Fourier Analysis on Boolean Functions and its Applications

Presented by Zhiwei Zhang
Oct 18, 2018

Boolean Functions

$f : \{0,1\}^n \to \{0,1\}$

Alternative representation: $f : \{1,-1\}^n \to \{1,-1\}$

$$MAX_2 : \{+1,-1\}^2 \to \{+1,-1\}$$

$$MAX_2(+1,+1) = +1$$
$$MAX_2(-1,+1) = +1$$
$$MAX_2(+1,-1) = +1$$
$$MAX_2(-1,-1) = -1$$

Majority: $Maj(x_1, ..., x_n) = sgn(x_1 + ... + x_n)$

Dictator: $Dic_i(x_1, ..., x_n) = x_i$

Voting Scheme

- n(odd number) voters, 2 candidates: “Majority”
- 3 or more candidates?

    35% $a>b>c$

    25% $b>a>c$

    40% $c>a>b$

  60% people hate c!

Voting Scheme

- n(odd number) voters, 2 candidates: "Majority"
- 3 or more candidates?

35% a>b>c	(+1,+1,-1)
25% b>a>c	(-1,+1,-1)
40% c>a>b	(+1,-1,+1)

$(f(x),f(y), f(z))$

"rational result" $(f(x),f(y),f(z)) \neq (-1,-1,-1)$ or $(1,1,1)$

"Condorcet Election" Always has a rational result?

Majority doesn't always work

$$Maj(x_1, ..., x_n) = sgn(x_1 + ... + x_n)$$

34% a>b>c
33% b>a>c
32% c>a>b

f is Maj

No Condorcet Winner!

Does "ideal" function exist?

Ideal Voting Scheme?

Arrow's Impossibility Theorem

Suppose $f : \{-1,1\}^n \to \{-1,1\}$

is a voting rule used in a 3-candidate Condorcet election such that there is always a Condorcet winner and f is unanimous.

Then f is a dictator function

$$f(-1,-1,...,-1) = -1$$
$$f(1,1,...,1) = 1$$

$$Dic_i(x_1, ..., x_n) = x_i$$

Proved by Fourier Analysis on Boolean functions

Outline

1. Basic Concepts and Notations

2. Examples of Applications
   2.1 Probabilistic Inference
   2.2 Model Counting
   2.3 Other Applications

3. Summary

Vector Representation of Boolean Functions

$$MAX_2(+1,+1) = +1$$
$$MAX_2(-1,+1) = +1$$
$$MAX_2(+1,-1) = +1$$
$$MAX_2(-1,-1) = -1$$

$$\begin{bmatrix} +1 \\ +1 \\ +1 \\ -1 \end{bmatrix}$$

Vector Representation of Boolean functions

$f : \{1,-1\}^n \to \{1,-1\}$ can be represented by vector basis of R^{2^n}

Inner Product of Two Boolean Functions

$$< f, g >= \frac{1}{2^n} \sum_{x \in \{+1,-1\}^n} f(x)g(x)$$

$$= E_x[f(x)g(x)]$$

f and g are orthonormal iff $< f, g >= 0$

Representation Boolean Function by Orthonormal Basis

$f : \{1, -1\}^n \to R$ can be represented by

$$v \in R^{2^n}$$

n=2	$[1, 0, 0, 0]$	$[0, 1, 0, 0]$	$[0, 0, 1, 0]$	$[0, 0, 0, 1]$
	$\mathbb{I}_{(+1,+1)}$	$\mathbb{I}_{(+1,-1)}$	$\mathbb{I}_{(-1,+1)}$	$\mathbb{I}_{(-1,-1)}$

$$f(x) = \sum_{a \in \{-1,1\}^n} \mathbb{I}_a(x) f(a)$$

Represent $MAX_2(x)$

n=2 $\quad [1,0,0,0] \quad [0,1,0,0] \quad [0,0,1,0] \quad [0,0,0,1]$

$\quad \mathbb{I}_{(+1,+1)} \quad \mathbb{I}_{(+1,-1)} \quad \mathbb{I}_{(-1,+1)} \quad \mathbb{I}_{(-1,-1)}$

$$\boxed{f(x) = \sum_{a \in \{-1,1\}^n} \mathbb{I}_a(x) f(a)}$$

$$\begin{aligned} MAX_2(x) &= (\frac{1+x_1}{2})(\frac{1+x_2}{2})1 \\ &+(\frac{1-x_1}{2})(\frac{1+x_2}{2})1 \\ &+(\frac{1+x_1}{2})(\frac{1-x_2}{2})1 \\ &+(\frac{1-x_1}{2})(\frac{1-x_2}{2})(-1) = \boxed{\frac{1}{2} + \frac{1}{2}x_1 + \frac{1}{2}x_2 - \frac{1}{2}x_1x_2} \end{aligned}$$

Multilinear Polynomials

$$MAX_2(x) = \boxed{\frac{1}{2} + \frac{1}{2}x_1 + \frac{1}{2}x_2 - \frac{1}{2}x_1x_2}$$

multilinear polynomials!

$$1, x_1, x_2, x_1x_2$$

$$\emptyset \;\; \{1\} \; \{2\} \;\; \{1,2\}$$

Every term corresponds to a subset of [n]!

Parity Functions

$$\chi_S = \prod_{i \in S} x_i, S \subseteq [n]$$ e.g. $x_1 x_2 x_5$, $x_3 x_7 x_9 x_{10}$

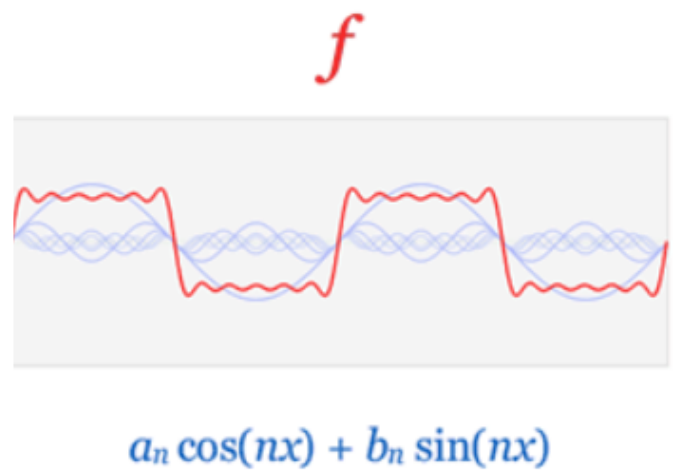
Compute the parity of number of -1s in x_S

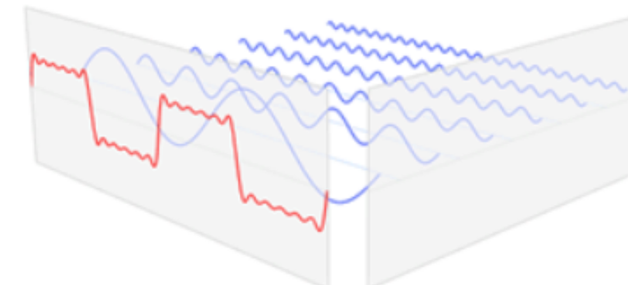
$$< \chi(S), \chi(T) >= \begin{cases} 0, S \neq T \\ 1, S = T \end{cases}$$

Parity functions $\{S \in [n] | \prod_{i \in S} x_i\}$

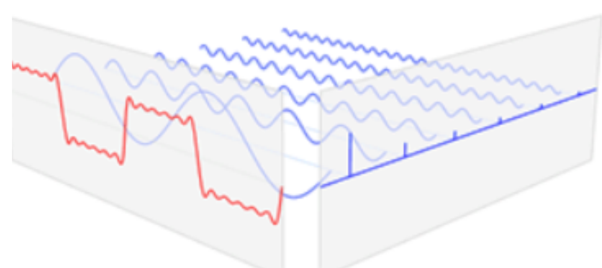
are orthonormal basis!

Fourier Analysis in Signal Processing





Time domain



Frequency domain

$$f(t) = \frac{1}{2}a_0 + \sum_{w=1}^{\infty}(a_w cos(wx) + b_w sin(wx))$$

Fourier Analysis in Signal Processing

$$f(t) = \frac{1}{2}a_0 + \sum_{w=1}^{\infty}(a_w cos(wx) + b_w sin(wx))$$

$$a_w = \frac{1}{T}\int_0^T f(t) sin(\frac{2\pi wt}{T})$$

$$MAX_2(x) = \boxed{\frac{1}{2} + \frac{1}{2}x_1 + \frac{1}{2}x_2 - \frac{1}{2}x_1x_2}$$

$$\frac{1}{2}\ \chi_{\emptyset}\ + \frac{1}{2}\ \chi_{\{x_1\}} + \frac{1}{2}\ \chi_{\{x_2\}} - \frac{1}{2}\ \chi_{\{x_1,x_2\}}$$

Fourier Transform on Boolean Function

$$f(t) = \frac{1}{2}a_0 + \sum_{w=1}^{\infty}(a_w cos(wx) + b_w sin(wx))$$

$$a_w = \frac{1}{T}\int_0^T f(t) sin(\frac{2\pi wt}{T})$$

$$MAX_2(x) = \frac{1}{2}\,\chi_\emptyset + \frac{1}{2}\,\chi_{\{x_1\}} + \frac{1}{2}\,\chi_{\{x_2\}} - \frac{1}{2}\,\chi_{\{x_1,x_2\}}$$

$$\hat{f}(\emptyset) \qquad \hat{f}(\{x_1\}) \qquad \hat{f}(\{x_2\}) \qquad \hat{f}(\{x_1, x_2\})$$

$$\boxed{\hat{f}(S) =< \chi_S, f >= \frac{1}{2^n}\sum_x \chi_S(x) f(x)}$$

Projection of f on χ_S !

Fourier Transform on Boolean Function

$$\hat{f}(S) = < \chi_S, f > = \frac{1}{2^n} \sum_x \chi_S(x) f(x)$$

Fourier Transform

$$f(x) = \sum_{S \subseteq [n]} \hat{f}(S) \chi_S(x)$$

Fourier Transform on Boolean Function

$$\boxed{\hat{f}(S) = <\chi_S, f> = \frac{1}{2^n}\sum_x \chi_S(x) f(x)}$$

Fourier Transform

$$f(x) = \sum_{S\subseteq[n]} \hat{f}(S)\chi_S(x)$$

$$MAX_2(x) = \frac{1}{2} + \frac{1}{2}x_1 + \frac{1}{2}x_2 - \frac{1}{2}x_1x_2$$

$$\widehat{MAX}2(\{x_1\}) = <MAX_2, \chi_{\{x_1\}}> = \frac{1}{4} \begin{bmatrix} +1 \\ +1 \\ +1 \\ -1 \end{bmatrix} \begin{bmatrix} +1 \\ -1 \\ +1 \\ -1 \end{bmatrix} = \frac{1}{2}$$

Plancherel and Parseval

Plancherel's Theorem

$$< f, g >= \sum_{S \in [n]} \hat{f}(S)\hat{g}(S)$$

Parseval's Theorem

$$\sum_{S \in [n]} \hat{f}(S)^2 =< f, f >= 1$$

(Conservation of Energy)

Fourier Spectrum

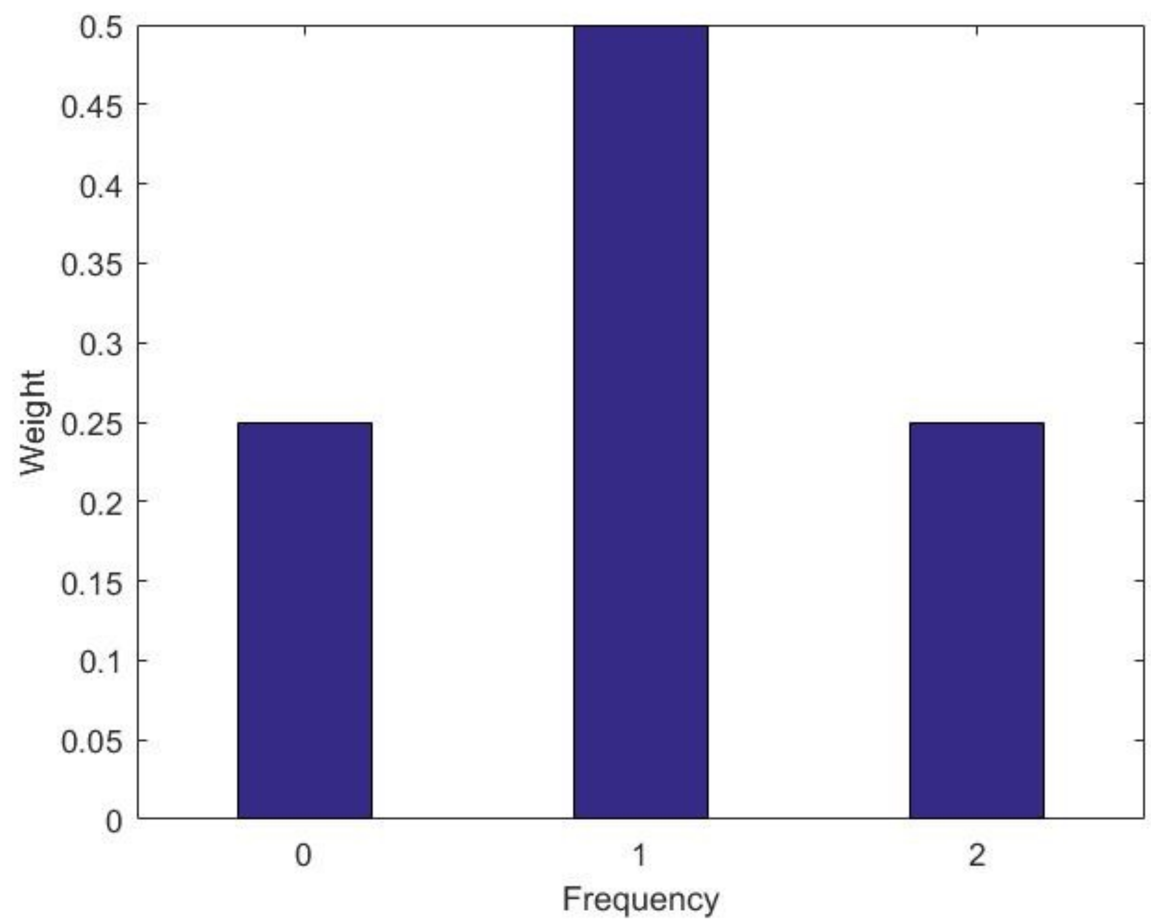
$$W_k = \sum_{|S|=k} \hat{f}(S)^2$$

$$\sum_k W_k = 1$$

Fourier Spectrum

$$W_k = \sum_{|S|=k} \hat{f}(S)^2$$

$$MAX_2(x) \boxed{= \frac{1}{2} + \frac{1}{2}x_1 + \frac{1}{2}x_2 - \frac{1}{2}x_1 x_2}$$



$k = 0 \;\; : \; \hat{f}(\emptyset) = E_x(f)$ "DC component"

small k: "Low frequency component"

large k: "High frequency component"

Noise Sensitivity

y: changing each bit of x with probability p

$$NS_p(f) = Pr[f(x) \neq f(y)]$$

$$= \frac{1}{2} - \frac{1}{2} \sum_{S \subseteq [n]} (1-2p)^{|S|} \hat{f}(S)^2$$

$NS_p(f)$ is large

$\Updownarrow$

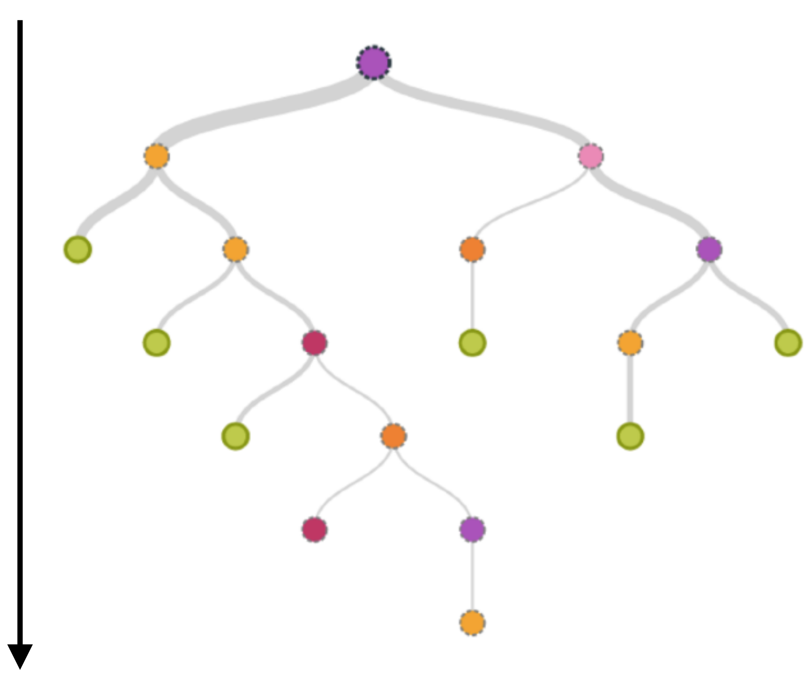
f has large weights on high-frequency component

Many Real-life Functions are $\frac{\text{Not Sensitive}}{\text{with Low Frequency}}$

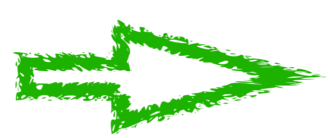
O'Donnell, 2008

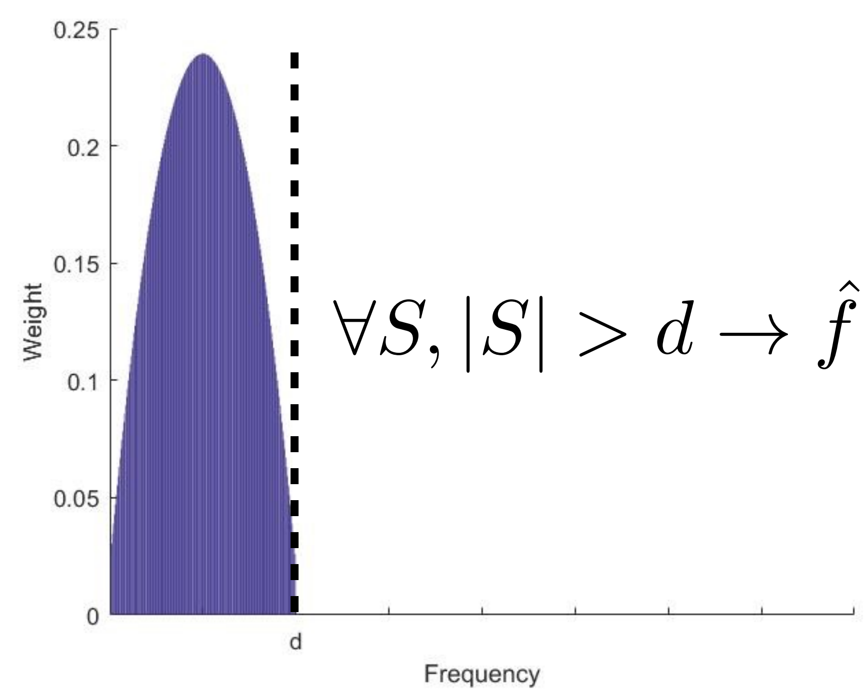
f can be represented by a decision tree of depth d

$$\implies \forall S, |S| > d \rightarrow \hat{f}(S) = 0$$



$depth = d$





$\forall S, |S| > d \rightarrow \hat{f}(S) = 0$

Many Real-life Functions are $\dfrac{\text{Not Sensitive}}{\text{with Low Frequency}}$

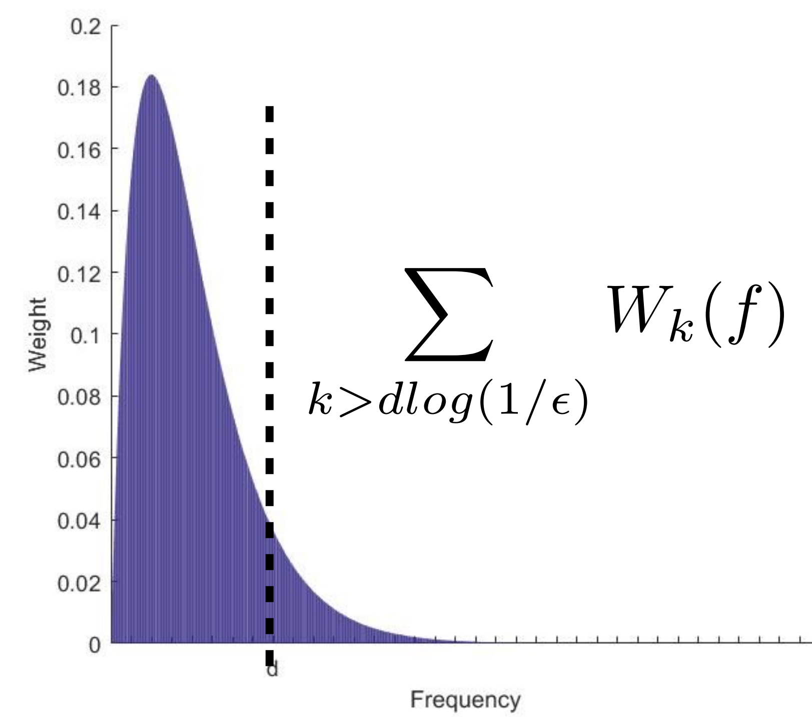
Linial, 1993

f can be represented by a CNF(DNF) with width d

$$\Longrightarrow \sum_{k > d log(1/\epsilon)} W_k(f) < \epsilon$$

$$\bigvee_i (x_{i_1} \wedge x_{i_2} \wedge \cdots x_{i_d})$$

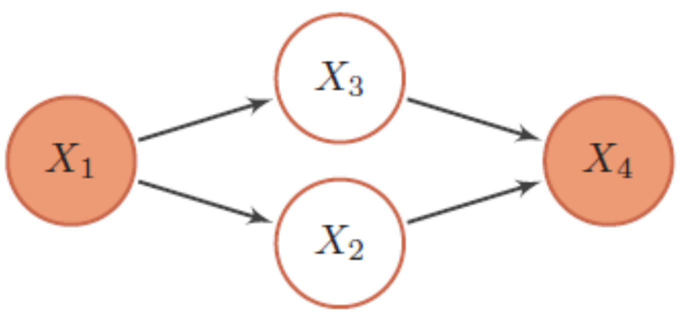
$$width = d$$



$$\sum_{k > d log(1/\epsilon)} W_k(f) < \epsilon$$

Application 1: Fourier Analysis in Probabilistic Inference

Graphic probabilistic Model



$$Pr[x_4 = 1 | x_2 = 1, x_3 = 0] = 0.3$$

$$Pr(x_4) = Pr(x_4|x_2, x_3)Pr(x_3|x_1)Pr(x_2|x_1)Pr(x_1)$$

$$Pr(x) = \frac{1}{Z}\prod_{i=1}^{k} \underline{f_i(x)}$$

"factor"

$$\boxed{\text{"Partition function": } Z = \sum_x \prod_{i=1}^{k} f_i(x)}$$

Counting is a special case of computing partition function

Variable Elimination

$$Z = \sum_x \prod_{i=1}^{k} f_i(x)$$

For every x_i compute

$$f_{x_i} = \prod_{x_i \in AP(f_i)}^{k} f_i(x, x_i = 0) + \prod_{x_i \in AP(f_i)}^{k} f_i(x, x_i = 1)$$

to eliminate x_i in all factors
until all variables are eliminated

Sum and Production of boolean functions

Exact computation of Z is intractable!

$f_i(x)$ usually concentrate on low frequency component

Keep Main Component

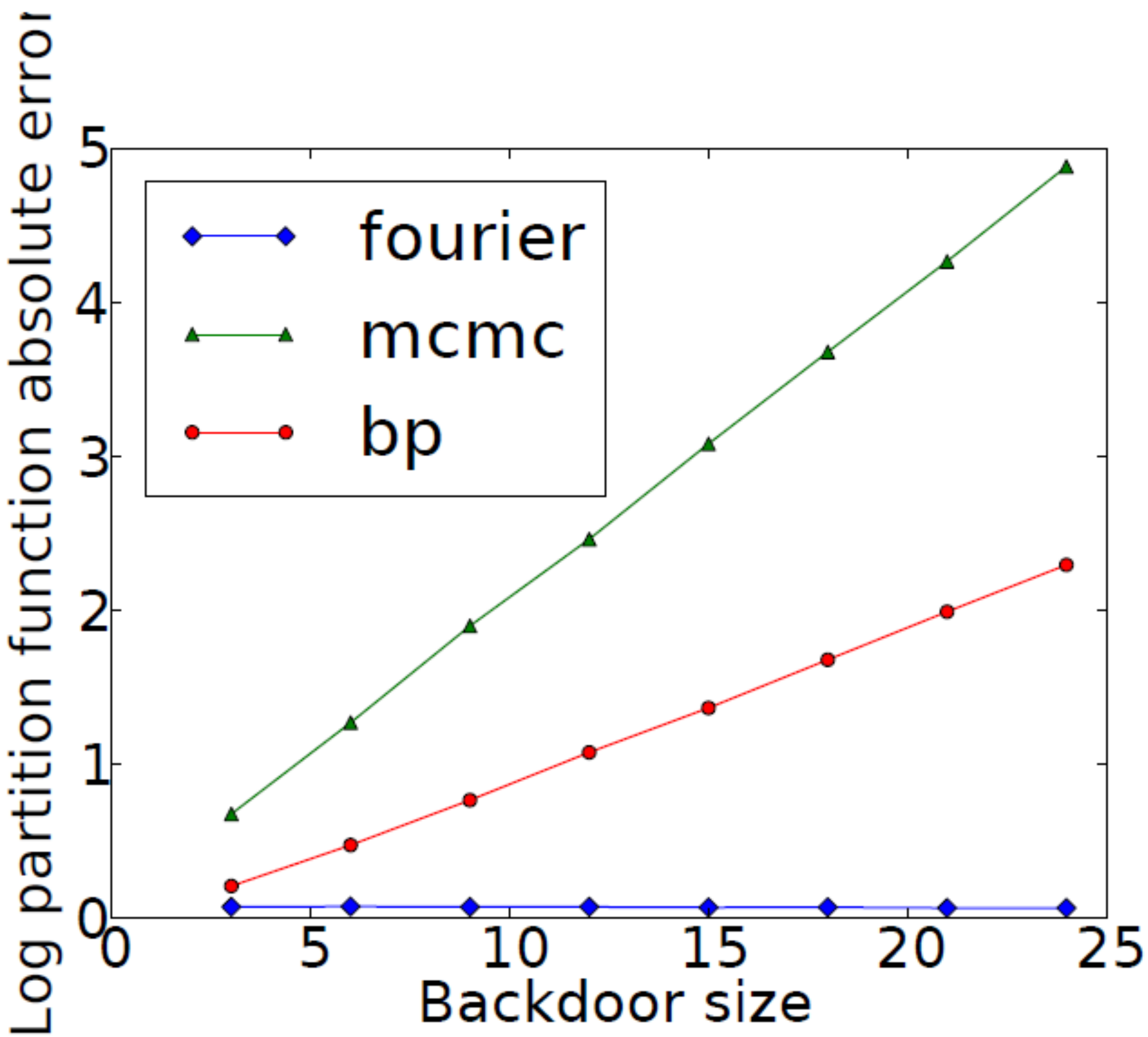
"Partition function" $Z = \sum_x \prod_{i=1}^{k} f_i(x)$

$f_i(x)$ usually concentrate on low frequency component

$\boxed{\text{Only keep Fourier coefficients with } |S|<d}$

	Point-value	$\overset{O(n2^n)}{\Longleftrightarrow}$	Fourier Coefficient
Add	$O(2^n)$		$O(n^d)$
Mul	$O(2^n)$		$O(n^{2d})$

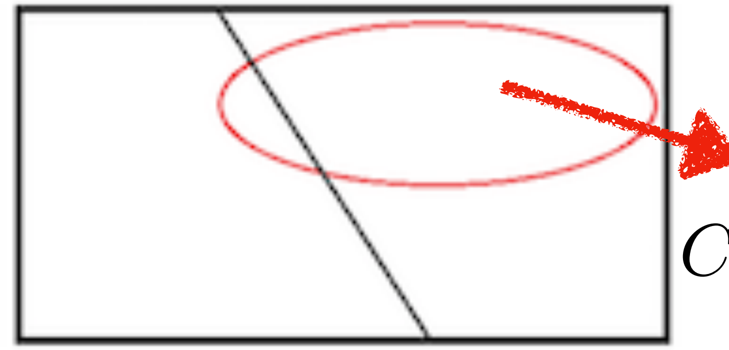
Computing Partition Function with Fourier Representation



Fourier method beats MCMC & Brief Propagation

Application 2: Hash-based Model Counting

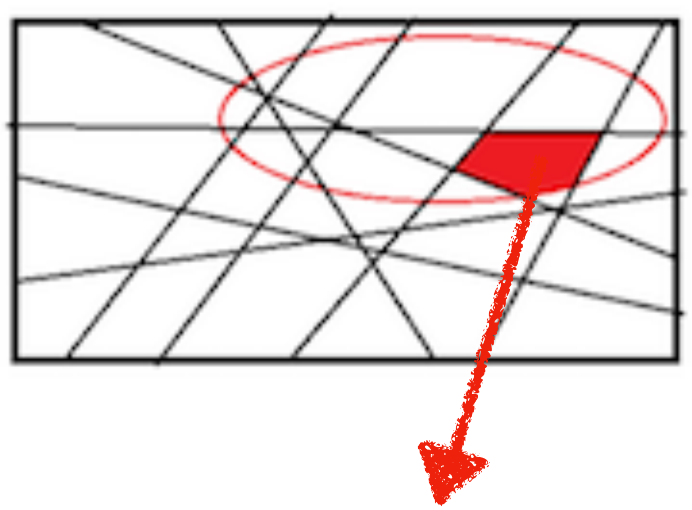
Model Counting: $\#(\tau)$, $F(\tau) = 1$



$C_1 = F \cap h_1^{-1}(a_1)$

$$E(|C_m|) = \frac{|F|}{2^m}$$

$$\boxed{C = 2^m |C_m|}$$



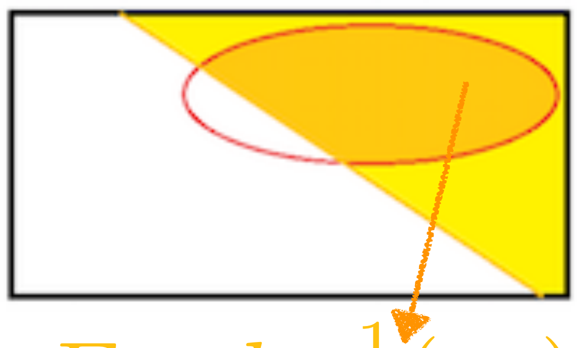
$C_m = F \cap h_1^{-1}(a) \cap \cdots h_m^{-1}(a_m)$

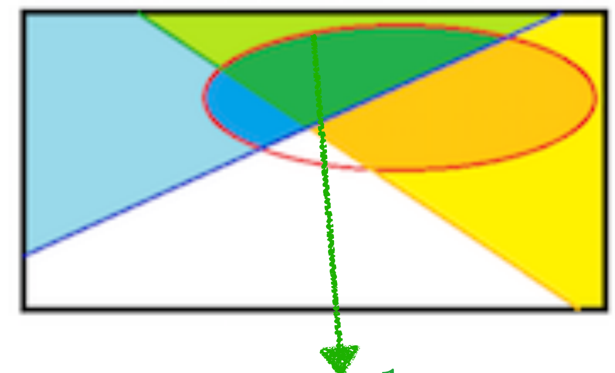
Arbitrary hash function
cannot guarantee each cell
contain approximately
same number of solutions

$Var(|C_m|)$ need to be controlled!

What is a Good Hash Function?

Use Majority as hash function:





$Var(|F \cap h^{-1}(a)|)$ is large!

A good hash function should be able to "split" close points

What is a Good Hash Function?

A good hash function should be able to "split" close points

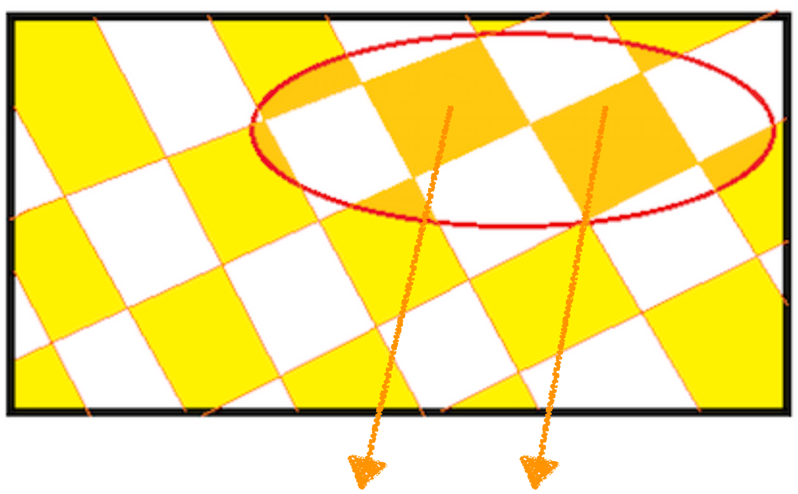
For "close" x and y ($distance(x, y) = d$ for small d)

$$NS'_d(f) = Pr[h(x) \neq h(y)] \sim W_{High\,freq}$$

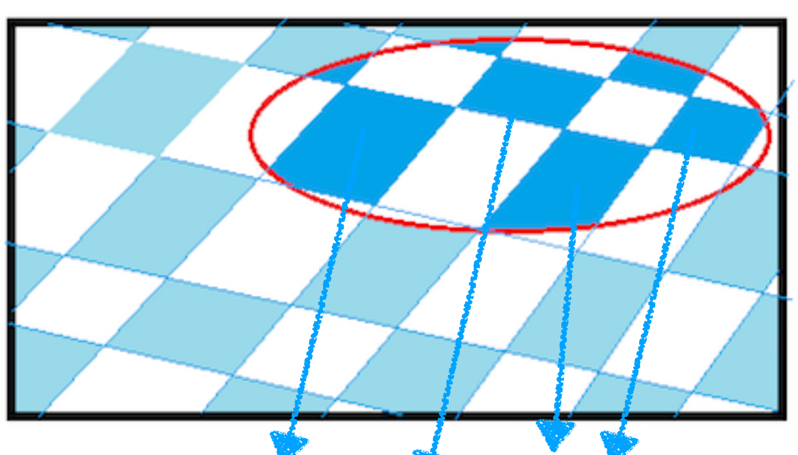
XOR is known to have maximum NS'_1

Counting by XORs

XOR is known to have maximum NS'_1

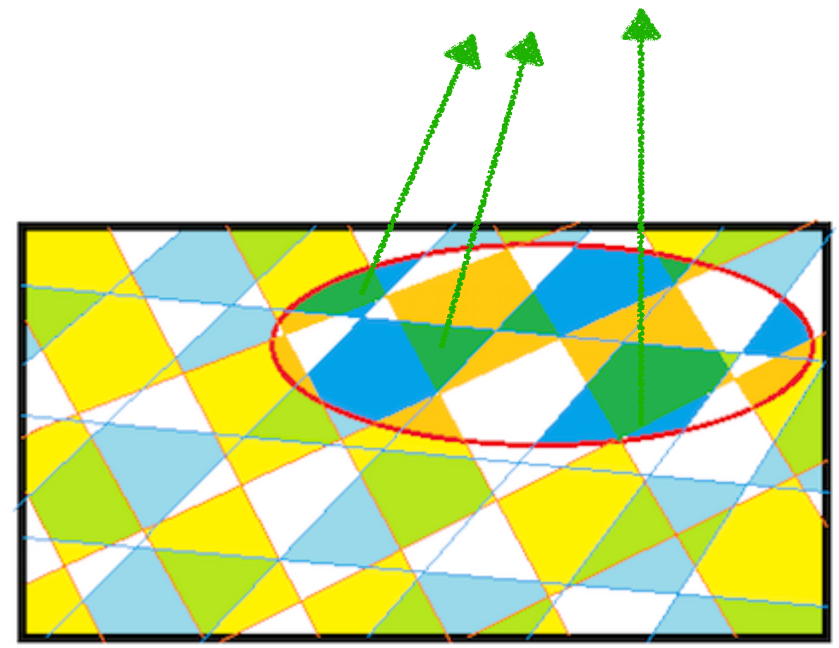


$F \cap h_1^{-1}(a_1)$



$F \cap h_2^{-1}(a_2)$

$C_2 = F \cap h_1^{-1}(a_1) \cap h_1^{-1}(a_2)$



Close points are "split"

$Var(|F \cap h^{-1}(a)|)$ is small!

Beyond XORs

- "Hard Problems are almost everywhere in CNF-XOR formulas"
- Monotone Boolean function: $f(x) \leq f(x, x_i = 1)$

  e.g. Majority

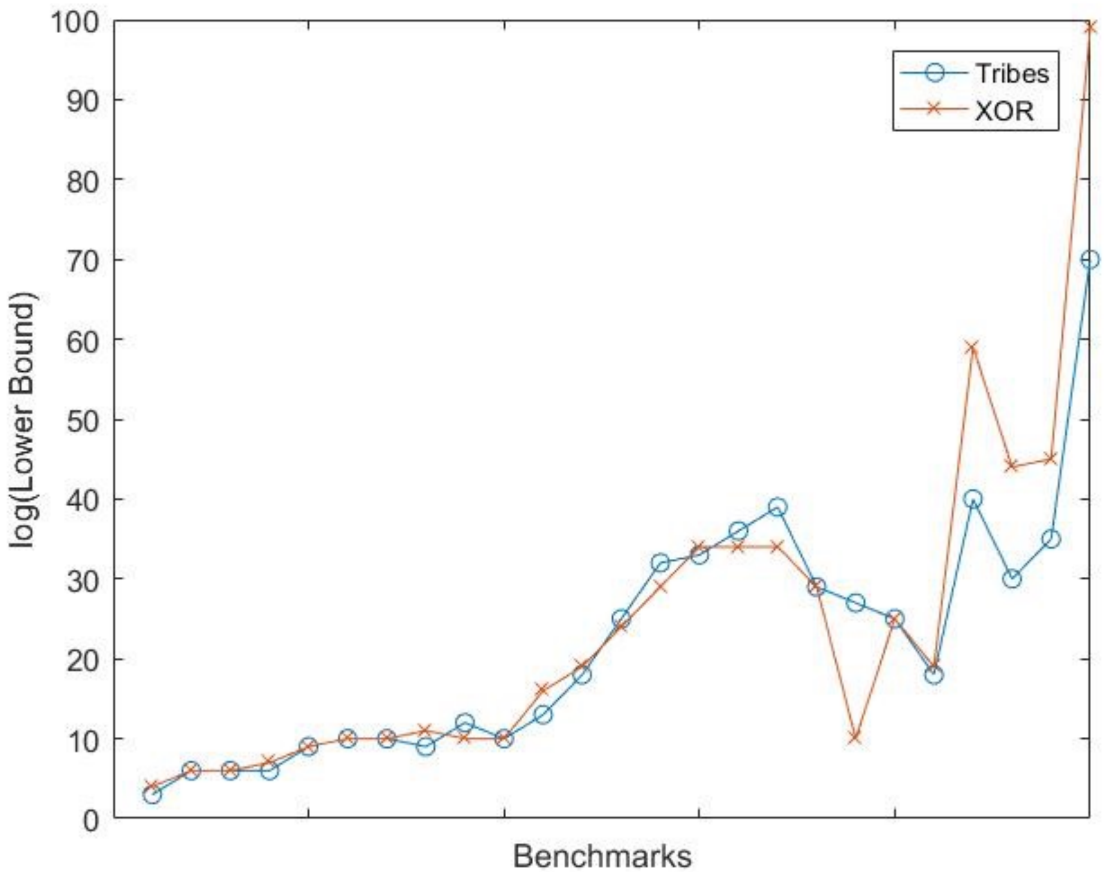
  Monotone functions are easy to optimize

- Tribes Function:

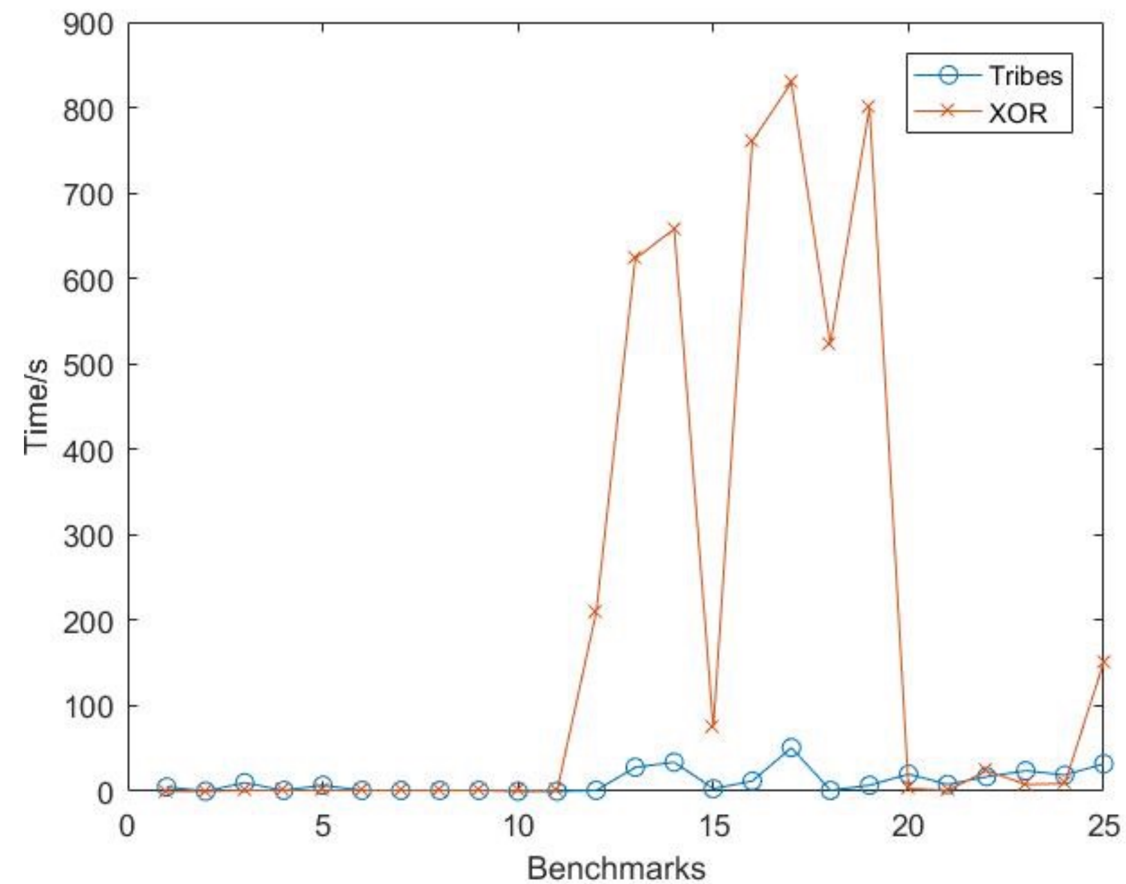
$$TRIBES(x_1, ..., x_n) = \bigvee_{i=1}^{b} (\bigwedge_{j=1}^{a} x_{A_{ij}})$$

- Tribes has maximum noise sensitivity among monotone functions

Hashing with XORs and Tribes



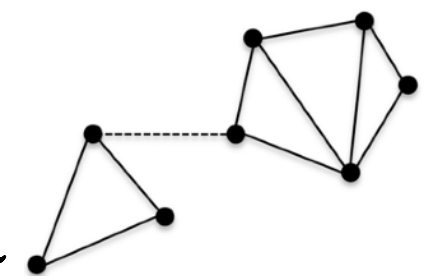
Lower Bound Estimation

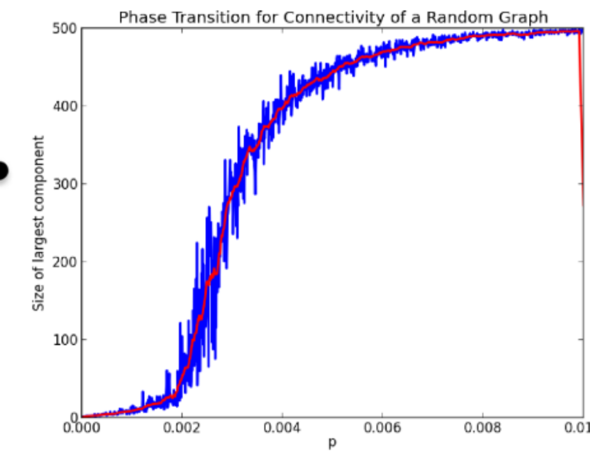


Time

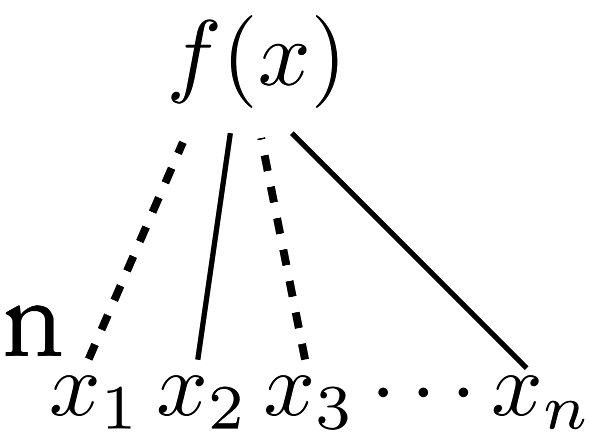
Other Applications

- Phase Transition Phenomena





How sharp is the phase transition of connectivity on random graph?



- Properties testing of Boolean Function

Test whether a boolean function only relies on k variables efficiently

- Proving hardness of Approximation

$$\begin{cases} x_1 + x_5 + x_3 = 0 \\ x_2 + x_8 + x_9 = 1 \\ x_3 + x_4 + x_7 = 0 \end{cases}$$

Max E3-LIN2

It's NP-hard to $(\frac{1}{2} + \epsilon)$ -approximate Max E3-LIN2 for any $\epsilon > 0$

Summary

Fourier Analysis on Boolean function is:

- A way to characterize boolean function on frequency view
- A powerful and versatile tool of both theoretical and practical interest
- An active field in theoretical computer science in last two decades

References

[1]Ryan O'Donnell, Analysis of Boolean Functions
[2]Meel, Kuldeep S. et al. Constrained Sampling and Counting: Universal Hashing Meets SAT Solving
[3]Supratik Chakraborty et.al. Distribution-Aware Sampling andWeighted Model Counting for SAT
[4]Yexiang Xue et.al. Variable Elimination in the Fourier Domain
[5]Gil Kalai et.al. Threshold Phenomena and Influence with Some Perspectives from Mathematics, Computer Science, and Economics
[7]Tudor Achim et.al. Beyond Parity Constraints: Fourier Analysis of Hash Functions for Inference
[8]Ronald de Wolf A Brief Introduction to Fourier Analysis on the Boolean Cube
[9]Sam Spiro, Fourier Analysis of Boolean Functions
[10]ColinWei et.al. General Bounds on Satisfiability Thresholds for Random CSPs via Fourier Analysis
[11]Jeffery M. Dudek et.al. The Hard Problems Are Almost Everywhere For Random CNF-XOR Formulas