E0 206: Theorist's Toolkit December 15, 2020

Lecture 22-23: Analysis of Boolean Functions

Instructor: Anand Louis *Scribe: Bhargav Thankey*

A boolean function is a function defined as $f : \{0,1\}^n \to \{0,1\}$ or more generally as $f : \{0,1\}^n \to \mathbb{R}$. Boolean functions can be used to capture a lot of situations arising in theoretical computer science. For instance, a boolean function can be used to describe the computation of a boolean circuit: the n inputs of the circuit correspond to the n arguments of the function and for any $x \in \{0,1\}^n$, $f(x)$ is the output of the circuit on input x. One can also model an election between two candidates using a boolean function as follows: identify the first candidate with 0 and the second with 1. For each of the n voters, let x_i represent their vote. Then the function f models the election rule, $f(x_1, x_2, .., x_n)$ is the winner of the election when the votes cast are $(x_1, x_2, ..., x_n)$.

Notice that in the second example we could as well have identified the candidates with -1 and 1. As this example suggests, while $f : \{0,1\}^n \to \{0,1\}$ is a common way to define boolean functions, it is not the only way. In fact, in most cases, the exact domain of the function doesn't matter as long as it is binary. For the purpose of this lecture we will use another common way to define boolean functions, $f : \{-1,1\}^n \to \{-1,1\}$ or more generally as $f : \{-1,1\}^n \to \mathbb{R}$. Whatever we will learn for this setting can also be extended to any binary domain.

Note that the set of all boolean functions $f : \{-1,1\}^n \to \mathbb{R}$ is a vector space under the following operations: for any boolean functions f_1, f_2 and any $\alpha \in \mathbb{R}$, $(f_1 + f_2)(x) = f_1(x) + f_2(x)$ and $(\alpha \cdot f_1)(x) = \alpha \cdot f_1(x)$ for all $x \in \{-1,1\}^n$. One way of seeing why this is a vector space is by treating f as a vector in $\mathbb{R}^{2^n}$ whose components are values of f on the vertices of the boolean hypercube $\{-1,1\}^n$. Then, the addition and scalar multiplication operations that we just defined above, are analogous to the normal vector addition and scalar multiplication in $\mathbb{R}^{2^n}$. This also tells us that the dimension of the space of all boolean functions is 2^n.

1 Boolean Functions as Polynomials

Every boolean function can be expressed as a polynomial. For example, consider the function $f : \{-1,1\} \to \mathbb{R}$. Notice that

$$\frac{1}{2}(1-x) = \begin{cases} 0 & \text{if } x = 1 \\ 1 & \text{if } x = -1 \end{cases} \qquad \text{and} \qquad \frac{1}{2}(1+x) = \begin{cases} 1 & \text{if } x = 1 \\ 0 & \text{if } x = -1. \end{cases}$$

Therefore,

$$f(x) = f(-1)\frac{1}{2}(1-x) + f(1)\frac{1}{2}(1+x).$$

More generally, for any $a \in \{-1,1\}^n$,

$$\left(\frac{1+a_1x_1}{2}\right) \cdot \left(\frac{1+a_2x_2}{2}\right) \cdot \ \cdots \ \cdot \left(\frac{1+a_nx_n}{2}\right) = \begin{cases} 1 & \text{if } x_i = a_i \ \forall i \in [n], \\ 0 & \text{otherwise.} \end{cases}$$

To see why this is true, observe that if $x_i = a_i \ \forall i \in [n]$, then $a_ix_i = a_i^2 = 1$ as $a_i = \{-1,1\}^n$ and

$$\frac{1+a_ix_i}{2} = 1$$

for all $i \in [n]$. On the other hand, if there exists an $i \in [n]$ such that $a_i \neq x_i$, then $a_i x_i = -1$ and

$$\frac{1 + a_i x_i}{2} = 0.$$

Thus,

$$f(x) = \sum_{a \in \{-1,1\}^n} f(a) \prod_{i \in [n]} \left(\frac{1 + a_i x_i}{2} \right). \tag{1}$$

Note that the polynomial on the right is a multilinear polynomial i.e. the highest power of each x_i is at most 1. Simplifying this polynomial and writing it in the sum of monomials representation, we get the Fourier expansion of f,

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

where $\chi_S := \prod_{i \in S} x_i$ is a parity function and $\widehat{f}_S \in \mathbb{R}$ is called the Fourier coefficient of f on S. The Fourier expansion is unique: by simplifying 1 we can write every function $f : \{-1,1\}^n \to \{-1,1\}$ as a linear combination of parity functions. Thus, $\{\chi_S\}_{S \subseteq [n]}$ spans the space of all boolean functions. Moreover, recall that the dimension of this space $2^n = \left| \{\chi_S\}_{S \subseteq [n]} \right|$. So, $\{\chi_S\}_{S \subseteq [n]}$ must be a linearly independent set and therefore form a basis of the space of boolean functions. Hence, every boolean function can be expressed as a unique linear combination of the parity functions.

Let us now look at some functions and their Fourier expansions.

Example 1. *Define*

$$\max(x_1, x_2) = \begin{cases} -1 & \text{if } x_1 = x_2 = -1 \\ 1 & \text{otherwise.} \end{cases}$$

Then,

$$\begin{aligned} \max(x_1, x_2) = \; & \max(1,1) \left(\frac{1 + x_1}{2} \right) \left(\frac{1 + x_1}{2} \right) + \max(1,-1) \left(\frac{1 + x_1}{2} \right) \left(\frac{1 - x_2}{2} \right) \\ & + \max(-1,1) \left(\frac{1 - x_1}{2} \right) \left(\frac{1 + x_2}{2} \right) + \max(-1,-1) \left(\frac{1 - x_1}{2} \right) \left(\frac{1 - x_2}{2} \right) \\ = \; & \frac{1}{4}(1 + x_1)(1 + x_1) + \frac{1}{4}(1 + x_1)(1 - x_2) + \frac{1}{4}(1 - x_1)(1 + x_2) - \frac{1}{4}(1 - x_1)(1 - x_2) \\ = \; & \frac{1}{2} + \frac{1}{2} x_1 + \frac{1}{2} x_2 - \frac{1}{2} x_1 x_2. \end{aligned}$$

Here note that the Fourier coefficients are not in $\{-1,1\}$ even though the range of max is $\{-1,1\}$. In general, the Fourier coefficients can be any real numbers, even when the range of the function is $\{-1,1\}$.

Example 2. *Define*

$$f(x_1, x_2, x_3) = \begin{cases} -1 & \text{if } x \text{ has an odd numer of -1s,} \\ 1 & \text{has an even number of -1s.} \end{cases}$$

It can be verified that $f(x_1, x_2, x_3) = x_1 x_2 x_3$. More generally, if f is analogously defined over $\{-1,1\}^n$, then $f(x) = \prod_{i \in [n]} x_i$.

As the Fourier expansion is just a linear combination of parity functions, let us study these functions in more detail.

2 Parity Functions

As we have already seen, the set of all boolean functions $f : \{-1,1\}^n \to \mathbb{R}$ forms a vector space of dimension 2^n over $\mathbb{R}$. Now, for any vector space over $\mathbb{R}$ (in fact, over the set of complex numbers $\mathbb{C}$) we can define an inner product. For two functions f, g, their inner product is defined as follows:

$$\langle f, g \rangle := \mathop{\mathbb{E}}_{x \sim \{-1,1\}^n} [f(x)g(x)] .$$

Then, for any $S, T \subseteq [n]$,

$$\begin{aligned}
\langle \chi_S, \chi_T \rangle &= \mathop{\mathbb{E}}_{x \sim \{-1,1\}^n} \left[\prod_{i \in S} x_i \prod_{j \in T} x_j \right] \\
&= \mathop{\mathbb{E}}_{x \sim \{-1,1\}^n} \left[\prod_{i \in S \cap T} x_i^2 \prod_{j \in S \Delta T} x_j \right] \\
&= \mathop{\mathbb{E}}_{x \sim \{-1,1\}^n} \left[\prod_{j \in S \Delta T} x_j \right]
\end{aligned}$$

since $x_i^2 = 1$. Thus,

$$\langle \chi_S, \chi_T \rangle = \begin{cases} 1 & \text{if } S = T \\ 0 & \text{if } S \neq T. \end{cases}$$

As we have already seen, $\{\chi_S\}_{S \subseteq [n]}$ is a basis of the space of all boolean functions. This along with what we have just shown, yields the following theorem.

Theorem 3. *$\{\chi_S\}_{S \subseteq [n]}$ forms an orthonormal basis of the vector space of functions $f : \{-1,1\}^n \to \mathbb{R}$.*

Moreover, as we saw in the last lecture, $\{\chi_S\}_{S \subseteq [n]}$ are the eigenfunctions of the hypercube graph.

Proposition 4. $\widehat{f_S} = \langle f, \chi_S \rangle$.

Proof.

$$\langle f, \chi_S \rangle = \left\langle \sum_{T \subseteq [n]} \widehat{f_T} \chi_T, \chi_S \right\rangle = \mathop{\mathbb{E}}_{x} \left[\left(\sum_{T \subseteq [n]} \widehat{f_T} \chi_T \right) \cdot \chi_S \right] = \widehat{f_S} \langle \chi_S, \chi_S \rangle = \widehat{f_S}.$$

□

Also, $\|f\|_2 = \sqrt{\langle f, f \rangle} = \sqrt{\mathbb{E}_x [f(x)^2]}$. Thus, for $f : \{-1,1\}^n \to \{-1,1\}$, $\|f\|_2 = 1$.

The distance between two boolean functions f and g - denoted by $\text{dist}(f,g)$ - is defined as $\text{dist}(f,g) := \Pr_x [f(x) \neq g(x)]$. Then, we can relate the distance to the inner product as follows:

$$\langle f, g \rangle = \mathop{\mathbb{E}}_{x} [f(x)g(x)]$$

$$
\begin{aligned}
&= \Pr_x[f(x) = g(x)] - \Pr_x[f(x) \neq g(x)] && (\text{as } f(x)g(x) \in \{-1,1\}) \\
&= 1 - 2\Pr_x[f(x) \neq g(x)] && (\text{as } \Pr_x[f(x) = g(x)] + \Pr_x[f(x) \neq g(x)] = 1) \\
&= 1 - 2\text{dist}(f,g).
\end{aligned}
$$

Now let us prove two important theorems, Plancherel's theorem and its special case Parseval's theorem.

Theorem 5 (Plancherel's theorem)**.** *For any $f, g : \{-1,1\}^n \to \mathbb{R}$, $\langle f, g\rangle = \sum_{S\subseteq[n]} \widehat{f}_S \widehat{g}_S$.*

Proof.

$$
\begin{aligned}
\langle f, g\rangle &= \left\langle \sum_{S\subseteq[n]} \widehat{f}_S \chi_S, \sum_{T\subseteq[n]} \widehat{g}_T \chi_T \right\rangle \\
&= \mathop{\mathbb{E}}_x \left[\sum_{S,T\subseteq[n]} \widehat{f}_S \widehat{g}_T \chi_S \chi_T \right] \\
&= \sum_{S,T\subseteq[n]} \widehat{f}_S \widehat{g}_T \mathop{\mathbb{E}}_x [\chi_S \chi_T] \\
&= \sum_{S\subseteq[n]} \widehat{f}_S \widehat{g}_S.
\end{aligned}
$$

□

Theorem 6 (Parseval's theorem)**.** *For any $f : \{-1,1\}^n \to \mathbb{R}$, $\langle f, f\rangle = \sum_{S\subseteq[n]} \widehat{f}_S^2$.*

Proof. Use Plancherel's theorem with $g = f$. □

We now relate the expected value, variance and covariance of functions with their Fourier coefficients. Let $g(x) = 1$ for all $x \in \{-1,1\}^n$. Then, $\widehat{g}_S = 1$ if and only if $S = \emptyset$. So,

$$
\mathop{\mathbb{E}}_x[f(x)] = \langle f, g\rangle = \sum_{S\subseteq[n]} \widehat{f}_S \widehat{g}_S = \widehat{f}_\emptyset.
$$

Also,

$$
\text{var}(f(x)) = \mathop{\mathbb{E}}_x[f(x)^2] - \left(\mathop{\mathbb{E}}_x[f(x)]\right)^2 = \langle f, f\rangle - \left(\widehat{f}_\emptyset\right)^2 = \sum_{S\neq\emptyset} \widehat{f}_S.
$$

Similarly,

$$
\text{Cov}(f(x), g(x)) = \mathop{\mathbb{E}}_x[f(x)g(x)] - \mathop{\mathbb{E}}_x[f(x)] \mathop{\mathbb{E}}_x[g(x)] = \sum_{S\neq\emptyset} \widehat{f}_S \widehat{g}_S.
$$

3 Fourier expansion over the domain $\{0,1\}^n$

For $b \in \mathbb{F}_2$, let $\chi(b) := (-1)^b$, i.e. $\chi(0_{\mathbb{F}_2}) = 1$ and $\chi(1_{\mathbb{F}_2}) = -1$. For $S \in [n]$, define

$$
\chi_S := \prod_{i\in S} \chi(x_i) = (-1)^{\sum_{i\in S} x_i}.
$$

Then, it can be shown that, just as before,

$$f(x) = \sum_{a \in \{0,1\}^n} f(a) \prod_{i \in [n]} \left(\frac{1 + \chi(a_i)\chi(x_i)}{2} \right)$$
$$= \sum_{S \subseteq [n]} \widehat{f}_S \chi_S.$$

Having seen the basics of analysis for boolean functions, we now look at an application to property testing.

4 Property Testing for Boolean Functions

Property testing is the problem of finding whether a given function has a certain property or not, for example, whether it is linear or not. We will only be interested in property testing for boolean functions. In many cases, we can check whether a boolean function has a certain property or not in time $2^{O(n)}$ by querying the function value at all possible inputs. So, our goal will be to do property testing in sub-exponential time, ideally, in polynomial time. However, notice that this is not a realistic goal: checking whether $f(x) = 1$ for all $x \in \{-1, 1\}^n$ can not be done without querying the function at all 2^n possible inputs. Hence, we will aim for the following more realistic goal: Given black box access to a boolean function f:

1. If f has property P, output YES with "high" probability.
2. If f is "far" from property P, output NO with "high" probability.

To make the goal more precise, we now define what if means for a function to be "far" from a property. We say that two functions, $f, g : \{-1, 1\}^n \to \{-1, 1\}$ are $\varepsilon-$close to each other if $\text{dist}(f, g) \leq \varepsilon$. Let $\mathcal{P}$ be the set of all functions satisfying property P. Then, f is $\varepsilon-$close to P, if

$$\text{dist}(f, \mathcal{P}) := \min_{g \in \mathcal{P}} \text{dist}(f, g) \leq \varepsilon.$$

Now let us see a simple example of property testing, that of checking whether $f = 1\ \forall x \in \{-1, 1\}^n$. Consider the following algorithm:

Algorithm 1: A simple property testing algorithm

Input: Blackbox access to f.
Sample k independent random inputs $x_1, ..., x_k$.
if $f(x_i) = 1\ \forall i \in [k]$ **then**
  return YES
end
else
  return NO
end

Analysis: If $f = 1$, then Pr[Algorithm 1 outputs YES] $= 1$. On the other hand, if f is $\varepsilon-$far from 1, then $\Pr_x[f(x_i) \neq 1] \geq \varepsilon$. Thus, $f(x) = 1$ for at most $1 - \varepsilon$ fraction of inputs . Hence,

$$\Pr[\text{Algorithm 1 outputs YES}] \leq (1 - \varepsilon)^k.$$

So, for $k \geq 1/\varepsilon$, Pr[Algorithm 1 outputs YES] $\leq 1/e$. This probability can be made arbitrarily close to 0 by picking larger values of k.

4.1 Linearity Testing

A function $g : \mathbb{F}_2^n \to \mathbb{F}_2$ is said to be linear if for all $x, y \in \mathbb{F}_2^n$, $g(x+y) = g(x) + g(y)$.[1] Alternatively, g is said to be linear if for all $x \in \mathbb{F}_2^n$, $g(x) = \sum_{i\in[n]} a_i x_i$ for some $a_1, ..., a_n \in \mathbb{F}_2$. It is easy to show that these two definitions are equivalent. If $g(x) = \sum_{i\in[n]} a_i x_i$, then

$$g(x+y) = \sum_{i\in[n]} a_i(x_i + y_i) = \sum_{i\in[n]} a_i x_i + \sum_{i\in[n]} a_i y_i = g(x) + g(y).$$

Conversely, if $g(x+y) = g(x) + g(y)\ \forall x, y$, then

$$\begin{aligned} g(x) &= g(x_1 e_1 + \cdots + x_n e_n) \\ &= g(\underbrace{e_1 + \cdots + e_1}_{x_1 \text{ times}}) + \cdots + g(\underbrace{e_n + \cdots + e_n}_{x_n \text{ times}}) \\ &= x_1 g(e_1) + \cdots + x_n g(e_n) \\ &= a_1 x_1 + \cdots + a_n x_n, \end{aligned}$$

where e_i is the vector whose $i-$th coordinate is 1 and all other coordinates are 0 and $a_i = g(e_i)$. Having seen two equivalent definitions of a linear function, let us now see the Blum, Luby, Rubinfeld (BLR) linearity test.

Algorithm 2: BLR linearity test

Input: Blackbox access to g.

1 Sample $x, y \sim F_2^n$.
2 **if** $g(x+y) = g(x) + g(y)$ **then**
3 | return YES
4 **end**
5 **else**
6 | return NO
7 **end**

Analysis: It is clear that if g is linear, then Pr[Algorithm 2 outputs YES] = 1. To prove the converse we will first construct a function $f : \{-1,1\}^n \to \{-1,1\}$ such that f is a parity function if and only if g is a linear function and then show that the BLR test outputs YES with high probability only if f is a parity function.

For any $y \in \{-1,1\}^n$, let $x \in \mathbb{F}_2^n$ be a string such that for all $i \in [n], y_i = (-1)^{x_i}$ and define $f(y) = (-1)^{g(x)}$. If g is linear, then $g(x) = \sum_{i\in[n]} a_i x_i$ and

$$f(y) = (-1)^{\sum_{i\in[n]} a_i x_i} = \prod_{\substack{i\in[n]: \\ a_i=1}} (-1)^{x_i} = \prod_{\substack{i\in[n]: \\ a_i=1}} y_i.$$

Conversely, if $f(y) = \prod_{i\in S} y_i$, then

$$f(y) = (-1)^{\sum_{i\in S} x_i} = (-1)^{\sum_{i\in[n]} a_i x_i},$$

where $a_i = 0$ for $i \notin S$ and $a_i = 1$ for $i \in S$. Since, by definition, $f(y) = (-1)^{g(x)}$, we have that $g(x) = \sum_{i\in[n]} a_i x_i \mod 2 = \sum_{i\in[n]} a_i x_i$, because we are working over $\mathbb{F}_2$.

[1] Here the sum is over $\mathbb{F}_2$, i.e. modulo 2.

Observe that, when $f = \chi_S$, for some $S \subseteq [n]$, $f(x \cdot y) = f(x)f(y)$, where $(x \cdot y)_i := x_i y_i$. Hence, the following algorithm is equivalent to the BLR linearity test.

Algorithm 3: BLR linearity test - equivalent form

Input: Blackbox access to f.

Sample $x, y \sim \{-1, 1\}^n$.
if $f(x \cdot y) = f(x)f(y)$ **then**
  return YES
end
else
  return NO
end

Note that the test outputs 1 if and only if $f(x)f(y)f(x \cdot y) = 1$. To analyse the test, we will use the fact that

$$\frac{1}{2} + \frac{1}{2} f(x)f(y)f(x \cdot y)$$

is an indicator for whether or not the test outputs YES. Thus,

$$\begin{aligned}
\Pr[\text{Algorithm 3 outputs YES}] &= \mathop{\mathbb{E}}_{x,y}\left[\frac{1}{2} + \frac{1}{2} f(x)f(y)f(x \cdot y)\right] \\
&= \mathop{\mathbb{E}}_{x,y}\left[\frac{1}{2} + \frac{1}{2}\left(\sum_{S \subseteq [n]} \widehat{f}_S \prod_{i \in S} x_i\right)\left(\sum_{T \subseteq [n]} \widehat{f}_T \prod_{i \in T} y_i\right)\left(\sum_{U \subseteq [n]} \widehat{f}_U \prod_{i \in U} x_i y_i\right)\right] \\
&= \frac{1}{2} + \frac{1}{2} \sum_{S,T,U \subseteq [n]} \widehat{f}_S \widehat{f}_T \widehat{f}_U \mathop{\mathbb{E}}_{x,y}\left[\prod_{i \in S} x_i \prod_{i \in T} y_i \prod_{i \in U} x_i y_i\right] \\
&= \frac{1}{2} + \frac{1}{2} \sum_{S,T,U \subseteq [n]} \widehat{f}_S \widehat{f}_T \widehat{f}_U \mathop{\mathbb{E}}_{x,y}\left[\underbrace{\prod_{i \in S \cap U} x_i^2}_{=1} \prod_{i \in S \Delta U} x_i \underbrace{\prod_{i \in T \cap U} y_i^2}_{=1} \prod_{i \in T \Delta U} y_i\right] \\
&= \frac{1}{2} + \frac{1}{2} \sum_{S,T,U \subseteq [n]} \widehat{f}_S \widehat{f}_T \widehat{f}_U \mathop{\mathbb{E}}_{x}\left[\prod_{i \in S \Delta U} x_i\right] \mathop{\mathbb{E}}_{y}\left[\prod_{i \in T \Delta U} y_i\right] \\
&= \frac{1}{2} + \frac{1}{2} \sum_{S \subseteq [n]} \widehat{f}_S^3,
\end{aligned}$$

where the last equality follows $\mathbb{E}_x\left[\prod_{i \in S \Delta U} x_i\right] = 0$ when $S \neq U$ and $\mathbb{E}_y\left[\prod_{i \in T \Delta U} y_i\right] = 0$ when $T \neq U$. Thus,

$$\begin{aligned}
\Pr[\text{Algorithm 3 outputs YES}] &\leq \frac{1}{2} + \frac{1}{2}\left(\sum_{S \subseteq [n]} \widehat{f}_S^2\right) \max_{S \subseteq [n]} \widehat{f}_S \\
&= \frac{1}{2} + \frac{1}{2} \max_{S \subseteq [n]} \widehat{f}_S,
\end{aligned}$$

where the last equality follows from Parseval's theorem. Hence, if $\Pr[\text{Algorithm 3 outputs YES}] \geq 1 - \varepsilon$, then $\max_{S \subseteq [n]} \widehat{f}_S \geq 1 - 2\varepsilon$. Recall that, $\widehat{f}_S = \langle f, \chi_S \rangle = 1 - 2\,\mathrm{dist}(f, \chi_S)$. Thus, for $S^* = \mathrm{argmax}_{S \subseteq [n]} \widehat{f}_S$, we have $\mathrm{dist}(f, \chi_{S^*}) \leq \varepsilon$.

5 Local Correctability

As seen in the previous section, by making just 3 queries to a function we can say with high probability whether it is $\varepsilon-$close to a parity function. However, the BLR test does not tell us which parity function χ_{S^*} it is close to. Notice that we can indeed compute χ_{S^*} correctly on at least $1-\varepsilon$ fraction of inputs by simply outputting $f(x)$. We now prove the following theorem.

Theorem 7. *If f is $\varepsilon-$close to χ_{S^*}, then there exists an algorithm that makes only two queries to f and for all $x \in \{-1,1\}^n$, outputs $\chi_{S^*}(x)$ with probability at least $1-2\varepsilon$.*

Proof. Consider the following algorithm.

Algorithm 4: Local Correctability

Input: $x \in \{-1,1\}^n$ and blackbox access to f.

1 Sample $y \sim \{-1,1\}^n$.

2 Output $f(y)f(x \cdot y)$.

Analysis: While y and $x \cdot y$ are not independent, they are uniformly distributed over $\{-1,1\}^n$. Hence, $\Pr_y[f(y) = \chi_{S^*}(y)] \geq 1-\varepsilon$ and $\Pr_y[f(x \cdot y) = \chi_{S^*}(x \cdot y)] \geq 1-\varepsilon$. So, by union bound, with probability as least $1-2\varepsilon$ over the choice of y, $f(y) = \chi_{S^*}(y)$ and $f(x \cdot y) = \chi_{S^*}(x \cdot y)$. In this case,

$$f(y)f(x \cdot y) = \chi_{S^*}(y)\chi_{S^*}(x \cdot y) = \chi_{S^*}(x).$$

□

6 Other Applications of Analysis of Boolean Functions

The analysis of boolean functions has found applications in multiple areas of theoretical computer science, including but not limited to:

- Social choice theory
- Learning theory
- Hardness of approximation and PCPs.

More applications can be found in [O'D14].

References

[O'D14] Ryan O'Donnell, *Analysis of boolean functions*, Cambridge University Press, 2014.