Property Testing, with a focus on Invariances

Arnab Bhattacharyya
Massachusetts Institute of Technology

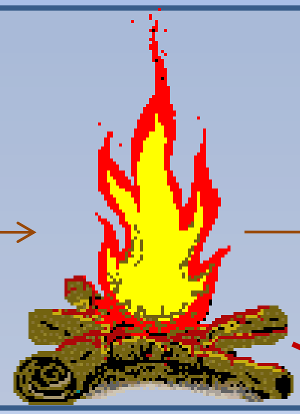
The two faces of computational science

Computational Model

Parameters →  → *Outputs*

<u>Simulation</u>

The two faces of computational science

Parameters →  → *Outputs*

Learning Algorithm

Computational Model

Learning

Theoretical underpinnings of simulation

- Given model for physical system, how do you efficiently implement it?
  - Algorithm design
  - Computational Complexity

- Kicked off field of theoretical computer science (Turing 1936)

Theoretical underpinnings of learning

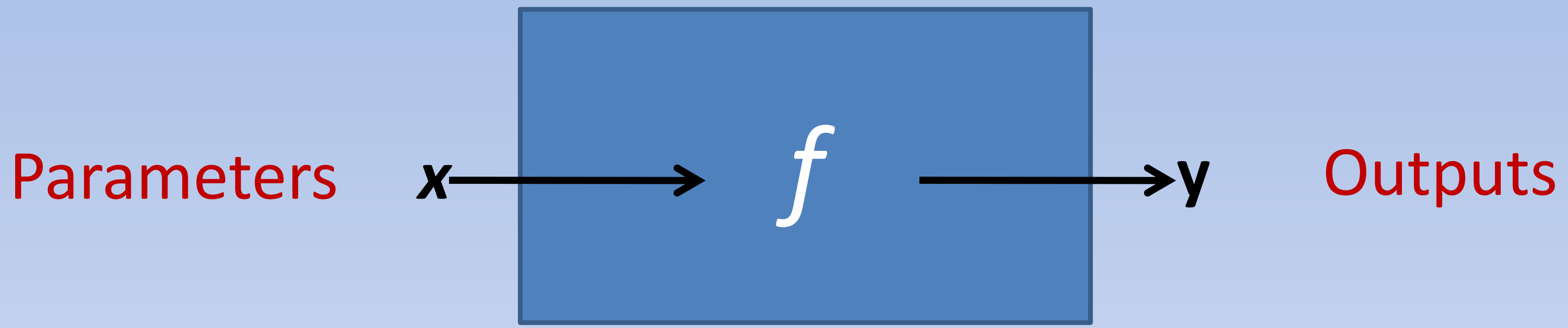
- How does one define learning formally?
- There is always a trivial learning algorithm but this is uninteresting
- Question first formalized by Leslie Valiant, 1984

What is learning?

- Searching for a <u>simple</u> description of process

- Formally, we ask that the learned model satisfy some natural property *P*
  - E.g., first order differential equation, or linear function, or small depth AND-OR-NOT circuit

Formal description of learning



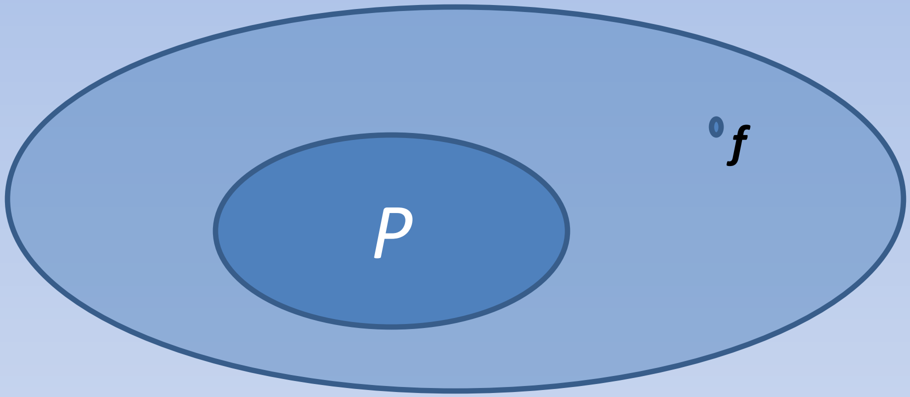
- What is *g* satisfying *P* such that $|g - f| < \varepsilon$?
- Want to find *g* <u>efficiently</u>
- Implicit assumption: *f* satisfies *P*

Property Testing: A relaxation

- Can such a *g* be found at all?

- Formal setup:
  - Given property *P* consisting of functions mapping *X* to *Y*
  - Given query access to function *f* mapping *X* to *Y*, decide efficiently if *f* has property *P* or is far from all functions with property *P*
- Question first asked in [Blum-Luby-Rubinfeld 1991]

Formal Tester Guarantees



- If *f* satisfies *P*, then tester should accept with probability at least 2/3
- If *f* disagrees on at least ε fraction of the domain with every *g* satisfying *P*, then tester should reject with probability at least 2/3

Easy Examples

- Property of a function being constant on all of the domain
  - Need $\Theta(1/\varepsilon)$ samples

- Property given by choosing a random subset of all functions on the domain
  - Need $\Omega(n)$ samples

More nontrivial testable properties

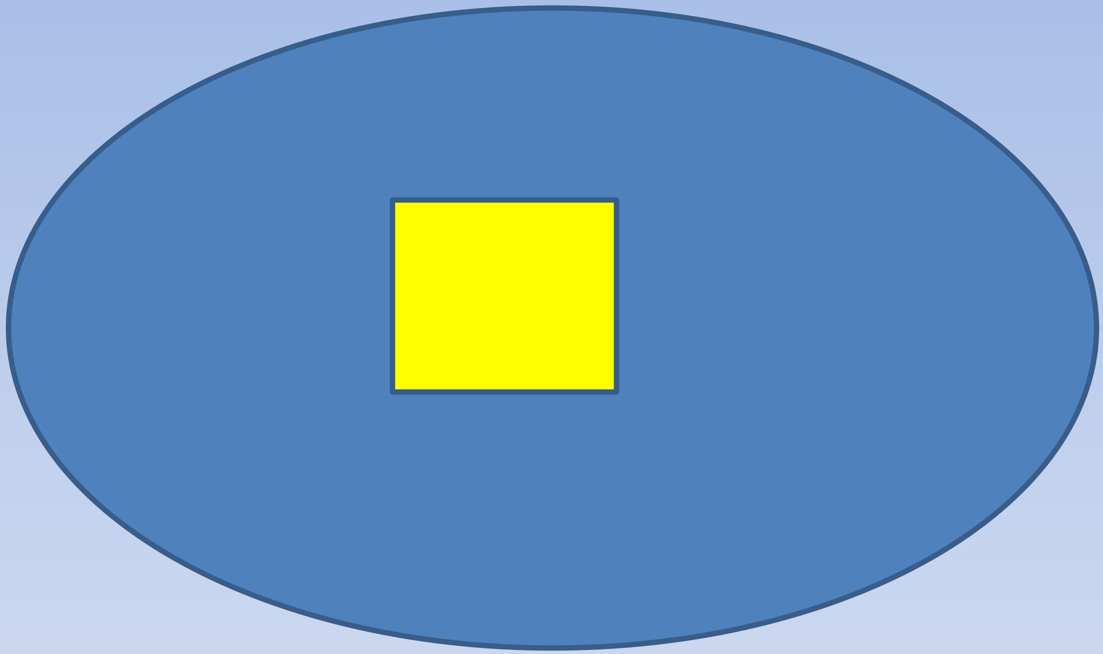
- Linearity [Blum-Luby-Rubinfeld 1991]
  - Property that a function satisfies $f(x) + f(y) = f(x+y)$ for all x,y in the domain
  - Testing linearity shown to require only $\Theta(1/\varepsilon)$ samples (independent of n)
- Low degree polynomials
  - Again shown to be testable using remarkable analysis
  - Simple proof recently found in [Bhattacharyya et al, 2010] for the case of finite fields

More nontrivial testable properties

- A *graph property* is a collection of unlabeled graphs
  - *Examples: acyclicity, bipartiteness, planarity*

- [Goldreich-Goldwasser-Ron 1997] showed that a vast collection of graph properties are testable, including all of the above!

Perspective: Invariance



If there is enough invariance, the tester can look at a random subset of the domain and get adequate information

Characterization of Testable Graph Properties

- Graph properties have a very large set of invariances
  - Always enough for the tester to inspect only a random induced subgraph
- [Alon et al, 2005] *characterized* exactly the class of testable graph properties using extremal additive combinatorics

Characterization of Testable Algebraic Properties

- Linear invariant properties
  - Functions on a vector space that are closed under linear transformations, such as linear functions, low degree polynomials, sum-free sets, and many other previously considered natural properties

- Current line of work with co-authors heading towards a characterization of testable linear invariant properties
  - Roughly, a linear invariant property is testable if and only if it is closed under restrictions to linear subspaces

Thanks!!