CSCI699: Machine Learning Theory Fall 2023

Lecture 4: Rademacher Complexity, Stability

Instructor: Vatsal Sharan

These lecture notes are based on an initial version scribed by Ali Omrani and Sai Anuroop Kesanapalli.

1 Rademacher Complexity

We begin by recalling the definition of Rademacher complexity.

Definition 1 (Rademacher Complexity)**.** *Let $\mathcal{F}$ be a family of real-valued functions $f : \mathcal{Z} \to \mathbb{R}$ where $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$. Then the Rademacher Complexity $RC(\mathcal{F})$ is defined as:*

$$RC(\mathcal{F}) = \frac{1}{n}\mathbb{E}_{\sigma\sim\{\pm1\}^n}\left[\sup_{f\in\mathcal{F}}\sum_{i=1}^{n}\sigma_i f(z_i)\right].$$

More generally, given a (possibly infinite) set of vectors $A \subseteq \mathbb{R}^n$, the Rademacher Complexity $RC(A)$ is defined as:

$$RC(A) = \frac{1}{n}\mathbb{E}_{\sigma\sim\{\pm1\}^n}\left[\sup_{a\in A}\sum_{i=1}^{n}\sigma_i a_i\right].$$

We stated last time how we can use Rademacher complexity to get upper bound the expected difference between the test and train error using a symmetrization argument.

Lemma 2 (Symmetrization with Rademacher)**.**

$$\mathop{\mathbb{E}}_{S\sim\mathcal{D}^n}\sup_{h\in\mathcal{H}}(R(h) - \hat{R}_S(h)) \le 2\mathop{\mathbb{E}}_{S\sim\mathcal{D}^n}\left[RC(\ell\circ\mathcal{H}\circ S)\right]$$

Proof. The proof follows from the same argument that we used to motivate the definition of Rademacher complexity.

$$\begin{aligned}
\mathop{\mathbb{E}}_{S\sim\mathcal{D}^n}\sup_{h\in\mathcal{H}}(R(h) - \hat{R}_S(h)) &\le \mathop{\mathbb{E}}_{S,S'}\sup_{h\in\mathcal{H}}\frac{1}{n}\left(\sum_{i=1}^{n}(\ell(h,z_i) - \ell(h,z_i'))\right)\\
&= \mathop{\mathbb{E}}_{S,S',\sigma_{1:n}}\sup_{h\in\mathcal{H}}\frac{1}{n}\left(\sum_{i=1}^{n}\sigma_i(\ell(h,z_i) - \ell(h,z_i'))\right)\\
&\le \mathop{\mathbb{E}}_{S}\mathop{\mathbb{E}}_{\sigma_{1:n}}\sup_{h\in\mathcal{H}}\frac{1}{n}\sum_{i=1}^{n}\sigma_i\ell(h,z_i)\\
&+ \mathop{\mathbb{E}}_{S'}\mathop{\mathbb{E}}_{\sigma_{1:n}}\sup_{h\in\mathcal{H}}\frac{1}{n}\sum_{i=1}^{n}(-\sigma_i)\ell(h,z_i').
\end{aligned}$$

Therefore we get that,

$$\mathop{\mathbb{E}}_{S} \sup_{h\in\mathcal{H}}(R(h)-\hat{R}_S(h)) \le 2 \mathop{\mathbb{E}}_{S,\sigma_{1:n}} \sup_{h\in\mathcal{H}} \frac{1}{n}\sum_{i=1}^{n}\sigma_i \ell(h,z_i) = 2 \mathop{\mathbb{E}}_{S\sim\mathcal{D}^n} RC(\ell\circ\mathcal{H}\circ S).$$

■

1.1 Generalization bounds using Rademacher complexity

We now show how Rademacher complexity can be used to upper bound the generalization gap with high probability.

Theorem 3 (Excess risk bounds using Rademacher Complexity)**.** *Assume that for all $z \in \mathcal{X}\times\mathcal{Y}$ and $h\in\mathcal{H}$ we have that $|\ell(h,z)| \le C$. Then with probability at least $(1-\delta)$ over $S\sim\mathcal{D}^n$,*

(1)

$$\sup_{h\in\mathcal{H}}(R(h)-\hat{R}_S(h)) \le 2 \mathop{\mathbb{E}}_{S'}\left[RC(\ell\circ\mathcal{H}\circ S')\right] + c\sqrt{\frac{2\log(1/\delta)}{n}}$$

(2)

$$\sup_{h\in\mathcal{H}}(R(h)-\hat{R}_S(h)) \le 2RC(\ell\circ\mathcal{H}\circ S) + 3c\sqrt{\frac{2\log(2/\delta)}{n}}$$

(3) For any $h\in\mathcal{H}$,

$$R(h_{ERM}) - R(h) \le 2RC(\ell\circ\mathcal{H}\circ S) + 4c\sqrt{\frac{2\log(4/\delta)}{n}}.$$

(in particular, this holds for $h=\tilde{h}=\arg\min_{h\in\mathcal{H}} R(h)$)

Proof. We will keep using McDiarmid's inequality throughout the proof.

(1) Note that $\sup_{h\in\mathcal{H}}(R(h)-\hat{R}_S(h))$ satisfies the bounded differences property with constant $2c/n$. (changing any (x_i,y_i) changes the loss by at most $2c/n$). Therefore, using McDiarmid's inequality

$$\sup_{h\in\mathcal{H}}(R(h)-\hat{R}_S(h)) \le \mathbb{E}\left[\sup_{h\in\mathcal{H}}(R(h)-\hat{R}_S(h))\right] + \epsilon$$

with probability at least

$$1-\exp\left(\frac{-2\epsilon^2}{n(2c/n)^2}\right) = 1-\underbrace{\exp\left(-\frac{n\epsilon^2}{2c^2}\right)}_{\delta}.$$

We choose $\epsilon = c\sqrt{\frac{2\log(1/\delta)}{n}}$ to set the error probability to be δ. Therefore we get that with probability $1-\delta$,

$$\sup_{h\in\mathcal{H}}(R(h)-\hat{R}_S(h)) \le \mathbb{E}\left[\sup_{h\in\mathcal{H}}(R(h)-\hat{R}_S(h))\right] + c\sqrt{\frac{2\log(1/\delta)}{n}}.$$

We now use Lemma 2 (Symmetrization with Rademacher Complexity), and the result follows.

(2) Note that

$$RC(\ell \circ \mathcal{H} \circ S) = \mathop{\mathbb{E}}_{\sigma_{1:n}} \left(\sup_{h \in \mathcal{H}} \frac{1}{n} \sum_{i=1}^{n} \sigma_i \ell(h, z_i) \right)$$

also satisfies bounded differences with constant $2c/n$ (swapping σ_i by σ_i' changes the value by $\leq 2c/n$). Therefore with probability $1 - \delta$,

$$RC(\ell \circ \mathcal{H} \circ S) \geq \mathop{\mathbb{E}}_{S'} \left[RC(\ell \circ \mathcal{H} \circ S') \right] - c\sqrt{\frac{2\log(1/\delta)}{n}}.$$

So

$$\mathop{\mathbb{E}}_{S'} \left[RC(\ell \circ \mathcal{H} \circ S') \right] \leq RC(\ell \circ \mathcal{H} \circ S) + c\sqrt{\frac{2\log(1/\delta)}{n}}.$$

Now set $\delta = \delta'/2$, with probability $1 - \frac{\delta'}{2}$,

$$\mathop{\mathbb{E}}_{S'} \left[RC(\ell \circ \mathcal{H} \circ S') \right] \leq RC(\ell \circ \mathcal{H} \circ S) + c\sqrt{\frac{2\log(2/\delta')}{n}},$$

$$\sup_{h \in \mathcal{H}} (R(h) - \hat{R}_S(h)) \leq \mathop{\mathbb{E}}_{S'} \left[RC(\ell \circ \mathcal{H} \circ S') \right] + c\sqrt{\frac{2\log(2/\delta)}{n}} \text{ (from part (1))}.$$

The result now follows by doing a union bound and combining the above results. We get that with probability $1 - \delta$,

$$\sup_{h \in \mathcal{H}} (R(h) - \hat{R}_S(h)) \leq 2RC(\ell \circ \mathcal{H} \circ S) + 3c\sqrt{\frac{2\log(2/\delta)}{n}}$$

proving the result we wanted.

(3) By doing a familiar decomposition,

$$R(h_{ERM}) - R(h^*) = \underbrace{R(h_{ERM}) - \hat{R}_S(h_{ERM})}_{\text{bounded by part (2)}} + \underbrace{\hat{R}_S(h_{ERM}) - \hat{R}_S(h^*)}_{\leq 0} + \underbrace{\hat{R}_S(h^*) - R(h^*)}_{\text{Hoeffding's}}.$$

With probability $1 - \delta/2$,

$$\hat{R}_S(h^*) - R(h^*) \leq c\sqrt{\frac{2\log(2/\delta)}{n}}.$$

$$\implies R(h_{ERM}) - R(h^*) \leq 2RC(\ell \circ \mathcal{H} \circ S) + 4c\sqrt{\frac{2\log(4/\delta)}{n}}.$$

■

Some takeways from this result:

- **Rademacher complexity bound could be much better than the VC bound**: Rademacher complexity takes the data distribution into account, whereas the VC dimension is only a property of the hypothesis class and does not depend on the data distribution. Therefore, on natural or nice distributions, Rademacher complexity could give tighter bounds than the VC dimesion bounds.

- **Data-dependent bound**. The Rademacher complexity can also be measured rather easily empirically. In particular, (3) in Theorem 3 uses the same training set S both for learning a hypothesis from $\mathcal{H}$, and for estimating its generalization error.

Remark 4. *A now famous experiment by [1] evaluated the efficacy of a data-dependent Rademacher complexity bound for modern neural networks on datasets that they succeed at (such as image classification datasets such as CIFAR-10). They showed that neural networks can get close to 0 training error even if the labels of all training datapoints are completely re-randomized. This implies that their Rademacher complexity—even on datasets that they can generalize well on—can be very large. This was one of the early results (another one was [2]) which pointed out that neural networks seem to be behaving quite differently in terms of their generalization behavior. Here Rademacher complexity acts as a useful lens to uncover interesting behavior, even though it does not itself explain that behavior.*

1.2 Rademacher calculus

Rademacher complexity has several nice properties and works well with various function operations. Here we discuss a *calculus* of how various functions change the Rademacher complexity.

Claim 5 (Translation and Scaling)**.** *Let* $A' = \{\rho a + v, a \in A\}$. *Then* $RC(A') = \rho RC(A)$.

Exercise: Prove this bound.

Here is another bound that we can show for a finite collection of vectors.

Lemma 6 (Massart Lemma)**.** *Let* $A = \{v_1, \ldots, v_m\}$ *be a finite set of vectors in* $\mathbb{R}^n$. *Let* $\overline{v} = \frac{1}{m}\sum_{i=1}^{m} v_i$. *Then*

$$RC(A) \leq \max_i \|v_i - \overline{v}\|_2 \frac{\sqrt{2\log m}}{n}$$

Exercise: Prove this bound. First, by translation invariance, you can take $\overline{v} = 0$ without loss of generality. Then use the max of sub-Gaussian result from last time.

We note that this result gives a bound for finite hypothesisc classes. A good exercise is to verify that for the case of the zero-one loss, by combining Lemma 6 and Theorem 3 we can recover the previous bound that we have shown via uniform convergence for finite hypothesis classes.

Our next result on Rademacher calculus allows us to handle function compositions.

Lemma 7 (Contraction lemma)**.** *For each* $i \in [m]$, *let* $\phi_i : \mathbb{R} \to \mathbb{R}$ *be a* ρ*-Lipschitz function i.e.* $|\phi_i(x) - \phi_i(y)| \leq \rho|x - y| \ \forall \ x, y \in \mathbb{R}$. *For any* $a \in \mathbb{R}^n$ *define* $\phi(a) \in \mathbb{R}^n$ *as*

$$\phi(a) = (\phi_1((a)_1), \ldots, \phi_n((a)_n)).$$

For a set A, *let* $\phi \circ A = \{\phi(a) : a \in A\}$. *Then*

$$R(\phi \circ A) \leq \rho R(A).$$

Refer to Lemma 26.9 in [3] for the proof. One way that this Lemma comes in handy is when the loss function is L-Lipschitz, because we can then just bound the Rademacher complexity of the restriction of the hypothesis class to the dataset to get generalization bounds with Theorem 3.

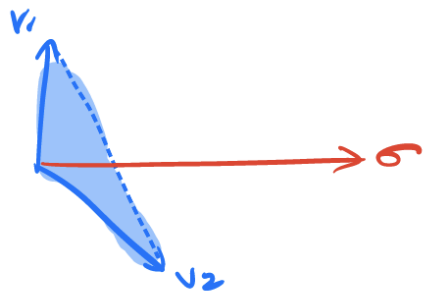

Figure 1: Another elementary result that we can show is that enlarging a set of vectors by taking their convex hull does not increasse the Rademacher complexity: $RC(\{\text{convex hull of } A\}) = RC(A)$. This is because the sup in rhe inner product is always one of the vertice of the convex hull, which were already present in the set.

1.3 Rademacher complexity of linear classes

We now consider two simple hypothesis classes and compute their Rademacher complexity. The two classes are linear predictions aiwth a L_1 and L_2 bound on the weight vectors.

- $\mathcal{H}_1 = \{h_w(x) = \langle w, x\rangle\} : \|w\|_1 \leq B_1\}$
- $\mathcal{H}_2 = \{h_w(x) = \langle w, x\rangle : \|w\|_2 \leq B_2\}$

Lemma 8 (ℓ_2 bounded linear predictor)**.** *Let* $S = (x_1, \ldots, x_n)$*. Define*

$$H_2 \circ S = \{(\langle w, x_1\rangle, \ldots, \langle w, x_n\rangle) : \|w\|_2 \leq B_2\}.$$

Then

$$RC(\mathcal{H}_2 \circ S) \leq \frac{B_2 \max_i \|x_i\|_2}{\sqrt{n}}$$

Proof. By Cauchy-Schwartz: $\langle w, v\rangle \leq \|w\|_2\|v\|_2$.

$$\begin{aligned}
\therefore nRC(\mathcal{H}_2 \circ S) &= \mathop{\mathbb{E}}_{\sigma}\left[\sup_{\mathcal{H}_2 \circ S} \sum_{i=1}^{n} \sigma_i a_i\right] \\
&= \mathop{\mathbb{E}}_{\sigma}\left[\sup_{w:\|w\|_2 \leq B_2} \sum_{i=1}^{n} \sigma_i \langle w, x_i\rangle\right] \\
&= \mathop{\mathbb{E}}_{\sigma}\left[\sup_{w:\|w\|_2 \leq B_2} \langle w, \sum_{i=1}^{n} \sigma_i x_i\rangle\right] \\
&\leq B_2 \cdot \mathop{\mathbb{E}}_{\sigma}\left[\|\sum_{i=1}^{n} \sigma_i x_i\|_2\right]. \qquad (1)
\end{aligned}$$

Using Jensen's,

$$\mathop{\mathbb{E}}_{\sigma}\left[\|\sum_{i=1}^{n} \sigma_i x_i\|_2\right] = \mathop{\mathbb{E}}_{\sigma}\left[\left(\|\sum_{i=1}^{n} \sigma_i x_i\|_2^2\right)^{1/2}\right] \leq \left(\mathop{\mathbb{E}}_{\sigma}\left[\|\sum_{i=1}^{n} \sigma_i x_i\|_2^2\right]\right)^{1/2} \qquad (2)$$

$$\mathop{\mathbb{E}}_{\sigma}\left[\|\sum_{i=1}^{n}\sigma_i x_i\|_2^2\right] = \mathop{\mathbb{E}}_{\sigma}\left[\sum_{i,j}\sigma_i\sigma_j\left\langle x_i, x_j\right\rangle\right].$$

Since σ_i are independent,

$$\mathop{\mathbb{E}}_{\sigma}[\sigma_i, \sigma_j] = 0 \quad \forall\, i \neq j$$

$$\implies \mathop{\mathbb{E}}_{\sigma}\left[\|\sum_{i=1}^{n}\sigma_i x_i\|_2^2\right] = \sum_{i=1}^{n}\|x_i\|_2^2 \leq n\max_i\|x_i\|_2^2. \tag{3}$$

The proof follows by combining (1), (2) and (3). ■

Lemma 9 (ℓ_1 bounded linear model)**.** *Let $S = (x_1, \ldots, x_n)$ where $x_i \in \mathbb{R}^d$ $\forall i \in [n]$ Then*

$$RC(\mathcal{H}_1 \circ S) \leq B_1 \max_i\|x_i\|_\infty\sqrt{\frac{2\log(2d)}{n}}$$

Proof. By Holder's inequality $\langle w, v\rangle \leq \|w\|_1\|v\|_\infty$. Therefore,

$$\begin{aligned}
nRC(\mathcal{H}_1 \circ S) &= \mathop{\mathbb{E}}_{\sigma}\left[\sup_{a\in H_1\circ S}\sum_{i=1}^{n}\sigma_i a_i\right] \\
&= \mathop{\mathbb{E}}_{\sigma}\left[\sup_{w:\|w\|_1\leq B_1}\sum_{i=1}^{n}\sigma_i\langle w_i, x_i\rangle\right] \\
&= \mathop{\mathbb{E}}_{\sigma}\left[\sup_{w:\|w\|_1\leq B_1}\langle w, \sum_{i=1}^{n}\sigma_i x_i\rangle\right] \\
&\leq B_1\cdot\mathop{\mathbb{E}}_{\sigma}\left[\|\sum_{i=1}^{n}\sigma_i x_i\|_\infty\right] \\
&= B_1\mathop{\mathbb{E}}_{\sigma}\left[\max_{j\in[d]}\left|\sum_{i=1}^{n}\sigma_i(x_i)_j\right|\right].
\end{aligned}$$

Note that each term $\sigma_i(x_i)_j$ is $|(x_i)_j|$ sub-Gaussian. Since $|(x_i)_j| \leq \max_i\|x_i\|_\infty$, each term $\sigma_i(x_i)_j$ is $\max_i\|x_i\|_\infty$ sub-Gaussian. The sum $\sum_{i=1}^{n}\sigma_i(x_i)_j$ is sub-Gaussian with parameter

$$\left(\sum_{i=1}^{n}\left(\max_i\|x_i\|_\infty\right)^2\right)^{1/2} \leq \sqrt{n}\cdot\max_i\|x_i\|_\infty.$$

By our bound on the expected value of the maximum of sub-Gaussian random variables (including negations of the orignal random variables to take care of the absolute value function), we have

$$nRC(\mathcal{H}_1 \circ S) \leq B_1\sqrt{n}\max_i\|x_i\|_\infty\sqrt{2\log(2d)}.$$

■

Lemma 8 and 9 bound the Rademacher complexity of the hypothesis class composed with the training set. If the loss function $\ell(h, z)$ is 1-Lipschitz (e.g. hinge loss or absolute value loss), then we can get a bound on the Rademacher complexity with the loss function factored in. For e.g. by the contraction lemma,

$$\mathcal{R}(\ell \circ \mathcal{H}_2 \circ S) \leq \mathcal{R}(\mathcal{H}_2 \circ S)$$

We can now get generalization bounds using the excess risk bound (Theorem 3).

We also note that the VC-dimension bound for linear predictors in $\mathbb{R}^d$ is $O(d)$. The Rademacher bound **does not directly depend polynomially on dimension**. Hence, it can be much smaller than the VC dimension.

1.4 Regularization

The above analysis for L_2 and L_1 bounded linear predictors is a good example of how choosing a suitable function class, and the related technique of *regularization* can be useful.

- Notice that the Rademacher complexity bound for $\mathcal{H}_2$ (Lemma 8) depends on $B_2 \max_i \|x_i\|_2$. So, if decide learn over a small ℓ_2-norm ball, we can have better generalization. In other words, less data could sufficefor getting small gap between training and test error if we restrict our predictor to have small ℓ_2 norm.

  Recall from Lecture 1 though that small generalization gap is not the only goal in supervised learning. The overall risk depends on the sum of the representation error, the optimization error, and the generalization error. We are ignoring the optimization error for the time being, by saying that we can find the empirical risk minimizer. So the goal becomes to balance the representation error and the generalization error. If we choose to restrict our hypothesis class to linear predictors which have L_2 norm bounded by B_2, and the best possible predictor w^* also has bounded norm bounded by B_2, then we can can get small representation error. In general we want to choose B_2 suitably such that the representation error and generalization error are simultaneously small.

- We also note that the Rademacher complexity bound for $\mathcal{H}_1$ (Lemma 9) depends on $B_1 \max_i \|x_i\|_\infty$. How does this compare with the bound for $\mathcal{H}_2$?

  Suppose $x_i \in \{\pm 1\}^d \implies \max_i \|x_i\|_\infty = 1$ & $\max_i \|x_i\|_2 = \sqrt{d}$. Also, suppose the true w^* is in $\{-1, 0, 1\}^d$, and it is also k-sparse. Then, $\|w^*\|_2 = \sqrt{k}$ and $\|w^*\|_1 = k$.

  To have good representation error, we should choose B_2 and B_1 for $\mathcal{H}_2$ and $\mathcal{H}_2$ respectively such that w^* lies in those hypothesis classes. Therefore, we should choose $B_2 = \sqrt{k}$ and $B_1 = k$. Then $B_1 \max_i \|x_i\|_\infty = k$ and $B_2 \max_i \|x_i\|_2 = \sqrt{kd}$. Therefore if $k << d$, working over the ℓ_1 ball could be much better than working over the ℓ_2 ball.

This discussion motivates how working with the right hypothesis class (which is also known as having the right inductive bias) can lead to better accuracy. Regularization is a technique which allows us to smoothly control the complexity of the model we learned. In empirical risk minimization,

we associate the same complexity with any hypothesis $h \in \mathcal{H}$, since we find any minimizer of the empirical risk:

$$\min_{h \in \mathcal{H}} \Big(\frac{1}{n} \sum_{i=1}^{n} \ell(h(x_i), y_i) \Big).$$

Structural risk minimization (SRM) in contrast, does not necessarily regard all the hypothesis in $\mathcal{H}$ as being equivalent. We define some function $\psi(h)$, which measures complexity of any $h \in \mathcal{H}$. Simpler hypothesis in $\mathcal{H}$ should have smaller values for $\psi(h)$. In SRM, the objective function is to minimize the sum of the empirical risk and the complexity of the learned hypothesis.

Definition (Structural Risk Minimization (SRM))**.** *For a given hypothesis class $\mathcal{H}$, complexity function $\psi(h)$ for any $h \in \mathcal{H}$ training dataset $\{(x_i, y_i), i \in [n]\}$ and $\lambda >= 0$, we solve the following optimization problem:*

$$\min_{h \in \mathcal{H}} \Big(\frac{1}{n} \sum_{i=1}^{n} \ell(h(x_i), y_i) + \lambda \psi(h) \Big).$$

λ is known as the regularization strength.

λ controls how much weight to put on the regularization term. If $\lambda = 0$, then the SRM problem is the same as the ERM problem. If $\lambda \to \infty$, then the objective function only minimizes complexity, and disregards the empirical risk on the training set.

As an example, consider linear predictors once more, so $h_w(x) = w^T x$. Two possible complexity functions or regularization functions $\psi(h)$ are $\psi(h) = \|w\|_2^2$ (which is known as L_2 regularization) and $\psi(h) = \|w\|_1$ (which is known as L_1 regularization). Using the method of Langrange multipliers, it can be shown that the SRM objective

$$\arg\min_{h \in \mathcal{H}} \Big(\frac{1}{n} \sum_{i=1}^{n} \ell(h(x_i), y_i) + \lambda \psi(h) \Big),$$

is equivalent to the following constrained optimization problem:

$$\arg\min_{h \in \mathcal{H}} \frac{1}{n} \sum_{i=1}^{n} \ell(h(x_i), y_i)$$
$$\text{subject to } \psi(h) \leq \beta$$

for some suitable β depending on λ. Therefore, we can see doing regularization is equivalent to controlling the radius of the L_2 or L_1 ball (B_2 and B_1 respectively) from which we choose our predictor. Therefore, choosing the radius (and hence λ) suitably can help balance the tradeoff between the representation error and the generalization error.

To choose λ, we generally rely on a **validation set**. Earlier we talked about the training/test split, the validation set is another split which is usually employed to tune the regularization and other hyperparameters, and model selection more broadly. The reason that a validation set is necessary from the persepective of regularization is because our eventual goal is supervised learning is to get low error on the unseen test set. Choosing smaller λ in the SRM objective will usually help the ERM objective (since more weight is put on the ERM objective). However, this does not mean that the solution we find generalize well, since we could end up with a overly complex model which overfits on the training set.

2 Algorithmic Stability

So far, we have seen measures to bound the complexity of a given hypothesis classes (such as the size of a finite hypothesis class, the VC dimensions etc.). We also saw how regularization provides a more fine-grained knob to control the complexity of models within this hypothesis class as well (such as based on the L_2 or L_1 norm for linear predictors). For modern hypothesis classes such as hugely overparameterized neural networks, direct measures of complexity can often by overly pessimistic since they suggest a data requirement which is much larger than what seems sufficient for generalization in datasets of interest. Though regularization techniques (such as L_2 regularization on the weights, known as weight decay in that context) are still quite useful in practice for neural networks, their role from the perspective of generalization seems unclear.

It appears instead that the algorithms that we use for training these models have a possibly large role to play in helping these large models generalize. We will now see a new notion, called **algorithmic stability**, which captures when a particular algorithm can be expected to generalize.

We begin with some notation. As before let $S = \{(x_1, y_1), \ldots, (x_n, y_n)\}$ be a training set of n examples drawn i.i.d. from $\mathcal{D}$. Let z_i be the labelled example, $z_i = (x_i, y_i)$. Let $S' = \{z'_i, \ldots, z'_n\}$ be another dataset of n i.i.d. examples drawn from $\mathcal{D}$. We also define a hybrid dataset $S^{(i)} = \{z_1, \ldots, z_{i-1}, x'_i, z_{i+1}, \ldots, z_n\}$, where we substitute the i-th example from S with the corresponding example in S'. We are now ready to define our first notion of stability, known as average stability.

Definition (Average Stability)**.** *For any algorithm A which outputs the predictor $A(S)$ on training set S, the average stability $\Delta(A)$ is*

$$\Delta(A) = \mathbb{E}_{S,S'}\Big[\frac{1}{n}\sum_{i=1}^{n}(\ell(A(S), z'_i) - \ell(A(S^{(i)}), z'_i))\Big]$$

One way to understand this definition is that it measures how sensitive the algorithm to changes in one training datapoint. This is average over all training datapoints, since the algorithm could be sensitive to any one of them.

Notice in this definition that in the first term $\ell(A(S), z'_i)$, z'_i is unseen to an algorithm which trains on the set S. In the second term $\ell(A(S^{(i)}), z'_i)$, z'_i is seen to the algorithm since it is part of the training set S'. Therefore this definition measures how much the predictions of an algorithm on a datapoint change when that datapoint is part of the training set. Intuitively, if an algorithm generalizes well then it's predictions on some test point should not change if that test point is included in the training set, since generalization requires that the algorithm behaves similarly on the training and test set. This can in fact be formalized, and it turns out that the expected generalization gap is *exactly* equal to the average stability of the algorithm.

Proposition 10 (Expected generalization gap equals average stability)**.** *Define $\Delta_{gen}(h)$ to be the gap between test and traing errors, $\Delta_{gen}(h) = R(h) - \hat{R}_S(h)$. Then*

$$\mathbb{E}_S[\Delta_{gen}(A(S))] = \Delta(A).$$

Proof. Since this is an exact equality, the proof just involves expanding out the definition and reinterpretating the terms that we get. The expression we want to analyze is,

$$\mathbb{E}_S[\Delta_{gen}(A(S))] = \mathbb{E}_S[R(A(S)) - \hat{R}_S(A(S))]$$

We begin with the first term, which involves a datapoint unseen to the algorithm. Since $\mathop{\mathbb{E}}_{z_i'} \ell(A(S), z_i') = \mathbb{E}_S[R(A(S))]$ by definition (since z_i' is i.i.d. from D), by linearity of expectations

$$\mathbb{E}_S[R(A(S))] = \mathbb{E}_{S,S'}[\frac{1}{n}\sum_{i=1}^{n} \ell(A(S), z_i')].$$

We now consider the second term, which involves a previously seen datapoint.

$$\mathbb{E}_S[\hat{R}_S(A(S))] = \mathbb{E}_S\left[\frac{1}{n}\sum_{i=1}^{n} \ell(A(S), z_i)\right]$$

Note that $\mathbb{E}_S[\ell(A(S), z_i)] = \mathbb{E}_{S,S'}[\ell(A(S^{(i)}, z_i')]$ since S and S' are sampled from the same distribution. Therefore,

$$\mathbb{E}_S[\hat{R}_S(A(S))] = \mathbb{E}_{S,S'}\left[\frac{1}{n}\sum_{i=1}^{n} \ell(A(S^{(i)}), z_i')\right]$$

which finishes the proof. ■

Average stability requires taking an average over training sets drawn from the distribution. It is often easier to bound the maximum value instead of the average, which leads to the notion of uniform stability.

Definition 11 (Uniform Stability)**.** *The uniform stability* $\Delta_{\sup}(A)$ *of an algorithm* A *is defined as*

$$\Delta_{\sup}(A) = \sup_{\substack{S,S' \in (\mathcal{X}\times\mathcal{Y})^n \\ \text{s.t. } S,S' \text{ differ in one point}}} \sup_{z \in \mathcal{X}\times\mathcal{Y}} |\ell(A(S), z) - \ell(A(S'), z)|.$$

Notice that $\Delta(A) \leq \Delta_{\sup}(A)$ by definition. Therefore, you can verify that uniform stability can also be used to upper bound the generalization error.

Claim 12. $\mathbb{E}[\Delta_{gen}(A(S)] \leq \Delta_{\sup}(A)$

As we discussed in the beginning of this section, in contrast to our previous generalization measures stability is a property of an algorithm instead of the hypothesis class. Many algorithms are known to be stable. A simple example is SRM with L_2 regularization. Consider some hypothesis class parameterized by $w \in \mathbb{R}^d$. Let the loss function $\ell(w, z)$ for $z = (x, y)$ be

1. Convex in w. An example of this is linear prediction with a convex loss, such as $\ell(w, z) = (w^T x - y)^2$ or $\ell(w, z) = |w^T x - y|$.
2. L-Lipschitz in w, i.e. $|\ell(w_1, z) - \ell(w_2, z)| \leq L \cdot \|w_1 - w_2\|_2$.

We define the SRM objective as

$$F_S(w) = \frac{1}{n}\sum_{i=1}^{n} \ell(w, z_i) + \lambda\|w\|_2^2. \tag{4}$$

Theorem 13 shows that the algorithm which minimizes the structural risk is uniformly stable. Therefore due to Claim 12, this algorithm will have small generalization gap.

Theorem 13. *Assume $\ell(w, z)$ is convex and L-Lipschitz. Then the SRM algorithm (which minizes (4)) satisfies*

$$\Delta_{\sup}(SRM) \leq \frac{2L^2}{\lambda n}$$

Proof. Let $\hat{w}_S = \arg\min_w F_S(w)$. Let S and S' be two sets of n examples which differ at index $i \in [n]$. We need to show that $|\ell(\hat{w}_S, z) - \ell(\hat{w}_{S'}, z)| \leq \frac{2L^2}{\lambda n}$.
Since ℓ is L-Lipschitz, it suffices to show that $\|\hat{w}_S - \hat{w}_{S'}\|_2 \leq \frac{2L}{\lambda n}$.

Claim 14. *For any w, $F_S(w) - F_S(\hat{w}_S) \geq \lambda \|w - \hat{w}_S\|_2^2$*

Proof. The proof follows by strong convexity, which we will define and study further later in the class.

To get some intuition for this in the meantime, consider the case when w is univariate. Then we can do a Taylor series expansion around the minizer $\hat{w}_S$ to write,

$$F_S(w) = F_{S(\hat{w}_S)} + \frac{\partial F_S(w)}{\partial w}\Big|_{w=\hat{w}_S}(w - \hat{w}_S) + \frac{1}{2!}\frac{\partial^2 F_S(w)}{\partial^2 w}\Big|_{w=\hat{w}_S}(w - \hat{w}_S)^2 + \cdots .$$

Note that the first derivative at $\hat{w}_S$ is 0 if $\hat{w}_S$ is the minimizer of $F_S(w)$. The second derivative is at least 2, since $\ell(w, z)$ is convex in w (and hence has non-negative 2nd derivative) and w^2 has second derivative 2. ■

Using this result for $w = \hat{w}_{S'}$, we get,

$$F_S(\hat{w}_{S'}) - F_S(\hat{w}_S) \geq \lambda \|\hat{w}_S - \hat{w}_{S'}\|_2^2 \tag{5}$$

We can also rewrite and bound $F_S(\hat{w}_{S'}) - F_S(\hat{w}_S)$ in another way,

$$\begin{aligned} F_S(\hat{w}_{S'}) - F_S(\hat{w}_S) &= \frac{1}{n}(\ell(\hat{w}_{S'}, z_i) - \ell(\hat{w}_S, z_i)) + \frac{1}{n}\sum_{j \neq i}(\ell(\hat{w}_{S'}, z_j) - \ell(\hat{w}_S, z_j)) \\ &\quad + \lambda\|\hat{w}_{S'}\|_2^2 - \lambda\|\hat{w}_S\|_2^2 \\ &= \frac{1}{n}(\ell(\hat{w}_{S'}, z_i) - \ell(\hat{w}_S, z_i)) - \frac{1}{n}(\ell(\hat{w}_{S'}, z_i') - \ell(\hat{w}_S, z_i'))) \\ &\quad + \frac{1}{n}\sum_j (\ell(\hat{w}_{S'}, z_j') - \ell(\hat{w}_S, z_j')) + \lambda\|\hat{w}_{S'}\|_2^2 - \lambda\|\hat{w}_S\|_2^2. \end{aligned}$$

Note that $\frac{1}{n}\sum_{j \neq i} \ell(w, z_j) + \lambda\|w\|_2^2 = F_{S'}(w)$. Moreover, by definition $\hat{w}_{S'}$ is the minimizer of $F_{S'}(w)$, therefore $\frac{1}{n}\sum_j (\ell(\hat{w}_{S'}, z_j') - \ell(\hat{w}_S, z_j')) + \lambda\|\hat{w}_{S'}\|_2^2 - \lambda\|\hat{w}_S\|_2^2 \leq 0$. Hence we can write,

$$\begin{aligned} F_S(\hat{w}_{S'}) - F_S(\hat{w}_S) &\leq \frac{1}{n}|\ell(\hat{w}_{S'}, z_i) - \ell(\hat{w}_S, z_i)| + \frac{1}{n}|\ell(\hat{w}_{S'}, z_i') - \ell(\hat{w}_S, z_i')| \\ &\leq \frac{2L}{n}\|\hat{w}_{S'} - \hat{w}_S\|_2 \end{aligned} \tag{6}$$

Combing (5) and (6), we get that $\|\hat{w}_S - \hat{w}_{S'}\|_2 \leq \frac{2L}{\lambda n}$, which proves the result.

■

3 Further reading

Rademacher complexity is Chapter 26 of the book [3]. Structural risk minimization is discussed in Chapter 7 of the book. Stability and the role of regularization in stability is in Chapter 13. A lot of our discussion of stability is based on Chapter 6 of [4].

References

[1] Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning (still) requires rethinking generalization. *Communications of the ACM*, 64(3):107–115, 2021.

[2] Behnam Neyshabur, Ryota Tomioka, and Nathan Srebro. In search of the real inductive bias: On the role of implicit regularization in deep learning. *arXiv preprint arXiv:1412.6614*, 2014.

[3] Shai Shalev-Shwartz and Shai Ben-David. *Understanding machine learning: From theory to algorithms*. Cambridge university press, 2014.

[4] Moritz Hardt and Benjamin Recht. Patterns, predictions, and actions: A story about machine learning. *arXiv preprint arXiv:2102.05242*, 2021.