TTIC 31250
An Introduction to the Theory of Machine Learning

Characterizing SQ-learnability

Avrim Blum
05/02/22
**Statistical Query Recap**

- **Target function** $c(x)$. No noise.

- **Algorithm asks**: “what is the probability a labeled example will have property $\chi$? Please tell me up to additive error $\tau$."

  - Formally, $\chi: X \times \{0,1\} \rightarrow \{0,1\}$. Must be poly-time computable. $\tau \geq 1/poly(\ldots)$.
  - Let $P_\chi = \Pr_{x \sim D}[\chi(x, c(x)) = 1]$.
  - World responds with $P'_\chi \in [P_\chi - \tau, P_\chi + \tau]$.
    [can extend to $E[\chi]$ for $[0,1]$-valued or vector-valued $\chi$]

- May repeat poly(...) times. Can also ask for unlabeled data. Must output $h$ of error $\leq \epsilon$. No $\delta$ in this model.
Statistical Query Recap

- Examples of query:
  - What is the error rate of my current hypothesis h?
    \[ \chi(x, y) = 1 \iff h(x) \neq y \]

- Get back answer to \( \pm \tau \). Can simulate from \( \approx 1/\tau^2 \) examples. [That's why need \( \tau \geq 1/\text{poly}(\ldots) \).]
Characterizing what’s learnable using SQ algorithms

• Say that $f, g$ uncorrelated if $\Pr_{x \sim D} [f(x) = g(x)] = \frac{1}{2}$.

Def: the SQ-dimension of a class $C$ wrt $D$ is the size of the largest set $C' \subseteq C$ s.t. for all $f, g \in C'$,

$$\left| \Pr_D [f(x) = g(x)] - \frac{1}{2} \right| < \frac{1}{|C'|}.$$ 

(size of largest set of nearly uncorrelated functions in $C$)

• Theorem 1: if $\text{SQDIM}_D(C) \leq \text{poly}(n)$ then you can weak-learn $C$ over $D$ by SQ algs. [error rate $\leq \frac{1}{2} - \frac{1}{\text{poly}(n)}$]

• Theorem 2: if $\text{SQDIM}_D(C) > \text{poly}(n)$ then you can’t weak-learn $C$ over $D$ by SQ algs.
Characterizing what's learnable using SQ algorithms

Example: Parity functions \( c(x) = c \cdot x \mod 2 \)

- Let \( D \) be uniform on \( \{0,1\}^n \).
- Any two parity functions are uncorrelated.
- So, \( \text{SQ-dim}_D(\{\text{Parity functions}\}) = 2^n \).
- Any parity function of size \( \lg(n) \) can be described as a size-\( n \) decision tree. So, \( \text{SQ-dim}_D(\{\text{size-}n \ \text{DTs}\}) \geq \binom{n}{\lg n} \).

So, poly-sized decision trees are not SQ-learnable either.

- Theorem 1: if \( \text{SQDIM}_D(C) \leq \text{poly}(n) \) then you can weak-learn \( C \) over \( D \) by SQ algs. [error rate \( \leq \frac{1}{2} - \frac{1}{\text{poly}(n)} \)]
- Theorem 2: if \( \text{SQDIM}_D(C) > \text{poly}(n) \) then you can't weak-learn \( C \) over \( D \) by SQ algs.
Characterizing what’s learnable using SQ algorithms

Can anyone think of a non-SQ algorithm to learn parity functions?

- Theorem 1: if $\text{SQDIM}_D(C) \leq \text{poly}(n)$ then you can weak-learn $C$ over $D$ by SQ algs. [error rate $\leq \frac{1}{2} - \frac{1}{\text{poly}(n)}$]
- Theorem 2: if $\text{SQDIM}_D(C) > \text{poly}(n)$ then you can’t weak-learn $C$ over $D$ by SQ algs.
Characterizing what’s learnable using SQ algorithms

Theorem 1 is easier - let’s prove it first.

- Let $d = \text{SQDIM}_D(C)$.
- Let $H \subseteq C$ be a maximal subset s.t. for all $h_i, h_j \in H$, we have $|\Pr_D[h_i(x) = h_j(x)] - \frac{1}{2}| < \frac{1}{d+1}$. So, $|H| \leq d$.
- To learn, just try each $h_i \in H$ and use an SQ to estimate its error. At least one $h_i$ (or $\neg h_i$) must be a weak predictor.

- Theorem 1: if $\text{SQDIM}_D(C) \leq \text{poly}(n)$ then you can weak-learn $C$ over $D$ by SQ algs. [error rate $\leq \frac{1}{2} - \frac{1}{\text{poly}(n)}$]
- Theorem 2: if $\text{SQDIM}_D(C) > \text{poly}(n)$ then you can’t weak-learn $C$ over $D$ by SQ algs.
Characterizing what’s learnable using SQ algorithms

Now, onto Theorem 2.

To keep things simpler, will change “nearly uncorrelated” to “uncorrelated”. I.e., we will assume there are more than \(\text{poly}(n)\) uncorrelated functions in \(C\).

- Theorem 1: if \(\text{SQDIM}_D(C) \leq \text{poly}(n)\) then you can weak-learn \(C\) over \(D\) by SQ algs. [error rate \(\leq \frac{1}{2} - \frac{1}{\text{poly}(n)}\)]
- Theorem 2: if \(\text{SQDIM}_D(C) > \text{poly}(n)\) then you can’t weak-learn \(C\) over \(D\) by SQ algs.
Characterizing what’s learnable using SQ algorithms

- **Key tool:** Fourier analysis of boolean functions.
- Sounds scary but it’s a cool idea!
- Let’s think of functions from \( \{0,1\}^n \rightarrow \{-1, +1\} \).
- View function \( f \) as a vector of \( 2^n \) entries:
  \[
  \left( \sqrt{D[000]}f(000), \sqrt{D[001]}f(001), ..., \sqrt{D[x]}f(x), ... \right)
  \]
  - In other words, the truth-table of \( f \), where entry \( x \) is weighted by the square-root of the probability of \( x \).
- What is \( \langle f, f \rangle \)? What is \( \langle f, g \rangle \)?
Characterizing what’s learnable using SQ algorithms

- **Key tool:** Fourier analysis of boolean functions.
- Sounds scary but it’s a cool idea!
- Let’s think of functions from \( \{0,1\}^n \rightarrow \{-1, +1\} \).
- View function \( f \) as a vector of \( 2^n \) entries:
  \[
  \left( \sqrt{D[000]}f(000), \sqrt{D[001]}f(001), \ldots, \sqrt{D[x]}f(x), \ldots \right)
  \]
  - In other words, the truth-table of \( f \), where entry \( x \) is weighted by the square-root of the probability of \( x \).
- **What is \( \langle f, f \rangle \)? What is \( \langle f, g \rangle \)?**
  - \( \langle f, f \rangle = 1 \).
  - \( \langle f, g \rangle = \sum_x \Pr(x) f(x)g(x) = E_D[f(x)g(x)] = \Pr(\text{agree}) - \Pr(\text{disagree}) \). Call this the correlation of \( f \) and \( g \).
Characterizing what’s learnable using SQ algorithms

• Key tool: Fourier analysis of boolean functions.
• Sounds scary but it’s a cool idea!
• Let’s think of functions from \( \{0,1\}^n \rightarrow \{-1, +1\} \).
• View function \( f \) as a vector of \( 2^n \) entries:

\[
(\sqrt{D[000]}f(000), \sqrt{D[001]}f(001), \ldots, \sqrt{D[x]}f(x), \ldots)
\]

- In other words, the truth-table of \( f \), where entry \( x \) is weighted by the square-root of the probability of \( x \).

• So, functions are unit-length vectors, and uncorrelated functions are orthogonal. Dot-product equals amount of correlation.
Characterizing what’s learnable using SQ algorithms

- **Fourier analysis** is just a way of saying we want to talk about what happens when we change basis.

- An **orthonormal basis** is a set of orthogonal unit vectors that span the space.

- E.g., in 2-d, let $x', y'$ be unit vectors in $x,y$ directions. $v = (2,3) = 2x' + 3y'$.

- If have two other orthogonal unit vectors $a, b$, then could write $v = \langle v, a \rangle a + \langle v, b \rangle b$. 
Characterizing what’s learnable using SQ algorithms

- We are in a $2^n$-dimensional space, so an orthonormal basis is a set of $2^n$ orthogonal unit vectors.

- Let’s fix one. $\varphi_1, \ldots, \varphi_{2^n}$.

- Given a vector $f$, let $f_i$ be the $i$th entry in the standard basis: $f_i = f(i)\sqrt{\Pr(i)}$.

- Then $\hat{f}_i = \langle f, \varphi_i \rangle$ is the $i$th entry in the $\varphi$ basis.

- For instance, can write vector $f$ as $f = \Sigma_i \hat{f}_i \varphi_i$

- The $\hat{f}_i$ are called the “Fourier coeffs of $f$” in the $\varphi$ basis.

- Since $f = \Sigma_i \hat{f}_i \varphi_i$, this means $f(x) = \Sigma_i \hat{f}_i \varphi_i (x)$. This is just saying the $x$th coordinates match.
Characterizing what's learnable using SQ algorithms

• Consider any Boolean function \( f \). Since it’s a unit-length vector, this means \( \sum_i f_i^2 = 1 \). Called “Parseval’s identity”

• At most \( t^2 \) of the \( \varphi_i \) can have \( |\langle f, \varphi_i \rangle| = |\hat{f}_i| \geq \frac{1}{t} \).

• I.e., any given Boolean function can have correlation \( \geq \frac{1}{t} \) with at most \( t^2 \) Boolean functions in an orthogonal set.

• In particular, any given \( f \) can be weakly correlated with at most a polynomial number of them.

• Since \( f = \sum_i \hat{f}_i \varphi_i \), this means \( f(x) = \sum_i \hat{f}_i \varphi_i (x) \). This is just saying the \( x \)th coordinates match.
Characterizing what’s learnable using SQ algorithms

- Consider any Boolean function $f$. Since it’s a unit-length vector, this means $\sum_i \hat{f}_i^2 = 1$. Called “Parseval’s identity”

- At most $t^2$ of the $\varphi_i$ can have $|\langle f, \varphi_i \rangle| = |\hat{f}_i| \geq \frac{1}{t}$.

- I.e., any given Boolean function can have correlation $\geq \frac{1}{t}$ with at most $t^2$ Boolean functions in an orthogonal set.

- In particular, any given $f$ can be weakly correlated with at most a polynomial number of them.

If $C$ has $n^{\omega(1)}$ uncorrelated functions, target is a random one of them, SQs all of form “what is correlation of target with my $h$ up to $\pm \frac{1}{\text{poly}(n)}$” then whp oracle can always answer 0.
Characterizing what’s learnable using SQ algorithms

• It turns out that any SQ can be converted into a portion that looks like this, and a portion that doesn’t depend on the target function at all.

If $C$ has $n^{\omega(1)}$ uncorrelated functions, target is a random one of them, SQs all of form “what is correlation of target with my $h$ up to $\pm \frac{1}{\text{poly}(n)}$” then whp oracle can always answer 0.
Proof of Theorem 2'

Theorem 2': If \( C \) has \( n^{\omega(1)} \) uncorrelated functions, and target is random one of them, then whp any SQ algo that makes \( poly(n) \) queries of tolerance \( \frac{1}{poly(n)} \) will fail to weak learn.

Proof:

• Let \( \varphi_1, ..., \varphi_m \) be orthogonal functions in \( C \). Extend arbitrarily to a basis \( \varphi_1, ..., \varphi_{2^n} \). (excess vectors may not be Boolean functions and may not be in \( C \))

• Now, consider a SQ \( \chi: \{0,1\}^n \times \{-1,1\} \rightarrow [-1,1] \). Can view this as a vector in \( 2^{n+1} \) dimensions.

• To apply Fourier analysis to this, need to extend our basis to this higher-dimensional space.
Proof of Theorem 2'

• Define distribution $D' = D \times uniform$ on $\{-1, +1\}$

• Define $\varphi_i(x, y) = \varphi_i(x)$ [ignore label]

Still orthogonal:

$$\Pr_D[\varphi_i(x, y) = \varphi_j(x, y)] = \Pr_D[\varphi_i(x) = \varphi_j(x)] = \frac{1}{2}$$

• Need $2^n$ more basis functions.

• Define $h_i(x, y) = y\varphi_i(x)$. Need to verify these work:
  - Check that $h_i$ and $h_j$ are orthogonal for $i \neq j$.
  - Check that $h_i$ and $\varphi_j$ are orthogonal even if $i = j$.

• Now do Fourier decomposition on $\chi(x, y)$. 
Proof of Theorem 2'

• \( \chi = \sum_i \alpha_i \varphi_i + \sum_i \beta_i h_i \) where \( \sum_i \alpha_i^2 + \sum_i \beta_i^2 = 1 \).

• So we can write the quantity we care about as:

\[
E_D[\chi(x, c(x))] = E_D \left[ \sum_i \alpha_i \varphi_i(x) + \sum_i \beta_i h_i(x, c(x)) \right]
\]

\[
= \sum_i \alpha_i E_D[\varphi_i(x)] + \sum_i \beta_i E_D[c(x)\varphi_i(x)]
\]

• First term doesn’t depend on target at all. Call it \( g(\chi, D) \).

• Recall that \( c \) is random from \( \{\varphi_1, \ldots, \varphi_m\} \). Say \( c = \varphi_{i*} \).

• What is the 2\textsuperscript{nd} term?

• Ans: 2\textsuperscript{nd} term = \( \beta_{i*} \). So whp, world can just return \( g(\chi, D) \).

• That’s it.
Stepping back

• If $C$ contains more than $\text{poly}(n)$ many uncorrelated functions, then can’t learn in SQ model. [holds also for “nearly uncorrelated” as in SQ-dim definition]

• Very last step of proof had adversary convert $g(\chi, D) + \text{tiny value}$ into $g(\chi, D)$. Can also make this work in “honest SQ” model, where it’s estimated from a random sample.

• Can also use SQ-dim to prove that certain $(C,D)$ pairs have no large-margin kernels (kernels where every $c$ in $C$ looks like a large-margin separator in the implicit space).