IFT 6085 - Lecture 9
Stability, Generalization and the Applications of Stability

This version of the notes has not yet been thoroughly checked. Please report any bugs to the scribes or instructor.

Scribes
Winter 2020: Zarour Mahdi
Winter 2019: Nishanth V Anand, Parviz Haggi,
Bhargav Kanuparthi, Jonathan Pilault
Winter 2018: Isabela Albuquerque and Nithin Vasisth,
Amy Zhang, William Fedus

Instructor: Ioannis Mitliagkas

1 Summary of the previous lecture

Previously we discussed the idea behind a few bounds in Statistical Learning Theory, the more advanced one being PAC-based Learning Theory. There we saw that we have to commit to a prior distribution on a hypothesis (P) and choose the posterior hypothesis (Q) after seeing the data. This is a powerful method as different choices of prior and posterior hypotheses can be made, each resulting in a new bound without us touching the algorithm. In a coming lecture we will discuss a concrete example of a PAC-based bound for Neural Networks. There, the discussion will also include the notion of stability-based bounds, which is the subject of this lecture.

The algorithm-agnostic bounds take into account the complexity of the hypothesis class of functions $\mathcal{H}$, without involving the algorithm or the actual distribution of the data. To put it correctly, for the Hoeffding bound to work, the distribution of the data was included in the analysis through the assumption that the data points were i.i.d.. Apart from that, however, no other information about the distribution was used.

Although we will not delve into the subject of distribution-agnostic bounds, in this lecture we will introduce the first class of bounds that take into account the algorithm. The analysis is largely dependent on the notion of stability which, simply put, says that a change in data distribution does not change the predictions.

2 PAC Learning

The setting of the Probably Approximately Correct (PAC) learning involves the same definitions as before but with a slight change in notation that will help us in our analysis: We introduce $z_i = (x_i, y_i)$ i.e we give each pair a name.

Definition 1 (Training set). *The training set consists of a set of values $z_i = (x_i, y_i)$, where x_i represents a feature vector and y_i the label of the i-th sample. Furthermore, z_i are assumed to be i.i.d. and sampled from an unknown data distribution $\mathcal{D}$.*

$$S = \{z_1, z_2, \dots, z_n\}$$

Next we define the loss function slightly differently.

Definition 2 (Loss Function). *The loss function $\ell(h(x), y)$ is defined as a function that takes two labels and produces a value between 0 and some constant M.*

$$\ell : \mathcal{Y} \times \mathcal{Y} \longrightarrow [0, M].$$

Equivalently, defining $\mathcal{Z} \triangleq \mathcal{X} \times \mathcal{Y}$, the loss function $\ell(h, z)$ can be defined as

$$\ell : \mathcal{H} \times \mathcal{Z} \longrightarrow [0, M].$$

Notice that, unlike its previous definition, the loss function is now assumed to be bounded by some constant M instead of 1. Note also that $\mathcal{Z}$ is the space of all tuples (x_i, y_i) and that the previous loss function $\ell(h(x), y)$ is now denoted as $\ell(h, z)$

3 Stability

We start this section by introducing two new notions, the first of which is the notion of perturbed datasets. This is a step-stone in introducing a new class of bounds that unlike the *algorithm-agnostic* bounds, are dependent on the algorithm.

Definition 3 (A Perturbed dataset)**.** *Given a dataset S (i.i.d.), a perturbed dataset $S^{i,z}$ is defined as*

$$S^{i,z} = \{z_1, z_2, \ldots, z_{i-1}, z, z_{i+1}, \ldots, z_n\}.$$

According to this definition, a perturbed dataset $S^{i,z}$ is defined by a set whose i-th element is replaced by an arbitrary sample z. We will see that some learning algorithms give essentially a hypothesis that makes the same predictions no matter if the algorithm is trained on the original dataset or the perturbed one.

Definition 4 (Algorithm)**.** *A learning algorithm $\mathcal{A}$ is defined as the following mapping*

$$\mathcal{A} : (\mathcal{X} \times \mathcal{Y})^n \to \mathcal{H}.$$

It is clear from this definition that an algorithm $\mathcal{A}$ used on a dataset S, produces a hypothesis class h_S i.e. $h_S = \mathcal{A}(S)$.

Definition 5 (Uniform stability)**.** *An algorithm $\mathcal{A}$ is β-uniformly stable with respect to the loss function ℓ if*

$$\forall (S, z) \in \mathcal{Z}^{n+1} \ \textit{and} \ \forall i \in \{1, 2, \ldots, n\} : \sup_{z' \in \mathcal{Z}} |\ell(h_S, z') - \ell(h_{S^{i,z}}, z')| \leq \beta.$$

This definition measures stability based on how much the predictions or the losses on the predictions change when we train using the perturbed dataset. Notice the two hypotheses: h_S that we get when using the unperturbed dataset and $h_{S^{i,z}}$ that we derive from the perturbed dataset. This definition holds for any dataset but there is no assumption on what particular distribution S comes from. The notion of a β-uniformly stable algorithm is reminiscent of the familiar notion of Lipschitz property on the loss function. Intuitively, an algorithm with this property can be understood as one that produces a hypothesis such that the loss function ℓ is not drastically affected by perturbing the dataset in this manner.

Definition 6 (Defect)**.**

$$D[h_S] = R[h_S] - \hat{R}_S[h_S].$$

Defect $D[h_S]$ for a hypothesis h_S derived from an algorithm after seeing the dataset S is defined as the difference between the population risk and the empirical risk.

While it is true that for an arbitrary hypothesis $h \in \mathcal{H}$, $\mathbb{E}[D[h]] = \mathbb{E}[R[h] - \hat{R}_S[h]] = 0$, this is not the case for $D[h_S]$ i.e. we will generally have

$$\mathbb{E}[D[h_S]] \neq 0.$$

This is due to second term $\hat{R}_S[h_S]$ which evaluates the empirical risk on the same dataset that is also used to extract the hypothesis.

$$\begin{aligned}
\mathbb{E}_S[\hat{R}_S[h_S]] &= \mathbb{E}_S[\frac{1}{n}\sum_{i=1}^{n} \ell(h_S(x_i), y_i)] \\
&= \frac{1}{n}\sum_{i=1}^{n} \mathbb{E}_S[\ell(h_S(x_i), y_i)] \\
&\neq \mathbb{E}_S[\mathbb{E}_{z\sim\mathcal{D}}\left[\ell(h_S(x), y)\right]] \triangleq \mathbb{E}_S[R[h_S]]
\end{aligned}$$

In the second line of the above equation, the expectation value is evaluated based on the dataset S and the hypothesis extracted from it. The expectation value of the population risk on the learned hypothesis (third line) is evaluated on a random variable z, independent from S and drawn from the distribution that the dataset is assumed to come from. The two entities are generally completely different simply because we need the independence assumption for the population risk.

A meaningful question to ask here is whether the expectation value of the defect, which we showed is generally non-zero, can be bounded. In what follows, we will show that the answer is yes and that this can be done under certain conditions.

Theorem 7 (Bounding the expectation value of the defect)**.** *If $\mathcal{A}$ is a β-uniformly stable algorithm, then*

$$-\beta \leq \mathbb{E}_S[D[h_S]] \leq \beta.$$

Proof. We prove this for one side of the inequality: $-\mathbb{E}_S[D[h_S]] \leq \beta$

$$\begin{aligned}
-\mathbb{E}_S[D[h_S]] &= \mathbb{E}_S\Big[\hat{R}_S[h_S] - R[h_S]\Big] \\
&= \mathbb{E}_S\Big[\frac{1}{n}\sum_{i=1}^{n} \ell(h_S, z_i) - \mathbb{E}_z \ell(h_S, z)\Big] \\
&= \mathbb{E}_{S,z}\Big[\frac{1}{n}\sum_{i=1}^{n} \Big[\ell(h_S, z_i) - \ell(h_S, z)\Big]\Big] \\
&= \mathbb{E}_{S,z}\Big[\frac{1}{n}\sum_{i=1}^{n} \Big[\ell(h_{S^{i,z}}, z) - \ell(h_S, z)\Big]\Big] \\
&\leq \mathbb{E}_{S,z}\Big[\frac{1}{n}\sum_{i=1}^{n} \beta\Big] = \beta
\end{aligned}$$

□

In the second line we inserted the population risk as defined above: $R[h_S] = \mathbb{E}_z \ell(h_S, z)$ In the third line, we used Fubini's theorem which allows us to change the order of the two expectations $\mathbb{E}_S$ and $\mathbb{E}_z$ as they are independent and bounded (due to the fact that the loss function is bounded by M). In the fourth line we rename a variable and finally we calculate the upper bound by using the definition of β-stability.

Note concerning the 4^{th} line of the proof:

- Getting a hypothesis from $(z_1, z_2, ..., z_n)$ which is h_S and computing the loss in z_1 for example.
- And Getting a hypothesis from $(z, z_2, ..., z_n)$ which is $h_{S^{1,z}}$ and computing the loss in z is the same thing.

And we generalize that for i in $\{1, 2, ..., n\}$

In conclusion we have proven the following relationship between the expectation values of the empirical and population risks:

Property 8 (The relationship between the empirical and the population risk)**.**

$$\mathbb{E}_S[R[h_S]] \leq \mathbb{E}_S[\hat{R}_S[h_S]] + \beta.$$

Note that this is a bound on the *expectation* value of the population risk. However, even if the expectation values of $R[h_S]$ and $\hat{R}_S[h_S]$ are close, this bound does not necessarily hold for all possible h_S. In what follows, we will demonstrate that for a β-uniformly stable algorithm, the population risk $R[h_S]$ can be shown to be bounded above by the empirical risk $\hat{R}_S[h_S]$ plus certain other quantities. To do so we will first introduce McDiramid's inequality, a well known concentration inequality.

Theorem 9 (McDiarmid's inequality)**.** *Let $V_1, V_2, V_3, \dots, V_n \in \mathcal{V}$ be independent random variables, and $v_1, v_2, v_3, \dots, v_n$ denote specific values (not independent). If a function $f : \mathcal{V}^n \to \mathbb{R}$ has the property that $\forall i \in \{1, 2, \dots, n\}$,*

$$\sup_{v_1, v_2, \dots, v_n, v_i{}'} \left| f(v_1, v_2, \dots, v_n) - f(v_1, \dots, v_{i-1}, v_i^{'}, v_{i+1}, \dots v_n) \right| \leq c_i$$

then

$$\mathbb{P}\bigg(\big| f(V_1, V_2, \dots, V_n) - \mathbb{E} f(V_1, V_2, \dots, V_n) \big| > \epsilon \bigg) \leq 2 \exp \frac{-2\epsilon^2}{\sum_i c_i^2}$$

Proof. See Appendix D.2 of [1]. □

This bound is useful because if we prove that an algorithm is β stable then we will have this property on a specific function.

Theorem 10 (Bound for the defect)**.** *Let "A" be a β uniformly stable learning algorithm with respect to a loss function $\ell : \mathcal{Y} \times \mathcal{Y} \to [0, M]$. The absolute difference of the defect calculated on a dataset S and on a perturbed version of this dataset $S^{i,z}$ is bounded by*

$$\left| D[h_S] - D[h_{S^{i,z}}] \right| \leq 2\beta + \frac{M}{n}.$$

This theorem gives us a bound on the gap between the actual dataset and the perturbed dataset. This is used as a stepping stone in the final theorem. In other words, the theorem tells us that the population risk and the empirical risk are close to each other. This theorem is an example of the use case of McDiramid's inequality.

Proof. Let us expand the following quantity using their definition:

$$|D[h_S] - D[h_{S^{i,z}}]| = |R[h_S] - \hat{R}_S[h_S] - R[h_{S^{i,z}}] + \hat{R}_{S^{i,z}}[h_{S^{i,z}}]| \tag{1}$$

Using the triangle inequality:

$$|D[h_S] - D[h_{S^{i,z}}]| \leq |R[h_S] - R[h_{S^{i,z}}]| + |\hat{R}_{S^{i,z}}[h_{S^{i,z}}] - \hat{R}_S[h_S]| \tag{2}$$

Now, we use the β uniform stability of algorithm $\mathcal{A}$ with respect to the loss function ℓ to find a bound for $|R[h_S] - R[h_{S^{i,z}}]|$:

$$\begin{aligned} \big| R[h_S] - R[h_{S^{i,z}}] \big| &= \big| \mathbb{E}_{z' \sim D}[\ell(h_S, z')] - \mathbb{E}_{z'}[\ell(h_{S^{i,z}}, z')] \big| \\ &= \big| \mathbb{E}_{z' \sim D}[\ell(h_S, z') - \ell(h_{S^{i,z}}, z'))] \big| \\ &\leq \beta \end{aligned} \tag{3}$$

We can find a bound for $|\hat{R}_{S^{i,z}}[h_{S^{i,z}}] - \hat{R}_S[h_S]|$ by expanding the quantities using their definition (notice that the perturbed dataset is only the non-perturbed one with subtracting z_i and adding z instead)

$$\begin{aligned} \big| \hat{R}_{S^{i,z}}[h_{S^{i,z}}] - \hat{R}_S[h_S] \big| &= \left| \frac{1}{n} \sum_{j=1}^{n} \ell(h_{S^{i,z}}, z_j) - \frac{1}{n} \ell(h_{S^{i,z}}, z_i) + \frac{1}{n} \ell(h_{S^{i,z}}, z) - \frac{1}{n} \sum_{j=1}^{n} \ell(h_S, z_j) \right| \\ &\leq \frac{1}{n} \left| \ell(h_{S^{i,z}}, z) - \ell(h_{S^{i,z}}, z_i) \right| + \frac{1}{n} \sum_j \left| \ell(h_{S^{i,z}}, z_j) - \ell(h_S, z_j) \right| \\ &\leq \frac{M}{n} + \beta \end{aligned} \tag{4}$$

We could do this last step because we know that $\left| \ell(h_{S^{i,z}}, z) - \ell(h_{S^{i,z}}, z_i) \right|$ is bounded by M from **Definition 2**

We now plug in the results obtained from Eq 3 and Eq 4 in Eq 2 to get the proof.

$$|D[h_S] - D[h_{S^{i,z}}]| \leq 2\beta + \frac{M}{n} \tag{5}$$

□

Theorem 11 (Bound for the population risk of a β-uniformly stable algorithm). *Consider a β-uniformly stable algorithm $\mathcal{A}$ with respect to a loss function $\ell : \mathcal{Y} \times \mathcal{Y} \rightarrow [0, M]$ and a hypothesis h_S with $|S| = n$. The following bound holds with probability $1 - \delta$:*

$$R[h_S] \leq \hat{R}_S[h_S] + \beta + \left(n\beta + \frac{M}{2}\right)\sqrt{\frac{2\log\frac{2}{\delta}}{n}}.$$

Note that:

- $\hat{R}_S[h_S]$ is empirical risk
- β is from theorem 7
- $\left(n\beta + \frac{M}{2}\right)\sqrt{\frac{2\log\frac{2}{\delta}}{n}}$ is a concentration inequality (McDiramid's)

Proof. Using theorem 10, we state McDiarmid's inequality for $D[h_S]$ and then use this result to find a high probability bound for $D[h_S]$:

$$\sup_{S,i,z} |D[h_S] - D[h_{S^{i,z}}]| \leq 2\beta + \frac{M}{n} \tag{6}$$

then

$$\begin{aligned} P(|D[h_S] - \mathbb{E}[D[h_S]]| > \epsilon) &\leq 2\exp\left(\frac{-2\epsilon^2}{\sum_{i=1}^n \left(2\beta + \frac{M}{n}\right)^2}\right), \\ &= 2\exp\left(\frac{-2n\epsilon^2}{\left(2n\beta + M\right)^2}\right), \\ &= 2\exp\left(\frac{-2n\epsilon^2}{4\left(n\beta + \frac{M}{2}\right)^2}\right), \\ &= 2\exp\left(\frac{-n\epsilon^2}{2\left(n\beta + \frac{M}{2}\right)^2}\right). \end{aligned} \tag{7}$$

Denoting $\delta = 2\exp\left(\frac{-n\epsilon^2}{2\left(n\beta + \frac{M}{2}\right)^2}\right)$ and solving this equation for ϵ, we obtain:

$$\begin{aligned} \delta = 2\exp\left(\frac{-n\epsilon^2}{2\left(n\beta + \frac{M}{2}\right)^2}\right) &\Rightarrow n\epsilon^2 = 2\log\frac{2}{\delta}\left(n\beta + \frac{M}{2}\right)^2 \\ &\Rightarrow \epsilon = \left(n\beta + \frac{M}{2}\right)\sqrt{\frac{2\log\frac{2}{\delta}}{n}}. \end{aligned} \tag{8}$$

Thus, with probability $1 - \delta$

$$\begin{aligned} |D[h_S] - \mathbb{E}[D[h_S]]| &\leq \epsilon \\ D[h_S] &\leq \mathbb{E}[D[h_S]] + \epsilon \\ D[h_S] &\leq \beta + \epsilon \end{aligned}$$

Replacing ϵ by the result previously obtained in Eq. 8

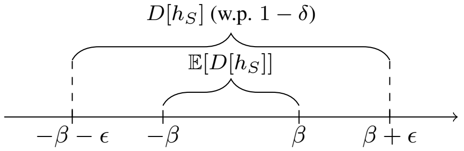
$$D[h_S] \leq \beta + \left(n\beta + \frac{M}{2}\right)\sqrt{\frac{2\log\frac{2}{\delta}}{n}} \tag{9}$$

we finally get the desired result,

$$R[h_S] \leq \hat{R}_S[h_S] + \beta + \left(n\beta + \frac{M}{2}\right)\sqrt{\frac{2\log\frac{2}{\delta}}{n}} \tag{10}$$

□

Some notes :
- Notice that as n goes up, our bound becomes less tight (even vacuous) which is not what we want. So we are not satisfied with our current results.
- The term $(\beta n + \frac{M}{2})\sqrt{\frac{2\ln 2/\delta}{n}}$ is $O(\beta\sqrt{n})$. Informally, an algorithm is stable if $\beta = O(\frac{1}{n})$. If stability is $O(\frac{1}{\sqrt{n}})$, this term is $O(1)$ and we can no longer show decrease in generalization gap with with increase in n.
The following illustration represents a summary of the bounds stated in Theorem 7 and 11 (in terms of the defect):



One can observe that despite the fact the bound for $\mathbb{E}[D[h_S]]$ is tighter, we have no guarantees that the actual of $D[h_S]$ lies in the interval $[-\beta, \beta]$. On the other hand, it possible to assure with probability $1-\delta$ will be in $[-\beta-\epsilon, \beta+\epsilon]$.

Summary

Sufficient condition: Given enough samples we can achieve a good enough generalization. However, typically in deep learning, we never have large enough data sets to get non-vacuous or meaningful bounds.

Last part	Next part
PAC Bounds Occam Bounds PAC Bayes Bounds Stability Bounds	Stability PAC Bayes (Practical) Generalization

How can we go from PAC Bayes to a non-vacuous generalization bound?

By sacrificing some data as part of a dedicated test set, we can measure test set generalization and achieve a tighter bound than the weak population bounds. See *Tutorial on Practical Prediction Theory for Classification* [**?**] for a comprehensive examination.

Empirical Risk Minimization + Regularization is Stable

Notation:

$$\hat{R}_S(w) \triangleq \hat{R}_S(h_w)$$

where h_w is a model parameterized by weights w.

$$l(h, z) \equiv l(h(x), y)$$
$$l(h_w, z) \equiv l(w, z)$$

Theorem 12 (ERM with regularization is β-stable)**.** *Under the assumption that $\hat{R}_S(w)$ is λ-convex and $l(\cdot|z)$ is L-Lipschitz $\forall z$, Empirical Risk Minimization and Regularization is β uniformly stable where*

$$\beta = \frac{4L^2}{\lambda n}$$

Proof. The objective function to be optimized can be written as

$$f_S(w) = \hat{R}_S(w) + \frac{\lambda}{2}||w||_2^2$$

Consider weights u, v for two different models.

$$f_S(v) - f_S(u) = [\hat{R}_S(v) + \frac{\lambda}{2}||v||_2^2] - [\hat{R}_S(u) + \frac{\lambda}{2}||u||_2^2]$$

We perturb the dataset by replacing the data point at i with z_i'. Now we get:

$$\begin{aligned} f_S(v) - f_S(u) &= \hat{R}_{S_{i,z_i'}}(v) + \frac{\lambda}{2}||v||_2^2 - (\hat{R}_{S_{i,z_i'}}(u) + \frac{\lambda}{2}||u||_2^2) + \frac{l(v, z_i) - l(v, z_i')}{n} - \frac{l(u, z_i) - l(u, z_i')}{n} \\ &= f_{S_{i,z_i'}}(v) - f_{S_{i,z_i'}}(u) + \frac{l(v, z_i) - l(v, z_i')}{n} - \frac{l(u, z_i) - l(u, z_i')}{n} \end{aligned}$$

Now we substitute $v = \mathcal{A}(S_{i,z_i'})$ and $u = \mathcal{A}(S)$.

$$\begin{aligned} f_S(\mathcal{A}(S_{i,z_i'})) - f_S(\mathcal{A}(S)) =& f_{S_{i,z_i'}}(\mathcal{A}(S_{i,z_i'})) - f_{S_{i,z_i'}}(\mathcal{A}(S)) \\ &+ \frac{l(\mathcal{A}(S_{i,z_i'}), z_i) - l(\mathcal{A}(S_{i,z_i'}), z_i')}{n} - \frac{l(\mathcal{A}(S), z_i) - l(\mathcal{A}(S), z_i')}{n} \end{aligned}$$

Because

$$\begin{aligned} f_{S_{i,z_i'}}(\mathcal{A}(S_{i,z_i'})) &= \min_w f_{S_{i,z_i'}}(w) \\ \implies \forall w f_{S_{i,z_i'}}(w) &\geq f(S_{i,z_i'})(\mathcal{A}(S_{i,z_i'})) \end{aligned}$$

Assumption 13. *$l(\cdot|z)$ is L-Lipschitz.*

$$\begin{aligned} f_S(\mathcal{A}(S_{i,z_i'})) - f_S(\mathcal{A}(S)) &\leq \frac{l(\mathcal{A}(S^{i,z_i'}), z_i) - l(\mathcal{A}(S), z_i)}{n} - \frac{l(\mathcal{A}(S^{i,z_i'}), z_i') - l(\mathcal{A}(S), z_i')}{n} \\ &\leq 2\frac{L}{n}||\mathcal{A}(S) - \mathcal{A}(S_{i,z_i'})||_2 \end{aligned} \tag{11}$$

Assumption 14. *$\hat{R}_S(w)$ is cvx.*

Which gives us $f_S(w)$ is λ-str cvx. Now we perform a Taylor expansion:

$$f_S(\mathcal{A}(S_{i,z_i'})) - f_S(\mathcal{A}(S)) \geq \frac{\lambda}{2}||\mathcal{A}(S_{i,z_i'}) - \mathcal{A}(S)||_2^2 \tag{12}$$

Since $\mathcal{A}(S)$ is the minimizer of f_s and λ-str cvx the first term disappears.
From 11 and 12 we get:

$$||\mathcal{A}(S) - \mathcal{A}(S_{i,z_i'})|| \leq \frac{4L}{\lambda n} \tag{13}$$

If we perturb the data by a single element, we learn $\mathcal{A}$ that can become arbitrarily close for large n.
We then use 13 and the L-Lipschitz property of $l(\cdot, z)$:

$$\implies \sup_z [l(\mathcal{A}(S), z) - l(\mathcal{A}(S_{i,z_i'}), z)| \leq \frac{4L^2}{\lambda n}$$

□

Stochastic Gradient Descent (SGD) is Stable

Stability Theorem

Recall the SGD update formula,

$$w_{t+1} = w_t - \alpha_t \nabla_w l(w_t, z_{i,t}), i_t \sim \text{uniform}(1, \cdots, n) \tag{14}$$

where w_t is the weight iterate at time t, α_t is an (annealing) learning rate at time t and $l(w_t, z_{i,t})$ is the computed loss for the current weight iterate for a particular example $z_{i,t}$.

Theorem 15. *If $f(\cdot, z)$ is γ-smooth, convex and L-Lipschitz, then Stochastic Gradient Descent is β-uniformly stable where*

$$\beta \leq \frac{2L^2}{n} \sum_{t=1}^{T} \alpha_t$$

Analysis:
We are no longer requiring the function to be strongly convex. Additionally, this result holds for a finite number of steps T.

Stability Proof (Rough Outline)

We will consider two runs of the SGD algorithm. One run will be on the original data set S and the other run will be on the data set S_{i,z'_i}. Recall, this indicates the same data set S only now with the i^{th} element swapped with element z'_i. In order to compare the stability between the two runs, we maintain the same order of element selection (same random seed) for $t = 1, \cdots, T$.

Definition 16.

$$\delta_t = ||w_t - w'_t||$$

where w'_t denotes the iterate for the SGD algorithm on the data set S_{i,z'_i}.
We can write the expectation of the difference δ_{t+1} as the following:

$$E[\delta_{t+1}] = P(i_t = i)E[\delta_{t+1}|i_t = i] + P(i_t \neq i)E[\delta_{t+1}|i_t \neq i] \tag{15}$$

We introduce two Lemmas

$$E[\delta_{t+1}|i_t \neq i] \leq E[\delta_t]$$

Proof. Convexity and γ-smoothness implies that the gradients are co-coercive for a function f:

$$\langle \nabla f(v) - \nabla f(w), v - w \rangle \geq \frac{1}{\gamma} ||\nabla f(v) - \nabla f(w)||^2$$

We conclude that the weight update can be expressed as:

$$||w_{t+1} - w'_{t+1}||^2 = ||w_t - w'_t||^2 - 2\alpha_t \langle \nabla f(w_t) - \nabla f(w'_t), w_t - w'_t \rangle + \alpha^2 ||\nabla f(w_t) - \nabla f(w'_t)||^2$$
$$\leq ||w_t - w'_t||^2 - (2\alpha_t/\gamma - \alpha_t^2)||\nabla f(w_t) - \nabla f(w'_t)||^2 \leq ||w_t - w'_t||^2$$

so we get, using definition 7 that:

$$||w_{t+1} - w'_{t+1}|| = \delta_{t+1} \leq ||w_t - w'_t|| = \delta_t$$

□

And for the index that has been swapped

$$E[\delta_{t+1}|i_t = i] \leq E[\delta_t] + 2\alpha_t L$$

where L is the Lipschitz value.

Proof. We know that

$$\delta_{t+1} = ||w_{t+1} - w'_{t+1}|| = ||w_t - \alpha_t \nabla l(w_t, z_{i_t}) - (w'_t - \alpha_t \nabla l(w'_t, z_{i_t}))||$$

Using the triangle inequality we can write

$$\delta_{t+1} \leq ||w_t - w'_t|| + \alpha_t ||\nabla l(w_t, z_{i_t}) - \nabla l(w'_t, z_{i_t})||$$

Since $l(\cdot, z)$ is L-lipschitz

$$\delta_{t+1} \leq \delta_t + 2\alpha_t L$$

Taking expectation on either side we get

$$E[\delta_{t+1}|i_t = i] \leq E[\delta_t] + 2\alpha_t L$$

□

Using Lemmas 3, 3, we may rewrite Equation 15 as:

$$E[\delta_{t+1}] \leq \left(1 - \frac{1}{n}\right) E[\delta_t] + \frac{1}{n}\left(E[\delta_t] + 2\alpha_t L\right) \tag{16}$$

which when recursively unrolled yields the following final δ_T

$$E[\delta_T] = E[||w_T - w'_T||] \leq \sum_{t=0}^{T-1} \frac{2\alpha_t L}{n} \tag{17}$$

We find that:

$$\begin{aligned} E[\delta_{t+1}] &= P(i_t = i)E[\delta_{t+1}|i_t = i] + P(i_t \neq i)E[\delta_{t+1}|i_t \neq i]] \\ &\leq \frac{1}{n}(E[\delta_t] + 2\alpha_t L) + E[\delta_t](1 + \frac{1}{n}) \leq E[\delta_t] + \frac{2\alpha_t L}{n} \end{aligned}$$

SGD is therefore **stable** since $\sum_{t=0}^{T-1} \frac{2\alpha_t L}{n} \equiv \beta$ is $O(\frac{1}{n})$ for n data points.

References

[1] M. Mohri, A. Rostamizadeh, and A. Talwalkar. *Foundations of machine learning*. MIT press, 2012.