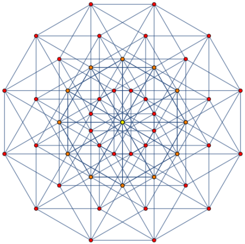
Learning low-degree functions on the discrete hypercube

Alexandros Eskenazis

Functional Analysis Seminar
Harbin Institute of Technology



The hypercube

The hypercube

Every function $f : \{-1,1\}^n \to \mathbb{R}$ admits a unique expansion

$$\forall\, x \in \{-1,1\}^n, \qquad f(x) = \sum_{S \subseteq \{1,\ldots,n\}} \hat{f}(S) w_S(x)$$

where the Walsh functions are given by $w_S(x) = \prod_{i \in S} x_i$.

The hypercube

Every function $f : \{-1,1\}^n \to \mathbb{R}$ admits a unique expansion

$$\forall\, x \in \{-1,1\}^n, \qquad f(x) = \sum_{S \subseteq \{1,\dots,n\}} \hat{f}(S) w_S(x)$$

where the Walsh functions are given by $w_S(x) = \prod_{i \in S} x_i$. The corresponding Fourier coefficients are then given by

$$\forall\, S \subseteq \{1,\dots,n\}, \qquad \hat{f}(S) = \mathbb{E}\big[f(x) w_S(x)\big],$$

where x is uniformly distributed on $\{-1,1\}^n$.

The hypercube

Every function $f : \{-1,1\}^n \to \mathbb{R}$ admits a unique expansion

$$\forall\, x \in \{-1,1\}^n, \qquad f(x) = \sum_{S \subseteq \{1,\dots,n\}} \hat{f}(S) w_S(x)$$

where the Walsh functions are given by $w_S(x) = \prod_{i \in S} x_i$. The corresponding Fourier coefficients are then given by

$$\forall\, S \subseteq \{1,\dots,n\}, \qquad \hat{f}(S) = \mathbb{E}\big[f(x) w_S(x)\big],$$

where x is uniformly distributed on $\{-1,1\}^n$. We say that f has *degree* at most d if $\hat{f}(S) = 0$ when $|S| > d$.

Learning

Learning

Let $\mathscr{F}$ be a class of functions on $\{-1,1\}^n$ and fix an unknown function $f \in \mathscr{F}$. Given access to data of the form

$$(X_1, f(X_1)), \ldots, (X_Q, f(X_Q))$$

where $X_1, \ldots, X_Q \in \{-1,1\}^n$, we want to algorithmically construct a hypothesis function $h : \{-1,1\}^n \to \mathbb{R}$ which well-approximates f.

Learning

Let $\mathscr{F}$ be a class of functions on $\{-1,1\}^n$ and fix an unknown function $f \in \mathscr{F}$. Given access to data of the form

$$(X_1, f(X_1)), \ldots, (X_Q, f(X_Q))$$

where $X_1, \ldots, X_Q \in \{-1,1\}^n$, we want to algorithmically construct a hypothesis function $h : \{-1,1\}^n \to \mathbb{R}$ which well-approximates f.

Query model. The algorithm can sequentially request any selection of samples $X_1, X_2, \ldots$.

Learning

Let $\mathscr{F}$ be a class of functions on $\{-1,1\}^n$ and fix an unknown function $f \in \mathscr{F}$. Given access to data of the form

$$(X_1, f(X_1)), \ldots, (X_Q, f(X_Q))$$

where $X_1, \ldots, X_Q \in \{-1,1\}^n$, we want to algorithmically construct a hypothesis function $h : \{-1,1\}^n \to \mathbb{R}$ which well-approximates f.

Query model. The algorithm can sequentially request any selection of samples $X_1, X_2, \ldots$.

Random example model. The samples $X_1, X_2, \ldots$ are i.i.d. random variables, uniformly distributed on the hypercube. In this model, the output function h is random and we want it to be a good approximation of f with high probability.

Learning

Question. How many samples do we need?

Learning

Question. How many samples do we need?

Query model. Denote by $Q(\mathscr{F}, \varepsilon)$ the least number of queries such that we can always output a function h with $\|h - f\|_2^2 \leq \varepsilon$.

Learning

Question. How many samples do we need?

Query model. Denote by $Q(\mathscr{F}, \varepsilon)$ the least number of queries such that we can always output a function h with $\|h - f\|_2^2 \leq \varepsilon$.

Random example model. Denote by $Q_r(\mathscr{F}, \varepsilon, \delta)$ the least number of queries such that we can always output a *random* function h satisfying $\|h - f\|_2^2 \leq \varepsilon$ with probability at least $1 - \delta$.

Learning

Question. How many samples do we need?

Query model. Denote by $Q(\mathscr{F}, \varepsilon)$ the least number of queries such that we can always output a function h with $\|h - f\|_2^2 \leq \varepsilon$.

Random example model. Denote by $Q_r(\mathscr{F}, \varepsilon, \delta)$ the least number of queries such that we can always output a *random* function h satisfying $\|h - f\|_2^2 \leq \varepsilon$ with probability at least $1 - \delta$.

Some structure is needed! If $\mathscr{F} = \{f : \{-1, 1\}^n \to \{0, 1\}\}$, one needs at least $(1 - \varepsilon)2^n$ values of an unknown $f \in \mathscr{F}$ in order to make an accurate hypothesis for f up to error ε.

Learning

Question. How many samples do we need?

Query model. Denote by $Q(\mathscr{F}, \varepsilon)$ the least number of queries such that we can always output a function h with $\|h - f\|_2^2 \leq \varepsilon$.

Random example model. Denote by $Q_r(\mathscr{F}, \varepsilon, \delta)$ the least number of queries such that we can always output a *random* function h satisfying $\|h - f\|_2^2 \leq \varepsilon$ with probability at least $1 - \delta$.

Some structure is needed! If $\mathscr{F} = \{f : \{-1, 1\}^n \to \{0, 1\}\}$, one needs at least $(1 - \varepsilon)2^n$ values of an unknown $f \in \mathscr{F}$ in order to make an accurate hypothesis for f up to error ε.

Structure = Low Complexity

Learning polynomials

Learning polynomials

One of the first concept classes $\mathscr{F}$ that was rigorously studied was

$$\mathscr{F}_{n,d} = \left\{ f : \{-1,1\}^n \to [-1,1] : \ \mathrm{deg}(f) \leq d \right\}.$$

Learning polynomials

One of the first concept classes $\mathscr{F}$ that was rigorously studied was

$$\mathscr{F}_{n,d} = \big\{ f : \{-1,1\}^n \to [-1,1] : \ \deg(f) \le d \big\}.$$

Why? Polynomials can be characterized by few values.

Learning polynomials

One of the first concept classes $\mathscr{F}$ that was rigorously studied was

$$\mathscr{F}_{n,d} = \big\{ f : \{-1,1\}^n \to [-1,1] : \ \deg(f) \le d \big\}.$$

Why? Polynomials can be characterized by few values.

Toy result. $Q(\mathscr{F}_{n,d}, 0) = \sum_{j=0}^{d} \binom{n}{j}$

Learning polynomials

One of the first concept classes $\mathscr{F}$ that was rigorously studied was

$$\mathscr{F}_{n,d} = \{f : \{-1,1\}^n \to [-1,1] : \deg(f) \le d\}.$$

Why? Polynomials can be characterized by few values.

Toy result. $Q(\mathscr{F}_{n,d}, 0) = \sum_{j=0}^{d} \binom{n}{j}$

Proof. It suffices to check that any degree-d polynomial is fully characterized by its values on a Hamming ball of radius d, e.g.

$$B_d(\mathbf{1}) = \{x \text{ with at most } d \text{ coordinates equal to } -1\}.$$

Learning polynomials

One of the first concept classes $\mathscr{F}$ that was rigorously studied was

$$\mathscr{F}_{n,d} = \big\{ f : \{-1,1\}^n \to [-1,1] : \ \deg(f) \leq d \big\}.$$

Why? Polynomials can be characterized by few values.

Toy result. $Q(\mathscr{F}_{n,d}, 0) = \sum_{j=0}^{d} \binom{n}{j}$

Proof. It suffices to check that any degree-d polynomial is fully characterized by its values on a Hamming ball of radius d, e.g.

$$B_d(\mathbf{1}) = \big\{ x \text{ with at most } d \text{ coordinates equal to } -1 \big\}.$$

To see that this many samples are also needed, observe that with fewer data points, the system would be undertermined. □

The Low-Degree Algorithm

Question. What about the random case?

The Low-Degree Algorithm

Question. What about the random case?

This question was first addressed in a fundamental result:

Low-Degree Algorithm (Linial, Mansour, Nisan, 1989) We have

$$Q_r(\mathscr{F}_{n,d}, \varepsilon, \delta) \leq \frac{2n^d}{\varepsilon} \log\left(\frac{2n^d}{\delta}\right).$$

The Low-Degree Algorithm

Question. What about the random case?

This question was first addressed in a fundamental result:

Low-Degree Algorithm (Linial, Mansour, Nisan, 1989) We have

$$Q_r(\mathscr{F}_{n,d}, \varepsilon, \delta) \leq \frac{2n^d}{\varepsilon} \log\left(\frac{2n^d}{\delta}\right).$$

Proof. Let $X_1, \ldots, X_Q$ i.i.d. random samples. For a subset S, let

$$\alpha_S = \frac{1}{Q} \sum_{j=1}^{Q} f(X_j) w_S(X_j),$$

which is a sum of bounded indep. variables with $\mathbb{E}[\alpha_S] = \hat{f}(S)$.

The Low-Degree Algorithm

Therefore, by the Chernoff bound, for $b > 0$ we have

$$\mathbb{P}\{|\alpha_S - \hat{f}(S)| \geq b\} \leq 2\exp(-Qb^2/2).$$

The Low-Degree Algorithm

Therefore, by the Chernoff bound, for $b > 0$ we have

$$\mathbb{P}\{|\alpha_S - \hat{f}(S)| \geq b\} \leq 2\exp(-Qb^2/2).$$

By the union bound,

$$\mathbb{P}\{|\alpha_S - \hat{f}(S)| \leq b,\ \forall\ S\} \geq 1 - 2\sum_{j=0}^{d}\binom{n}{j}\exp(-Qb^2/2) \geq 1 - \delta$$

for

$$Q = \left\lceil \frac{2}{b^2}\log\left(\frac{2}{\delta}\sum_{j=0}^{d}\binom{n}{j}\right)\right\rceil.$$

The Low-Degree Algorithm

Therefore, by the Chernoff bound, for $b > 0$ we have

$$\mathbb{P}\left\{|\alpha_S - \hat{f}(S)| \geq b\right\} \leq 2\exp(-Qb^2/2).$$

By the union bound,

$$\mathbb{P}\left\{|\alpha_S - \hat{f}(S)| \leq b,\ \forall\ S\right\} \geq 1 - 2\sum_{j=0}^{d}\binom{n}{j}\exp(-Qb^2/2) \geq 1-\delta$$

for

$$Q = \left\lceil \frac{2}{b^2}\log\left(\frac{2}{\delta}\sum_{j=0}^{d}\binom{n}{j}\right)\right\rceil.$$

How large can we take b?

The Low-Degree Algorithm

Consider the function

$$\forall\ x \in \{-1, 1\}^{,} \qquad h_b(x) = \sum_{|S| \leq d} \alpha_S w_S(x).$$

The Low-Degree Algorithm

Consider the function

$$\forall\, x \in \{-1,1\}^n \qquad h_b(x) = \sum_{|S|\le d} \alpha_S w_S(x).$$

Then, if the high probability event holds

$$\|f - h_b\|_2^2 = \sum_{|S|\le d} \left(\alpha_S - \hat{f}(S)\right)^2 \le \sum_{j=0}^{d} \binom{n}{j} b^2 \le \varepsilon$$

for $b^2 \le \varepsilon / \sum_{j=0}^{d} \binom{n}{j}$ which completes the proof. □

Learning polynomials

Question. Are $O(n^d \log n)$ samples too many?

Learning polynomials

Question. Are $O(n^d \log n)$ samples too many?

E.–Ivanisvili–Streck (2022). $Q_r(\mathscr{F}_{n,d}, 0, \delta) \leq 2^{O(d)} n^d \log\left(\frac{n}{\delta}\right)$.

Learning polynomials

Question. Are $O(n^d \log n)$ samples too many?

E.–Ivanisvili–Streck (2022). $Q_r(\mathscr{F}_{n,d}, 0, \delta) \leq 2^{O(d)} n^d \log\left(\frac{n}{\delta}\right)$.

The first advance for $\varepsilon > 0$ was a result of:

Iyer–Rao–Reis–Rothvoss–Yehudayoff (2021).

$$Q_r(\mathscr{F}_{n,d}, \varepsilon, \delta) = O_{d,\varepsilon,\delta}(n^{d-1} \log n).$$

Learning polynomials

Question. Are $O(n^d \log n)$ samples too many?

E.–Ivanisvili–Streck (2022). $Q_r(\mathscr{F}_{n,d}, 0, \delta) \leq 2^{O(d)} n^d \log\left(\frac{n}{\delta}\right)$.

The first advance for $\varepsilon > 0$ was a result of:

Iyer–Rao–Reis–Rothvoss–Yehudayoff (2021).

$$Q_r(\mathscr{F}_{n,d}, \varepsilon, \delta) = O_{d,\varepsilon,\delta}(n^{d-1} \log n).$$

The correct answer turns out to be much better.

E.–Ivanisvili (2021). $Q_r(\mathscr{F}_{n,d}, \varepsilon, \delta) = O_{d,\varepsilon,\delta}(\log n)$.

Learning polynomials

Question. Are $O(n^d \log n)$ samples too many?

E.–Ivanisvili–Streck (2022). $Q_r(\mathscr{F}_{n,d}, 0, \delta) \leq 2^{O(d)} n^d \log\left(\frac{n}{\delta}\right)$.

The first advance for $\varepsilon > 0$ was a result of:

Iyer–Rao–Reis–Rothvoss–Yehudayoff (2021).

$$Q_r(\mathscr{F}_{n,d}, \varepsilon, \delta) = O_{d,\varepsilon,\delta}(n^{d-1} \log n).$$

The correct answer turns out to be much better.

E.–Ivanisvili (2021). $Q_r(\mathscr{F}_{n,d}, \varepsilon, \delta) = O_{d,\varepsilon,\delta}(\log n)$.

E.–Ivanisvili–Streck (2022). $Q_r(\mathscr{F}_{n,d}, \varepsilon, \delta) = \Omega_{d,\varepsilon,\delta}(\log n)$.

Learning polynomials

Question. Are $O(n^d \log n)$ samples too many?

E.–Ivanisvili–Streck (2022). $Q_r(\mathscr{F}_{n,d}, 0, \delta) \le 2^{O(d)} n^d \log\left(\frac{n}{\delta}\right)$.

The first advance for $\varepsilon > 0$ was a result of:

Iyer–Rao–Reis–Rothvoss–Yehudayoff (2021).

$$Q_r(\mathscr{F}_{n,d}, \varepsilon, \delta) = O_{d,\varepsilon,\delta}(n^{d-1} \log n).$$

The correct answer turns out to be much better.

E.–Ivanisvili (2021). $Q_r(\mathscr{F}_{n,d}, \varepsilon, \delta) = O_{d,\varepsilon,\delta}(\log n)$.

E.–Ivanisvili–Streck (2022). $Q_r(\mathscr{F}_{n,d}, \varepsilon, \delta) = \Omega_{d,\varepsilon,\delta}(\log n)$.

Tweaking the Low-Degree Algorithm

Where did we lose in the proof?

Tweaking the Low-Degree Algorithm

Where did we lose in the proof? Since $\|f\|_2 \leq 1$, we have

$$\sum_{|S| \leq d} \hat{f}(S)^2 \leq 1$$

so unless $b^2 \lesssim n^{-d}$ there is not much to gain by incorporating *all* the empirical coefficients α_S in the hypothesis function h_b. We should just make sure to include the few influential ones, say those larger than a. By Markov's inequality there are

$$\#\{S : \ |\hat{f}(S)| > a\} \leq \frac{1}{a^2} \sum_{S\colon \ |\hat{f}(S)| > a} \hat{f}(S)^2 \leq \frac{1}{a^2}.$$

Tweaking the Low-Degree Algorithm

Where did we lose in the proof? Since $\|f\|_2 \leq 1$, we have

$$\sum_{|S| \leq d} \hat{f}(S)^2 \leq 1$$

so unless $b^2 \lesssim n^{-d}$ there is not much to gain by incorporating *all* the empirical coefficients α_S in the hypothesis function h_b. We should just make sure to include the few influential ones, say those larger than a. By Markov's inequality there are

$$\#\{S : \ |\hat{f}(S)| > a\} \leq \frac{1}{a^2} \sum_{S\colon \ |\hat{f}(S)| > a} \hat{f}(S)^2 \leq \frac{1}{a^2}.$$

Then, we are left to estimate a term of the form

$$\sum_{S\colon \ |\hat{f}(S)| \leq a} \hat{f}(S)^2 \overset{??}{\ll} \varepsilon(a).$$

Digression: Littlewood, BH,...

Digression: Littlewood, BH,...

Trivially, for $a_1, a_2, \ldots \in \mathbb{R}$,

$$\sum_{i \geq 1} |a_i| = \sup \left\{ \Big| \sum_{i \geq 1} a_i x_i \Big| : \; \|x\|_\infty \leq 1 \right\}.$$

Digression: Littlewood, BH,...

Trivially, for $a_1, a_2, \ldots \in \mathbb{R}$,

$$\sum_{i\geq 1} |a_i| = \sup \Big\{ \Big| \sum_{i\geq 1} a_i x_i \Big| : \ \|x\|_\infty \leq 1 \Big\}.$$

Littlewood's $\frac{4}{3}$-inequality. For $a_{ij} \in \mathbb{R}$, where $i, j \geq 1$

$$\Big(\sum_{i,j\geq 1} |a_{ij}|^{\frac{4}{3}} \Big)^{\frac{3}{4}} \leq \sqrt{2} \sup \Big\{ \Big| \sum_{i,j\geq 1} a_{ij} x_i y_j \Big| : \ \|x\|_\infty, \|y\|_\infty \leq 1 \Big\}.$$

Digression: Littlewood, BH,...

Bohnenblust–Hille inequality. For a degree-d polynomial $p(x) = \sum_{|\alpha|\leq d} c_\alpha x^\alpha$ on infinitely many variables,

$$\Big(\sum_{|\alpha|\leq d} |c_\alpha|^{\frac{2d}{d+1}}\Big)^{\frac{d+1}{2d}} \leq C_d \sup\big\{|p(x)| : \ \|x\|_\infty \leq 1\big\}.$$

Digression: Littlewood, BH,...

Bohnenblust–Hille inequality. For a degree-d polynomial $p(x) = \sum_{|\alpha| \leq d} c_\alpha x^\alpha$ on infinitely many variables,

$$\Big(\sum_{|\alpha| \leq d} |c_\alpha|^{\frac{2d}{d+1}} \Big)^{\frac{d+1}{2d}} \leq C_d \sup \big\{ |p(x)| : \ \|x\|_\infty \leq 1 \big\}.$$

If p is a multilinear polynomial representing $f : \{-1,1\}^n \to \mathbb{R}$, the maximum on the RHS is attained at a vertex of $\{-1,1\}^n$. Thus, we can get an estimate on the hypercube

$$\Big(\sum_{|S| \leq d} |\hat{f}(S)|^{\frac{2d}{d+1}} \Big)^{\frac{d+1}{2d}} \leq B_d \|f\|_\infty$$

for functions of degree at most d.

Proof of the logarithmic bound on the queries

Proof of the logarithmic bound on the queries

The idea of introducing a cutoff for the spectrum first appeared in an algorithm of Kushilevitz and Mansour (1993). Fix $b > 0$ and set

$$Q = \left\lceil \frac{2}{b^2} \log \left(\frac{2}{\delta} \sum_{j=0}^{d} \binom{n}{j} \right) \right\rceil$$

so that

$$\mathbb{P}\left\{ |\alpha_S - \hat{f}(S)| \leq b, \ \forall \ S \right\} \geq 1 - 2 \sum_{j=0}^{d} \binom{n}{j} \exp(-Qb^2/2) \geq 1 - \delta.$$

Proof of the logarithmic bound on the queries

The idea of introducing a cutoff for the spectrum first appeared in an algorithm of Kushilevitz and Mansour (1993). Fix $b > 0$ and set

$$Q = \left\lceil \frac{2}{b^2} \log \left(\frac{2}{\delta} \sum_{j=0}^{d} \binom{n}{j} \right) \right\rceil$$

so that

$$\mathbb{P}\left\{ |\alpha_S - \hat{f}(S)| \leq b, \ \forall \ S \right\} \geq 1 - 2 \sum_{j=0}^{d} \binom{n}{j} \exp(-Qb^2/2) \geq 1 - \delta.$$

Consider the random collection of sets

$$\Sigma_b = \left\{ S : \ |\alpha_S| > 2b \right\}.$$

Proof of the logarithmic bound on the queries

Then, on the high probability event, we have

$$\forall\ S \in \Sigma_b, \qquad |\hat{f}(S)| > b$$

and

$$\forall\ S \notin \Sigma_b, \qquad |\hat{f}(S)| \leq 3b.$$

Proof of the logarithmic bound on the queries

Then, on the high probability event, we have

$$\forall\ S \in \Sigma_b, \qquad |\hat{f}(S)| > b$$

and

$$\forall\ S \notin \Sigma_b, \qquad |\hat{f}(S)| \leq 3b.$$

If we define $h_b = \sum_{S \in \Sigma_b} \alpha_S w_S$, then

$$\|f - h_b\|_2^2 = \sum_{S \in \Sigma_b} \left(\alpha_S - \hat{f}(S)\right)^2 + \sum_{S \notin \Sigma_b} \hat{f}(S)^2 = (1) + (2).$$

Proof of the logarithmic bound on the queries

Then, on the high probability event, we have

$$\forall\ S \in \Sigma_b, \qquad |\hat{f}(S)| > b$$

and

$$\forall\ S \notin \Sigma_b, \qquad |\hat{f}(S)| \leq 3b.$$

If we define $h_b = \sum_{S \in \Sigma_b} \alpha_S w_S$, then

$$\|f - h_b\|_2^2 = \sum_{S \in \Sigma_b} \left(\alpha_S - \hat{f}(S)\right)^2 + \sum_{S \notin \Sigma_b} \hat{f}(S)^2 = (1) + (2).$$

To bound (1), observe that

$$|\Sigma_b| \leq b^{-\frac{2d}{d+1}} \sum_{S \in \Sigma_b} \hat{f}(S)^{\frac{2d}{d+1}} \leq B_d^{\frac{2d}{d+1}} b^{-\frac{2d}{d+1}}$$

so that $(1) \leq B_d^{\frac{2d}{d+1}} b^{\frac{2}{d+1}}$.

Proof of the logarithmic bound on the queries

To bound (2), write

$$(2) = \sum_{S \notin \Sigma_b} \hat{f}(S)^2 \leq (3b)^{\frac{2}{d+1}} \sum_{S \notin \Sigma_b} |\hat{f}(S)|^{\frac{2d}{d+1}} \leq 3B_d^{\frac{2d}{d+1}} b^{\frac{2}{d+1}}.$$

Proof of the logarithmic bound on the queries

To bound (2), write

$$(2) = \sum_{S \notin \Sigma_b} \hat{f}(S)^2 \leq (3b)^{\frac{2}{d+1}} \sum_{S \notin \Sigma_b} |\hat{f}(S)|^{\frac{2d}{d+1}} \leq 3B_d^{\frac{2d}{d+1}} b^{\frac{2}{d+1}}.$$

Putting everything together

$$\|f - h_b\|_2^2 \leq 4B_d^{\frac{2d}{d+1}} b^{\frac{2}{d+1}} \leq \varepsilon$$

for $b^2 \leq (\varepsilon/4)^{d+1} B_d^{-\frac{2d}{d+1}}$. □

Remarks

E.–Ivanisvili (2021). $Q_r(\mathscr{F}_{n,d}, \varepsilon, \delta) = O_{d,\varepsilon,\delta}(\log n)$.

E.–Ivanisvili–Streck (2022). $Q_r(\mathscr{F}_{n,d}, \varepsilon, \delta) = \Omega_{d,\varepsilon,\delta}(\log n)$.

Remarks

E.–Ivanisvili (2021). $Q_r(\mathscr{F}_{n,d}, \varepsilon, \delta) = O_{d,\varepsilon,\delta}(\log n)$.

E.–Ivanisvili–Streck (2022). $Q_r(\mathscr{F}_{n,d}, \varepsilon, \delta) = \Omega_{d,\varepsilon,\delta}(\log n)$.

In fact, for n large enough,

$$c(1-\sqrt{\varepsilon})2^d \log\left(\frac{n}{\delta}\right) \le Q_r(\mathscr{F}_{n,d}, \varepsilon, \delta) \le \frac{B_d^{2d}}{\varepsilon^{d+1}} \log\left(\frac{n}{\delta}\right).$$

Remarks

E.–Ivanisvili (2021). $Q_r(\mathscr{F}_{n,d}, \varepsilon, \delta) = O_{d,\varepsilon,\delta}(\log n)$.

E.–Ivanisvili–Streck (2022). $Q_r(\mathscr{F}_{n,d}, \varepsilon, \delta) = \Omega_{d,\varepsilon,\delta}(\log n)$.

In fact, for n large enough,

$$c(1-\sqrt{\varepsilon})2^d \log\left(\frac{n}{\delta}\right) \le Q_r(\mathscr{F}_{n,d}, \varepsilon, \delta) \le \frac{B_d^{2d}}{\varepsilon^{d+1}} \log\left(\frac{n}{\delta}\right).$$

• The best known bound for B_d is $B_d \le \exp(C\sqrt{d \log d})$. A (conjectured) polynomial bound on B_d would give almost optimal dependence on d also.

• The dependence on ε can be improved to ε^{-1} if the unknown function is a priori known to be Boolean.

Beyond polynomials?

Beyond polynomials?

Pro. Correct query complexity of polynomials.

Beyond polynomials?

Pro. Correct query complexity of polynomials.

Con. Too rigid: hard to imagine other concept classes for which BH-type arguments would be applicable.

Beyond polynomials?

Pro. Correct query complexity of polynomials.

Con. Too rigid: hard to imagine other concept classes for which BH-type arguments would be applicable.

What about the class of bounded *approximate* polynomials,

$$\mathscr{F}_{n,d}(t) = \left\{ f : \{-1,1\}^n \to [-1,1] : \sum_{|S|>d} \hat{f}(S)^2 \leq t \right\} ?$$

Beyond polynomials?

Pro. Correct query complexity of polynomials.

Con. Too rigid: hard to imagine other concept classes for which BH-type arguments would be applicable.

What about the class of bounded *approximate* polynomials,

$$\mathscr{F}_{n,d}(t) = \left\{ f : \{-1,1\}^n \to [-1,1] : \sum_{|S|>d} \hat{f}(S)^2 \leq t \right\} ?$$

E.–Ivanisvili–Streck (2022). There exists $\eta = \eta(t,d) > 0$ s.t.

$$Q_r(\mathscr{F}_{n,d}(t), \eta + \varepsilon, \delta) \lesssim_{t,d,\varepsilon} \log\left(\frac{n}{\delta}\right).$$

Beyond polynomials?

Pro. Correct query complexity of polynomials.

Con. Too rigid: hard to imagine other concept classes for which BH-type arguments would be applicable.

What about the class of bounded *approximate* polynomials,

$$\mathscr{F}_{n,d}(t) = \Big\{ f : \{-1,1\}^n \to [-1,1] : \sum_{|S|>d} \hat{f}(S)^2 \le t \Big\} \ ?$$

E.–Ivanisvili–Streck (2022). There exists $\eta = \eta(t,d) > 0$ s.t.

$$Q_r(\mathscr{F}_{n,d}(t), \eta + \varepsilon, \delta) \lesssim_{t,d,\varepsilon} \log\left(\frac{n}{\delta}\right).$$

Warning! This is useful only when $\eta(t,d)$ is small.

Beyond polynomials?

More concretely, consider $\mathscr{B}_{n,d}(t)$ the subclass of $\mathscr{F}_{n,d}(t)$ consisting of Boolean functions.

Beyond polynomials?

More concretely, consider $\mathscr{B}_{n,d}(t)$ the subclass of $\mathscr{F}_{n,d}(t)$ consisting of Boolean functions.

E.–Ivanisvili–Streck (2022). We have

$$t = o\Big(\frac{1}{\sqrt{d}}\Big) \quad \Longrightarrow \quad Q_r(\mathscr{B}_{n,d}(t), \varepsilon, \delta) \lesssim_{t,d,\varepsilon} \log\Big(\frac{n}{\delta}\Big)$$

for $\varepsilon > 0$ arbitrarily small constant.

Beyond polynomials?

More concretely, consider $\mathscr{B}_{n,d}(t)$ the subclass of $\mathscr{F}_{n,d}(t)$ consisting of Boolean functions.

E.–Ivanisvili–Streck (2022). We have

$$t = o\Big(\frac{1}{\sqrt{d}}\Big) \quad \Longrightarrow \quad Q_r(\mathscr{B}_{n,d}(t), \varepsilon, \delta) \lesssim_{t,d,\varepsilon} \log\Big(\frac{n}{\delta}\Big)$$

for $\varepsilon > 0$ arbitrarily small constant.

Conversely, we can also prove that

$$t = \Omega\Big(\frac{1}{\sqrt{d}}\Big) \quad \Longrightarrow \quad Q_r\big(\mathscr{B}_{n,d}(t), \tfrac{1}{3}, \tfrac{1}{3}\big) \gtrsim_{t,d} n.$$

Linear threshold functions

Linear threshold functions

A Boolean function of the form $f(x) = \text{sign}(\langle x, \theta \rangle)$ for a fixed vector $\theta \in \mathbb{R}^n$ is called a linear threshold function.

Linear threshold functions

A Boolean function of the form $f(x) = \mathrm{sign}(\langle x, \theta \rangle)$ for a fixed vector $\theta \in \mathbb{R}^n$ is called a linear threshold function. Peres' noise sensitivity theorem (2004) asserts that any LTF satisfies

$$\forall\ t > 0, \qquad \sum_{|S| > \Omega(1/t^2)} \hat{f}(S)^2 \leq t.$$

Linear threshold functions

A Boolean function of the form $f(x) = \mathrm{sign}(\langle x, \theta \rangle)$ for a fixed vector $\theta \in \mathbb{R}^n$ is called a linear threshold function. Peres' noise sensitivity theorem (2004) asserts that any LTF satisfies

$$\forall\ t > 0, \qquad \sum_{|S| > \Omega(1/t^2)} \hat{f}(S)^2 \leq t.$$

As this estimate is in general optimal, the existing algorithm does not allow us to efficiently learn LTFs.

DNF formulas

DNF formulas

A disjunctive normal form (DNF) is a logical $\vee$ of terms, each of which is a logical $\wedge$ of Boolean variables x_i or their negations $\neg x_i$,

$$(x_1 \wedge x_2) \vee (\neg x_2 \wedge \neg x_3) \vee (\neg x_1 \wedge x_3).$$

DNF formulas

A disjunctive normal form (DNF) is a logical $\vee$ of terms, each of which is a logical $\wedge$ of Boolean variables x_i or their negations $\neg x_i$,

$$(x_1 \wedge x_2) \vee (\neg x_2 \wedge \neg x_3) \vee (\neg x_1 \wedge x_3).$$

The number of terms is the size of the DNF (=3 in the example).

DNF formulas

A disjunctive normal form (DNF) is a logical $\vee$ of terms, each of which is a logical $\wedge$ of Boolean variables x_i or their negations $\neg x_i$,

$$(x_1 \wedge x_2) \vee (\neg x_2 \wedge \neg x_3) \vee (\neg x_1 \wedge x_3).$$

The number of terms is the size of the DNF (=3 in the example).

It is known that any DNF form of size s satisfies

$$\forall\ t > 0, \qquad \sum_{|S| > \Omega(\log(s/t)^2)} \hat{f}(S)^2 \leq t$$

DNF formulas

A disjunctive normal form (DNF) is a logical $\vee$ of terms, each of which is a logical $\wedge$ of Boolean variables x_i or their negations $\neg x_i$,

$$(x_1 \wedge x_2) \vee (\neg x_2 \wedge \neg x_3) \vee (\neg x_1 \wedge x_3).$$

The number of terms is the size of the DNF (=3 in the example).

It is known that any DNF form of size s satisfies

$$\forall\ t > 0, \qquad \sum_{|S| > \Omega(\log(s/t)^2)} \hat{f}(S)^2 \leq t$$

and plugging this choice of d, one obtains new learning results for the class of DNF formulas.

Thank you!