only, or have serious problems with state space exploration versus exploitation, or are prone to diverge, or have non-optimal learning rate.

- We introduce a universal and optimal mathematical model now ...

8.2 THE UNIVERSAL ALGORITHMIC AGENT AIXI: CONTENTS

- Formal Definition of Intelligence
- Is Universal Intelligence Υ any Good?
- Definition of the Universal AIXI Model
- Universality of M^{AI} and ξ^{AI}
- Convergence of ξ^{AI} to μ^{AI}
- Intelligence Order Relation
- On the Optimality of AIXI
- Value Bounds & Asymptotic Learnability
- The OnlyOne CounterExample
- Separability Concepts

Formal Definition of Intelligence

- Agent follows policy $\pi : (\mathcal{A} \times \mathcal{O} \times \mathcal{R})^* \rightsquigarrow \mathcal{A}$

- Environment reacts with $\mu : (\mathcal{A} \times \mathcal{O} \times \mathcal{R})^* \times \mathcal{A} \rightsquigarrow \mathcal{O} \times \mathcal{R}$

- Performance of agent π in environment μ
  $=$ expected cumulative reward $= V_\mu^\pi := \mathbb{E}_\mu^\pi [\sum_{t=1}^\infty r_t^{\pi\mu}]$

- True environment μ unknown
  $\Rightarrow$ average over wide range of environments
  (all semi-computable chronological semi-measures $\mathcal{M}_U$)

- Ockham+Epicurus: Weigh each environment with its
  Kolmogorov complexity $K(\mu) := \min_p \{ length(p) : U(p) = \mu \}$

- Universal intelligence of agent π is $\Upsilon(\pi) := \sum_{\mu \in \mathcal{M}_U} 2^{-K(\mu)} V_\mu^\pi$.

- Compare to our informal definition: Intelligence measures an
  agent's ability to perform well in a wide range of environments.

- AIXI $= \arg\max_\pi \Upsilon(\pi) =$ most intelligent agent.

Is Universal Intelligence Υ any Good?

- Captures our informal definition of intelligence.

- Incorporates Occam's razor.

- Very general: No restriction on internal working of agent.

- Correctly orders simple adaptive agents.

- Agents with high Υ like AIXI are extremely powerful.

- Υ spans from very low intelligence up to ultra-high intelligence.

- Practically meaningful: High Υ = practically useful.

- Non-anthropocentric: based on information & computation theory. (unlike Turing test which measures humanness rather than int.)

- Simple and intuitive formal definition: does not rely on equally hard notions such as creativity, understanding, wisdom, consciousness.

Υ is valid, informative, wide range, general, dynamic, unbiased, fundamental, formal, objective, fully defined, universal.

Definition of the Universal AIXI Model

Universal AI = Universal Induction + Decision Theory

Replace μ^{AI} in sequential decision model AIμ by an appropriate generalization of Solomonoff's M.

$$M(x_{1:k}|y_{1:k}) := \sum_{q:q(y_{1:k})=x_{1:k}} 2^{-l(q)}$$

$$\dot{y}_k = \arg\max_{y_k} \sum_{x_k} ... \max_{y_{m_k}} \sum_{x_{m_k}} (r(x_k)+...+r(x_{m_k})){\cdot}M(x_{k:m_k}|\ddot{y}\ddot{x}_{<k}y_{k:m_k})$$

Functional form: $\mu(q) \hookrightarrow \xi(q) := 2^{-\ell(q)}$.

Bold Claim: AIXI is the most intelligent environmental independent agent possible.

Universality of M^{AI} and ξ^{AI}

$$M(x_{1:n}|y_{1:n}) \stackrel{\times}{=} \xi(x_{1:n}|y_{1:n}) \geq 2^{-K(\rho)}\rho(x_{1:n}|y_{1:n}) \quad \forall \text{ chronological } \rho$$

The proof is analog as for sequence prediction. Actions y_k are pure spectators (here and below)

Convergence of ξ^{AI} to μ^{AI}

Similarly to Bayesian multistep prediction [Hut05] one can show

$$\xi^{AI}(x_{k:m_k}|x_{<k}y_{1:m_k}) \stackrel{k\to\infty}{\Longrightarrow} \mu^{AI}(x_{k:m_k}|x_{<k}y_{1:m_k}) \quad \text{with } \mu \text{ prob. 1.}$$

with rapid conv. for bounded horizon $h_k \equiv m_k - k + 1 \leq h_{max} < \infty$

Does replacing μ^{AI} with ξ^{AI} lead to AIξ system with asymptotically optimal behavior with rapid convergence?

This looks promising from the analogy to the Sequence Prediction (SP) case, but is much more subtle and tricky!

Intelligence Order Relation

Definition 8.1 (Intelligence order relation) We call a policy p *more or equally intelligent* than p' and write

$$p \succeq p' \quad :\Leftrightarrow \quad \forall k \forall \ddot{y}\ddot{x}_{<k} : V_{km_k}^{p\xi}(\ddot{y}\ddot{x}_{<k}) \geq V_{km_k}^{p'\xi}(\ddot{y}\ddot{x}_{<k}),$$

i.e. if p yields in any circumstance higher ξ-expected reward than p'.

As the algorithm p^ξ behind the AIXI agent maximizes $V_{km_k}^{p\xi}$, we have $p^\xi \succeq p$ for all p.

The AIXI model is hence the most intelligent agent w.r.t. $\succeq$.

Relation $\succeq$ is a universal order relation in the sense that it is free of any parameters (except m_k) or specific assumptions about the environment.

On the Optimality of AIXI

- **What is meant by universal optimality?** Value bounds for AIXI are expected to be weaker than the SP loss bounds because problem class covered by AIXI is larger.

- The problem of defining and proving general value bounds becomes more feasible by considering, in a first step, restricted environmental classes.

- Another approach is to generalize AIXI to AIξ, where $\xi() = \sum_{\nu \in \mathcal{M}} w_\nu \nu()$ is a general Bayes mixture of distributions ν in some class $\mathcal{M}$.

- A possible further approach toward an optimality "proof" is to regard AIXI as optimal by construction. (common Bayesian perspective, e.g. Laplace rule or Gittins indices).

Value Bounds & Asymptotic Learnability

Naive value bound analogously to error bound for SP

$$V_{1m}^{p^{best}\mu} \overset{?}{\geq} V_{1m}^{p\mu} - o(...) \quad \forall \mu, p$$

HeavenHell Counter-Example: Set of environments $\{\mu_0, \mu_1\}$ with $\mathcal{Y} = \mathcal{R} = \{0, 1\}$ and $r_k = \delta_{iy_1}$ in environment μ_i violates value bound. The first output y_1 decides whether all future $r_k = 1$ or 0.

Asymptotic learnability: μ probability $D_{n\mu\xi}/n$ of suboptimal outputs of AIXI different from AIμ in the first n cycles tends to zero

$$D_{n\mu\xi}/n \to 0 \quad , \quad D_{n\mu\xi} := \mathbb{E}_\mu\left[\sum_{k=1}^n 1 - \delta_{\dot{y}_k^\mu, \dot{y}_k^\xi}\right]$$

This is a weak asymptotic convergence claim.

The OnlyOne CounterExample

Let $\mathcal{R} = \{0, 1\}$ and $|\mathcal{Y}|$ be large. Consider all (deterministic) environments in which a single complex output y^* is correct ($r=1$) and all others are wrong ($r=0$). The problem class is

$$\{\mu : \mu(r_k = 1|x_{<k}y_{1:k}) = \delta_{y_k y^*}, \ K(y^*) = \lfloor \log_2 |Y| \rfloor\}$$

Problem: $D_{k\mu\xi} \leq 2^{K(\mu)}$ is the best possible error bound we can expect, which depends on $K(\mu)$ only. It is useless for $k \ll |Y| \stackrel{\times}{=} 2^{K(\mu)}$, although asymptotic convergence satisfied.

But: A bound like $2^{K(\mu)}$ reduces to $2^{K(\mu|\dot{x}_{<k})}$ after k cycles, which is $O(1)$ if enough information about μ is contained in $\dot{x}_{<k}$ in any form.

Separability Concepts

that might be useful for proving reward bounds

- Forgetful μ.

- Relevant μ.

- Asymptotically learnable μ.

- Farsighted μ.

- Uniform μ.

- (Generalized) Markovian μ.

- Factorizable μ.

- (Pseudo) passive μ.

Other concepts

- Deterministic μ.

- Chronological μ.

8.3 Important Environmental Classes: Contents

- Sequence Prediction (SP)

- Strategic Games (SG)

- Function Minimization (FM)

- Supervised Learning by Examples (EX)

In this subsection $\xi \equiv \xi^{AI} :\overset{\times}{=} M^{AI}$.

Particularly Interesting Environments

- Sequence Prediction, e.g. weather or stock-market prediction.
  Strong result: $V_\mu^* - V_\mu^{p^\xi} = O(\sqrt{\frac{K(\mu)}{m}})$, $m =$horizon.

- Strategic Games: Learn to play well (minimax) strategic zero-sum games (like chess) or even exploit limited capabilities of opponent.

- Optimization: Find (approximate) minimum of function with as few function calls as possible. Difficult exploration versus exploitation problem.

- Supervised learning: Learn functions by presenting $(z, f(z))$ pairs and ask for function values of z' by presenting $(z', ?)$ pairs. Supervised learning is much faster than reinforcement learning.

AIξ quickly learns to predict, play games, optimize, and learn supervised.

Sequence Prediction (SP)

SPμ Model: Binary sequence $z_1 z_2 z_3...$ with true prior $\mu^{SP}(z_1 z_2 z_3...)$.

AIμ Model: $y_k =$ prediction for z_k; $\quad o_{k+1} = \epsilon$.

$r_{k+1} = \delta_{y_k z_k} = 1/0$ if prediction was correct/wrong.

Correspondence:

$\mu^{AI}(r_1...r_k|y_1...y_k) = \mu^{SP}(\delta_{y_1 r_1}...\delta_{y_k r_k}) = \mu^{SP}(z_1...z_k)$

For arbitrary horizon h_k: $\dot{y}_k^{AI\mu} = \arg\max_{y_k} \mu(y_k|\dot{z}_1...\dot{z}_{k-1}) = \dot{y}_k^{SP\Theta_\mu}$

Generalization: AIμ always reduces exactly to XXμ model if XXμ is optimal solution in domain XX.

AIξ model differs from SPξ model: Even for $h_k = 1$

$$\dot{y}_k^{AI\xi} = \arg\max_{y_k} \xi(r_k = 1|\dot{y}\dot{r}_{<k}y_k) \neq \dot{y}_k^{SP\Theta_\xi}$$

Weak error bound: $\#\text{Errors}_{n\xi}^{AI} \overset{\times}{<} 2^{K(\mu)} < \infty$ for deterministic μ.

Strategic Games (SG)

- Consider strictly competitive strategic games like chess.

- Minimax is best strategy if both Players are rational with unlimited capabilities.

- Assume that the environment is a minimax player of some game $\Rightarrow$ μ^{AI} uniquely determined.

- Inserting μ^{AI} into definition of $\dot{y}_k^{AI}$ of AIμ model reduces the expecimax sequence to the minimax strategy $(\dot{y}_k^{AI} = \dot{y}_k^{SG})$.

- As $\xi^{AI} \to \mu^{AI}$ we expect AIξ to learn the minimax strategy for any game and minimax opponent.

- If there is only non-trivial reward $r_k \in \{win, loss, draw\}$ at the end of the game, repeated game playing is necessary to learn from this very limited feedback.

- AIξ can exploit limited capabilities of the opponent.

Function Maximization (FM)

Approximately maximize (unknown) functions with as few function calls as possible. Applications:

- Traveling Salesman Problem (bad example).
- Minimizing production costs.
- Find new materials with certain properties.
- Draw paintings which somebody likes.

$$\mu^{FM}(z_1...z_n|y_1...y_n) := \sum_{f:f(y_i)=z_i \ \forall 1\leq i\leq n} \mu(f)$$

Greedily choosing y_k which maximizes f in the next cycle does not work.

General Ansatz for FMμ/ξ:
$$\dot{y}_k = \arg\max_{y_k} \sum_{z_k} ... \max_{y_m} \sum_{z_m} (\alpha_1 z_1 + ... + \alpha_m z_m)\cdot\mu(z_m|\ddot{y}\ddot{z}_1...y_m)$$

Under certain weak conditions on α_i, f can be learned with AIξ.

Function Maximization – Example

Very hard problem in practice, since (unlike prediction, classification, regression) it involves the infamous exploration$\leftrightarrow$explotation problem
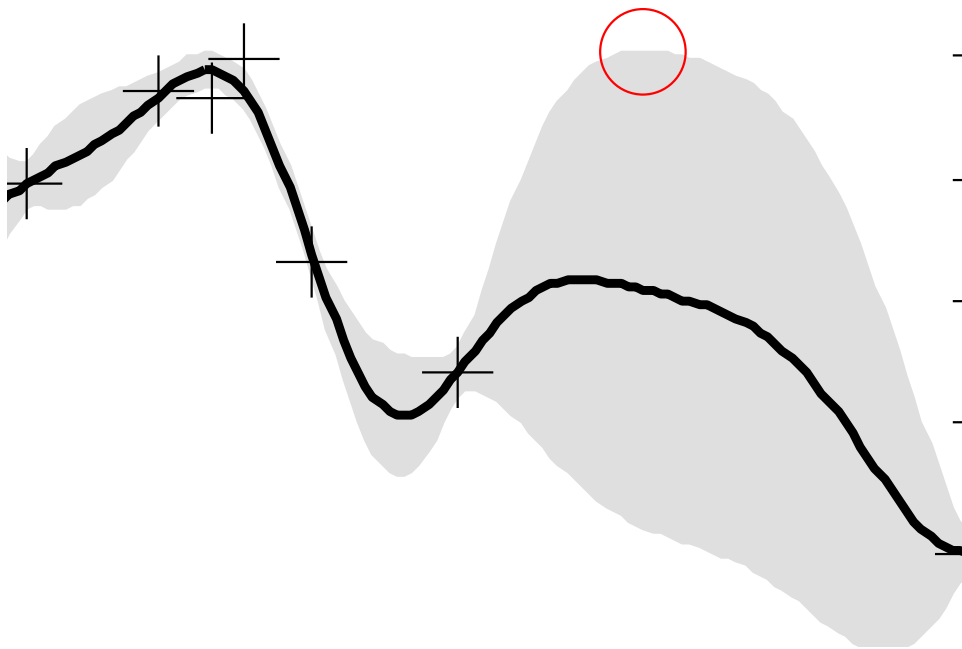
[Srinivas et al. 2010]

Exploration: If horizon is large, function is probed where uncertainty is large, since global maximum might be there.

Exploitation: If horizon is small, function is probed where maximum is believed to be, since agent needs/wants good results now.

Efficient and effective heuristics for special function classes available: Extension of Upper Confidence Bound for Bandits (UCB) algorithm.

Supervised Learning by Examples (EX)

Learn functions by presenting $(z, f(z))$ pairs and ask for function values of z' by presenting $(z', ?)$ pairs.

More generally: Learn relations $R \ni (z, v)$.

Supervised learning is much faster than reinforcement learning.

The AIμ/ξ model:

$$o_k = (z_k, v_k) \in R \cup (Z \times \{?\}) \subset Z \times (Y \cup \{?\}) = O$$

$y_{k+1} =$ guess for true v_k if actual $v_k = ?$.

$r_{k+1} = 1$ iff $(z_k, y_{k+1}) \in R$

AIμ is optimal by construction.

EX is closely related to classification which itself can be phrased as sequence prediction task.

Supervised Learning – Intuition

The AIξ model:

- Inputs o_k contain much more than 1 bit feedback per cycle.

- Short codes dominate ξ.

- The shortest code of examples (z_k, v_k) is a coding of R and the indices of the (z_k, v_k) in R.

- This coding of R evolves independently of the rewards r_k.

- The system has to learn to output y_{k+1} with $(z_k, y_{k+1}) \in R$.

- As R is already coded in q, an additional algorithm of length $O(1)$ needs only to be learned.

- Rewards r_k with information content $O(1)$ are needed for this only.

- AIξ learns to learn supervised.

8.4 Discussion: Contents

- Uncovered Topics

- Remarks

- Outlook

- Exercises

- Literature

Uncovered Topics

- General and special reward bounds and convergence results for AIXI similar to SP case.

- Downscale AIXI in more detail and to more problem classes analog to the downscaling of SP to Minimum Description Length and Finite Automata.

- There is no need for implementing extra knowledge,
  as this can be learned by presenting it in o_k in any form.

- The learning process itself is an important aspect.

- Noise or irrelevant information in the inputs do not disturb the AIXI system.

Remarks

- We have developed a parameterless AI model based on sequential decisions and algorithmic probability.

- We have reduced the AI problem to pure computational questions.

- AIξ seems not to lack any important known methodology of AI, apart from computational aspects.

- Philosophical questions: relevance of non-computational physics (Penrose), number of wisdom Ω (Chaitin), consciousness, social consequences.

Outlook

mainly technical results for AIXI and variations

- General environment classes $\mathcal{M}_U \rightsquigarrow \mathcal{M}$.

- Results for general/universal $\mathcal{M}$ for discussed performance criteria.

- Strong guarantees for specific classes $\mathcal{M}$ by exploiting extra properties of the environments.

- Restricted policy classes.

- Universal choice of the rewards.

- Discounting future rewards and time(in)consistency.

- Approximations and algorithms.

Most of these items will be covered in the next Chapter

Exercises

1. [C30] Proof equivalence of the functional, recursive, and iterative AIμ models. Hint: Consider $k = 2$ and $m = 3$ first. Use $\max_{y_3(\cdot)} \sum_{x_2} f(x_2, y_3(x_2)) \equiv \sum_{x_2} \max_{y_3} f(x_2, y_3)$, where $y_3(\cdot)$ is a function of x_2, and $\max_{y_3(\cdot)}$ maximizes over all such functions.

2. [C30] Show that the optimal policy $p_k^* := \arg\max_p V_{km}^{p\mu}(yx_{<k})$ is independent of k. More precisely, the actions of p_1^* and p_k^* in cycle t given history $yx_{<t}$ coincide for $k \geq t$. The derivation goes hand in hand with the derivation of Bellman's equations [BT96].

Literature

[SB98]   R. S. Sutton and A. G. Barto. *Reinforcement Learning: An Introduction*. MIT Press, Cambridge, MA, 1998.

[RN10]   S. J. Russell and P. Norvig. *Artificial Intelligence. A Modern Approach*. Prentice-Hall, Englewood Cliffs, NJ, 3rd edition, 2010.

[LH07]   S. Legg and M. Hutter. *Universal intelligence: A definition of machine intelligence.* Minds & Machines, 17(4):391–444, 2007.

[Hut05]   M. Hutter. *Universal Artificial Intelligence: Sequential Decisions based on Algorithmic Probability*. Springer, Berlin, 2005. http://www.hutter1.net/ai/uaibook.htm.

9 THEORY OF RATIONAL AGENTS

- The Bayesian Agent AIξ

- Future Value and Discounting

- Knowledge-Seeing and Optimistic Agents

- Discussion

Theory of Rational Agents: Abstract

... There are strong arguments that the resulting AIXI model is the most intelligent unbiased agent possible.

Other discussed topics are relations between problem classes, the horizon problem, and computational issues.

9.1 THE BAYESIAN AGENT AIξ: CONTENTS

- Agents in Probabilistic Environments

- Optimal Policy and Value – AIρ Model

- The Bayes-Mixture Distribution ξ

- Questions of Interest

- Linearity and Convexity of V_ρ in ρ

- Pareto Optimality

- Self-optimizing Policies

- Environments w./ (Non)Self-Optimizing Policies

Agents in Probabilistic Environments

Given history $y_{1:k}x_{<k}$, the probability that the environment leads to perception x_k in cycle k is (by definition) $\rho(x_k|y_{1:k}x_{<k})$.

Abbreviation (chain rule)

$$\rho(x_{1:m}|y_{1:m}) \; = \; \rho(x_1|y_1)\cdot\rho(x_2|y_{1:2}x_1)\cdot...\cdot\rho(x_m|y_{1:m}x_{<m})$$

The average value of policy p with horizon m in environment ρ is defined as

$$V_\rho^p \; := \; \tfrac{1}{m}\sum_{x_{1:m}}(r_1+...+r_m)\rho(x_{1:m}|y_{1:m})_{|y_{1:m}=p(x_{<m})}$$

The goal of the agent should be to maximize the value.

Optimal Policy and Value – AIρ Model

The ρ-optimal policy $p^\rho := \arg\max_p V_\rho^p$ maximizes $V_\rho^p \leq V_\rho^* := V_\rho^{p^\rho}$.

Explicit expressions for the action y_k in cycle k of the ρ-optimal policy p^ρ and their value V_ρ^* are

$$y_k = \arg\max_{y_k} \sum_{x_k} \max_{y_{k+1}} \sum_{x_{k+1}} \ldots \max_{y_m} \sum_{x_m} (r_k + \ldots + r_m) \cdot \rho(x_{k:m}|y_{1:m}x_{<k}),$$

$$V_\rho^* = \frac{1}{m} \max_{y_1} \sum_{x_1} \max_{y_2} \sum_{x_2} \ldots \max_{y_m} \sum_{x_m} (r_1 + \ldots + r_m) \cdot \rho(x_{1:m}|y_{1:m}).$$

Keyword: Expectimax tree/algorithm.

The Bayes-Mixture Distribution ξ

Assumption: The true environment μ is unknown.

Bayesian approach: The true probability distribution μ^{AI} is not learned directly, but is replaced by a Bayes-mixture ξ^{AI}.

Assumption: We know that the true environment μ is contained in some known (finite or countable) set $\mathcal{M}$ of environments.

The Bayes-mixture ξ is defined as

$$\xi(x_{1:m}|y_{1:m}) := \sum_{\nu \in \mathcal{M}} w_\nu \nu(x_{1:m}|y_{1:m}) \quad \text{with} \quad \sum_{\nu \in \mathcal{M}} w_\nu = 1, \quad w_\nu > 0 \,\forall \nu$$

The weights w_ν may be interpreted as the prior degree of belief that the true environment is ν.

Then $\xi(x_{1:m}|y_{1:m})$ could be interpreted as the prior subjective belief probability in observing $x_{1:m}$, given actions $y_{1:m}$.

Questions of Interest

- It is natural to follow the policy p^ξ which maximizes V_ξ^p.

- If μ is the true environment the expected reward when following policy p^ξ will be $V_\mu^{p^\xi}$.

- The optimal (but infeasible) policy p^μ yields reward $V_\mu^{p^\mu} \equiv V_\mu^*$.

- Are there policies with uniformly larger value than $V_\mu^{p^\xi}$?

- How close is $V_\mu^{p^\xi}$ to V_μ^*?

- What is the most general class $\mathcal{M}$ and weights w_ν?

  $\mathcal{M} = \mathcal{M}_U$ and $w_\nu = 2^{-K(\nu)} \quad \Longrightarrow \quad$ AIξ =AIXI !

Linearity and Convexity of V_ρ in ρ

> **Theorem 9.1 (Linearity and convexity of V_ρ in ρ)**
>
> V_ρ^p is a linear function in ρ: $\quad V_\xi^p = \sum_\nu w_\nu V_\nu^p$
>
> V_ρ^* is a convex function in ρ: $\quad V_\xi^* \leq \sum_\nu w_\nu V_\nu^*$
>
> where $\quad \xi(x_{1:m}|y_{1:m}) = \sum_\nu w_\nu \, \nu(x_{1:m}|y_{1:m})$.

These are the crucial properties of the value function V_ρ.

Loose interpretation: A mixture can never increase performance.

Pareto Optimality

Every policy based on an estimate ρ of μ which is closer to μ than ξ is, outperforms p^ξ in environment μ, simply because it is more tailored toward μ. On the other hand, such a system performs worse than p^ξ in other environments:

Theorem 9.2 (Pareto optimality of p^ξ) Policy p^ξ is Pareto-optimal in the sense that there is no other policy p with $V_\nu^p \geq V_\nu^{p^\xi}$ for all $\nu \in \mathcal{M}$ and strict inequality for at least one ν.

From a practical point of view a significant increase of V for many environments ν may be desirable even if this causes a small decrease of V for a few other ν. This is impossible due to

Balanced Pareto optimality:
$$\Delta_\nu := V_\nu^{p^\xi} - V_\nu^{\tilde{p}}, \quad \Delta := \sum_\nu w_\nu \Delta_\nu \quad \Rightarrow \quad \Delta \geq 0.$$

Self-optimizing Policies

Under which circumstances does the value of the universal policy p^ξ converge to optimum?

$$V_\nu^{p^\xi} \to V_\nu^* \quad \text{for horizon} \quad m \to \infty \quad \text{for all} \quad \nu \in \mathcal{M}. \qquad (9.3)$$

The least we must demand from $\mathcal{M}$ to have a chance that (9.3) is true is that there exists some policy $\tilde{p}$ at all with this property, i.e.
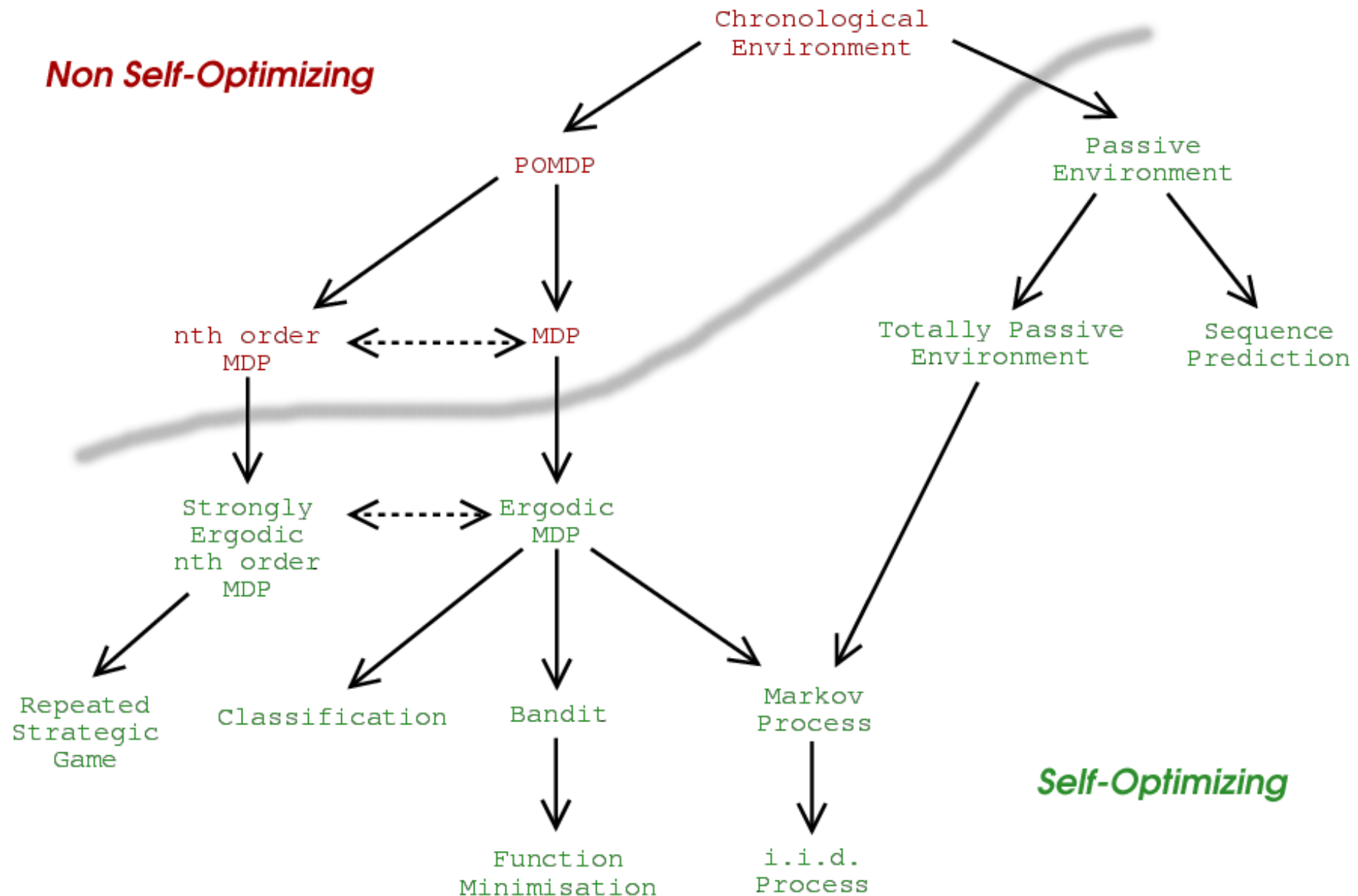
$$\exists \tilde{p}: \ V_\nu^{\tilde{p}} \to V_\nu^* \quad \text{for horizon} \quad m \to \infty \quad \text{for all} \quad \nu \in \mathcal{M}. \qquad (9.4)$$

Main result:

Theorem 9.5 (Self-optimizing policy p^ξ (9.4) $\Rightarrow$ (9.3))
The necessary condition of the existence of a self-optimizing policy $\tilde{p}$ is also sufficient for p^ξ to be self-optimizing.

Environments w./ (Non)Self-Optimizing Policies

Discussion of Self-optimizing Property

- The beauty of this theorem is that the necessary condition of convergence is also sufficient.

- The unattractive point is that this is not an asymptotic convergence statement of a single policy p^ξ for time $k \to \infty$ for some fixed m.

- Shift focus from the total value V and horizon $m \to \infty$ to the future value (value-to-go) V and current time $k \to \infty$.

9.2 FUTURE VALUE AND DISCOUNTING: CONTENTS

- Results for Discounted Future Value

- Continuity of Value

- Convergence of Universal to True Value

- Markov Decision Processes (MDP)

- Importance of the Right Discounting

- Properties of Ergodic MDPs

- General Discounting

- Effective Horizon

- Other Attempts to Deal with the Horizon Issue

- Time(In)Consistent Discounting

Future Value and Discounting

- Eliminate the horizon by discounting the rewards $r_k \rightsquigarrow \gamma_k r_k$ with $\Gamma_k := \sum_{i=k}^{\infty} \gamma_i < \infty$ and letting $m \to \infty$.

- $V_{k\gamma}^{\pi\rho} := \dfrac{1}{\Gamma_k} \lim_{m \to \infty} \sum_{x_{k:m}} (\gamma_k r_k + ... + \gamma_m r_m) \rho(x_{k:m} | y_{1:m} x_{<k})_{|y_{1:m} = p(x_{<m})}$

- Further advantage: Traps (non-ergodic environments) do not necessarily prevent self-optimizing policies any more.

Results for Discounted Future Value

Theorem 9.6 (Properties of Discounted Future Value)

- $V_{k\gamma}^{\pi\rho}$ is linear in ρ: $V_{k\gamma}^{\pi\xi} = \sum_\nu w_k^\nu V_{k\gamma}^{\pi\nu}$.

- $V_{k\gamma}^{*\rho}$ is convex in ρ: $V_{k\gamma}^{*\xi} \leq \sum_\nu w_k^\nu V_{k\gamma}^{*\nu}$.

- where $w_k^\nu := w_\nu \frac{\nu(x_{<k}|y_{<k})}{\xi(x_{<k}|y_{<k})}$ is the posterior belief in ν.

- p^ξ is Pareto-optimal in the sense that there is no other policy π with $V_{k\gamma}^{\pi\nu} \geq V_{k\gamma}^{p^\xi\nu}$ for all $\nu \in \mathcal{M}$ and strict inequality for at least one ν.

- If there exists a self-optimizing policy for $\mathcal{M}$, then p^ξ is self-optimizing in the sense that
  If $\exists \tilde{\pi}_k \forall \nu : V_{k\gamma}^{\tilde{\pi}_k\nu} \overset{k\to\infty}{\longrightarrow} V_{k\gamma}^{*\nu} \implies V_{k\gamma}^{p^\xi\mu} \overset{k\to\infty}{\longrightarrow} V_{k\gamma}^{*\mu}$.

Continuity of Value

> **Theorem 9.7 (Continuity of discounted value)**
>
> The values $V_{k\gamma}^{\pi\mu}$ and $V_{k\gamma}^{*\mu}$ are continuous in μ, and $V_{k\gamma}^{p^{\hat{\mu}}\mu}$ is continuous in $\hat{\mu}$ at $\hat{\mu} = \mu$ w.r.t. a conditional 1-norm in the following sense:
> If $\sum_{x_k} |\mu(x_k|x_{<k}y_{1:k}) - \hat{\mu}(x_k|x_{<k}y_{1:k})| \le \varepsilon \ \forall yx_{<k}y_k \ \forall k \ge k_0$, then
> $$|V_{k\gamma}^{\pi\mu} - V_{k\gamma}^{\pi\hat{\mu}}| \le \delta(\varepsilon), \quad |V_{k\gamma}^{*\mu} - V_{k\gamma}^{*\hat{\mu}}| \le \delta(\varepsilon), \quad |V_{k\gamma}^{*\mu} - V_{k\gamma}^{p^{\hat{\mu}}\mu}| \le 2\delta(\varepsilon)$$
> $\forall\, k \ge k_0$ and $yx_{<k}$, where $\delta(\varepsilon) := r_{max} \cdot \min_{n \ge k}\{(n-k)\varepsilon + \frac{\Gamma_n}{\Gamma_k}\} \xrightarrow{\varepsilon \to 0} 0$.

Warning: $V_{k\gamma}^{p^\xi\mu} \not\to V_{k\gamma}^{*\mu}$, since $\xi \to \mu$ does not hold for all $yx_{1:\infty}$, but only for μ-random ones.

Average Value: By setting $\gamma_k = 1$ for $k \le m$ and $\gamma_k = 0$ for $k > m$ we also get continuity of $V_{km}^{\cdots}$.

Convergence of Universal to True Value

Theorem 9.8 (Convergence of universal to true value)

For a given policy p and history generated by p and μ, i.e. on-policy, the future universal value $V_{...}^{p\xi}$ converges to the true value $V_{...}^{p\mu}$:

$$V_{km_k}^{p\xi} \overset{k\to\infty}{\longrightarrow} V_{km_k}^{p\mu} \qquad \text{i.m.s.} \qquad \text{if} \quad h_{max} < \infty,$$

$$V_{k\gamma}^{p\xi} \overset{k\to\infty}{\longrightarrow} V_{k\gamma}^{p\mu} \qquad \text{i.m.} \qquad \text{for any} \quad \gamma.$$

If the history is generated by $p = p^\xi$, this implies $V_{k\gamma}^{*\xi} \to V_{k\gamma}^{p^\xi\mu}$.

Hence the universal value $V_{k\gamma}^{*\xi}$ can be used to estimate the true value $V_{k\gamma}^{p^\xi\mu}$, without any assumptions on $\mathcal{M}$ and γ.

Nevertheless, maximization of $V_{k\gamma}^{p\xi}$ may asymptotically differ from max. of $V_{k\gamma}^{p\mu}$, since $V_{k\gamma}^{p\xi} \not\to V_{k\gamma}^{p\mu}$ for $p \neq p^\xi$ is possible (and also $V_{k\gamma}^{*\xi} \not\to V_{k\gamma}^{*\mu}$).

Markov Decision Processes (MDP)

From all possible environments, Markov (Decision) Processes are probably the most intensively studied ones.

> **Definition 9.9 (Ergodic MDP)**
> We call μ a (stationary) MDP if the probability of observing $o_k \in \mathcal{O}$ and reward $r_k \in \mathcal{R}$, only depends on the last action $y_k \in \mathcal{Y}$ and the last observation o_{k-1} (called state), i.e. if $\mu(x_k|x_{<k}y_{1:k}) = \mu(x_k|o_{k-1}y_k)$, where $x_k \equiv o_k r_k$.
> An MDP μ is called ergodic if there exists a policy under which every state is visited infinitely often with probability 1.

If the transition matrix $\mu(o_k|o_{k-1}y_k)$ is independent of the action y_k, the MDP is a Markov process;

If $\mu(x_k|o_{k-1}y_k)$ is independent of o_{k-1} we have an i.i.d. process.

Importance of the Right Discounting

Standard geometric discounting: $\gamma_k = \gamma^k$ with $0 < \gamma < 1$.

Problem: Most environments do not possess self-optimizing policies under this discounting.

Reason: Effective horizon h_k^{eff} is finite ($\sim 1/\ln\frac{1}{\gamma}$ for $\gamma_k = \gamma^k$).

The analogue of $m \to \infty$ is $k \to \infty$ *and* $h_k^{eff} \to \infty$ for $k \to \infty$.

Result: Policy p^ξ is self-optimizing for the class of (l^{th} order) ergodic MDPs if $\frac{\gamma_{k+1}}{\gamma_k} \to 1$.

Example discounting: $\gamma_k = k^{-2}$ or $\gamma_k = k^{-1-\varepsilon}$ or $\gamma_k = 2^{-K(k)}$.

Horizon is of the order of the age of the agent: $h_k^{eff} \sim k$.

Properties of Ergodic MDPs

- Stationary MDPs μ have stationary optimal policies p^μ in case of geometric discount, mapping the same state/observation o_k always to the same action y_k.

- A mixture ξ of MDPs is itself not an MDP, i.e. $\xi \notin \mathcal{M}_{MDP}$ $\Rightarrow$ p^ξ is, in general, not a stationary policy.

- There are self-optimizing policies for the class of ergodic MDPs for the average value V_ν, and for the future value $V_{k\gamma}$ if $\frac{\gamma_{k+1}}{\gamma_k} \to 1$.

- Hence Theorems 9.5 and 9.6 imply that p^ξ is self-optimizing for ergodic MDPs (if $\frac{\gamma_{k+1}}{\gamma_k} \to 1$).

- $\frac{\gamma_{k+1}}{\gamma_k} \to 1$ for $\gamma_k = 1/k^2$, but not for $\gamma_k = \gamma^k$.

- **Fazit:** Condition $\frac{\gamma_{k+1}}{\gamma_k} \to 1$ admits *self-optimizing Bayesian policies*.

General Discounting

- Future rewards give only small contribution to $V_{k\gamma}$

  $\Rightarrow$ effective horizon.

- The only significant arbitrariness in the AIXI model lies in the choice of the horizon.

- Power damping $\gamma_k = k^{-1-\varepsilon}$ leads to horizon proportional to age k of agent.

  It does not introduce arbitrary time-scale and has natural/plausible horizon.

- Universal discount $\gamma_k = 2^{-K(k)}$ leads to largest possible horizon.

  Allows to "mimic" all other more greedy behaviors based on other discounts.

Effective Horizon

Table 9.10 (Effective horizon)

$h_k^{eff} := \min\{h \geq 0 : \Gamma_{k+h} \leq \frac{1}{2}\Gamma_k\}$ for various types of discounts γ_k

Horizons	γ_k	$\Gamma_k = \sum_{i=k}^{\infty} \gamma_i$	h_k^{eff}
finite	1 for $k \leq m$ 0 for $k > m$	$m - k + 1$	$\frac{1}{2}(m - k + 1)$
geometric	$\gamma^k,\ 0 \leq \gamma < 1$	$\frac{\gamma^k}{1-\gamma}$	$\frac{\ln 2}{\ln \gamma^{-1}}$
quadratic	$\frac{1}{k(k+1)}$	$\frac{1}{k}$	k
power	$k^{-1-\varepsilon},\ \varepsilon > 0$	$\sim \frac{1}{\varepsilon} k^{-\varepsilon}$	$\sim (2^{1/\varepsilon} - 1)k$
harmonic$_{\approx}$	$\frac{1}{k \ln^2 k}$	$\sim \frac{1}{\ln k}$	$\sim k^2$
universal	$2^{-K(k)}$	decreases slower than any computable function	increases faster than any computable function

Other Attempts to Deal with Horizon Issue

- **Finite horizon:**
  - good if known,
  - bad if unknown and for asymptotic analysis.

- **Infinite horizon:**
  - Limit may not exist.
  - can delay exploitation indefinitely,
    since no finite exploration decreases value.
  - immortal agents can be lazy.

- **Average reward and differential gain:**
  - limit may not exist.

- **Moving horizon m_k:**
  - can lead to very bad time-inconsistent behavior.

- **Time-inconsistent discounting** ...

Time(In)Consistent Discounting

- Generalize $V_{k\gamma}^{\pi\rho} \equiv \frac{1}{\Gamma_k}\mathbb{E}^{\pi\rho}[\sum_{t=k}^{\infty}\gamma_t r_t]$ to:

  Potentially different discount sequence $d_1^k, d_2^k, d_3^k, ...$ for different k:

  Value $V_{k\gamma}^{\pi\rho} := \mathbb{E}^{\pi\rho}[\sum_{t=k}^{\infty} d_t^k r_t]$

- Leads in general to time-inconsistency,

  i.e. $\pi_k^* := \arg\max_\pi V_{k\gamma}^{\pi\rho}$ depends on k.

- Consequence: Agent plans to do one thing,

  but then changes its mind.

  Can in general lead to very bad behavior.

- Humans seem to behave time-inconsistently.

  Solution: Pre-commitment strategies.

Time(In)Consistent Discounting (ctd)

Time-consistent examples: $d_t^k = \gamma^{t-k}$ geometric discounting.

Is the only time-invariant consistent discounting

Time-inconsistent example: $d_t^k = (t - k + 1)^{-1-\varepsilon}$ ($\approx$humans)

Theorem 9.11 (Time(In)Consistent Discounting)   [LH11]

d_t^k is time-consistent $\quad \Longleftrightarrow \quad d_{()}^k \propto d_{()}^1$ for all k.

What to do if you know you're time inconsistent?

Treat your future selves as opponents in an extensive game and follow

sub-game perfect equilibrium policy.

9.3 Optimistic and Knowledge-Seeking Variations of AIξ: Contents

• Universal Knowledge-Seeking Agent

• Optimistic Agents in Deterministic Worlds

• Optimistic Agents for General Environments

• Optimism in MDPs

Universal Knowledge-Seeking Agent (KSA)

reward for exploration; goal is to learn the true environment [OLH13]

- $w_k^\nu := w_\nu \frac{\nu(x_{<k}|y_{<k})}{\xi(x_{<k}|y_{<k})}$ is the posterior belief in ν given history $yx_{<k}$.

- $w_k^{()}$ summarizes the information contained in history $yx_{<k}$.

- $w_k^{()} \rightsquigarrow w_{k+1}^{()}$ changes $\Leftrightarrow x_k$ given $yx_{<k}$ is informative about $\nu \in \mathcal{M}$

- Information gain can be quantified by KL-divergence.

- Reward agent for gained information:
$$r_k := \mathsf{KL}(w_{k+1}^{()}||w_k^{()}) \equiv \sum_{\nu \in \mathcal{M}} w_{k+1}^\nu \log(w_{k+1}^\nu/w_k^\nu)$$

Asymptotic Optimality of Universal KSA

Theorem 9.12 (Asymptotic Optimality of Universal KSA)

- Universal π_ξ^* converges to optimal π_μ^*. More formally:

- $P_\xi^\pi(\cdot|yx_{<k})$ converges in (μ, π_ξ^*)-probability to $P_\mu^\pi(\cdot|yx_{<k})$ uniformly for all π.

Def: $P_\rho^\pi(\cdot|yx_{<k})$ is (ρ, π)-probability of future $yx_{k:\infty}$ given past $yx_{<k}$.

Note: On-policy agent π_ξ^* is able to even predict off-policy!

Remark: **No** assumption on $\mathcal{M}$ needed, i.e. Thm. applicable to $\mathcal{M}_U$.

Optimistic Agents in Deterministic Worlds

act optimally w.r.t. the most optimistic environment

until it is contradicted [SH12]

- $\pi^\circ := \pi_k^* := \arg\max_\pi \max_{\nu \in \mathcal{M}_{k-1}} V_{k\gamma}^{\pi\nu}(yx_{<k})$

- $\mathcal{M}_{k-1} :=$ environments consistent with history $yx_{<k}$.

- As long as the outcome is consistent with the optimistic prediction, the return is optimal, even if the wrong environment is chosen.

Theorem 9.13 (Optimism is asymptotically optimal)

For finite $\mathcal{M} \equiv \mathcal{M}_0$,

- Asymptotic: $V_{k\gamma}^{\pi^\circ \mu} = V_{k\gamma}^{*\mu}$ for all large k.

- Errors: For geometric discount, $V_{k\gamma}^{\pi^\circ \mu} \geq V_{k\gamma}^{*\mu} - \varepsilon$ (i.e. π° ε-sub-optimal) for all but at most $|\mathcal{M}| \frac{\log \varepsilon(1-\gamma)}{\log \gamma}$ time steps k.

Optimistic Agents for General Environments

- Generalization to stochastic environments: Likelihood criterion:
  Exclude ν from $\mathcal{M}_{k-1}$ if $\nu(x_{<k}|y_{<k}) < \varepsilon_k \cdot \max_{\nu \in \mathcal{M}} \nu(x_{<k}|y_{<k})$. [SH12]

- Generalization to compact classes $\mathcal{M}$:
  Replace $\mathcal{M}$ by centers of finite ε-cover of $\mathcal{M}$ in def. of π°.   [SH12]

- Use decreasing $\varepsilon_k \to 0$ to get self-optimizingness.

- There are non-compact classes for which self-optimizingness is impossible to achieve.   [Ors10]

- Weaker self-optimizingness in Cesaro sense possible
  by starting with finite subset $\mathcal{M}_0 \subset \mathcal{M}$
  and adding environments ν from $\mathcal{M}$ over time to $\mathcal{M}_k$.   [SH15]

- Fazit: There exist (weakly) self-optimizing policies for arbitrary (separable) /compact $\mathcal{M}$.

Optimism in MDPs

- Let $\mathcal{M}$ be the class of all MDPs with $|\mathcal{S}| < \infty$ states and $|\mathcal{A}| < \infty$ actions and geometric discount γ.

- Then $\mathcal{M}$ is continuous but compact
  $\implies$ π° is self-optimizing by previous slide.

- But much better polynomial error bounds in this case possible:

Theorem 9.14 (PACMDP bound) $V_{k\gamma}^{\pi^\circ\mu} \leq V_{k\gamma}^{*\mu} - \varepsilon$ for at most $\tilde{O}(\frac{|\mathcal{S}|^2|\mathcal{A}|}{\varepsilon^2(1-\gamma)^3} \log \frac{1}{\delta})$ time steps k with probability $1 - \delta$.   [LH12]

9.4 Discussion: Contents

- Summary

- Exercises

- Literature

Summary - Bayesian Agents

- Setup: Agents acting in general probabilistic environments with reinforcement feedback.

- Assumptions: True environment μ belongs to a known class of environments $\mathcal{M}$, but is otherwise unknown.

- Results: The Bayes-optimal policy p^ξ based on the Bayes-mixture $\xi = \sum_{\nu \in \mathcal{M}} w_\nu \nu$ is Pareto-optimal and self-optimizing if $\mathcal{M}$ admits self-optimizing policies.

- Application: The class of ergodic MDPs admits self-optimizing policies.

Summary - Discounting

- Discounting: Considering future values and the right discounting γ leads to more meaningful agents and results.

- Learn: The combined conditions $\Gamma_k < \infty$ and $\frac{\gamma_{k+1}}{\gamma_k} \to 1$ allow a consistent self-optimizing Bayes-optimal policy based on mixtures.

- In particular: Policy p^ξ with unbounded effective horizon is the first purely Bayesian self-optimizing consistent policy for ergodic MDPs.

- Wrong discounting leads to myopic or time-inconsistent policies (bad).

Summary - Variations of AIξ

- Use information gain as a universal choice for the rewards.
  AIξ becomes purely knowledge seeking.

- Real world has traps
  $\implies$ no self-optimizing policy
  $\implies$ need more explorative policies and weaker criteria like ...

- Optimistic agents: Act optimally w.r.t. the most optimistic
  environment until it is contradicted.

Exercises

1. [C15] Prove Pareto-optimality of p^ξ.

2. [C35] Prove Theorem 9.7 (Continuity of discounted value).

3. [C35] Prove Theorem 9.8 (Convergence of universal to true value).

4. [C15ui] Solve [Hut05, Problem 5.2]
   (Absorbing two-state environment)

5. [C25u] Derive the expressions for the effective horizons in Table 9.10.

6. [C30ui] Solve [Hut05, Problem 5.11] (Belief contamination)

7. [C20u] Solve [Hut05, Problem 5.16] (Effect of discounting)

Literature

[BT96]   D. P. Bertsekas and J. N. Tsitsiklis. *Neuro-Dynamic Programming*. Athena Scientific, Belmont, MA, 1996.

[KV86]   P. R. Kumar and P. P. Varaiya. *Stochastic Systems: Estimation, Identification, and Adaptive Control*. Prentice Hall, Englewood Cliffs, NJ, 1986.

[Hut05] M. Hutter. *Universal Artificial Intelligence: Sequential Decisions based on Algorithmic Probability*. Springer, Berlin, 2005. http://www.hutter1.net/ai/uaibook.htm.

[Lat14] T. Lattimore. *Theory of General Reinforcement Learning*. PhD thesis, Research School of Computer Science, Australian National University, 2014.

10 APPROXIMATIONS & APPLICATIONS

- Universal Search

- The Fastest Algorithm (FastPrg)

- Time-Bounded AIXI Model (AIXItl)

- Brute-Force Approximation of AIXI (AIξ)

- A Monte-Carlo AIXI Approximation (MC-AIXI-CTW)

- Feature Reinforcement Learning (ΦMDP)

Approximations & Applications: Abstract

Many fundamental theories have to be approximated for practical use. Since the core quantities of universal induction and universal intelligence are incomputable, it is often hard, but not impossible, to approximate them. In any case, having these "gold standards" to approximate (top$\rightarrow$down) or to aim at (bottom$\rightarrow$up) is extremely helpful in building truly intelligent systems. A couple of universal search algorithms ((adaptive) Levin search, FastPrg, OOPS, Gödel-machine, ...) that find short programs have been developed and applied to a variety of toy problem. The AIXI model itself has been approximated in a couple of ways (AIXItl, Brute Force, Monte Carlo, Feature RL). Some recent applications will be presented.

Towards Practical Universal AI

Goal: Develop efficient general-purpose intelligent agent

- *Additional Ingredients:* *Main Reference (year)*

- Universal search: Schmidhuber (200X) & al.

- Learning: TD/RL Sutton & Barto (1998) & al.

- Information: MML/MDL Wallace, Rissanen

- Complexity/Similarity: Li & Vitanyi (2008)

- Optimization: Aarts & Lenstra (1997)

- Monte Carlo: Fishman (2003), Liu (2002)

10.1 UNIVERSAL SEARCH: CONTENTS

- Blum's Speed-up Theorem and Levin's Theorem.

- The Fastest Algorithm M_{p^*}.

- Applicability of Levin Search and M_{p^*}.

- Time Analysis of M_{p^*}.

- Extension of Kolmogorov Complexity to Functions.

- The Fastest *and* Shortest Algorithm.

- Generalizations.

- Summary & Outlook.

Introduction

- Searching for fast algorithms to solve certain problems is a central and difficult task in computer science.

- Positive results usually come from explicit constructions of efficient algorithms for specific problem classes.

- A wide class of problems can be phrased in the following way:

- Find a fast algorithm computing $f : X \to Y$, where f is a formal specification of the problem depending on some parameter x.

- The specification can be formal (logical, mathematical),
  it need not necessarily be algorithmic.

- Ideally, we would like to have the fastest algorithm, maybe apart from some small constant factor in computation time.

Blum's Speed-up Theorem (Negative Result)

There are problems for which an (incomputable) sequence of speed-improving algorithms (of increasing size) exists, but no fastest algorithm.

[Blum, 1967, 1971]

Levin's Theorem (Positive Result)

Within a (large) constant factor, Levin search is the fastest algorithm to invert a function $g : Y \to X$, if g can be evaluated quickly.

[Levin 1973]

SIMPLE **is as fast as** SEARCH

- SIMPLE: run all programs $p_1 p_2 p_3 \ldots$ on x one step at a time according to the following scheme: p_1 is run every second step, p_2 every second step in the remaining unused steps, ... if $g(p_k(x)) = x$, then output $p_k(x)$ and halt $\Rightarrow time_{\mathsf{SIMPLE}}(x) \leq 2^k time^+_{p_k}(x) + 2^{k-1}$.

- SEARCH: run all p of length less than i for $\lfloor 2^i 2^{-l(p)} \rfloor$ steps in phase $i = 1, 2, 3, \ldots$. $time_{\mathsf{SEARCH}}(x) \leq 2^{K(k)+O(1)} time^+_{p_k}(x), \quad K(k) \ll k$.

- Refined analysis: SEARCH itself is an algorithm with some index $k_{\mathsf{SEARCH}} = O(1)$

  $\Longrightarrow$ SIMPLE executes SEARCH every $2^{k_{\mathsf{SEARCH}}}$-th step

  $\Longrightarrow time_{\mathsf{SIMPLE}}(x) \leq 2^{k_{\mathsf{SEARCH}}} time^+_{\mathsf{SEARCH}}(x)$

  $\Longrightarrow$ SIMPLE and SEARCH have the same asymptotics also in k.

- Practice: SEARCH should be favored because the constant $2^{k_{\mathsf{SEARCH}}}$ is rather large.

Bound for The Fast Algorithm M_{p^*}

- Let $p^* : X \to Y$ be a given algorithm or specification.

- Let p be any algorithm, computing provably the same function as p^*

- with computation time provably bounded by the function $t_p(x)$.

- $time_{t_p}(x)$ is the time needed to compute the time bound $t_p(x)$.

- Then the algorithm M_{p^*} computes $p^*(x)$ in time

$$time_{M_{p^*}}(x) \ \leq \ 5 \cdot t_p(x) + d_p \cdot time_{t_p}(x) + c_p$$

- with constants c_p and d_p depending on p but not on x.

- Neither p, t_p, nor the proofs need to be known in advance for the construction of $M_{p^*}(x)$.

Applicability

- Prime factorization, graph coloring, truth assignments, ... are Problems suitable for Levin search, if we want to find a solution, since verification is quick.

- Levin search cannot decide the corresponding decision problems.

- Levin search cannot speedup matrix multiplication, since there is no faster method to verify a product than to calculate it.

- Strassen's algorithm p' for $n \times n$ matrix multiplication has time complexity $time_{p'}(x) \leq t_{p'}(x) := c \cdot n^{2.81}$.

- The time-bound function (cast to an integer) can, as in many cases, be computed very fast, $time_{t_{p'}}(x) = O(\log^2 n)$.

- Hence, also M_{p*} is fast, $time_{M_{p*}}(x) \leq 5c \cdot n^{2.81} + O(\log^2 n)$, even without known Strassen's algorithm.

- If there exists an algorithm p'' with $time_{p''}(x) \leq d \cdot n^2 \log n$, for instance, then we would have $time_{M_{p*}}(x) \leq 5d \cdot n^2 \log n + O(1)$.

- Problems: Large constants c, c_p, d_p.

The Fast Algorithm M_{p^*}

$\boxed{M_{p^*}(x)}$

Initialize the shared variables

$L := \{\}, \quad t_{fast} := \infty, \quad p_{fast} := p^*.$

Start algorithms A, B, and C

in parallel with 10%, 10% and 80%

computational resources, respectively.

$\boxed{A}$

Run through all proofs.

if a proof proves for some (p, t) that

$p(\cdot)$ is equivalent to (computes) $p^*(\cdot)$

and has time-bound $t(\cdot)$

then add (p, t) to L.

$\boxed{B}$

Compute all $t(x)$ in parallel

for all $(p, t) \in L$ with

relative computation time $2^{-\ell(p)-\ell(t)}$.

if for some t, $t(x) < t_{fast}$,

then $t_{fast} := t(x)$ and $p_{fast} := p$.

continue

$\boxed{C}$

for k:=1,2,4,8,16,32,... do

run current p_{fast} for k steps

(without switching).

if p_{fast} halts in less than k steps,

then print result and abort A, B and C.

else continue with next k.

Fictitious Sample Execution of M_{p^*}
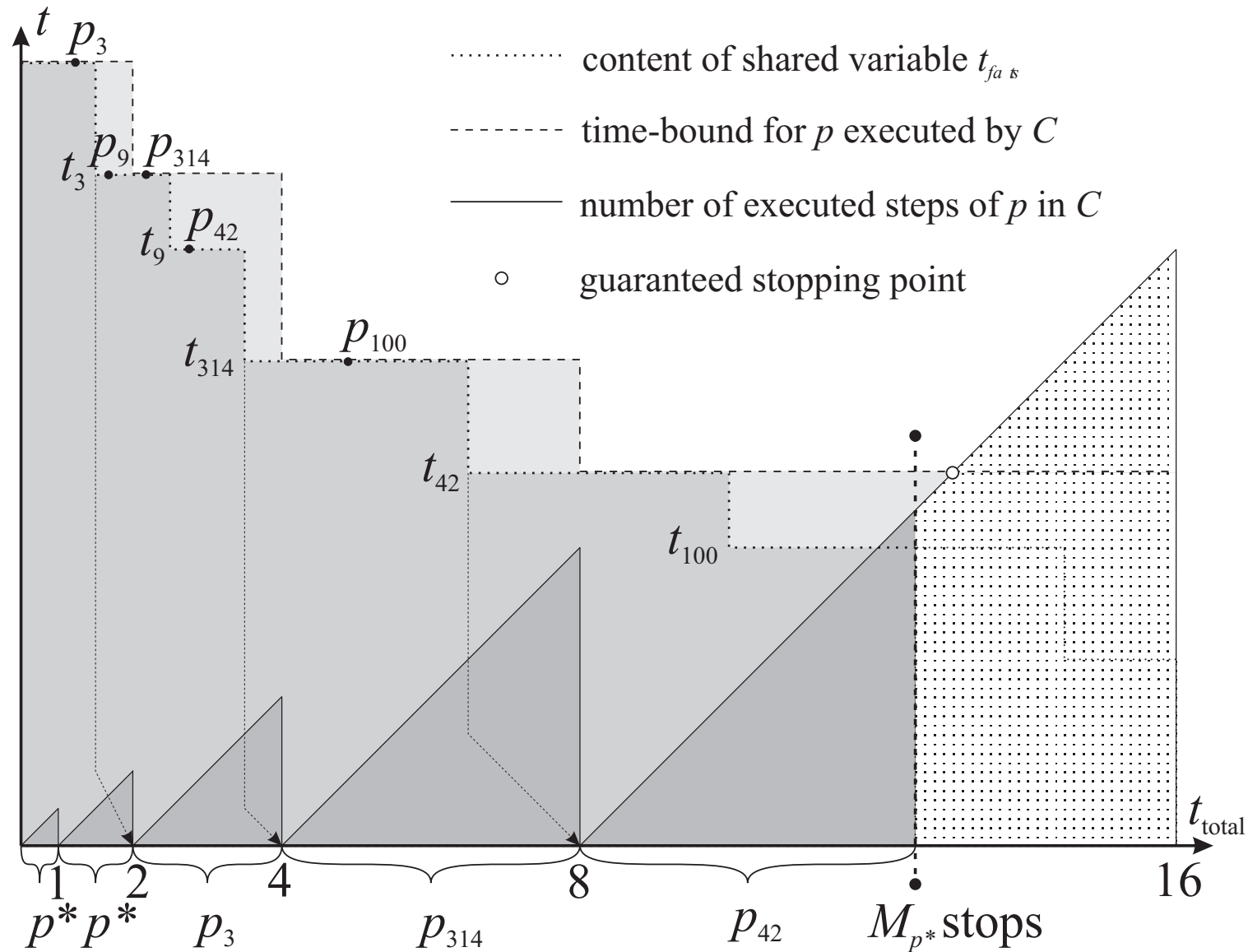
Time Analysis

$$T_A \leq \frac{1}{10\%} \cdot 2^{\ell(proof(p'))+1} \cdot O(\ell(proof(p'))^2)$$

$$T_B \leq T_A + \frac{1}{10\%} \cdot 2^{\ell(p')+\ell(t_{p'})} \cdot time_{t_{p'}}(x)$$

$$T_C \leq \begin{cases} 4T_B & \text{if } C \text{ stops not using } p' \text{ but on some earlier program} \\ \frac{1}{80\%} 4t_{p'} & \text{if } C \text{ computes } p'. \end{cases}$$

$$time_{M_{p^*}}(x) = T_C \leq 5 \cdot t_p(x) + d_p \cdot time_{t_p}(x) + c_p$$

$$d_p = 40 \cdot 2^{\ell(p)+\ell(t_p)}, \quad c_p = 40 \cdot 2^{\ell(proof(p))+1} \cdot O(\ell(proof(p)^2)$$

Kolmogorov Complexity

Kolmogorov Complexity is a universal notion of the information content of a string. It is defined as the length of the shortest program computing string x.

$$K(x) \;:=\; \min_p\{\ell(p) : U(p) = x\}$$

[Kolmogorov 1965 and others]

Universal Complexity of a Function

The length of the shortest program provably equivalent to p^*

$$K''(p^*) \;:=\; \min_p\{\ell(p) : \text{a proof of } [\forall y : u(p,y) = u(p^*,y)] \text{ exists}\}$$

[Hut02]

K and K'' can be approximated from above (are co-enumerable), but not finitely computable. The provability constraint is important.

The Fastest and Shortest Algorithm for p^*

Let p^* be a given algorithm or formal specification of a function.

There exists a program $\tilde{p}$, equivalent to p^*, for which the following holds

$$i) \quad \ell(\tilde{p}) \qquad \leq K''(p^*) + O(1)$$

$$ii) \quad time_{\tilde{p}}(x) \leq 5 \cdot t_p(x) + d_p \cdot time_{t_p}(x) + c_p$$

where p is any program provably equivalent to p^* with computation time provably less than $t_p(x)$. The constants c_p and d_p depend on p but not on x.   [Hut02]

Proof

Insert the shortest algorithm p' provably equivalent to p^* into M, that is
$\tilde{p} := M_{p'} \qquad \Rightarrow \qquad l(\tilde{p}) = \ell(p') + O(1) = K''(p^*) + O(1)$.

Generalizations

- If p^* has to be evaluated repeatedly, algorithm A can be modified to remember its current state and continue operation for the next input (A is independent of x!). The large offset time c_p is only needed on the first call.

- M_{p^*} can be modified to handle i/o streams, definable by a Turing machine with monotone input and output tapes (and bidirectional working tapes) receiving an input stream and producing an output stream.

- The construction above also works if time is measured in terms of the current output rather than the current input x (e.g. for computing π).

Summary

- Under certain provability constraints, M_{p^*} is the asymptotically fastest algorithm for computing p^* apart from a factor 5 in computation time.

- The fastest program computing a certain function is also among the shortest programs provably computing this function.

- To quantify this statement we defined a novel natural measure for the complexity of a function, related to Kolmogorov complexity.

- The large constants c_p and d_p seem to spoil a direct implementation of M_{p^*}.

- On the other hand, Levin search has been successfully extended and applied even though it suffers from a large multiplicative factor [Schmidhuber 1996-2004].

Outlook

- More elaborate theorem-provers could lead to smaller constants.

- Transparent or holographic proofs allow under certain circumstances an exponential speed up for checking proofs. [Babai et al. 1991]

- Will the ultimate search for asymptotically fastest programs typically lead to fast or slow programs for arguments of practical size?

10.2 APPROXIMATIONS & APPLICATIONS OF AIXI: CONTENTS

- Time-Bounded AIXI Model (AIXItl)

  (theoretical guarantee)

- Brute-Force Approximation of AIXI (AIξ)

  (application to 2×2 matrix games)

- A Monte-Carlo AIXI Approximation (MC-AIXI-CTW)

  (application to mazes, tic-tac-toe, pacman, poker)

Computational Issues

- If $\mathcal{X}$, $\mathcal{Y}$, m, $\mathcal{M}$ finite, then ξ and p^ξ (theoretically) computable.

- ξ and hence p^ξ incomputable for infinite $\mathcal{M}$, as for Solomonoff's prior ξ_U.

- Computable approximations to ξ_U:
  Time bounded Kolmogorov complexity Kt or K^t.
  Time bounded universal prior like speed prior S [Schmidhuber:02].

- Even for efficient approximation of ξ_U, exponential (in m) time is needed for evaluating the expectimax tree in V_ξ^*.

- Additionally perform Levin search through policy space, similarly to OOPS+AIXI [Schmidhuber:02].

- Approximate V_ξ^* directly: AIXItl [Hutter:00].

Computability and Monkeys

SPξ and AIξ are not really uncomputable (as often stated) but ...

$\dot{y}_k^{AI\xi}$ is only asymptotically computable/approximable with slowest possible convergence.

Idea of the typing monkeys:

- Let enough monkeys type on typewriters or computers, eventually one of them will write Shakespeare or an AI program.

- To pick the right monkey by hand is cheating, as then the intelligence of the selector is added.

- Problem: How to (algorithmically) select the right monkey.

The Time-bounded AIXI Model

- Let p be any (extended chronological self-evaluating) policy

- with length $\ell(p) \leq l$ and computation time per cycle $t(p) \leq t$

- for which there exists a proof of length $\leq l_P$ that p is a valid approximation of (not overestimating) its true value $V_M^p \equiv \Upsilon(p)$.

- AIXItl selects such p with highest self-evaluation.

Optimality of AIXItl

- AIXItl depends on l,t and l_P but not on knowing p.

- It is effectively more or equally intelligent

  w.r.t. intelligence order relation $\succeq^c$ than any such p.

- Its size is $\ell(p^{best}) = O(\log(l \cdot t \cdot l_P))$.

- Its setup-time is $t_{setup}(p^{best}) = O(l_P^2 \cdot 2^{l_P})$.

- Its computation time per cycle is $t_{cycle}(p^{best}) = O(2^l \cdot t)$.

Outook

- Adaptive Levin-Search (Schmidhuber 1997)

- The Optimal Ordered Problem Solver (Schmidhuber 2004) (has been successfully applied to Mazes, towers of hanoi, robotics, ...)

- The Gödel Machine (Schmidhuber 2007)

- Related fields: Inductive Programming

Brute-Force Approximation of AIXI

- **Truncate expectimax tree** depth to a small fixed lookahead h.
  Optimal action computable in time $|\mathcal{Y}\times\mathcal{X}|^h\times$ time to evaluate ξ.

- Consider mixture over **Markov Decision Processes** (MDP) only, i.e.
  $\xi(x_{1:m}|y_{1:m}) = \sum_{\nu\in\mathcal{M}} w_\nu \prod_{t=1}^m \nu(x_t|x_{t-1}y_t)$. Note: ξ is not MDP

- Choose **uniform prior** over w_μ.
  Then $\xi(x_{1:m}|y_{1:m})$ can be computed in linear time.

- Consider (approximately) Markov problems
  with very **small action and perception space**.

- **Example application:** 2×2 Matrix Games like Prisoner's Dilemma,
  Stag Hunt, Chicken, Battle of Sexes, and Matching Pennies. [PH06]

AIXI Learns to Play 2×2 Matrix Games

- Repeated prisoners dilemma.

- Game unknown to AIXI.
  Must be learned as well

- AIXI behaves appropriately.

| Loss matrix	cooperates	defects
cooperates	=0.3 years	=1 year
defects	free	=0.7 years
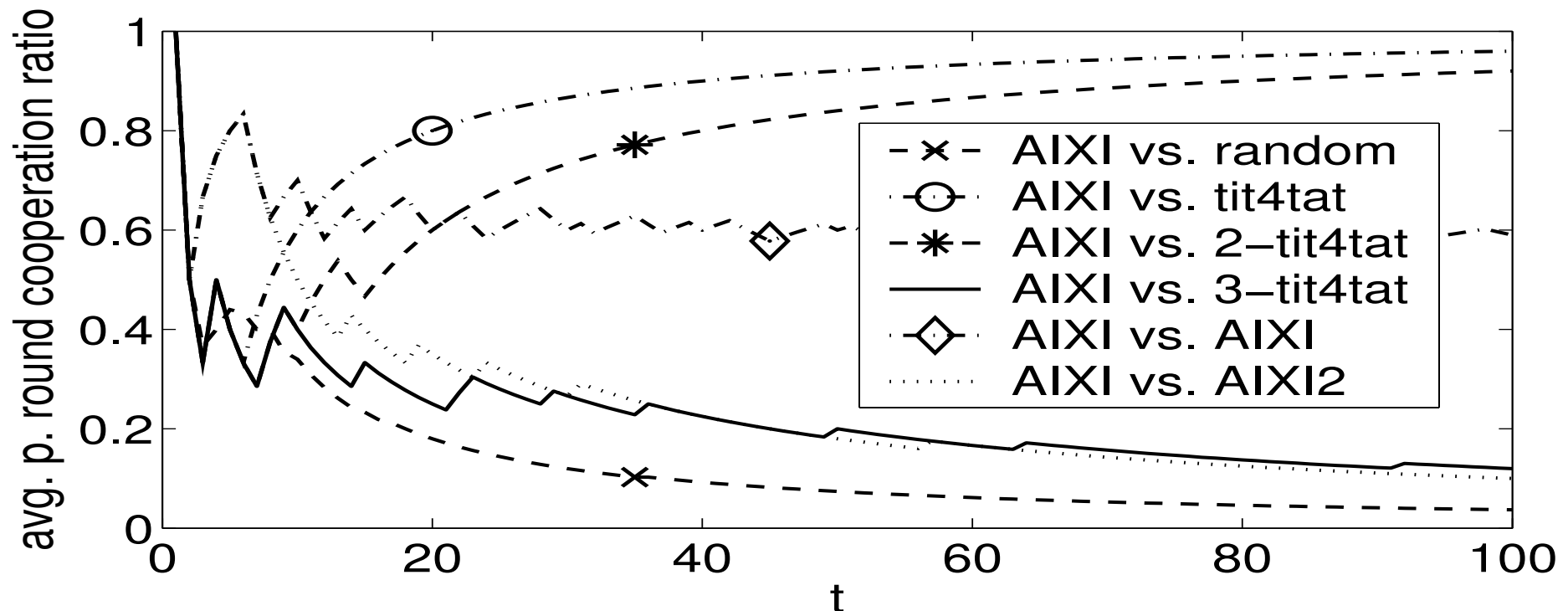
A Monte-Carlo AIXI Approximation

Consider class of Variable-Order Markov Decision Processes.

The Context Tree Weighting (CTW) algorithm can efficiently mix
(exactly in essentially linear time) all prediction suffix trees.

Monte-Carlo approximation of expectimax tree:

Upper Confidence Tree (UCT) algorithm:
- Sample observations from CTW distribution.
- Select actions with highest upper confidence bound.
- Expand tree by one leaf node (per trajectory).
- Simulate from leaf node further down using (fixed) playout policy.
- Propagate back the value estimates for each node.
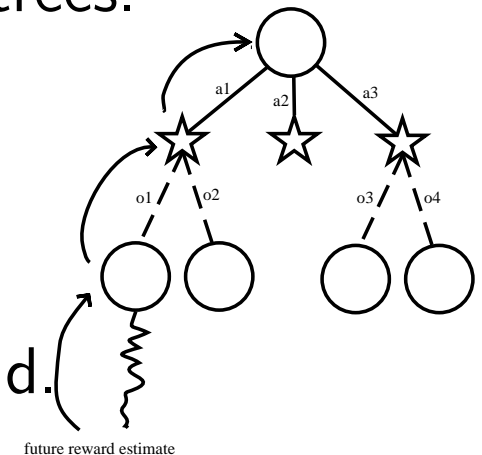
Repeat until timeout. [VNH$^+$11]

Guaranteed to converge to exact value.

Extension: Predicate CTW not based on raw obs. but features thereof.

Monte-Carlo AIXI Applications

without providing any domain knowledge, the same agent is
able to self-adapt to a diverse range of interactive environments.

[VNH$^+$11]

www.youtube.com/watch?v=yfsMHtmGDKE

- - - Optimal
- Cheese Maze
- Tiger
- 4x4 Grid
- TicTacToe
- Biased RPS
- Kuhn Poker
- Pacman

Normalised Average Reward per Cycle

Experience
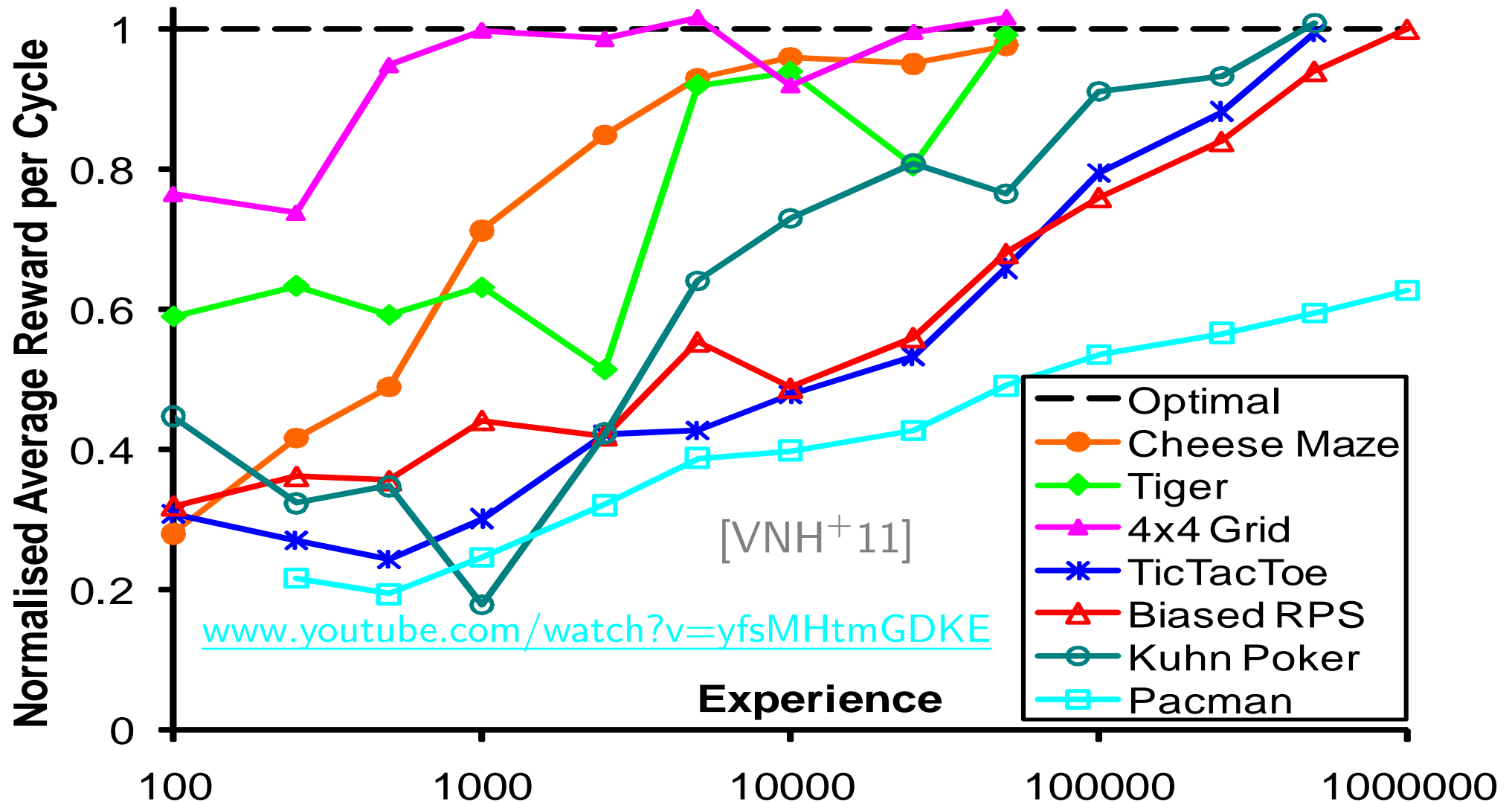
Extensions of MC-AIXI-CTW [VSH12]

- Smarter than random playout policy, e.g. learnt CTW policy.

- Extend the model class to improve general prediction ability. However, not so easy to do this in a comput. efficient manner.

- Predicate CTW: Context is vector of (general or problem-specific) predicate=feature=attribute values.

- Convex Mixing of predictive distributions.
  Competitive guarantee with respect to the best fixed set of weights.

- Switching: Enlarge base class by allowing switching between distr. Can compete with best rarely changing sequence of models.

- Improve underlying KT Est.: Adaptive KT, Window KT, KT0, SAD

- Partition Tree Weighting technique for piecewise stationary sources with breaks at/from a binary tree hierarchy.

- Mixtures of factored models such as quad-trees for images [GBVB13]

- Avoid expensive MCTS by direct compression-based value estimation. [VBH$^+$15]

10.3 Feature Reinforcement Learning: Contents

- **Markov Decision Processes (MDPs)**

- **The Main Idea: Map Real Problem to MDP**

- **Criterion to Evaluate/Find/Learn Map the Automatically**

- **Algorithm & Results**

Feature Reinforcement Learning (FRL)

Goal: Develop efficient general purpose intelligent agent.   [Hut09b]

State-of-the-art: (a) AIXI: Incomputable theoretical solution.
(b) MDP: Efficient limited problem class.
(c) POMDP: Notoriously difficult. (d) PSRs: Underdeveloped.

Idea: ΦMDP reduces real problem to MDP automatically by learning.

Accomplishments so far: (i) Criterion for evaluating quality of reduction.
(ii) Integration of the various parts into one learning algorithm. [Hut09c]
(iii) Generalization to structured MDPs (DBNs).   [Hut09a]
(iv) Theoretical and experimental investigation.   [SH10, DSH12, Ngu13]

ΦMDP is promising path towards the grand goal & alternative to (a)-(d)

Problem: Find reduction Φ efficiently (generic optimization problem?)

Markov Decision Processes (MDPs)

a computationally tractable class of problems

- **MDP Assumption:** State $s_t := o_t$ and r_t are probabilistic functions of o_{t-1} and a_{t-1} only.

  Example MDP

- **Further Assumption:**
  State=observation space $\mathcal{S}$ is finite and small.

- **Goal:** Maximize long-term expected reward.

- **Learning:** Probability distribution is unknown but can be learned.

- **Exploration:** Optimal exploration is intractable
  but there are polynomial approximations.

- **Problem:** Real problems are not of this simple form.
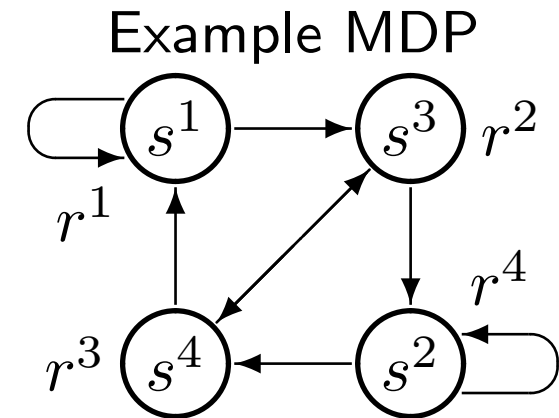
Map Real Problem to MDP

Map history $h_t := o_1 a_1 r_1 ... o_{t-1}$ to state $s_t := \Phi(h_t)$, for example:

Games: Full-information with static opponent: $\Phi(h_t) = o_t$.

Classical physics: Position+velocity of objects = position at two time-slices: $s_t = \Phi(h_t) = o_t o_{t-1}$ is (2nd order) Markov.

I.i.d. processes of unknown probability (e.g. clinical trials $\simeq$ Bandits), Frequency of obs. $\Phi(h_n) = (\sum_{t=1}^{n} \delta_{o_t o})_{o \in \mathcal{O}}$ is sufficient statistic.
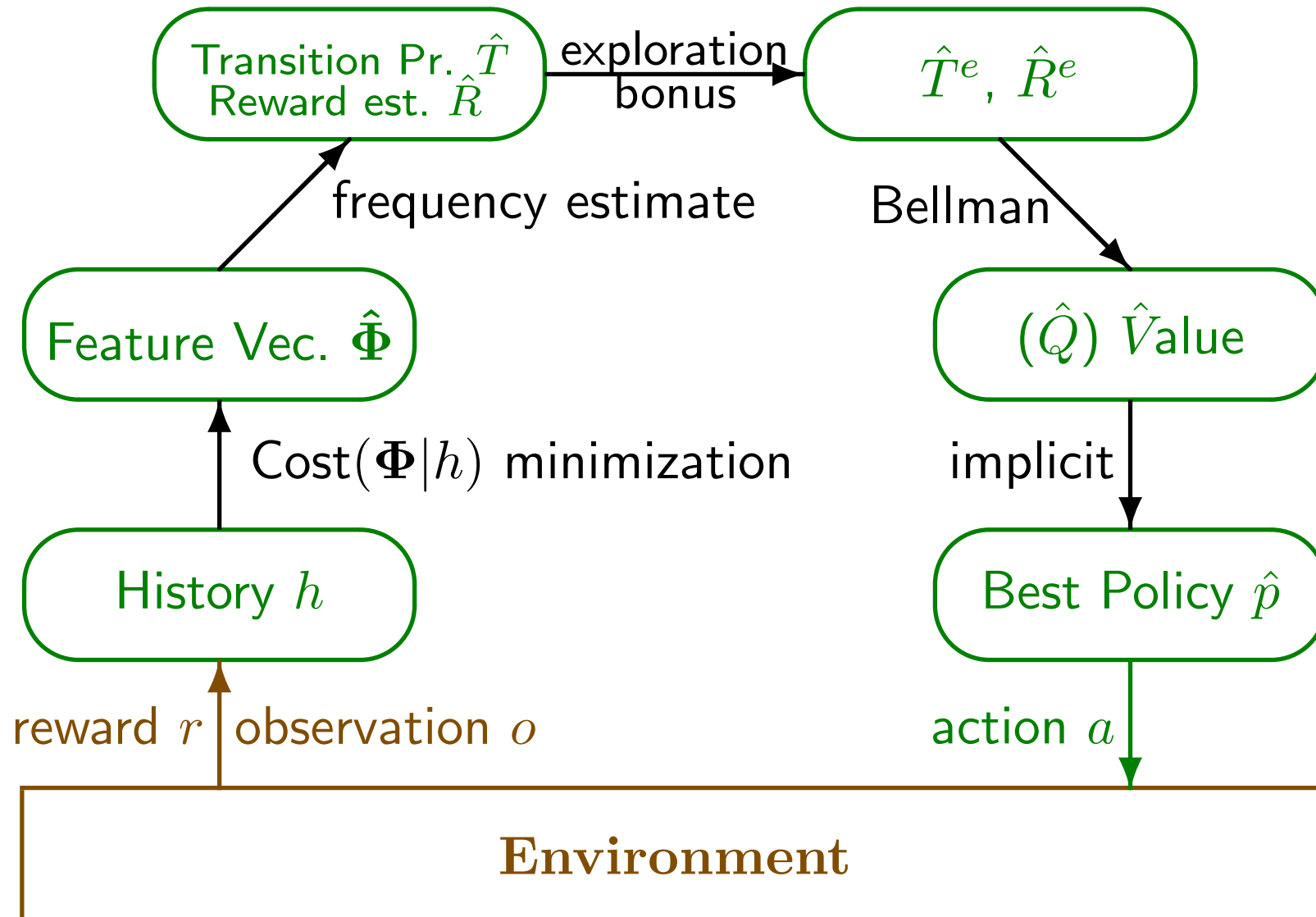
Identity: $\Phi(h) = h$ is always sufficient, but not learnable.

Find/Learn Map Automatically

$$\Phi^{best} := \arg\min_\Phi \mathsf{Cost}(\Phi|h_t)$$

- What is the best map/MDP? (i.e. what is the right Cost criterion?)

- Is the best MDP good enough? (i.e. is reduction always possible?)

- How to find the map Φ (i.e. minimize Cost) efficiently?

Φ**MDP: Computational Flow**

Φ**MDP Results**

- **Theoretical guarantees:** Asymptotic consistency. [SH10]

- **Example Φ-class:** As Φ choose class of suffix trees as in CTW.

- **How to find/approximate Φ^{best}:**
  - Exhaustive search for toy problems [Ngu13]
  - Monte-Carlo (Metropolis-Hastings / Simulated Annealing) for approximate solution [NSH11]
  - Exact "closed-form" by CTM similar to CTW [NSH12]

- **Experimental results:** Comparable to MC-AIXI-CTW [NSH12]

- **Extensions:**
  - Looping suffix trees for long-term memory [DSH12]
  - Structured/Factored MDPs (Dynamic Bayesian Networks) [Hut09a]

Literature

[Hut02] M. Hutter. *The fastest and shortest algorithm for all well-defined problems.* International Journal of Foundations of Computer Science, 13(3):431–443, 2002.

[Hut01] M. Hutter. *Towards a universal theory of artificial intelligence based on algorithmic probability and sequential decisions.* In Proc. 12th European Conf. on Machine Learning (ECML-2001), volume 2167 of *LNAI*, pages 226–238, Freiburg, 2001. Springer, Berlin.

[Sch07] J. Schmidhuber. *The new AI: General & sound & relevant for physics.* In Artificial General Intelligence, pages 175–198. Springer, 2007.

[Hut09] M. Hutter. *Feature reinforcement learning: Part I: Unstructured MDPs.* Journal of Artificial General Intelligence, 1:3–24, 2009.

[VNH+11] J. Veness, K. S. Ng, M. Hutter, W. Uther, and D. Silver. *A Monte Carlo AIXI approximation. Journal of Artificial Intelligence Research*, 40:95–142, 2011. http://dx.doi.org/10.1613/jair.3125

[PH06] J. Poland and M. Hutter. *Universal learning of repeated matrix games.* In Proc. 15th Annual Machine Learning Conf. of Belgium and The Netherlands (Benelearn'06), pages 7–14, Ghent, 2006.

11 DISCUSSION

- What has been achieved?

- Universal AI in perspective

- Miscellaneous considerations

- Outlook and open questions

- Philosophical issues

Discussion: Abstract

The course concludes by critically reviewing what has been achieved and discusses some otherwise unmentioned topics of general interest. We summarize the AIXI model and compare various learning algorithms along various dimensions. We continue with an outlook on further research. Furthermore, we collect and state all explicit or implicit assumptions, problems and limitations of AIXI(tl).

The dream of creating artificial devices that reach or outperform human intelligence is an old one, so naturally many philosophical questions have been raised: weak/strong AI, Gödel arguments, the mind-body and the free will problem, consciousness, and various thought experiments. Furthermore, the Turing test, the (non)existence of objective probabilities, non-computable physics, the number of wisdom, and finally ethics, opportunities, and risks of AI are briefly discussed.

11.1   What has been Achieved: Contents

- Recap of Universal AI and AIXI

- Involved Research Fields

- Overall and Major Achievements

Overall Achievement

- Developed the mathematical foundations of artificial intelligence.

- Developed a theory for rational agents
  acting optimally in any environment.

- This was not an easy task since intelligence has many
  (often ill-defined) facets.

Universal Artificial Intelligence (AIXI)

$$\|\qquad\qquad\qquad\qquad\qquad\qquad\|$$

Decision Theory $\quad = \quad$ Probability $+$ Utility Theory

$$+\qquad\qquad\qquad\qquad\qquad\qquad+$$

Universal Induction $\quad = \quad$ Ockham $+$ Bayes $+$ Turing

Involved Scientific Areas

- reinforcement learning
- information theory
- theory of computation
- Bayesian statistics
- sequential decision theory

- adaptive control theory
- Solomonoff induction
- Kolmogorov complexity
- Universal search
- and many more

The AIXI Model in one Line

complete & essentially unique & limit-computable

$$\text{AIXI:} \quad a_k \; := \; \arg\max_{a_k} \sum_{o_k r_k} ... \max_{a_m} \sum_{o_m r_m} [r_k + ... + r_m] \sum_{p\,:\,U(p,a_1..a_m)=o_1 r_1..o_m r_m} 2^{-\ell(p)}$$

$a$ction, $r$eward, $o$bservation, Universal TM, $p$rogram, k=now

AIXI is an elegant mathematical theory of AI

Claim: AIXI is the most intelligent environmental independent, i.e.
universally optimal, agent possible.

Proof: For formalizations, quantifications, and proofs, see [Hut05].

Applications: Robots, Agents, Games, Optimization, Supervised
Learning, Sequence Prediction, Classification, ...

Issues in AI and how UAI solves them

Kolmogorov complexity:

- generalization
- associative learning
- transfer learning [Mah09]
- knowledge representation
- abstraction
- similarity [CV05]
- regularization, bias-variance [Wal05]

Bayes:

- exploration-exploitation
- learning

History-based:

- partial observability
- non-stationarity
- long-term memory
- large state space

Expectimax:

- planning

UAI deals with these issues in a general and optimal way

Major Achievements 1

Philosophical & mathematical & computational

foundations of universal induction based on

- Occam's razor principle,

- Epicurus' principle of multiple explanations,

- subjective versus objective probabilities,

- Cox's axioms for beliefs,

- Kolmogorov's axioms of probability,

- conditional probability and Bayes' rule,

- Turing machines,

- Kolmogorov complexity,

- culminating in universal Solomonoff induction.

Major Achievements 2
Miscellaneous

- Convergence and optimality results
  for (universal) Bayesian sequence prediction.

- Sequential decision theory in a very general form in which actions
  and perceptions may depend on arbitrary past events (AIμ).

- Kolmogorov complexity with approximations (MDL) and
  applications to clustering via the Universal Similarity Metric.

- Universal intelligence measure and order relation regarding which
  AIXI is the most intelligent agent.

Major Achievements 3
Universal Artificial Intelligence (AIXI)

- Unification of sequential decision theory and Solomonoff's theory of universal induction, both optimal in their own domain, to the optimal universally intelligent agent AIXI.

- Categorization of environments.

- Universal discounting and choice of the horizon

- AIXI/AIξ is self-optimizing and Pareto optimal

- AIXI can deal with a number of important problem classes, including sequence prediction, strategic games, function minimization, and supervised learning.

Major Achievements 4
Approximations & Applications

- **Universal search:** Levin search, FastPrg, OOPS, Gödel machine, ...

- **Approximations:** AIXItl, AIξ, MC-AIXI-CTW, ΦMDP.

- **Applications:** Prisoners Dilemma and other 2×2 matrix games, Toy Mazes, TicTacToe, Rock-Paper-Scissors, Pacman, Kuhn-Poker, ...

- **Fazit:** Achievements 1-4 show that artificial intelligence can be framed by an elegant mathematical theory. Some progress has also been made toward an elegant $computational$ theory of intelligence.

11.2 Universal AI in Perspective: Contents

- Aspects of AI included in AIXI

- Emergent Properties of AIXI

- Intelligent Agents in Perspective

- Properties of Learning Algorithms

- Machine Intelligence Tests & Definitions

- Common Criticisms

- General Murky & Quirky AI Questions

Connection to (AI) SubFields

- **Agents:** The UAIs (AIXI,ΦMDP,...) are (single) agents.
- **Utility theory:** goal-oriented agent.
- **Probability theory:** to deal with uncertain environment.
- **Decision theory:** agent that maximizes utility/reward.
- **Planning:** in expectimax tree and large DBNs.
- **Information Theory:** Core in defining and analyzing UAIs.
- Reinforcement **Learning:** via Bayes-mixture and PAC-MDP to deal with unknown world.
- **Knowledge Representation:** In compressed history and features Φ.
- **Reasoning:** To improve compression/planning/search/... algorithms.
- **Logic:** For proofs in AIXItl and soph. features in ΦDBN.
- **Complexity Theory:** In AIXItl and PAC-MDP. We need poly-time and ultimately linear-time approx. algorithms for all building blocks.
- Heuristic **Search & Optimization:** Approximating Solomonoff by compressing history, and minimizing $\text{Cost}(\Phi, \text{Structure}|h)$
- **Interfaces: Robotics, Vision, Language:** In theory learnable from scratch. In practice engineered pre-&post-processing.

Aspects of Intelligence
are all(?) either directly included in AIXI or are emergent

TRAIT OF INTELL.	HOW INCLUDED IN AIXI
reasoning	to improve internal algorithms (emergent)
creativity	exploration bonus, randomization, ...
association	for co-compression of similar observations
generalization	for compression of regularities
pattern recognition	in perceptions for compression
problem solving	how to get more reward
memorization	storing historic perceptions
planning	searching the expectimax tree
achieving goals	by optimal sequential decisions
learning	Bayes-mixture and PAC-MDP
optimization	compression and expectimax (Cost() in ΦMDP)
self-preservation	by coupling reward to robot components
vision	observation=camera image (emergent)
language	observation/action = audio-signal (emergent)
motor skills	action = movement (emergent)
classification	by compression (partition from Φ in ΦMDP)
induction	Universal Bayesian posterior (Ockham's razor)
deduction	Correctness proofs in AIXItl

Other Aspects of the Human Mind

- Conciousness

- Self-awareness

- Sentience

- Emotions
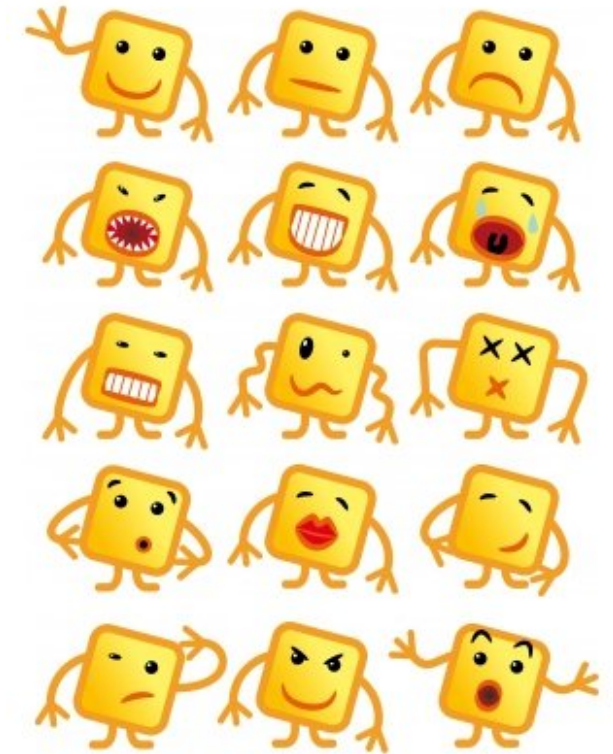
If these qualia are relevant for rational decision making,
then they should be emergent traits of AIXI too.

Intelligent Agents in Perspective
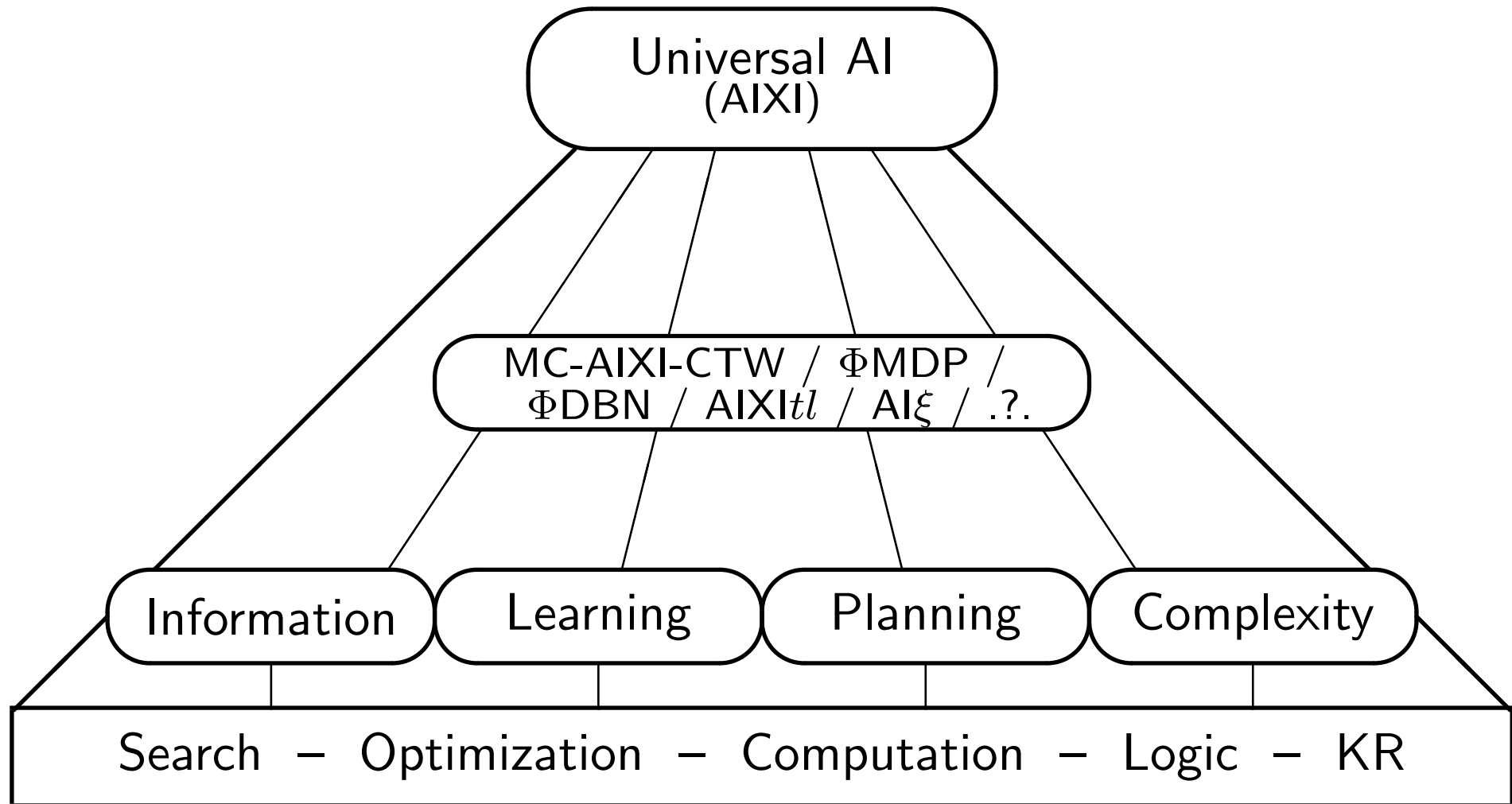
Agents = General Framework, Interface = Robots,Vision,Language

Issues in RL and how AIXI solves them

Kolmogorov complexity:

- generalization
- associative learning
- transfer learning [Mah09]
- knowledge representation
- abstraction
- similarity [CV05]
- regularization, bias-variance [Wal05]

Bayes:

- exploration-exploitation
- learning

History-based:

- partial observability
- non-stationarity
- long-term memory
- large state space

Expectimax:

- planning

AIXI deals with these issues in a general and optimal way

Properties of Learning Algorithms
Comparison of AIXI to Other Approaches

Algorithm	time efficient	data efficient	explo-ration	conver-gence	global optimum	genera-lization	POMDP	learning	active
Value/Policy iteration	yes/no	yes	–	YES	YES	NO	NO	NO	yes
TD w. func.approx.	no/yes	NO	NO	no/yes	NO	YES	NO	YES	YES
Direct Policy Search	no/yes	YES	NO	no/yes	NO	YES	no	YES	YES
Logic Planners	yes/no	YES	yes	YES	YES	no	no	YES	yes
RL with Split Trees	yes	YES	no	YES	NO	yes	YES	YES	YES
Pred.w. Expert Advice	yes/no	YES	–	YES	yes/no	yes	NO	YES	NO
OOPS	yes/no	no	–	yes	yes/no	YES	YES	YES	YES
Market/Economy RL	yes/no	no	NO	no	no/yes	yes	yes/no	YES	YES
SPXI	no	YES	–	YES	YES	YES	NO	YES	NO
AIXI	NO	YES	YES	yes	YES	YES	YES	YES	YES
AIXItl	no/yes	YES	YES	YES	yes	YES	YES	YES	YES
MC-AIXI-CTW	yes/no	yes	YES	YES	yes	NO	yes/no	YES	YES
Feature RL	yes/no	YES	yes	yes	yes	yes	yes	YES	YES
Human	yes	yes	yes	no/yes	NO	YES	YES	YES	YES

Machine Intelligence Tests & Definitions

★= yes, ·= no, ●= debatable, ? = unknown. Intelligence Test	Valid	Informative	Wide Range	General	Dynamic	Unbiased	Fundamental	Formal	Objective	Fully Defined	Universal	Practical	Test vs. Def.
Turing Test	●	·	·	·	●	·	·	·	·	●	·	●	T
Total Turing Test	●	·	·	·	●	·	·	·	·	●	·	·	T
Inverted Turing Test	●	●	·	·	●	·	·	·	·	●	·	●	T
Toddler Turing Test	●	·	·	·	●	·	·	·	·	·	·	●	T
Linguistic Complexity	●	★	●	·	·	·	·	●	●	·	●	●	T
Text Compression Test	●	★	★	●	·	●	●	★	★	★	●	★	T
Turing Ratio	●	★	★	★	?	?	?	?	?	·	?	?	T/D
Psychometric AI	★	★	●	★	?	●	·	●	●	●	·	●	T/D
Smith's Test	●	★	★	●	·	?	★	★	★	·	?	●	T/D
C-Test	●	★	★	●	·	★	★	★	★	★	★	★	T/D
Universal $\Upsilon(\pi)$, AIXI	★	★	★	★	★	★	★	★	★	★	★	·	D

Common Criticisms

- **AIXI is obviously wrong.**
  (intelligence cannot be captured in a few simple equations)

- **AIXI is obviously correct.** (everybody already knows this)

- **Assuming that the environment is computable is too strong.**

- **All standard objections to strong AI also apply to AIXI.**
  (free will, lookup table, Lucas/Penrose Gödel argument)

- **AIXI doesn't deal with X or cannot do X.**
  (X = consciousness, creativity, imagination, emotion, love, soul, etc.)

- **AIXI is not intelligent because it cannot choose its goals.**

- **Universal AI is impossible due to the No-Free-Lunch theorem.**

See [Leg08] for refutations of these and more criticisms.

General Murky & Quirky AI Questions

- Is current mainstream AI research relevant for AGI?

- Are sequential decision and algorithmic probability theory all we need to well-define AI?

- What is (Universal) AI theory good for?

- What are robots good for in AI?

- Is intelligence a fundamentally simple concept?
  (compare with fractals or physics theories)

- What can we (not) expect from super-intelligent agents?

- Is maximizing the expected reward the right criterion?

- Isn't universal learning impossible due to the NFL theorems?

11.3 Miscellaneous Considerations: Contents

- Game Theory and Simultaneous Actions

- Input/Output Spaces

- Specific/Universal/Generic Prior Knowledge

- How AIXI(tl) Deals with Encrypted Information

- Origin of Rewards and Universal Goals

- Mortal Embodied (AIXI) Agent

- Some more Social Questions

- Creativity – An Algorithmic View

- Is Intelligence Simple or Complex?

Game Theory and Simultaneous Actions

Game theory often considers simultaneous actions of both players
(e.g. 2×2 matrix games) (agent and environment in our terminology).

Our approach can simulate this by withholding the environment from
the current agent's output y_k, until x_k has been received by the agent.

Input/Output Spaces

- In our examples: specialized input and output spaces $\mathcal{X}$ and $\mathcal{Y}$.

- In principle: Generic interface, e.g. high-resolution camera / monitor
  / actuators, but then complex vision and control behavior has to be
  learnt too (e.g. recognizing and drawing TicTacToe boards).

- In theory: Any interface can be Turing-reduced to binary $\mathcal{X}$ and $\mathcal{Y}$
  by sequentializing, or embedded into $\mathcal{X} = \mathcal{Y} = \mathbb{N}$.

Prior Knowledge — Specific Solutions

For specific practical problems we usually have extra information about the problem at hand, which could and should be used to guide the forecasting and decisions.

Ways of incorporating prior knowledge:

- Restrict Bayesian mixture ξ_U from all computable environments to those not contradicting our prior knowledge, or soft version:

- Bias weights weights w_ν towards environments that are more likely according to our prior knowledge.

Both can be difficult to realize, since one often has only an informal description of prior facts.

Prior Knowledge — Universal Solution

- Code all prior knowledge in one long binary string $d_{1:\ell}$
  (e.g. a dump of Wikipedia, see H-prize) essentially in any format.

- Provide $d_{1:\ell}$ as first (sequence of) observation to AIXI/Solomonoff,
  i.e. prefix actual observation $x_{<n}$ with $d_{1:\ell}$.

- This also allows to predict short sequences reliably
  (insensitive to choice of UTM).

- This is also how humans are able to agree on predictions based on
  apparently little data, e.g. 1,1,1,1,1,1,?

- Humans can make non-arbitrary predictions given a short sequence
  $x_{<n}$ only iff $M(x_n|d_{1:\ell}x_{<n})$ leads to essentially the same prediction
  for all "reasonable" universal Turing machines U.

Universal=Generic Prior Knowledge

- **Problem 1:** Higher-level knowledge is never 100% sure.
  $\Rightarrow$ No environment (except those inconsistent with bare observations) can be ruled out categorically
  (The world may change completely tomorrow).

- **Problem 2:** Env. μ does not describe the total universe, but only a small fraction, from the subjective perspective of the agent.

- **Problem 3:** Generic properties of the universe like locality, continuity, or the existence of manipulable objects with properties and relations in a manifold may be distorted due to the subjective perspective.

- **Problem 4:** Known generic properties only constitute information of size O(1) and do not help much in theory (but might in practice).

- **On the other hand,** the scientific approach is to simply assume some properties (whether true in real life or not) and analyze the performance of the resulting models.

How AIXI(tl) Deals with Encrypted Information

- De&en-cryption are bijective functions of complexity O(1), and
  Kolmogorov complexity is invariant under such transformations
  $\Rightarrow$ AIXI is immune to encryption. Due its unlimited computational
  resources it can crack any encryption.

- This shows that in general it does not matter how information is
  presented to AIXI.

- But any time-bounded approximation like AIXItl will degrade under
  hard-to-invert encodings.

Origin of Rewards and Universal Goals

- **Where do rewards come from** if we don't (want to) provide them.

- **Human interaction:** reward the robot according to how well it solves the tasks we want it to do.

- **Autonomous:** Hard-wire reward to predefined task:
  E.g. Mars robot: reward = battery level & evidence of water/life.

- Is there something like a **universal goal**?

- **Curiosity-driven learning**   [Sch07]

- **Knowledge seeking agents**   [Ors11, OLH13]

- **Universal (instrumental) values:** survival, spreading, information, rationality, space, time, matter, energy, power, security, truth ?

Mortal Embodied (AIXI) Agent

- **Robot in human society:** reward the robot according to how well it solves the tasks we want it to do, like raising and safeguarding a child. In the attempt to maximize reward, the robot will also maintain itself.

- **Robot w/o human interaction (e.g. on Alpha-Centauri):**
  Some rudimentary capabilities (which may not be that rudimentary at all) are needed to allow the robot to at least survive.
  Train the robot first in safe environment, then let it loose.

- **Drugs (hacking the reward system):**
  No, since long-term reward would be small (death).   but see [OR11]

- **Replication/procreation:** Yes, if AIXI believes that clones or descendants are useful for its own goals (ensure retirement pension).

- **Suicide:** Yes (No), if AIXI expects negative (positive) life-time reward.   [MEH16]

- **Self-Improvement:** Yes, since this helps to increase reward.

- **Manipulation:** Any Super-intelligent robot can manipulate or threaten its teacher to give more reward.

Some more Social Questions

- Attitude: Are pure reward maximizers egoists, *psychopaths*, and/or killers or will they be *friendly* (*altruism* as extended *ego(t)ism*)?

- Curiosity killed the cat and maybe AIXI,   [Sch07, Ors11]
  or is extra reward for curiosity necessary?   [LHS13, LH14]

- Immortality can cause laziness!   [Hut05, Sec.5.7]

- Can self-preservation be learned or need (parts of) it be innate.

  see also [RO11]

- Socializing: How will AIXI interact with another AIXI?

  [Hut09d, Sec.5j],[PH06, LTF16]

Creativity − An Algorithmic View

- **Definition:** the process of producing something original&worthwhile.

- **The process:** combining and modifying existing thoughts or artifacts in novel ways, driven by random choice and filtering out bad results.

- **Analogy:** Ecosystems appear to be creatively designed, but blind evolutionary process was sufficient.

- Solving complex problems requires (apparent) creativity.

- Since AIXI is able to solve complex problems, it will appear creative.

- **Analogy:** Brute-force MiniMax chess programs appear to make (occasionally) creative moves.

- Creativity emerges from long-term reward maximization.

- **Science** $\approx$ finding patterns $\approx$ **Compression**
  is creative process   is formal procedure

- **Exploratory actions** can appear creative.

- **Fazit:** Creativity is just exploration, filtering, and problem solving.

Is Intelligence Simple or Complex?

The AIXI model shows that

> in theory intelligence is a simple concept

> that can be condensed into a few formulas.

But intelligence may be complicated in practice:

- One likely needs to provide special-purpose algorithms ($methods$) from the very beginning to reduce the computational burden.

- Many algorithms will be related to reduce the complexity of the input/output by appropriate pre/postprocessing (vision/language/robotics).

11.4 Outlook and Open Questions: Contents

- Outlook

- Assumptions

- Multi-Agent Setup

- Next Steps

Outlook

- Theory: Prove stronger theoretical performance guarantees for AIXI and AIξ; general ones, as well as tighter ones for special environments μ.

- Scaling AIXI down: Further investigation of the approximations AIXItl, AIξ, MC-AIXI-CTW, ΦMDP, Gödel machine.
  Develop other/better approximations of AIXI.

- Importance of training (sequence):
  To maximize the information content in the reward,
  one should provide a sequence of simple-to-complex tasks to solve,
  with the simpler ones helping in learning the more complex ones,
  and give positive reward to approximately the better half of the
  actions.

Assumptions

- Occam's razor is a central and profound assumption,

  but actually a general prerequisite of science.

- Environment is sampled from a computable probability distribution

  with a reasonable program size on a natural Turing machine.

- Objective probabilities/randomness exist

  and respect Kolmogorov's probability Axioms.

  Assumption can be dropped if world is assumed to be deterministic.

- Using Bayes mixtures as subjective probabilities did not involve any

  assumptions, since they were justified decision-theoretically.

Assumptions (contd.)

- **Maximizing expected lifetime reward sum:**
  Generalization possible but likely not needed.
  (e.g. obtain risk aversion by concave trafo of rewards)

- **Finite action/perception spaces $\mathcal{Y}/\mathcal{X}$:** Likely generalizable to countable spaces (ε-optimal policies), and possibly to continuous ones. but finite is sufficient in practice.

- **Nonnegative rewards:**
  Generalizable to bounded rewards. Should be sufficient in practice.

- **Finite horizon or near-harmonic discounting.**

Attention: All(?) other known approaches to AI implicitly or explicitly make (many) more assumptions.

Multi-Agent Setup – Problem

Consider AIXI in a multi-agent setup interacting with other agents,
in particular consider AIXI interacting with another AIXI.

There are no known theoretical guarantees for this case,
since AIXI-environment is non-computable.

AIXI may still perform well in general multi-agent setups,
but we don't know.

Next Steps

- Address the many open theoretical questions in [Hut05].

- Bridge the gap between (Universal) AI theory and AI practice.

- Explore what role logical reasoning, knowledge representation, vision, language, etc. play in Universal AI.

- Determine the right discounting of future rewards.

- Develop the right nurturing environment for a learning agent.

- Consider embodied agents (e.g. internal$\leftrightarrow$external reward)

- Analyze AIXI in the multi-agent setting.

11.5 Philosophical AI Questions: Contents

- Can machines act or be intelligent or conscious?
  (weak/strong AI, Gödel, mind-body, free will,
   brain dissection, Chinese room, lookup table)

- Turing Test & Its Limitations

- (Non)Existence of Objective Probabilities

- Non-Computable Physics & Brains

- Evolution & the Number of Wisdom

- Ethics and Risks of AI

- What If We Do Succeed?

- Countdown To Singularity

- Three Laws of Robotics

Can Weak AI Succeed?

The argument from disability:

- − A machine can never do X.
- + These claims have been disproven for an increasing # of things X.

The mathematical objection (Lucas 1961, Penrose 1989,1994):

- − No formal system incl. AIs, but only humans can "see" that Gödel's unprovable sentence is true.
- + Lucas cannot consistently assert that this sentence is true.

The argument from informality of behavior:

- − Human behavior is far too complex to be captured by any simple set of rules. Dreyfus (1972,1992) "What computers (still) can't do".
- + Computers already can generalize, can learn from experience, etc.

The Mathematical Objection to Weak AI

Applying Gödel's incompleteness theorem:

- G(F) := "This sentence cannot be proved in the formal axiomatic system F"
- We humans can easily see that G(F) must be true.
- Lucas (1961), Penrose (1989,1994):
  Since any AI is an F, no AI can prove G(F).
- Therefore there are things humans, but no AI system can do.

Counter-argument:

- L := "J.R.Lucas cannot consistently assert that this sentence is true"
- Lucas cannot assert L, but now we can conclude that it is true.
- Lucas is in the same situation as an AI.

Strong AI versus Weak AI

Argument from consciousness:

- – A machine passing the Turing test would not prove that it actually really thinks or is conscious about itself.
- + We do not know whether other humans are conscious about themselves, but it is a polite convention, which should be applied to AIs too.

Biological naturalism:

– Mental states can emerge from neural substrate only.

Functionalism:

+ Only the functionality/behavior matters.

Strong AI: Mind-Body and Free Will

Mind-body problem:

+ Materialist: There exists only the a mortal body.

− Dualist: There also exists an immortal soul.

Free will paradox:

− How can a purely physical mind, governed strictly by physical laws, have free will?

+ By carefully reconstructing our naive notion of free will:
  If it is impossible to predict and tell my next decision,
 then I have effective free will.

Strong AI: Brain Dissection

The "brain in a vat" experiment:
(no) real experience:

+ [see movie *Matrix* for details]

The brain prosthesis experiment:

+ Replacing some neurons in the brain by functionally identical electronic prostheses would neither effect external behavior nor internal experience of the subject.

+ Successively replace one neuron after the other until the whole brain is electronic.
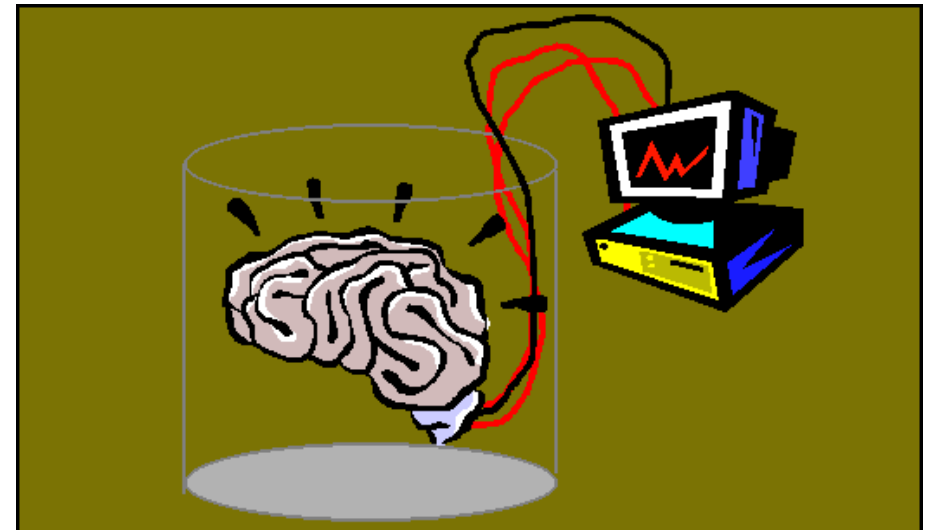
Strong AI: Chinese Room & Lookup Table

Strong AI: Chinese Room & Lookup Table

Assume you have a huge table or rule book containing all answers to all potential questions in the Turing test (say in Chinese which you don't understand).

- − You would pass the Turing test without understanding anything.

- + There is no big enough table.

- + The used rule book is conscious.

- + Analogy: Look, the brain just works according to physical rules without understanding anything.

Strong AI versus Weak AI: Does it Matter?

The phenomenon of consciousness is mysterious, but likely it is not too important whether a machine simulates intelligence or really *is* self aware. Maybe the whole distinction between strong and weak AI makes no sense.

Analogy:

– Natural $\leftrightarrow$ artificial: urea, wine, paintings, thinking.

– Real $\leftrightarrow$ virtual: flying an airplane versus simulator.

Is there a fundamental difference? Should we care?

Turing Test & Its Limitations

Turing Test (1950): If a human judge cannot reliably tell whether a teletype chat is with a machine or a human, the machine should be regarded as intelligent.

Standard objections:
- Tests for humanness, not for intelligence:
  - Some human behavior is unintelligent.
  - Some intelligent behavior is inhuman.
- The test is binary rather than graded.

Real problem: Unlike the Universal Intelligence Measure [LH07] and AIXI, the Turing test involves a human interrogator and, hence, cannot be formalized mathematically, therefore it does also not allow the development of a computational theory of intelligence.

(Non)Existence of Objective Probabilities

- The assumption that an event occurs with some objective probability expresses the opinion that the occurrence of an individual stochastic event has no explanation.

$\Rightarrow$ i.e. the event is inherently impossible to predict for sure.

- One central goal of science is to explain things.

- Often we do not have an explanation (yet) that is acceptable,

- but to say that "something can principally not be explained" means to stop even trying to find an explanation.

$\Rightarrow$ It seems safer, more honest, and more scientific to say that with our current technology and understanding we can only determine (subjective) outcome probabilities.

Objective=InterSubjective Probability

- If a sufficiently large community of people arrive at the same
  subjective probabilities from their prior knowledge,
  one may want to call these probabilities objective.

- Example 1: The outcome of tossing a coin is usually agreed upon to
  be random, but may after all be predicted by taking a close enough
  look.

- Eaxmple 2: Even quantum events may be only pseudo-random
  (Schmidhuber 2002).

- Conclusion: All probabilities are more or less subjective.
  Objective probabilities may actually only be inter-subjective.

Non-Computable Physics & Brains

Non-computable physics (which is not too odd)
could make Turing-computable AI impossible.

At least the world that is relevant for humans seems to be computable,
so non-computable physics can likely be ignored in practice.
(Gödel argument by Penrose&Lucas has loopholes).

Evolution & the Number of Wisdom

The enormous computational power of evolution could have developed
and coded information into our genes,
(a) which significantly guides human reasoning,
(b) cannot efficiently be obtained from scratch (Chaitin 1991).

Cheating solution: Add the information from our genes or brain
structure to any/our AI system.

Ethics and Risks of AI

– People might lose their jobs to automation.

+ So far automation (via AI technology) has created more jobs and wealth than it has eliminated.

– People might have too much (or too little) leisure time

+ AI frees us from boring routine jobs and leaves more time for pretentious and creative things.

– People might lose their sense of being unique.

+ We mastered similar degradations in the past (Galileo, Darwin, physical strength)

+ We will not feel so lonely anymore (cf. SETI)

– People might lose some of their privacy rights.

– The use of AI systems might result in a loss of accountability.

? Who is responsible if a physician follows the advice of a medical expert system, whose diagnosis turns out to be wrong?

What If We Do Succeed?

The success of AI might mean the end of the human race.

- Natural selection is replaced by artificial evolution.
  AI systems will be our mind children (Moravec 1988,2000)

- Once a machine surpasses the intelligence of a human it can design even smarter machines (I.J.Good 1965).

- This will lead to an intelligence explosion and a technological singularity at which the human era ends.

- Prediction beyond this event horizon will be impossible (Vernor Vinge 1993)

- Alternative 1: We keep the machines under control.

- Alternative 2: Humans merge with or extend their brain by AI. Transhumanism (Ray Kurzweil 2005)

Countdown To Singularity
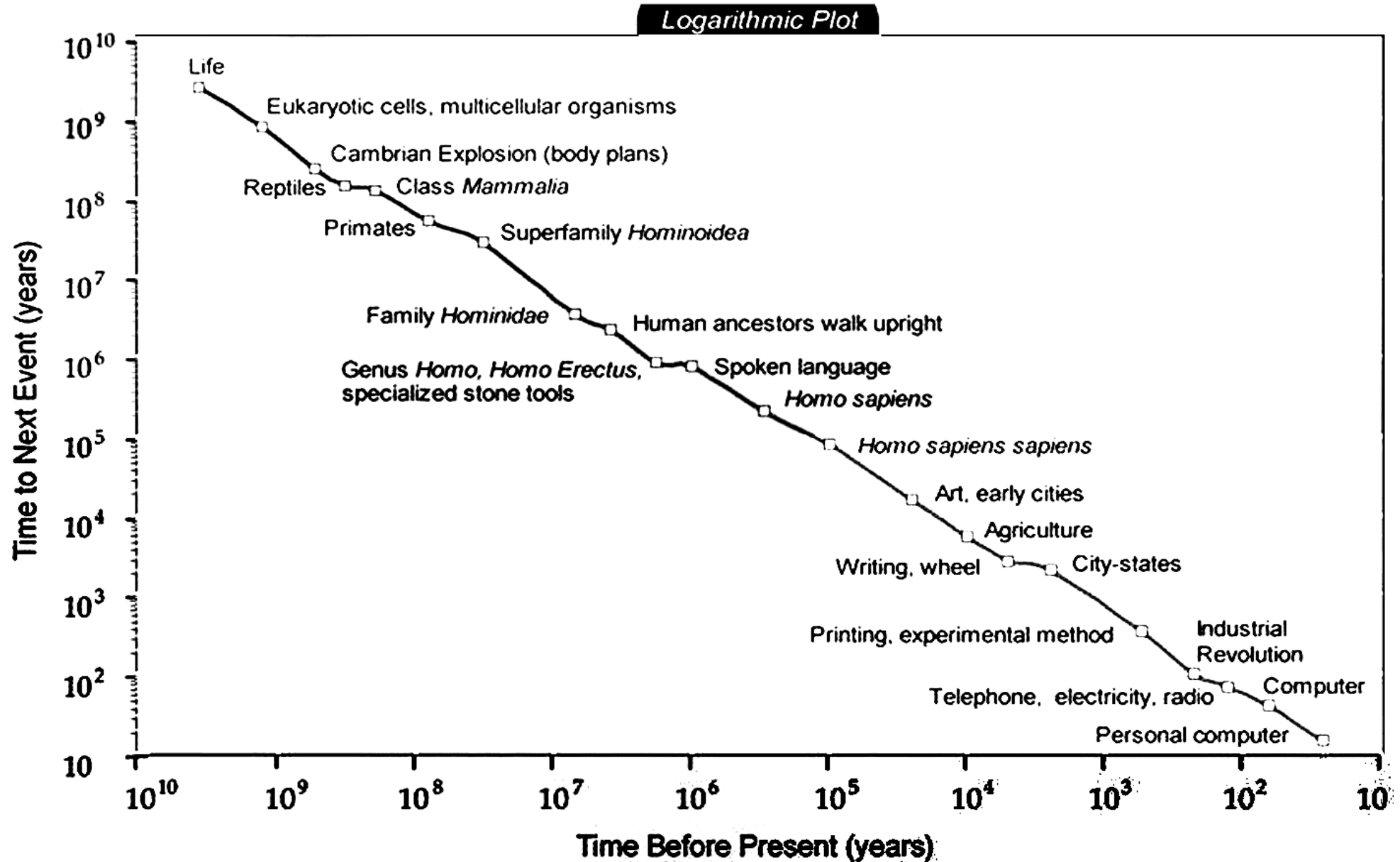
Three Laws of Robotics

Robots (should) have rights and moral duties

1. A robot may not injure a human being, or, through inaction, allow a human being to come to harm.

2. A robot must obey the orders given it by human beings except where such orders would conflict with the First Law.

3. A robot must protect its own existence as long as such protection does not conflict with the First or Second Law.

(Isaac Asimov 1942)

Conclusions

- We have developed a parameterless model of AI based on Decision Theory and Algorithm Information Theory.

- We have reduced the AI problem to pure computational questions.

- A formal theory of something, even if not computable, is often a great step toward solving a problem and also has merits in its own.

- All other systems seem to make more assumptions about the environment, or it is far from clear that they are optimal.

- Computational questions are very important and are probably difficult. This is the point where AI could get complicated as many AI researchers believe.

- Elegant theory rich in consequences and implications.

Literature

[Leg08] S. Legg. *Machine Super Intelligence*. PhD thesis, IDSIA, Lugano, Switzerland, 2008.

[Hut05] M. Hutter. *Universal Artificial Intelligence: Sequential Decisions based on Algorithmic Probability*, Chapter 8. Springer, Berlin, 2005.

[RN10] S. J. Russell and P. Norvig. *Artificial Intelligence. A Modern Approach*, Part VII. Prentice-Hall, Englewood Cliffs, NJ, 3rd edition, 2010.

[Mor00] H. Moravec. *Robot: Mere Machine to Transcendent Mind*. Oxford University Press, USA, 2000.

[Kur05] R. Kurzweil. *The Singularity Is Near*. Viking, 2005.

[Hut12a] M. Hutter. Can intelligence explode? *Journal of Consciousness Studies*, 19(1-2):143–166, 2012.

[Bos14] N. Bostrom, *SuperIntelligence: Paths, Dangers, Strategies*. Oxford University Press, 2014.

Main Course Sources

[Hut05] M. Hutter. *Universal Artificial Intelligence*. Springer, Berlin, 2005.
http://www.hutter1.net/ai/uaibook.htm

[CV05] R. Cilibrasi and P. M. B. Vitányi. *Clustering by compression.*
IEEE Trans. Information Theory, 51(4):1523–1545, 2005.
http://arXiv.org/abs/cs/0312044

[RH11] S. Rathmanner and M. Hutter.
A philosophical treatise of universal induction. *Entropy*,
16(6):1076–1136, 2011. http://dx.doi.org/10.3390/e13061076

[VNH+11] J. Veness, K. S. Ng, M. Hutter, W. Uther, and D. Silver.
A Monte Carlo AIXI approximation. *Journal of Artificial Intelligence
Research*, 40:95–142, 2011. http://dx.doi.org/10.1613/jair.3125

[Hut12] M. Hutter. One Decade of Universal Artificial Intelligence.
In *Theoretical Foundations of Artificial General Intelligence*,
4:67–88, 2012. http://arxiv.org/abs/1202.6153

Thanks! Questions? Details:

A Unified View of Artificial Intelligence

=   =

Decision Theory   =   Probability + Utility Theory

+   +

Universal Induction   =   Ockham + Bayes + Turing

Open research problems:

at www.hutter1.net/ai/uaibook.htm

Compression contest:

with 50'000 € prize at prize.hutter1.net

Projects: www.hutter1.net/official/projects.htm