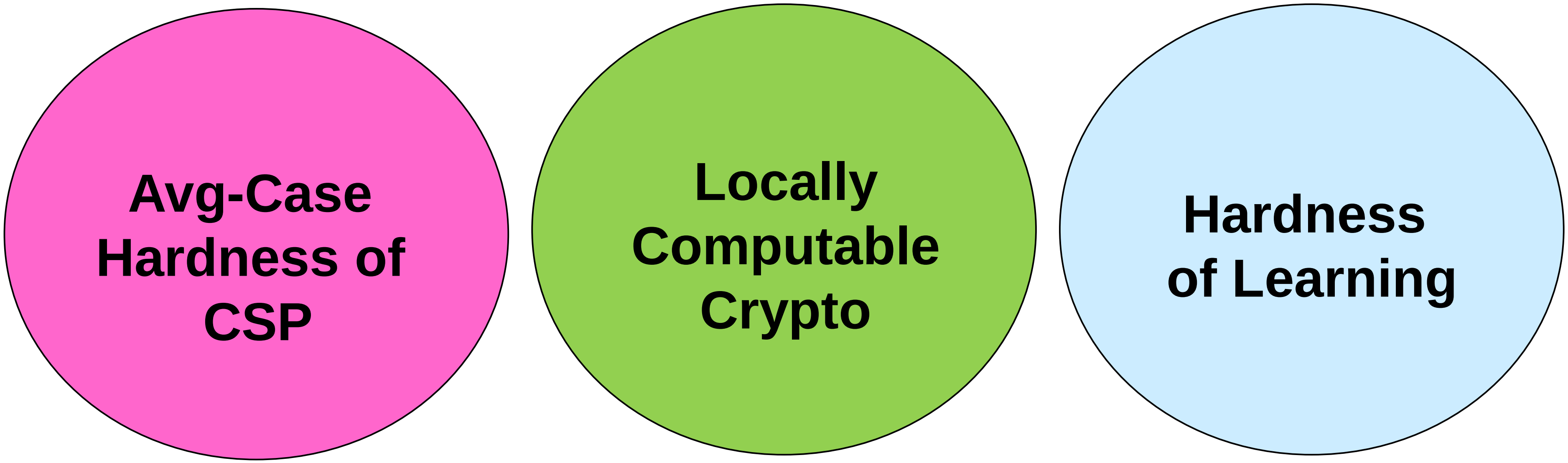
Cryptographic Hardness of Constraint Satisfaction Problems

Benny Applebaum
Tel-Aviv University

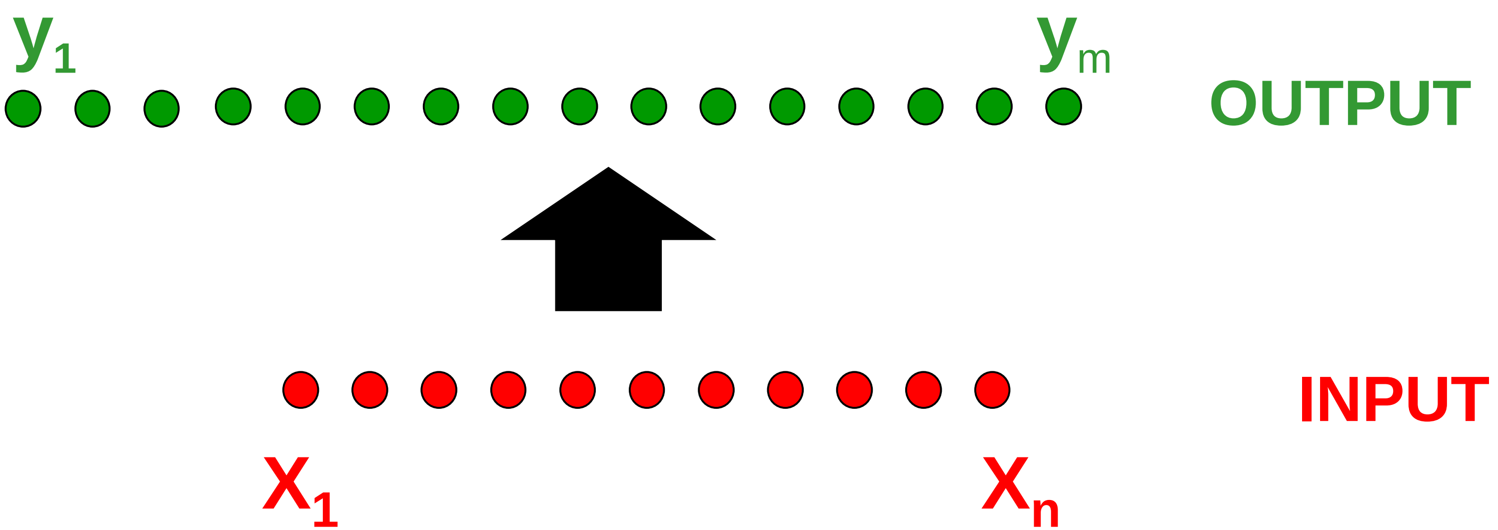
Average-Case Complexity: From Cryptography to Statistical Learning
Simons November 2021



Avg-Case Hardness of CSP
Locally Computable Crypto
Hardness of Learning

Pseudorandom Generator

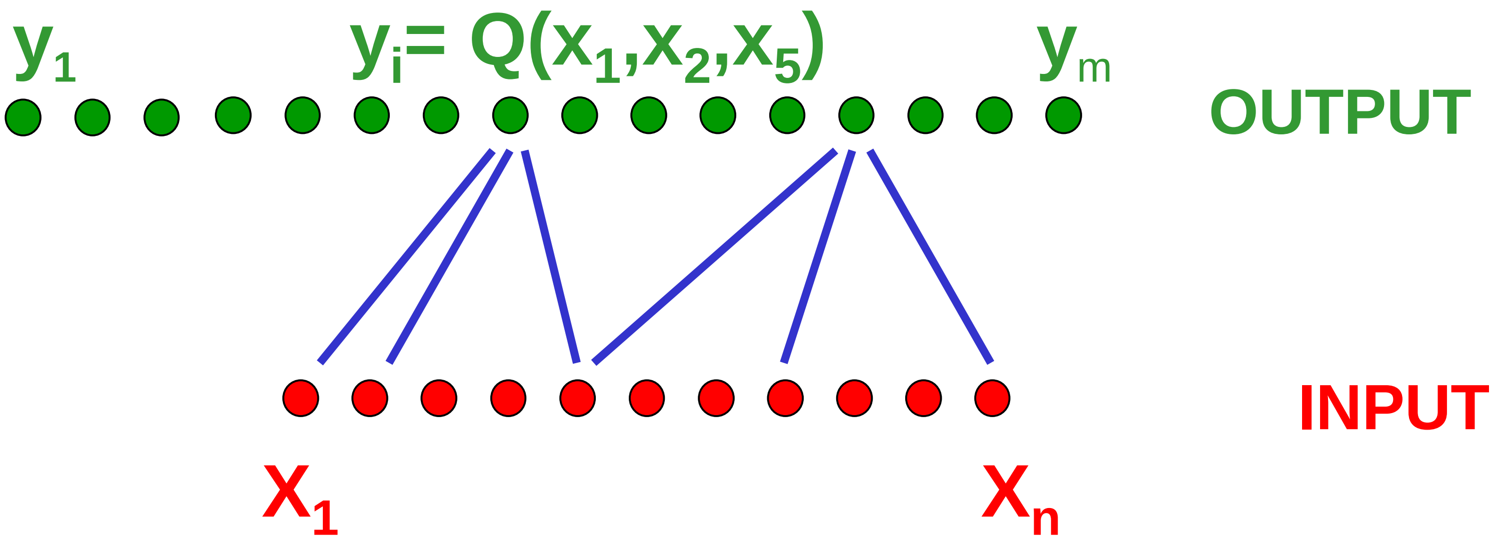
Expand n random bits into $m \gg n$ pseudorandom bits



Locally Computable PRGs

Expand n random bits into $m \gg n$ pseudorandom bits
each output depends on $d=O(1)$ inputs

- Well studied problem [CM02,Alekh03,MST03,AIK04-6, …]
- For poly-long output length, only known candidates based: **Random Local Functions** [Goldreich00]



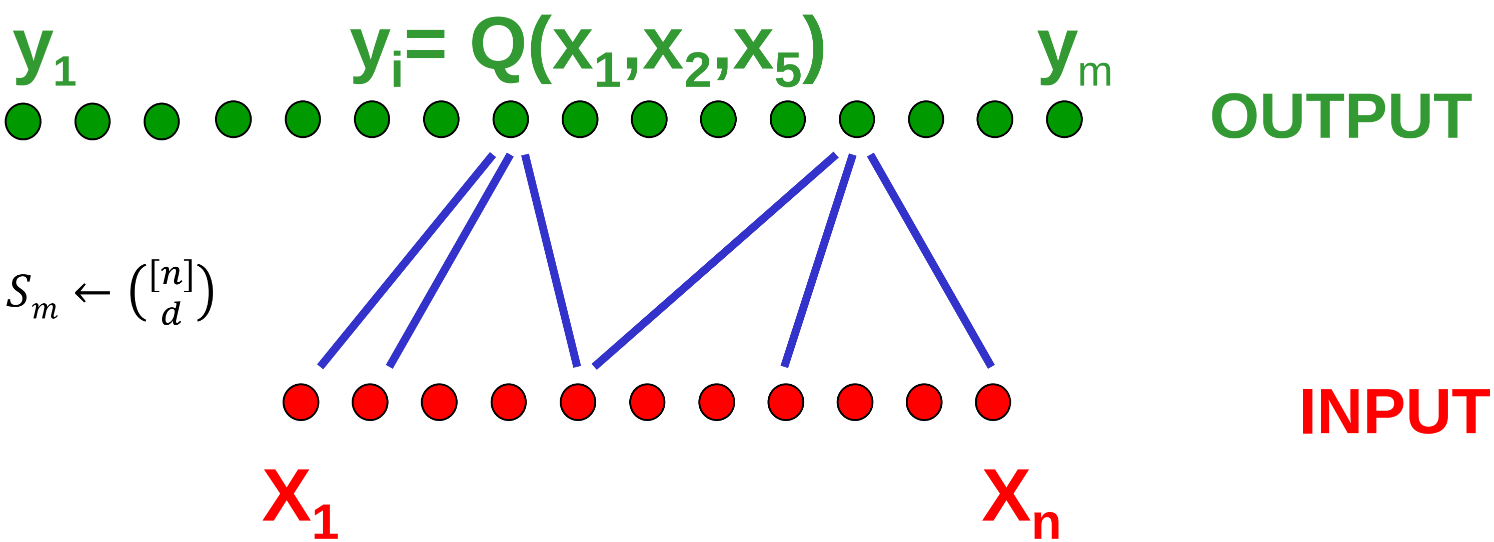
Random Local Functions are PRGs

conjecture: for **most graphs** and **properly chosen** predicate **Q**, the resulting function is a pseudorandom generator

Parameters: locality d, predicate Q, output length m

- E.g., d=10, Q=XOR+MAJ and $m=n^2$

Choose iid m d-tuples $S_1, \ldots, S_m \leftarrow \binom{[n]}{d}$

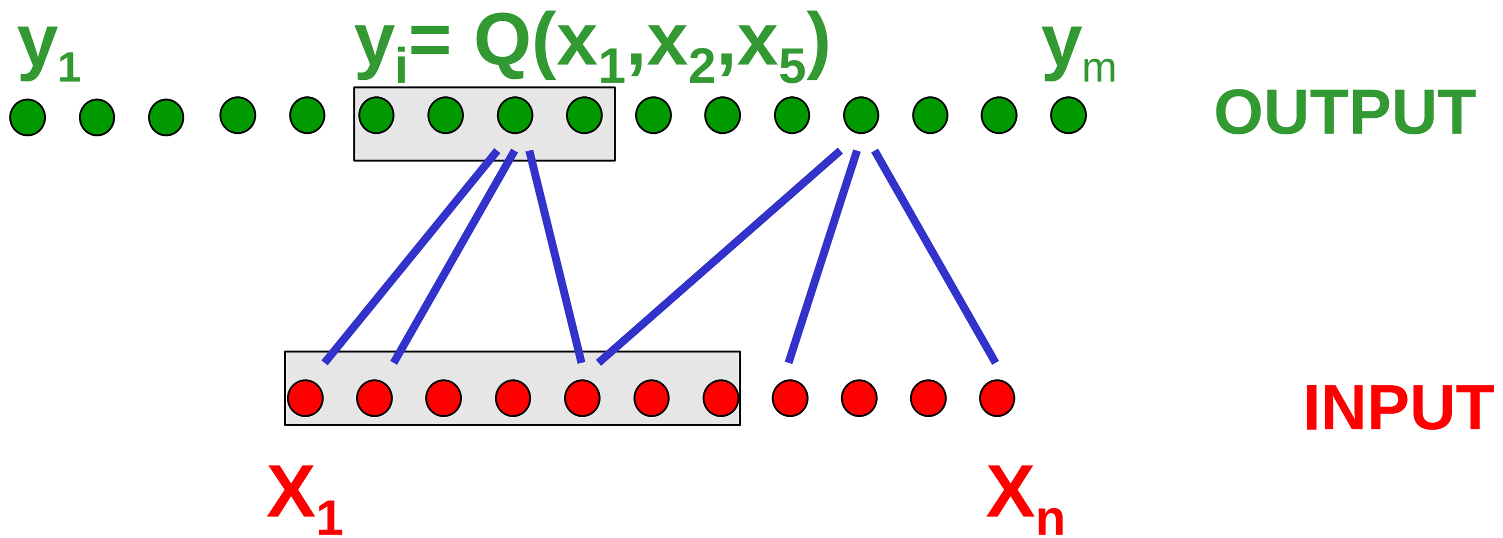


Random Local Functions are PRGs

conjecture: for **most graphs** and **properly chosen** predicate **Q**, the resulting function is a pseudorandom generator

Variants:

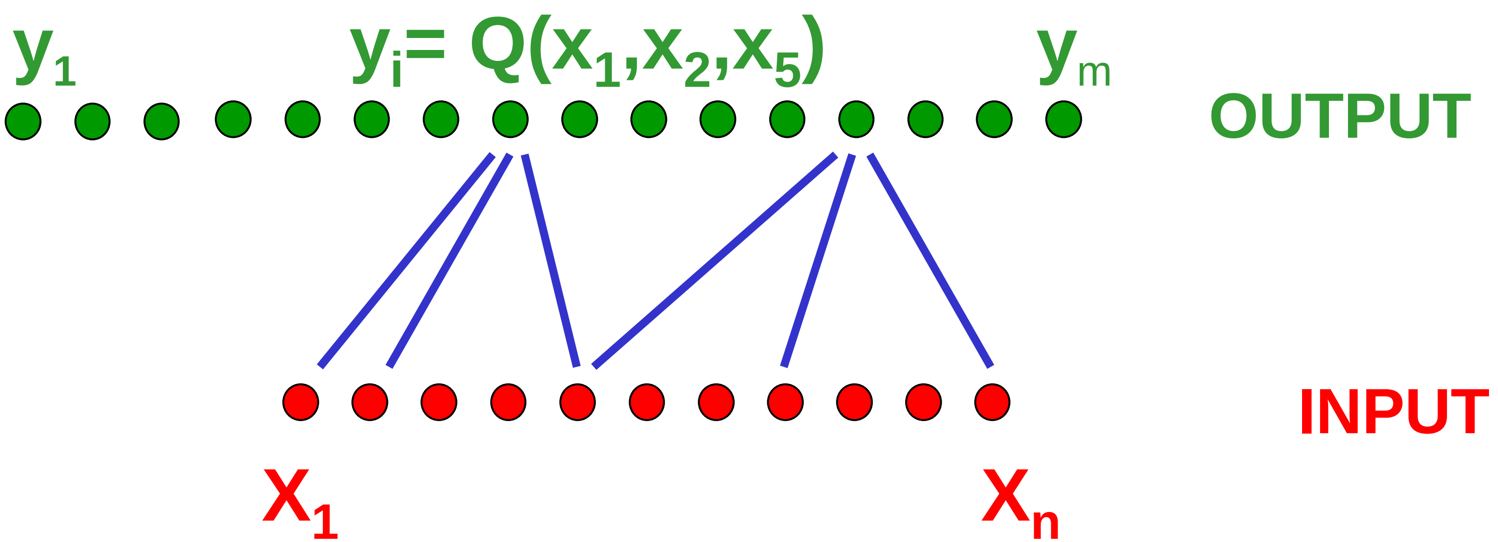
- Stronger variant: for **every sufficiently-good expander graph**
  - $(t, 0.9d)$-expansion => pseudorandom against $\exp(\Omega(t))$-time attacks?
- Weaker variant: there **exists some graph**



Random Local Functions are PRGs

conjecture: for **most graphs** and **properly chosen** predicate **Q**, the resulting function is a pseudorandom generator

- Studied in [CEMT09,ABW10,A12,ABR12,BR11,BQ12,OW14,FPV15,…] See survey [A15]
- Before assuming **crypto in P** understand crypto by **local functions**
- Interesting cryptographic/complexity-theoretic applications [IKOS08,A12,…,JLS21]
  - See Aayush's talk

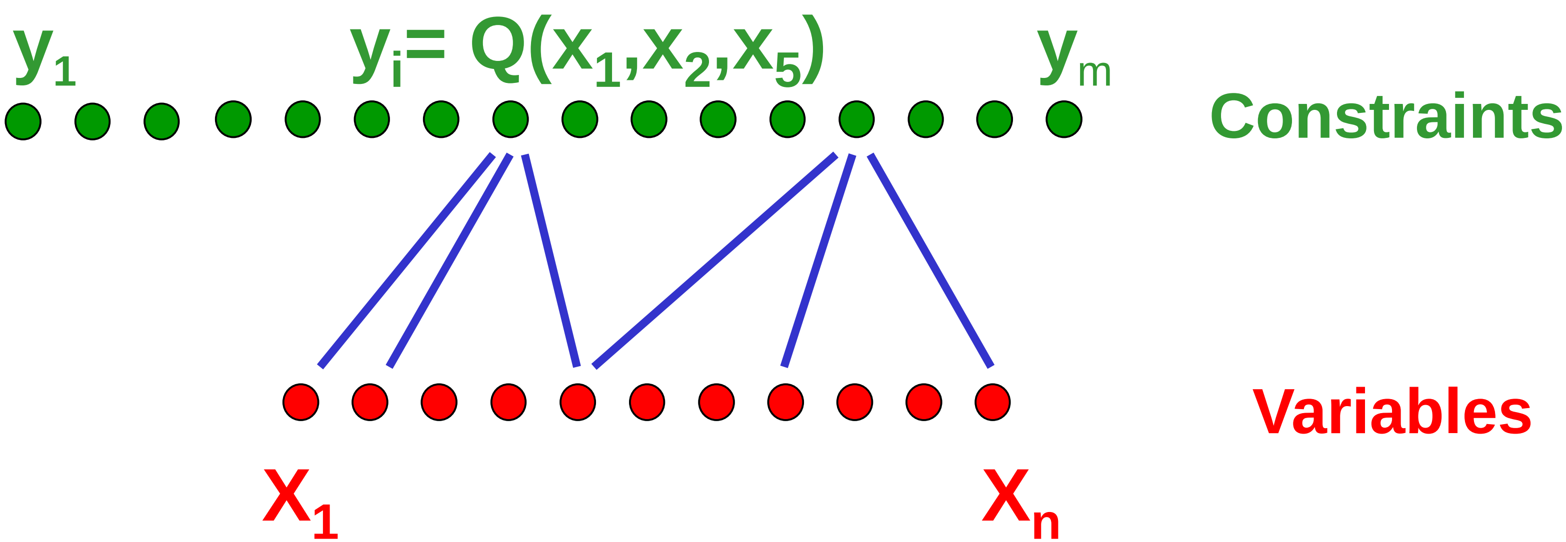


Some Implications

CSP Perspective [AIK06]

View (G,y) as CSP: i-th constraint $Q(x[S_i])=y_i$

- $y \leftarrow PRG(x)$ then CSP is satisfiable
- $y \leftarrow$ uniform then, whp, $value(CSP) \ll 1$
- $m = \omega(n)$ and Q is balanced=> $value(CSP) \approx 0.5$

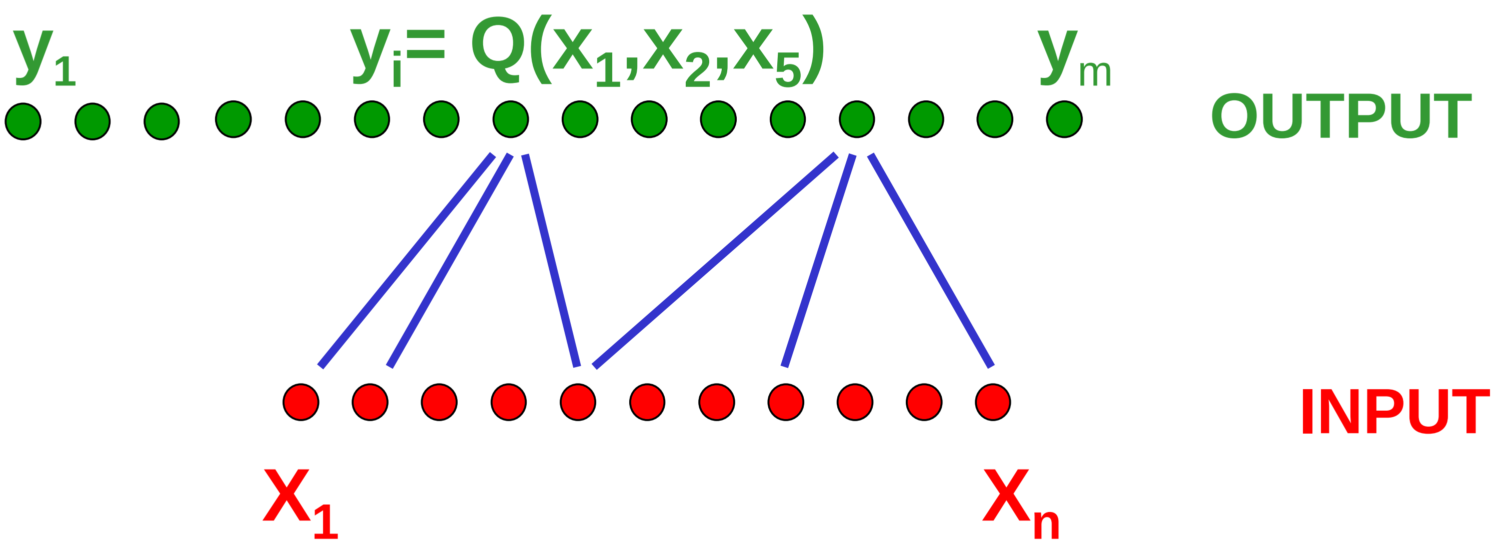


CSP Perspective [AIK06]

RLFs are PRGs

Distribution over CSPs which is:

Hard to approximate/satisfy/refute



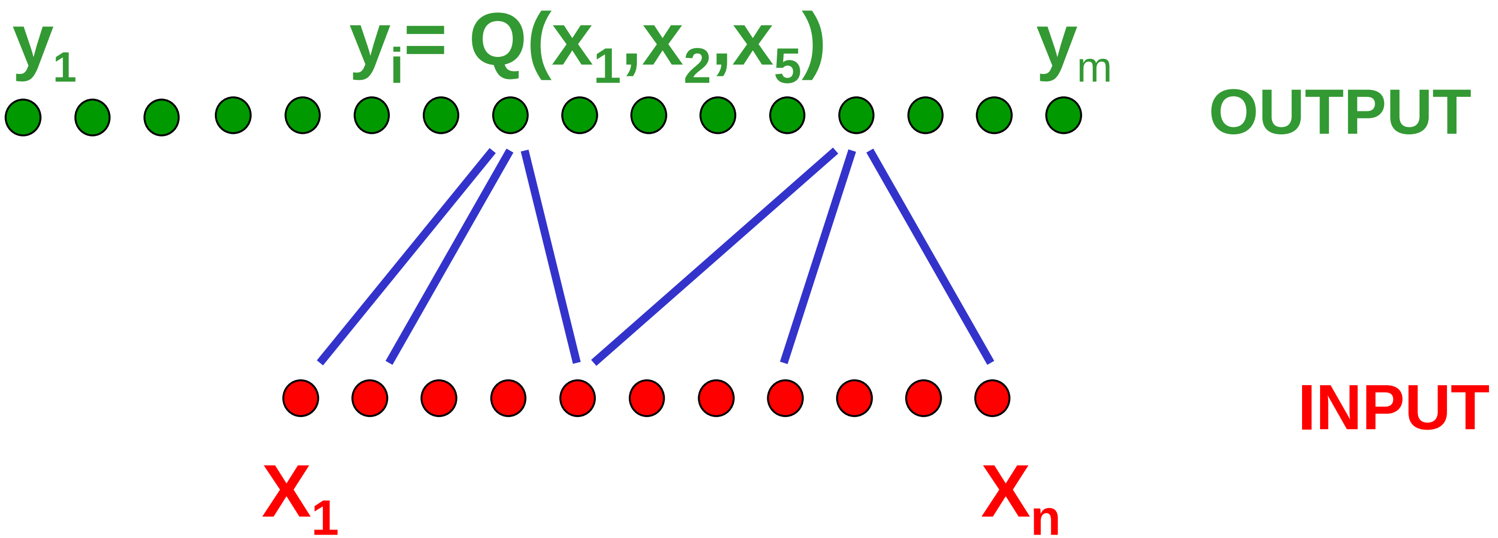
Learning Perspective [ABW10]

Define the function: $g_x : \{0,1\}^{d \log n} \rightarrow \{0,1\}$

- $g_x(i_1, .., i_d) = Q(x[i_1], x[i_2], \ldots, x[i_d])$

PAC-Learning:

Given m-1 random (inputs/output) pairs
predict g_x on fresh random input

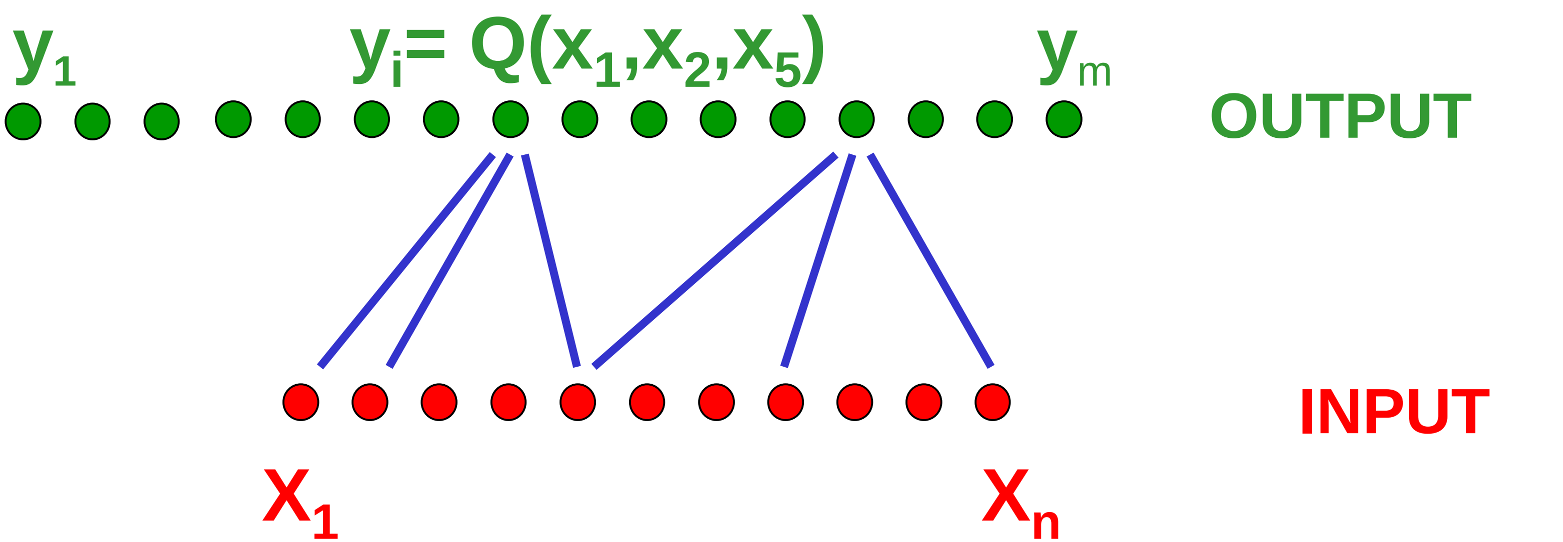


Learning Perspective [ABW10]

RLFs are PRGs

Distribution over functions with domain n^d which is:

Hard to PAC-learn given $m = n^s$ **samples for some 1<s<d**



Scaled-up Version [AR16]

RLFs are PRGs

$d = O(\log n), m = n^{\omega(1)}$

Supported by the "expanders are PRG" conjecture

Distribution over **depth-3 AC0** functions which is:

Hard to PAC-learn given **any poly(n) uniform** samples

Input:

$i_1 \in \{0,1\}^{\log n}$ **Log(n)-local function** $x[i_1]$

...

$i_d \in \{0,1\}^{\log n}$ **Log(n)-local function** $x[id]$

X_1 HARD-WIRED X_n

Q

d-local

y=

$Q(x_{i1},\ldots,x_{id})$

Scaled-up Version [AR16]

RLFs are PRGs

$d = O(\log n), m = n^{\omega(1)}$

Distribution over **depth-3 AC0** function which is:

Hard to PAC-learn given any **poly(n) uniform** samples

Even given sub-exponential time

[LMN93] learning algorithm is tight

(can't learn AC0 over uniform samples in quasi-poly time given poly uniform samples)

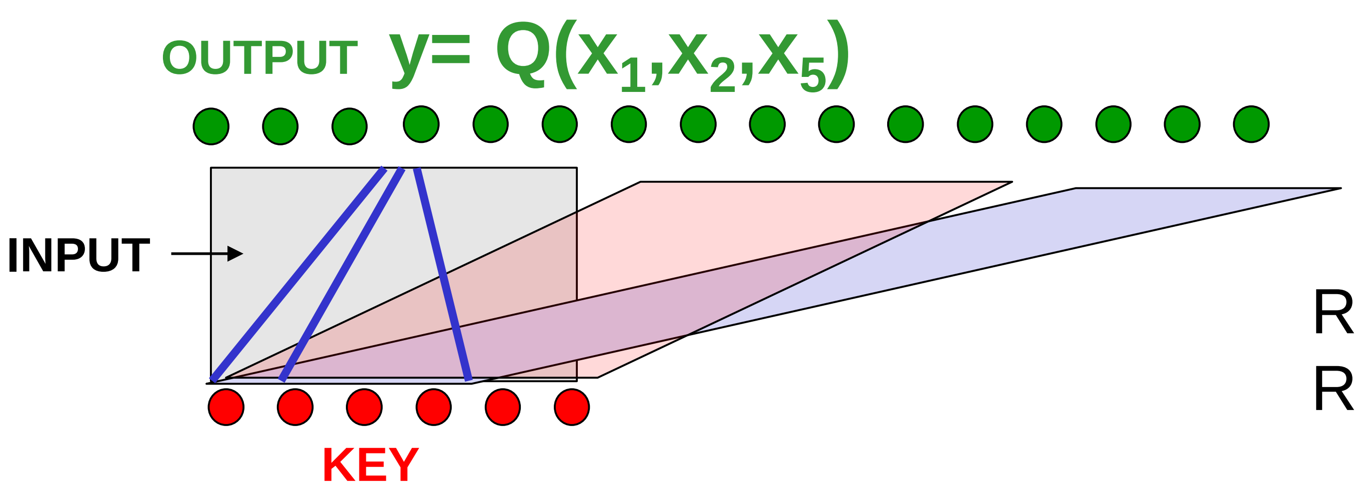
Simple PRF [AR16]

RLFs are PRGs

$d = O(\log n), m = n^{\omega(1)}$

Weak-PRF from n bits to $\widetilde{\Omega}(n)$ bits computable by

- AC0 circuit of depth-3
- Linear-time in RAM model



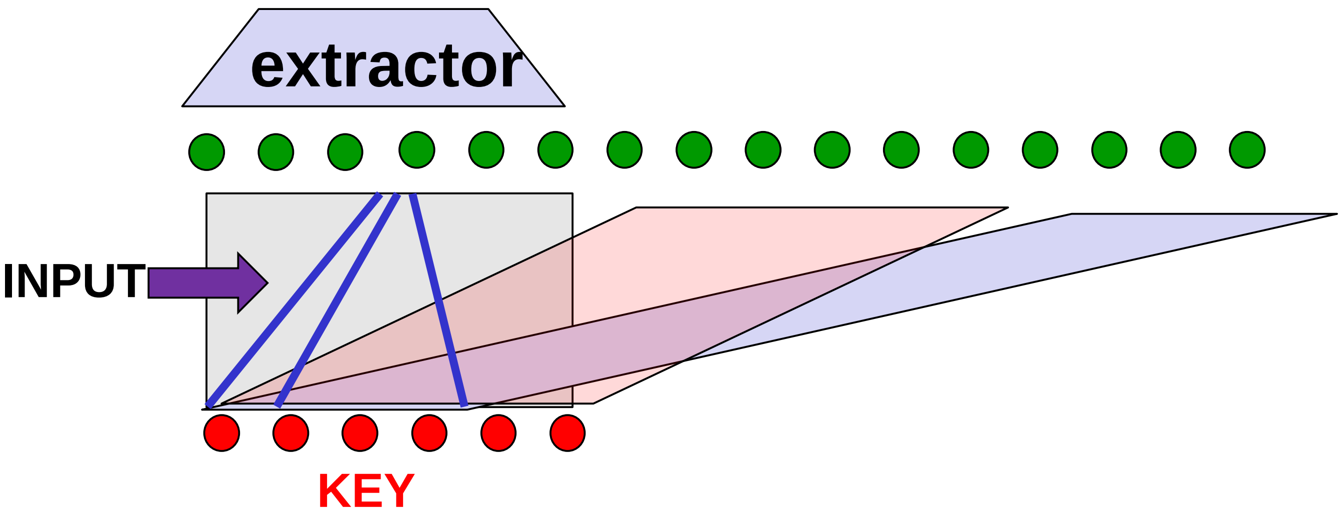
Random inputs $\Rightarrow$ Random graph

Handling non-random inputs? [AR16]

Fast low-bias generator

Arbitrary polynomialy many inputs

$\Rightarrow$ resulting graph is almost expanding



Fast PRF [AR16]

Expander-functions are one-way function

$d = O(\log n), m = n^{\omega(1)}$

Strong-PRF from n bits to $\widetilde{\Omega}(n)$ bits computable by

- Constant-depth circuit over AND, OR, MAJORITY
- Quasilinear Circuit
- Sub-exp security given poly(n) queries (beyond?)

[OlivSanthTell18]: Highly efficient Expander+predicate $\Rightarrow$

Too-efficient PRF $\Rightarrow$ breakable via natural properties

Barrier for eff expanders/natural properties OR conjecture too bold

Why should we believe the Conjecture?

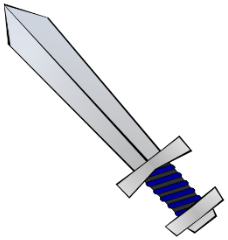
A1: Unconditional Security against concrete attacks

For $m=n^s$ which predicates satisfy the conjecture?

A2: Reduction to One-wayness

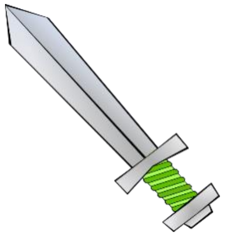
Which predicates yield PRGs?

Resiliency



“Local” attacks

“Degree”

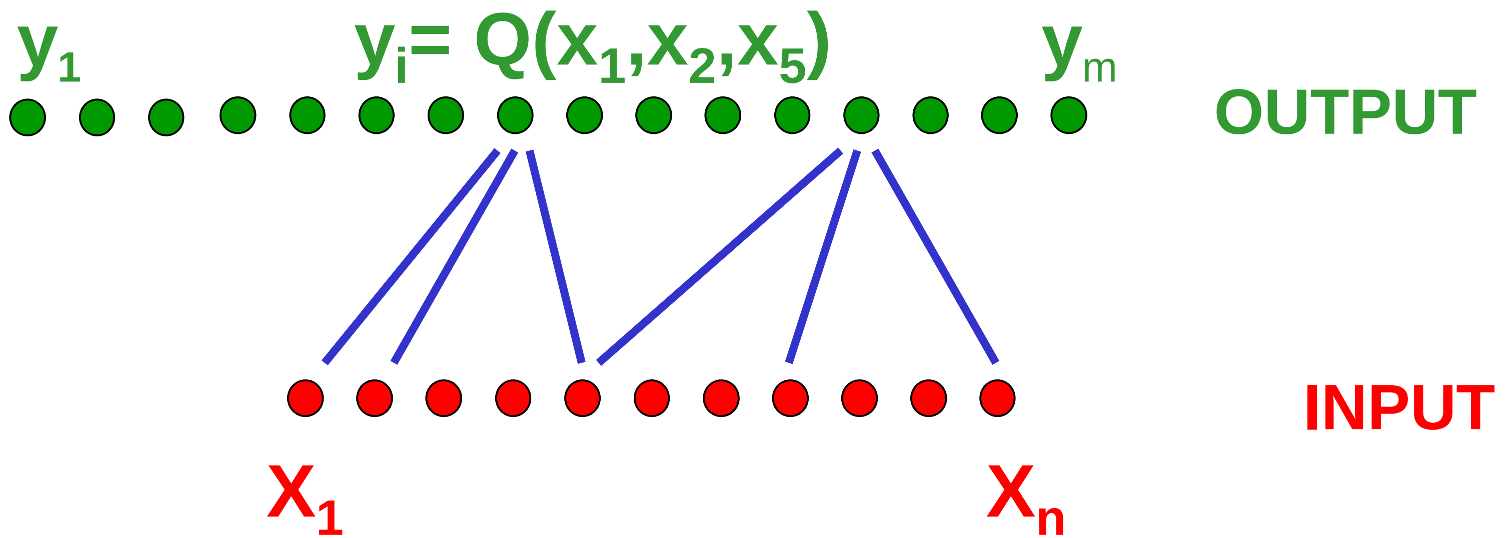


Linear algebra

Goal: Hard to distinguish y from random

More fragile than one-wayness:

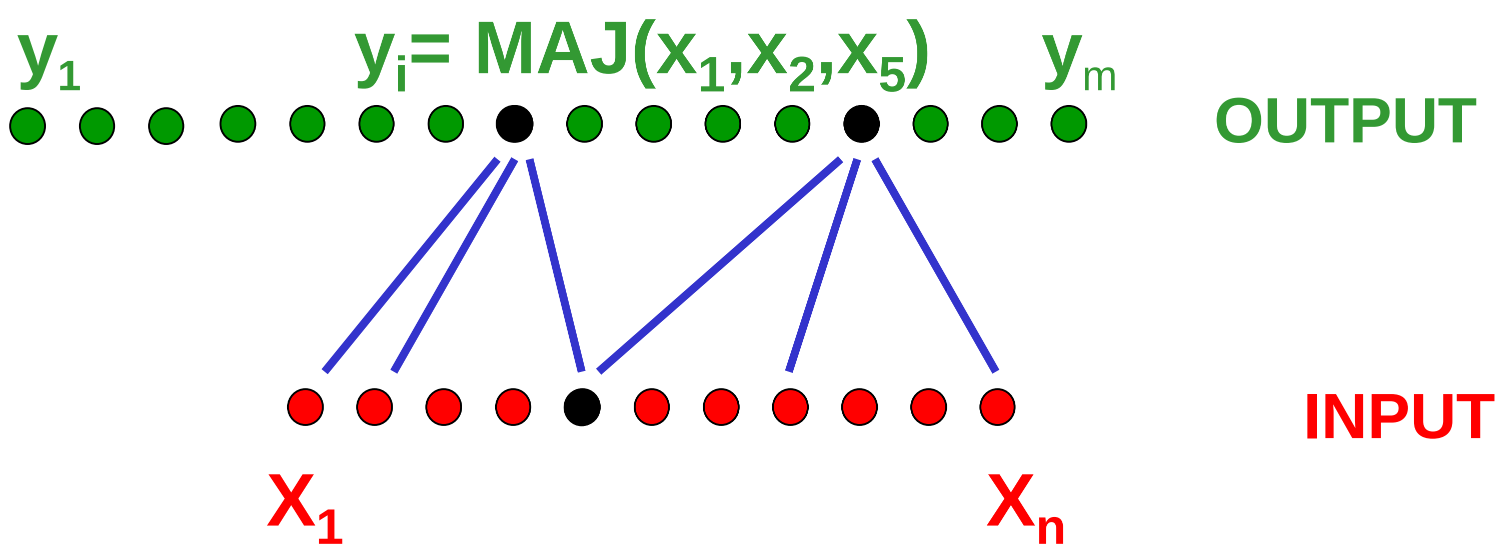
Predicate must be balanced



Goal: Hard to distinguish y from random

More fragile than one-wayness:

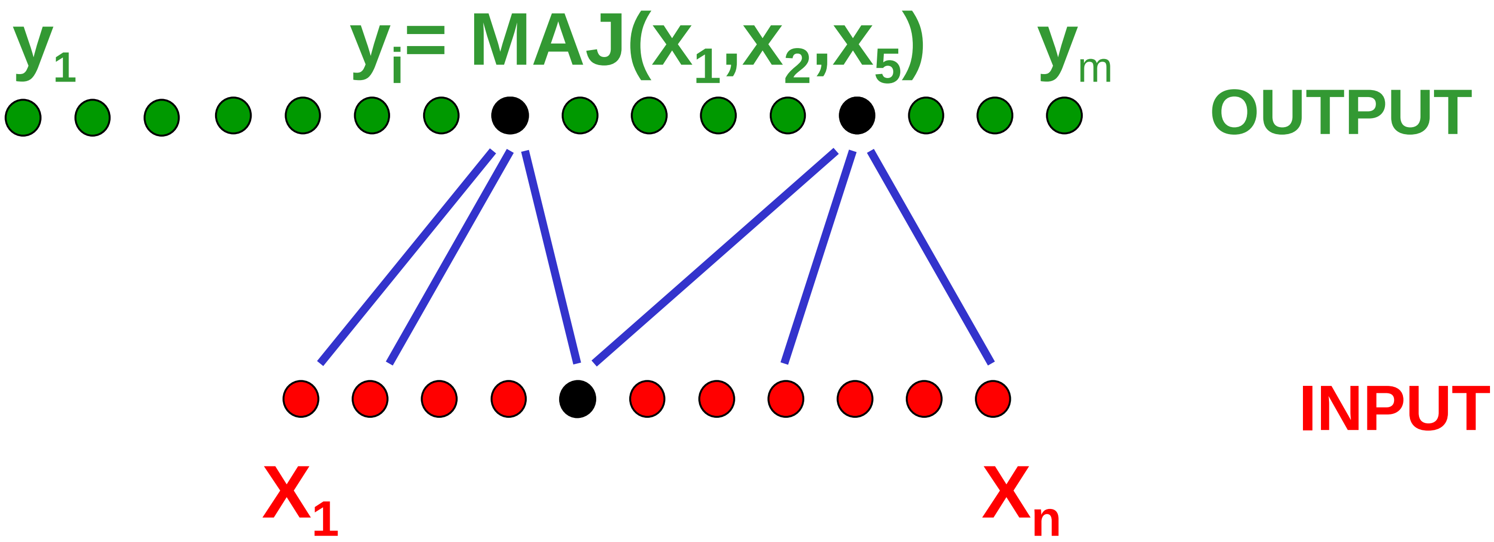
Predicate must be balanced even after fixing single input



Goal: Hard to distinguish y from random

k-resiliency [Cho-Gol-Has-Fre-Rud-Smo]:

Predicate must be balanced even after fixing **k** inputs

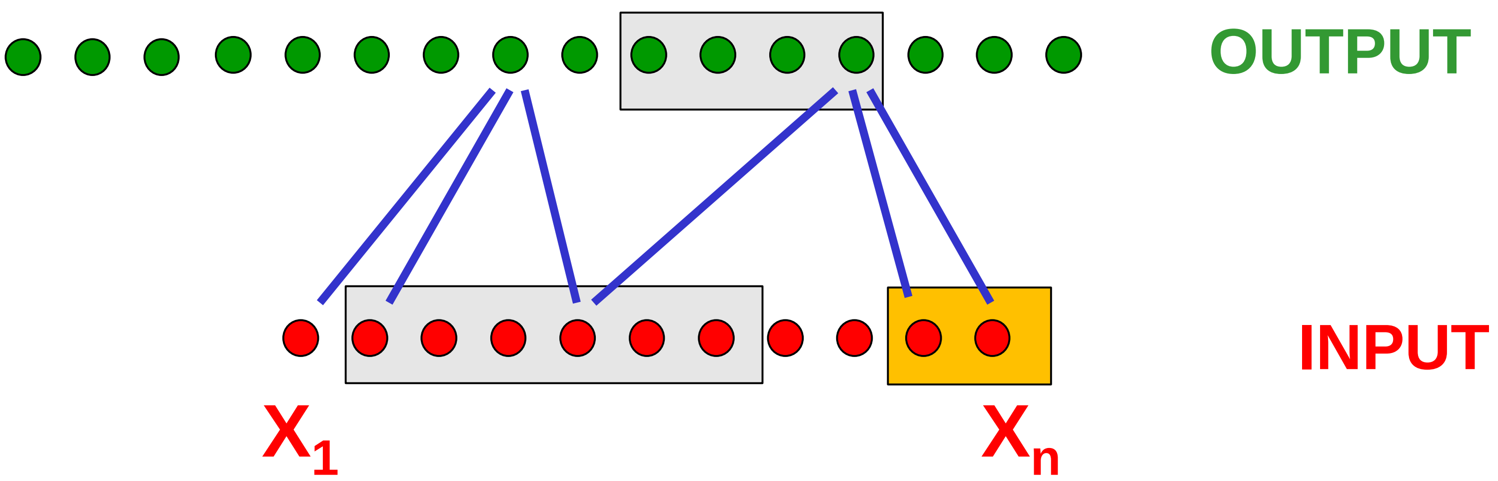


Resiliency defeats local attacks

[Mossel-Shpilka-Trevisan'03]

For m=n^s resiliency of k=2s-1 is necessary and sufficient against

- Sub-exponential AC0 circuits [A-Bogdanov-Rosen12]
- Semidefinite programs [O'Donnel Witmer14]
- Sum of Squares attacks [Kothari Mori O'Donnel Witmer17]
- Statistical algorithms [Feldman Perkins Vempala15]

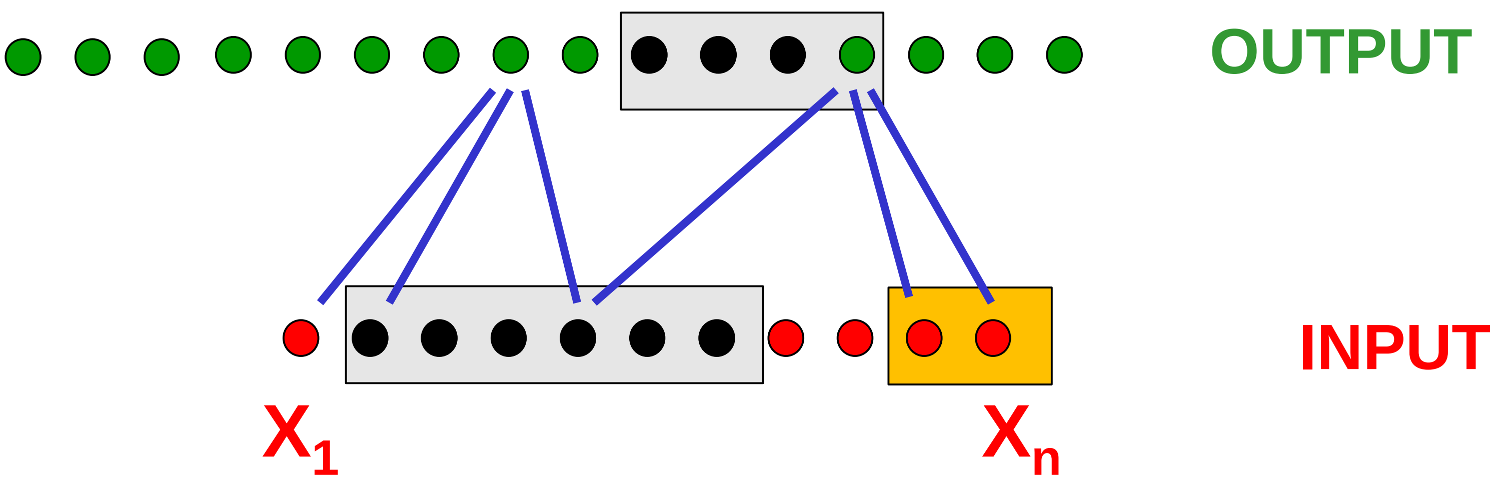


Resiliency defeats local attacks

For $m=n^s$ resiliency of $k=2s-1$ is necessary and sufficient against

- Sub-exponential AC0 circuits [A-Bogdanov-Rosen12]
- Semidefinite programs [O'Donnel Witmer14]
- Sum of Squares attacks [Kothari Mori O'Donnel Witmer17]
- Statistical algorithms [Feldman Perkins Vempala15]

Q: Order these attacks?

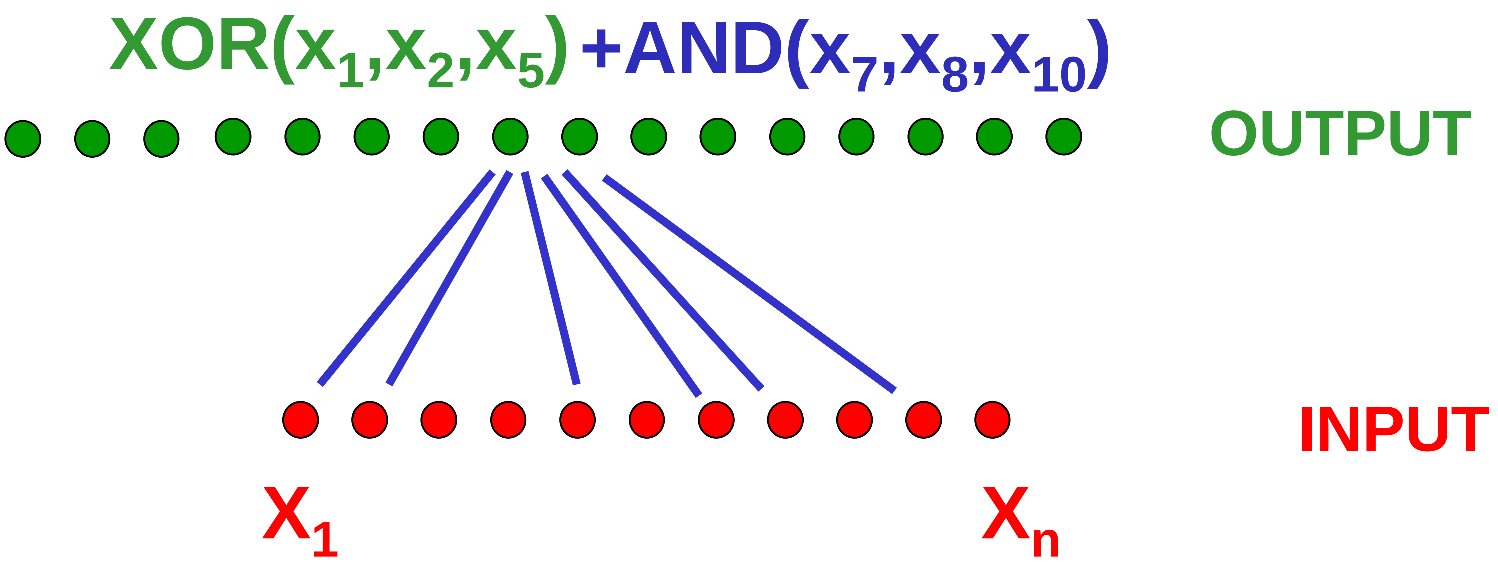


Defeating Linear Algebra

For $m=n^s$ need **algebraic degree** of s

Resiliency+Degree$\Rightarrow$Pseudorandomness? [OW14, A14, FPV15]

- Yes for $m<n^{5/4}$ and linear distinguishers [MST03, ABW10, ABR12]
  i.e., small-bias generator [NN]
- No for larger m’s [A-Lovett16]



Defeating Linear Algebra [AL16]

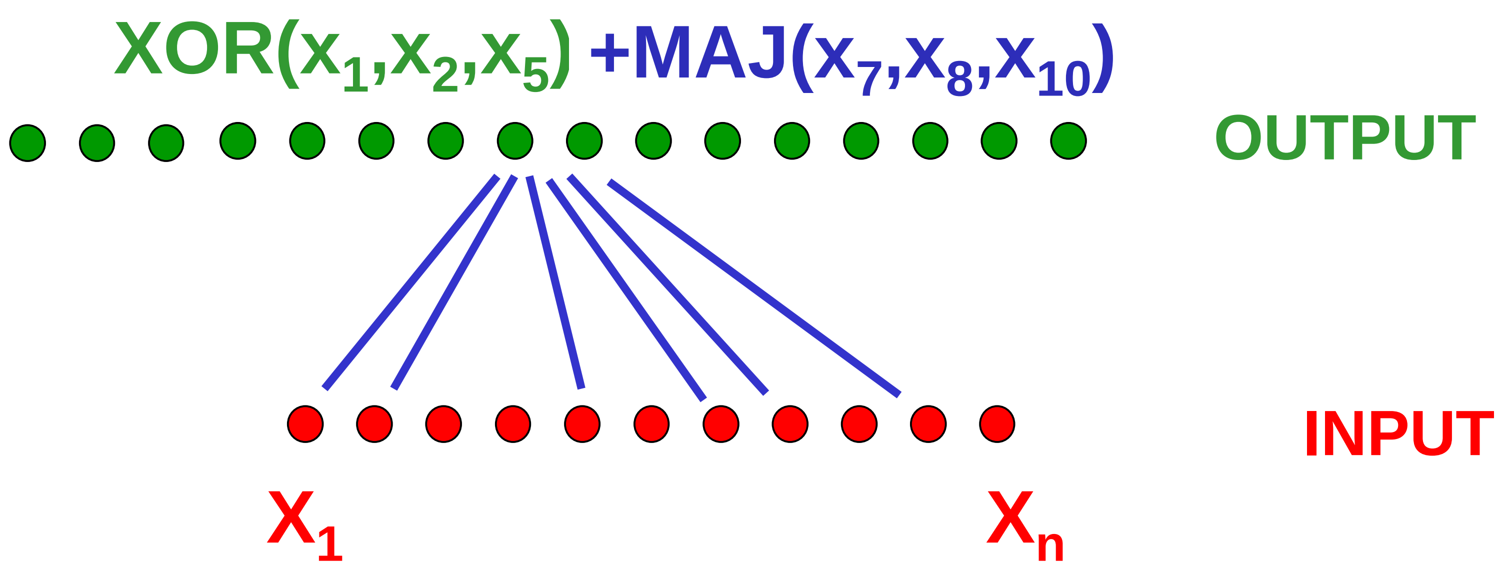
b-**fixing degree**: **algebraic degree** of b **even after fixing** b inputs

Thm: For $m=n^s$, $\Theta(s)$-bit fixing degree
necessary & sufficient against linear distinguishers

Stronger form of **rational-degree** is necessary & sufficient for defeating "algebraic attacks"

$Q(x)=0 \not\Rightarrow$

low-degree-equation



Defeating Linear Algebra [AL16]

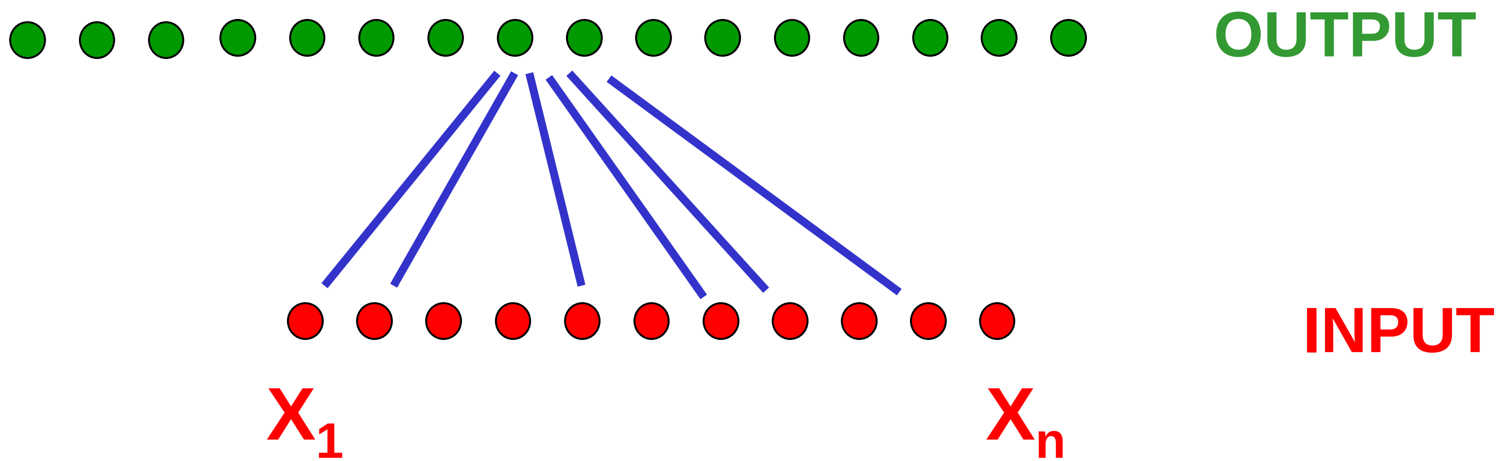
b-**fixing degree**: **algebraic degree** of b **even after fixing** b inputs

Thm: For $m=n^s$, $\Theta(s)$-bit fixing degree
necessary & sufficient against linear distinguishers

Stronger form of **rational-degree** is necessary & sufficient for defeating "algebraic attacks"

Refutation via polynomial-calculus proof system [CleggEdmondsImpag96]

$XOR(x_1,x_2,x_5) + MAJ(x_7,x_8,x_{10})$



Random Local Functions:
one-wayness $\Rightarrow$ unpredicatbility

One-Wayness $\Rightarrow$ Pseudorandomness [A11]

OWF Conjecture: f is one-way
for predicate Q, random graph with **m** outputs

Implication 1: f is 0.99-unpredictable
for predicate Q & random graph with **m** outputs

Implication 2: f is ε-pseudorandom
for predicate Q & random graph with $\mathbf{m^{1/3}/\varepsilon^2}$ outputs

Supports the PRG-conjecture

Extension to expanders [AR16]:

- One-wayness over all expanders
  $\Rightarrow$ Pseudorandomness over all expanders

One-Wayness $\Rightarrow$ Pseudorandomness [A11]

CSP Perspective: Search-to-decision reduction

- Solving the GAP-problem=> finding satisfying assignment
- Preserves the distribution (up to loss in the length)

One-Wayness $\Rightarrow$ Pseudorandomness [A11]

Learning Perspective: Proper-to-Improper reduction

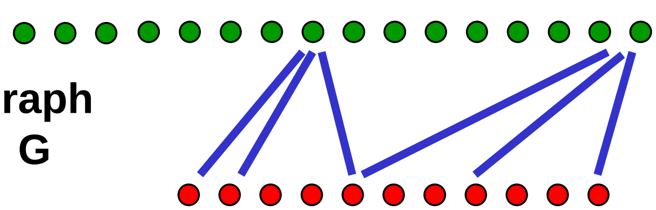
Predicting the target function by **some arbitrary** hypothesis

=> Recovering the **description** of the target function

Prediction $\Rightarrow$ Inversion

Simplifying Assumption:

$\mathbf{Q}(x_1,\ldots, x_d)=x_1\oplus\mathbf{P}(x_2,\ldots, x_d)$

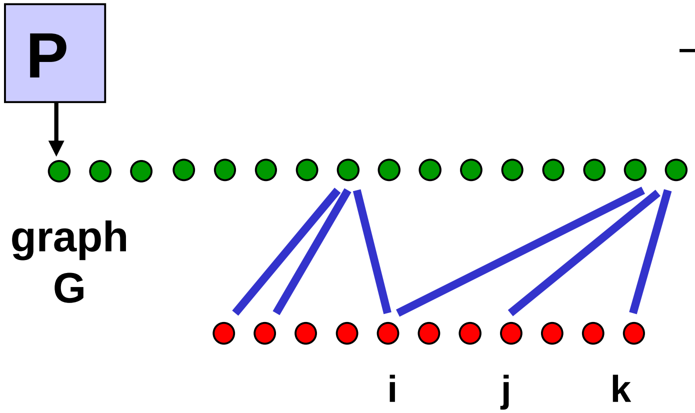


$y_m = Q(x_i,x_j,x_k)$

graph G

i j k

Prediction $\Rightarrow$ Inversion



$\rightarrow$ $b = Q(x_i,x_j,x_k)$ w.p $½+\delta$

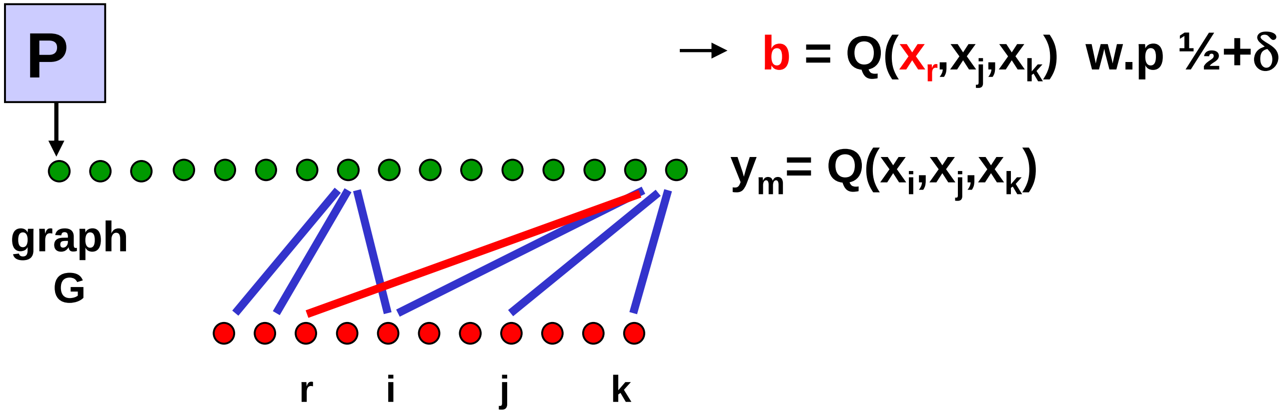
$y_m = Q(x_i,x_j,x_k)$

Prediction $\Rightarrow$ Inversion

Idea: Run the predictor on a modified graph G'

- Assuming P is right: $b=y_m$ iff $x_i=x_r$
- We learned a noisy 2-LIN equation $x_r \oplus x_i = \sigma$

Invert by collecting many eq's + error-correction + re-randomization



Crypto Implications

Conjecture: f is one-way
for predicate Q, random graph with m outputs

m=1.1n outputs

$m=n^{1.1}$ outputs

Thm: ∃ Linear-stretch local PRG

Thm [A-Kacholon19**]:**

∃ poly-stretch local PRG

Tool: Sampling highly-unbalanced expanders with $n^{-\omega(1)}$ err

Drawbacks:

- Polynomial security loss
- Yields collections of local primitives
- Relies on hardness for most graphs/most expanders

Q: from OWFs to PRGs Locally and **generically for arbitrary graph ?**

Almost…

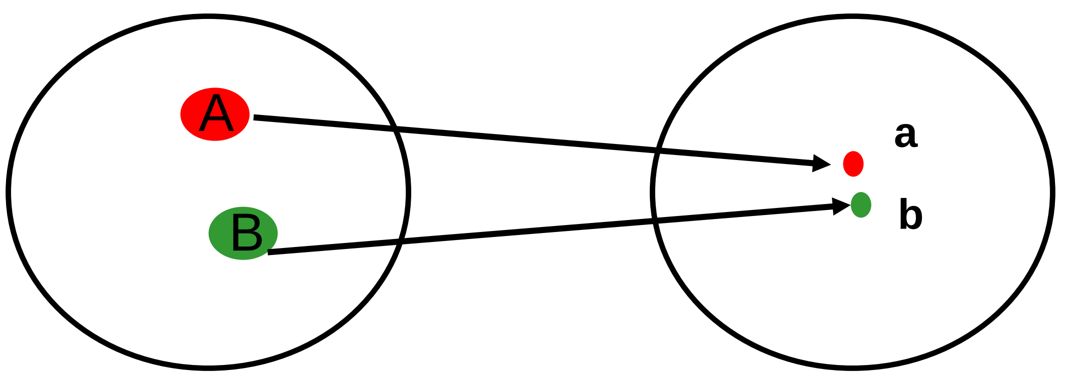
Exploiting exponential hardness [A17]

Thm: α**-almost** regular local OWF with **exp(6α n)** hardness
$\Rightarrow$ local exp–strong PRG with linear stretch

satisfied by Goldreich's original conjecture [Bar-Ish-Ost13]

- Can be based on single function
- Yields single function

|A|< exp(α n) |B|



f:X$\rightarrow$Y

Exploiting exponential hardness [A17]

Thm: α**-almost** regular local OWF with **exp(6α n)** hardness
$\Rightarrow$ local exp–strong PRG with linear stretch

Proof technique yields new candidates for optimal OWFs

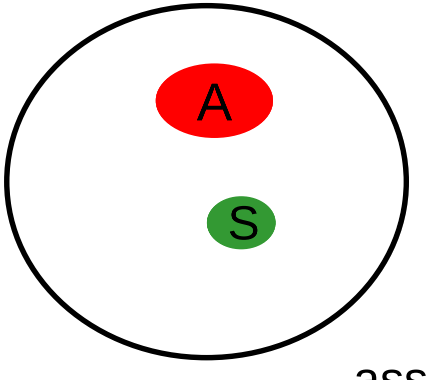
Exploiting exponential hardness [A17]

Thm: α**-almost** regular local OWF with **exp(6α n)** hardness
$\Rightarrow$ local exp–strong PRG with linear stretch

Proof extends to the worst-case setting

- If "smooth" 3-CNF are exponentially hard to satisfy
  Then 3-CNF are exponentially hard to approximate

- smooth-ETH $\Rightarrow$ Gap-ETH



|A|< exp(α n) |S|

assignments

S= Satisfying Assignments

A= Almost-satisfying assignments

Exploiting exponential hardness [A17]

Thm: α**-almost** regular local OWF with **exp(6α n)** hardness

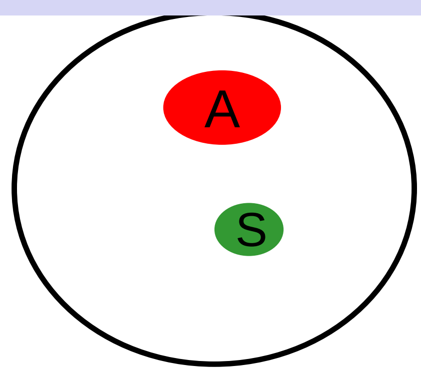
$\Rightarrow$ local exp–strong PRG with linear stretch

Proof extends to the worst-case setting

- If “smooth” 3-CNF are exponentially hard to satisfy

Typical instances are smooth:

local-OWF/random-CNFs $\Rightarrow$ smooth-ETH $\Rightarrow$ Gap-ETH



|A|< exp(α n) |S|

assignments

S= Satisfying Assignments

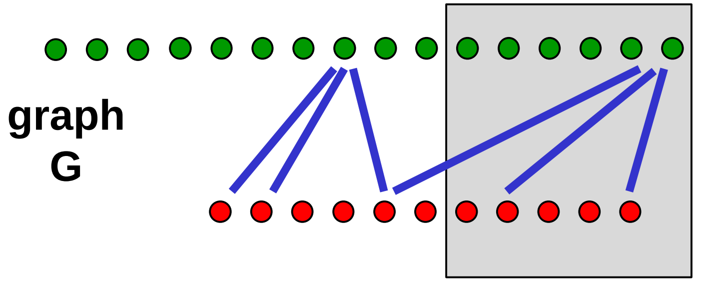
A= Almost-satisfying assignments

Exploiting exponential hardness [A17]

Thm: α**-almost** regular local OWF with **exp(6α n)** hardness
$\Rightarrow$ local exp–strong PRG with linear stretch

Proof relies on new local hardcore function

Assumption: Can't be inverted
Implication: Can't be predicted



Local hard-core function
"local encoding" of the [IKOS08] function– see Yuval's talk

Universal Hardcore Functions

Let f be 2^s -hard one-way function

Definition: g is **hardcore function** if:

- g(x,r) is pseudorandom given f(x)
- Expansion: $|g(x,r)|-|r|=\Omega(s)$

Complexity of hard-core functions:

- [GL89]: $O(n^2)$ randomness/circuit-size
- [Gol]: O(n) randomness, $\tilde{O}(n)$ circuit-size
- [BIO13]: O(n) randomness, O(n) circuit-size

  Building on [GL89,HMS04,IKOS08]; See Yuval's talk

All constructions are public-coins $g(x,r)=(r,g_r(x))$

Cannot be implemented locally !

New 3-local construction with O(n) private-coins

Building on [AIK04,BIO13]

Open Problems

- Pseudorandomness against low-degree $\mathbb{F}_2$ polynomials?

- Smoothness vs Hardness?
  - ETH=>smooth-ETH
  - Relate smoothness to graph structure

- Local exp-OWF must be somewhat-regular?
  - local exp-OWF => local exp-PRG?

- How much expansion is needed for security?
  - Aggressive relation => No fast expanders [OliveiraSanthanamTell18]

- Other implications of PRG-conjecture ?
  - Public-key encryption [ABW10]
  - Hardness of learning (log n)-juntas [ABW10]
  - Inapproximability of densest sub-hypergraph [A11]

Conclusion

Ambitious Goal:
Theory for Avg-case hardness over NATURAL Distributions

- Unconditional Hardness against concrete algorithms
- More Structural theory?
- Algorithmic Hierarchy?
- Distribution-preserving Reductions?

    (Hardness of Rand-SAT=>hardness of planted clique?)

Local Crypto provides some partial results along these lines
Forms strong hypothesis for avg-case hardness

Thank You!