Machine Learning and Neural Networks (MATH3431) Epiphany term, 2025

Lecture notes 3: Learnability and stability in learning problems

Lecturer & author: Georgios P. Karagiannis georgios.karagiannis@durham.ac.uk

Aim. To introduce concepts PAC, fitting vs stability trade off, stability, and their implementation in regularization problems and convex problems.

Reading list & references:

- Shalev-Shwartz, S., & Ben-David, S. (2014). Understanding machine learning: From theory to algorithms. Cambridge university press.
  - Ch. 2, 3, 13
- Vapnik, V. (2000). The nature of statistical learning theory. Springer science & business media.

1. Learnable Probably Approximately Correct (PAC) Learning

Note 1. We formally define the broad learning problem we will work on.

Note 2. Learning algorithms $\mathfrak{A}$ use training data sets $\mathcal{S}$ which may miss characteristics of the unknown data-generating process g. Essentially, approximations (aka errors) are inevitable; some characteristics of data generating process g will be missed even if we use a very representative training set $\mathcal{S}$. We cannot hope that the learning algorithm will find a hypothesis whose error is smaller than the minimal possible error.

Note 3. The PAC learning problem requires no prior assumptions about the data-generating process g. It requires that the learning algorithm $\mathfrak{A}$ will find a predictor $\mathfrak{A}(\mathcal{S})$ whose error is not much larger than the best possible error of a predictor in some given benchmark hypothesis class. So essentially, in practice, the researcher's effort falls on the hypothesis class $\mathcal{H}$.

Definition 4. (Agnostic PAC Learnability for General Loss Functions) A hypothesis class $\mathcal{H}$ is agnostic PAC learnable with respect to a domain $\mathcal{Z}$ and a loss function $\ell : \mathcal{H} \times \mathcal{Z} \to \mathbb{R}_+$, if there exist a function $m_{\mathcal{H}} : (0,1)^2 \to \mathbb{N}$ and a learning algorithm $\mathfrak{A}$ with the following property:

- for every $\epsilon \in (0,1)$, $\delta \in (0,1)$, and distribution g over $\mathcal{Z}$, when running algorithm $\mathfrak{A}$ given training set $\mathcal{S}_m = \{z_1, ..., z_m\}$ with $z_i \overset{\text{IID}}{\sim} g$ for $m \geq m_{\mathcal{H}}(\epsilon, \delta)$ then $\mathfrak{A}$ returns

$\mathfrak{A}(\mathcal{S}) \in \mathcal{H}$ such as

$$\Pr_{\mathcal{S}\sim g}\left(R_g\left(\mathfrak{A}\left(\mathcal{S}_m\right)\right) - \min_{h'\in\mathcal{H}}\left(R_g\left(h'\right)\right) \leq \epsilon\right) \geq 1-\delta \tag{1.1}$$

Note 5. About 1.1: The accuracy parameter ϵ determines how far the output rule $\mathfrak{A}(\mathcal{S}_m)$ of $\mathfrak{A}$ can be from the optimal one (this corresponds to the 'approximately correct'). The confidence parameter δ indicates how likely the classifier is to meet that accuracy requirement (corresponds to the "probably" part of "PAC").

Note 6. It may be easier to work with expectations: by using Markov inequality (1.1) becomes

$$\Pr_{\mathcal{S}\sim g}\left(R_g\left(\mathfrak{A}\left(\mathcal{S}_m\right)\right) - \min_{h'\in\mathcal{H}}\left(R_g\left(h'\right)\right) \leq \epsilon\right) \geq 1-\frac{1}{\epsilon}\mathrm{E}_{\mathcal{S}\sim g}\left(R_g\left(\mathfrak{A}\left(\mathcal{S}_m\right)\right) - \min_{h'\in\mathcal{H}}\left(R_g\left(h'\right)\right)\right) \tag{1.2}$$

and hence can just compute expectations and bound them from above.

Hint: Markov inequality $\Pr(X \geq a) \leq \frac{1}{a}\mathrm{E}(X)$ for $X > 0$.

2. Analysis of the risk based on the trade-off fitting vs stability

Note 7. Let $R^* = \min_{\forall h}(R(h))$ be an ideal/optimal (hence minimum) Risk, and $\mathfrak{A}(\mathcal{S})$ the learning rule from a learning algorithm $\mathfrak{A}$ trained against dataset $\mathcal{S}$. The Risk of a learning algorithm $\mathfrak{A}$ can be decomposed as

$$R_g\left(\mathfrak{A}\left(\mathcal{S}\right)\right) - R^* = \hat{R}_{\mathcal{S}}\left(\mathfrak{A}\left(\mathcal{S}\right)\right) - R^* + \underbrace{R_g\left(\mathfrak{A}\left(\mathcal{S}\right)\right) - \hat{R}_{\mathcal{S}}\left(\mathfrak{A}\left(\mathcal{S}\right)\right)}_{\text{over-fitting}} \tag{2.1}$$

Note 8. Over-fitting can be represented by $R_g(\mathfrak{A}(\mathcal{S})) - \hat{R}_{\mathcal{S}}(\mathfrak{A}(\mathcal{S}))$. However, $\hat{R}_{\mathcal{S}}(\cdot)$ is a random variable and, here, for our computational convenience, we focus on its expectation w.r.t. $\mathcal{S} \sim g$. Hence, we provide the following (arguable) definition for over-fitting on which we base our analysis.

Definition 9. For a learning algorithm $\mathfrak{A}$, as a measure of over-fitting we consider the expected difference between true Risk and empirical Risk

$$\mathrm{E}_{\mathcal{S}\sim g}\left(R_g\left(\mathfrak{A}\left(\mathcal{S}\right)\right) - \hat{R}_{\mathcal{S}}\left(\mathfrak{A}\left(\mathcal{S}\right)\right)\right) \tag{2.2}$$

Definition 10. We say that learning algorithm $\mathfrak{A}$ suffers from over-fitting when (2.2) is 'too' large.

Note 11. The expected Risk of a learning algorithm $\mathfrak{A}(\mathcal{S})$ can be decomposed as

$$\mathrm{E}_{\mathcal{S}\sim g}\left(R_g\left(\mathfrak{A}\left(\mathcal{S}\right)\right)\right) = \underbrace{\mathrm{E}_{\mathcal{S}\sim g}\left(\hat{R}_{\mathcal{S}}\left(\mathfrak{A}\left(\mathcal{S}\right)\right)\right)}_{\text{(I)}} + \underbrace{\mathrm{E}_{\mathcal{S}\sim g}\left(R_g\left(\mathfrak{A}\left(\mathcal{S}\right)\right) - \hat{R}_{\mathcal{S}}\left(\mathfrak{A}\left(\mathcal{S}\right)\right)\right)}_{\text{(II)}} \tag{2.3}$$

by applying expectations in (2.1) and ignoring R^*. (I) indicates how well $\mathfrak{A}(\mathcal{S})$ fits the training set $\mathcal{S}$, and (II) indicates the discrepancy between the true and empirical risks of $\mathfrak{A}(\mathcal{S})$. In Section 3, we argue that the over-fitting term (II) is directly related to a certain type of stability of $\mathfrak{A}(\mathcal{S})$.

Note 12. The following result connects the need for upper bounding (and minimizing this bound) the Expected Risk in (2.3) and PAC learning (Definition 4).

Proposition 13. *Let $\mathfrak{A}$ be a learning algorithm that guarantees the following:*

- *If $m \geq m_{\mathcal{H}}(\epsilon)$ then for every distribution g, it is*

$$E_{\mathcal{S}\sim g}\left(R_g\left(\mathfrak{A}\left(\mathcal{S}_m\right)\right)\right) \leq \min_{h'\in\mathcal{H}}\left(R_g\left(h'\right)\right) + \epsilon$$

Then $\mathfrak{A}$ satisfies the PAC guarantee in Definition 4:

- *for every $\delta \in (0,1)$, if $m \geq m_{\mathcal{H}}(\epsilon\delta)$* then

$$\Pr_{\mathcal{S}\sim g}\left(R_g\left(\mathfrak{A}\left(\mathcal{S}_m\right)\right) - \min_{h'\in\mathcal{H}}\left(R_g\left(h'\right)\right) \leq \epsilon\right) \geq 1-\delta$$

Proof. Let $\xi = R_g\left(\mathfrak{A}\left(\mathcal{S}_m\right)\right) - \min_{h'\in\mathcal{H}}\left(R_g\left(h'\right)\right)$. From Markov's inequality $\Pr\left(\xi \geq \mathrm{E}\left(\xi\right)/\delta\right) \leq \frac{1}{\mathrm{E}(\xi)/\delta}\mathrm{E}\left(\xi\right) = \delta$. Namely, $\Pr\left(\xi \leq \mathrm{E}\left(\xi\right)/\delta\right) = 1-\delta$. But it is given that if $m \geq m_{\mathcal{H}}\left((\epsilon\delta)\right)$ for every distribution g it is $\mathrm{E}\left(\xi\right) \leq (\epsilon\delta)$. So by substitution $\Pr\left(\xi \leq (\epsilon\delta)/\delta\right) = 1-\delta$ implies $\Pr\left(\xi \leq \epsilon\right) = 1-\delta$. Now substitute back ξ and we conclude the proof. □

Note 14. Hence, we aim to design a learning algorithm $\mathfrak{A}(\mathcal{S})$ that both fits the training set and is stable; i.e, keep both (I) and (II) in (2.3) small. As seen later, in certain learning problems, there may be a trade-off between empirical risk term (I) and (II) in (2.3). Consequently, we aim to upper bound (2.3) by upper bounding (I) and (II) individually and decide which term we should benefit against the other to achieve a certain total bound.

3. Stability and its association with over-fitting

Notation 15. Consider a learning problem $(\mathcal{H}, \mathcal{Z}, \ell)$. Let $\mathcal{S} = \{z_1, ..., z_m\}$ be a training sample, and $\mathfrak{A}$ be a learning algorithm with output $\mathfrak{A}(\mathcal{S})$.

Note 16. It is reasonable to consider that a learning algorithm $\mathfrak{A}$ can be stable if a small change of the algorithm input does not change the algorithm output much. Mathematically formalizing this, we can say that if $\mathcal{S}^{(i)} = \{z_1, ..., z_{i-1}, z', z_{i+1}, ..., z_m\}$ is another training dataset equal to $\mathcal{S}$ but the ith element being replaced by another independent example $z' \sim g$, then a good learning algorithm $\mathfrak{A}$ would produce a small value of

$$\ell\left(\mathfrak{A}\left(\mathcal{S}^{(i)}\right), z_i\right) - \ell\left(\mathfrak{A}\left(\mathcal{S}\right), z_i\right) \geq 0$$

Definition 17. A learning algorithm $\mathfrak{A}$ is **on-average-replace-one-stable** with rate a decreasing function $\epsilon : \mathbb{N} \to \mathbb{R}$ if for every distribution g

$$\mathrm{E}_{\substack{\mathcal{S} \sim g \\ z' \sim g, i \sim \mathrm{U}\{1,...,m\}}} \left(\ell\left(\mathfrak{A}\left(\mathcal{S}^{(i)}\right), z_i\right) - \ell\left(\mathfrak{A}\left(\mathcal{S}\right), z_i\right)\right) \leq \epsilon\left(m\right) \tag{3.1}$$

Note 18. The following result demonstrates the direct association of stability and over-fitting as defined in Definitions 10 & 17.

Theorem 19. *For any learning algorithm* $\mathfrak{A}$ *it is*

$$E_{\mathcal{S}\sim g}\left(R_g\left(\mathfrak{A}\left(\mathcal{S}\right)\right) - \hat{R}_{\mathcal{S}}\left(\mathfrak{A}\left(\mathcal{S}\right)\right)\right) = E_{\substack{\mathcal{S} \sim g,\, z' \sim g \\ i \sim U\{1,...,m\}}} \left(\ell\left(\mathfrak{A}\left(\mathcal{S}^{(i)}\right), z_i\right) - \ell\left(\mathfrak{A}\left(\mathcal{S}\right), z_i\right)\right) \tag{3.2}$$

where g *is a distribution,* $\mathcal{S} = \{z_1, ..., z_m\}$ *and* $\mathcal{S}^{(i)} = \{z_1, ..., z_{i-1}, z', z_{i+1}, ..., z_m\}$ are *training datasets with* $z', z_1, ..., z_m \overset{\text{iid}}{\sim} g$.

Proof. As $z', z_1, ..., z_m \overset{\text{iid}}{\sim} g$, then for every i

$$\mathrm{E}_{\mathcal{S}\sim g}\left(R_g\left(\mathfrak{A}\left(\mathcal{S}\right)\right)\right) = \mathrm{E}_{\substack{\mathcal{S} \sim g \\ z' \sim g}} \left(\ell\left(\mathfrak{A}\left(\mathcal{S}\right), z'\right)\right) = \mathrm{E}_{\substack{\mathcal{S} \sim g \\ z' \sim g}} \left(\ell\left(\mathfrak{A}\left(\mathcal{S}^{(i)}\right), z_i\right)\right)$$

and

$$\mathrm{E}_{\mathcal{S}\sim g}\left(\hat{R}_{\mathcal{S}}\left(\mathfrak{A}\left(\mathcal{S}\right)\right)\right) = \mathrm{E}_{\substack{\mathcal{S} \sim g \\ i \sim U\{1,...,m\}}} \left(\ell\left(\mathfrak{A}\left(\mathcal{S}^{(i)}\right), z_i\right)\right)$$

□

Note 20. Hence, we desire to design learning algorithms corresponding to as small as possible (3.2).

4. Implementation in regularized loss learning problems

Definition 21. A Regularized Loss Minimization (RLM) learning problem $(\mathcal{H}, \mathcal{Z}, \ell)$ with a regularization function $J : \mathcal{H} \to \mathbb{R}$ aims at a RLM learning rule that is the minimizer

$$h^* = \arg\min_{h \in \mathcal{H}} \left(\hat{R}_{\mathcal{S}}\left(h\right) + J\left(h\right)\right) \tag{4.1}$$

where $\hat{R}_{\mathcal{S}}\left(\cdot\right) = \frac{1}{m}\sum_{i=1}^{m} \ell\left(\cdot, z_i\right)$, and $\mathcal{S} = \{z_1, ..., z_m\} \subset$ an IID training sample.

Remark 22. The motivation for considering the regularization function J in (4.1) is to: (1.) control complexity and (2.) improve stability; as we will see later.

Note 23. Here, we make our example more specific and narrow it to the Ridge RLM learning problem (could have been LASSO etc.).

Definition 24. The Ridge RLM learning problem $(\mathcal{H}, \mathcal{Z}, \ell)$, with $\mathcal{H} = \mathcal{W} \subset \mathbb{R}^d$, uses regularization function $J(w; \lambda) = \lambda \|w\|_2^2$ with $\lambda > 0$, $w \in \mathcal{W}$ and produces learning rule

$$\mathfrak{A}(\mathcal{S}) = \arg\min_{w \in \mathcal{W}} \left(\hat{R}_{\mathcal{S}}(w) + \lambda \|w\|_2^2 \right) \tag{4.2}$$

Note 25. Recall (Term 1) how the regularization function in Ridge RLM learning problem $(\mathcal{H}, \mathcal{Z}, \ell)$ penalizes complexity. Essentially, it implies a sequence of hypothesis $\mathcal{H}_1 \subset \mathcal{H}_2 \subset ...$ with $\mathcal{H}_i = \left\{ w \in \mathbb{R}^d : \|w\|_2 < i \right\}$ due to duality of the corresponding minimization problem.

Note 26. Below, we will try to analyze the behavior of Ridge RLM learning rule (4.2) w.r.t. the Risk decomposition (2.3). In particular, to upper bounded w.r.t. the shrinkage term λ, training sample size m, and other characteristics.

4.1. **Bounding the empirical risk (I) in (2.3).**

Note 27. From (4.2), we have

$$\begin{aligned} \hat{R}_{\mathcal{S}}(\mathfrak{A}(\mathcal{S})) &\leq \hat{R}_{\mathcal{S}}(\mathfrak{A}(\mathcal{S})) + \lambda \|\mathfrak{A}(\mathcal{S})\|_2^2 \\ &\leq \hat{R}_{\mathcal{S}}(w) + \lambda \|w\|_2^2 ; \quad \forall w \in \mathcal{W} \end{aligned}$$

and by taking expectations w.r.t. $\mathcal{S}$, it is

$$\mathrm{E}_{\mathcal{S} \sim g} \left(\hat{R}_{\mathcal{S}}(\mathfrak{A}(\mathcal{S})) \right) \leq R_g(w) + \lambda \|w\|_2^2 ; \quad \forall w \in \mathcal{W} \tag{4.3}$$

because $\mathrm{E}_{\mathcal{S} \sim g} \left(\hat{R}_{\mathcal{S}}(\cdot) \right) = \frac{1}{m} \sum_{i=1}^{m} \mathrm{E}_{\mathcal{S} \sim g} (\ell(\cdot, z_i)) = R_g(\cdot)$.

Note 28. From (4.3), we observe that part (I) in the expected risk decomposition (2.3), (aka the upper bound of the expected empirical risk) increases with the regularization term $\lambda > 0$. (!!!) –Although anticipated, it did not start well.

4.2. **Bounding the empirical risk (II) in (2.3).**

Note 29. As we will see to bound (II) in (2.3) we will have to impose an additional condition on the behavior of loss function apart from convexity; e.g. Lipstchitzness.

Assumption 30. *The loss function $\ell(\cdot, z)$ in (4.2) is convex for any $z \in \mathcal{Z}$.*

Note 31. Let $\tilde{R}_{\mathcal{S}}(w) = \hat{R}_{\mathcal{S}}(w) + \lambda \|w\|_2^2$. $\tilde{R}_{\mathcal{S}}(\cdot)$ is 2λ-strongly convex as the sum of a convex function $\hat{R}_{\mathcal{S}}(\cdot)$ (Assumption 30) and a 2λ-strongly convex function $J(\cdot; \lambda) = \lambda \|\cdot\|_2^2$ (results directly from Definition 14 in Lecture notes 2).

Fact 32. *(To be used in Note 33) If f is* λ-strongly convex and u is a minimizer of f then for any w

$$f(w) - f(u) \geq \frac{\lambda}{2} \|w - u\|^2$$

Note 33. Let $\mathfrak{A}(\mathcal{S})$ be the Ridge learning algorithm output minimizing (4.2). Because $\tilde{R}_{\mathcal{S}}(\cdot)$ is 2λ-strongly convex and $\mathfrak{A}(\mathcal{S})$ is its minimizer, according to Fact 32, for $\mathfrak{A}(\mathcal{S})$ and any $w \in \mathcal{W}$, it is

$$\tilde{R}_{\mathcal{S}}(w) - \tilde{R}_{\mathcal{S}}(\mathfrak{A}(\mathcal{S})) \geq \lambda \|w - \mathfrak{A}(\mathcal{S})\|^2, \quad \forall w \in \mathcal{W} \tag{4.4}$$

Note 34. Also, for any $w, u \in \mathcal{W}$, it is

$$\begin{aligned}
\tilde{R}_{\mathcal{S}}(w) - \tilde{R}_{\mathcal{S}}(u) &= \left(\hat{R}_{\mathcal{S}}(w) + \lambda \|w\|_2^2\right) - \left(\hat{R}_{\mathcal{S}}(u) + \lambda \|u\|_2^2\right) \\
&= \left(\hat{R}_{\mathcal{S}^{(i)}}(w) + \lambda \|w\|_2^2\right) - \left(\hat{R}_{\mathcal{S}^{(i)}}(u) + \lambda \|u\|_2^2\right) \\
&\quad + \frac{\ell(w, z_i) - \ell(u, z_i)}{m} + \frac{\ell(u, z') - \ell(w, z')}{m} \\
&= \tilde{R}_{\mathcal{S}^{(i)}}(w) - \tilde{R}_{\mathcal{S}^{(i)}}(u) + \frac{\ell(w, z_i) - \ell(u, z_i)}{m} + \frac{\ell(u, z') - \ell(w, z')}{m}
\end{aligned}$$

Choosing $w = \mathfrak{A}\left(\mathcal{S}^{(i)}\right)$ and $u = \mathfrak{A}(\mathcal{S})$, and the fact that $\tilde{R}_{\mathcal{S}^{(i)}}\left(\mathfrak{A}\left(\mathcal{S}^{(i)}\right)\right) \leq \tilde{R}_{\mathcal{S}^{(i)}}(\mathfrak{A}(\mathcal{S}))$, it is

(4.5)

$$\tilde{R}_{\mathcal{S}}\left(\mathfrak{A}\left(\mathcal{S}^{(i)}\right)\right) - \tilde{R}_{\mathcal{S}}(\mathfrak{A}(\mathcal{S})) \leq \frac{\ell\left(\mathfrak{A}\left(\mathcal{S}^{(i)}\right), z_i\right) - \ell(\mathfrak{A}(\mathcal{S}), z_i)}{m} + \frac{\ell(\mathfrak{A}(\mathcal{S}), z') - \ell\left(\mathfrak{A}\left(\mathcal{S}^{(i)}\right), z'\right)}{m}$$

Note 35. Then (4.4) and (4.5) imply

(4.6)

$$\lambda \left\|\mathfrak{A}\left(\mathcal{S}^{(i)}\right) - \mathfrak{A}(\mathcal{S})\right\| \leq \frac{\ell\left(\mathfrak{A}\left(\mathcal{S}^{(i)}\right), z_i\right) - \ell(\mathfrak{A}(\mathcal{S}), z_i)}{m} + \frac{\ell(\mathfrak{A}(\mathcal{S}), z') - \ell\left(\mathfrak{A}\left(\mathcal{S}^{(i)}\right), z'\right)}{m}$$

Note 36. Now that we brought it in form (4.6), we can impose/use an additional assumption on the loss to bound it. In this example we use Lipstchitzness.

Assumption 37. *The loss function $\ell(\cdot, z)$ in (4.2) is convex and ρ-Lipschitz for any $z \in \mathcal{Z}$.*

Note 38. Given ρ-Lipschitzness in Assumption 37, it is

$$\ell\left(\mathfrak{A}\left(\mathcal{S}^{(i)}\right), z_i\right) - \ell(\mathfrak{A}(\mathcal{S}), z_i) \leq \rho \left\|\mathfrak{A}\left(\mathcal{S}^{(i)}\right) - \mathfrak{A}(\mathcal{S})\right\| \tag{4.7}$$

$$\ell(\mathfrak{A}(\mathcal{S}), z') - \ell\left(\mathcal{S}^{(i)}, z'\right) \leq \rho \left\|\mathfrak{A}\left(\mathcal{S}^{(i)}\right) - \mathfrak{A}(\mathcal{S})\right\| \tag{4.8}$$

and hence plugging 4.7 and (4.8) in (4.6) yields

$$\left\|\mathfrak{A}\left(\mathcal{S}^{(i)}\right) - \mathfrak{A}(\mathcal{S})\right\| \leq 2\frac{\rho}{\lambda m} \tag{4.9}$$

Note 39. Plugging (4.9) in (4.7) yields

$$\ell\left(\mathfrak{A}\left(\mathcal{S}^{(i)}\right), z_i\right) - \ell\left(\mathfrak{A}\left(\mathcal{S}\right), z_i\right) \leq 2\frac{\rho^2}{\lambda m}$$

Note 40. Using Theorem 19, we get an upper bound for the stability / over-fitting

$$\mathrm{E}_{\mathcal{S}\sim g}\left(R_g\left(\mathfrak{A}\left(\mathcal{S}\right)\right) - \hat{R}_{\mathcal{D}}\left(\mathfrak{A}\left(\mathcal{S}\right)\right)\right) = \mathrm{E}_{\substack{\mathcal{S}\sim g,\, z'\sim g \\ i\sim \mathrm{U}\{1,...,m\}}}\left(\ell\left(\mathfrak{A}\left(\mathcal{S}^{(i)}\right), z_i\right) - \ell\left(\mathfrak{A}\left(\mathcal{S}\right), z_i\right)\right) \leq 2\frac{\rho^2}{\lambda m}$$

Note 41. After this saga, the researcher could come to the conclusion that: A Ridge RLM learning problem $(\mathcal{H}, \mathcal{Z}, \ell)$ with the loss function which is convex and ρ-Lipschitz, regularizer $J(\cdot;\lambda) = \lambda\|\cdot\|^2$, $\lambda > 0$, and learning rule trained against an iid sample $\mathcal{S} = \{z_i\}_{i=1}^m$ from g is on-averave-replace-one-stable with rate $\epsilon(m) = 2\frac{\rho^2}{\lambda m}$; i.e.

$$\mathrm{E}_{\mathcal{S}\sim g}\left(R_g\left(\mathfrak{A}\left(\mathcal{S}\right)\right) - \hat{R}_{\mathcal{S}}\left(\mathfrak{A}\left(\mathcal{S}\right)\right)\right) \leq 2\frac{\rho^2}{\lambda m} \tag{4.10}$$

Note 42. From (4.10), we see that stability improves (and over-fitting decreases) as the shrinkage parameter λ increases.

4.3. **Bounding the Risk (2.3).**

Note 43. Given the bounds (4.3) and (4.10), the decomposition of the expected Risk in (2.3) yields that: A Ridge RLM learning problem $(\mathcal{H}, \mathcal{Z}, \ell)$ with the loss function which is convex and ρ-Lipschitz, regularizer $J(\cdot;\lambda) = \lambda\|\cdot\|^2$, $\lambda > 0$, and learning rule trained against an iid sample $\mathcal{S} = \{z_i\}_{i=1}^m$ from g has Expected Risk bound

$$\mathrm{E}_{\mathcal{S}\sim g}\left(R_g\left(\mathfrak{A}\left(\mathcal{S}\right)\right)\right) \leq \underbrace{R_g\left(w\right) + \lambda\|w\|_2^2}_{\text{(I)}} + \underbrace{2\frac{\rho^2}{\lambda m}}_{\text{(II)}}; \quad \forall w \in \mathcal{W} \tag{4.11}$$

Note 44. From (4.11), we see that there is a trade-off between Empirical Risk (I) and stability/overfitting (II) with regards the regularization parameter λ. It is desirable to use the optimal $\lambda > 0$ corresponding to the smallest bound in (4.11); it has to both fit the training data well (but perhaps not too well) and be very stable to different training data from the same g (but perhaps not too stable)!

Assumption 45. *Assume that the learning problem is additionally B-bounded by $B > 0$; i.e. $\mathcal{H} = \left\{w \in \mathbb{R}^d : \|w\|_2 \leq B\right\}$.*

Note 46. If additionally the learning problem is B-bounded then the upper bound in (4.11) becomes

$$\mathrm{E}_{\mathcal{S}\sim g}\left(R_g\left(\mathfrak{A}\left(\mathcal{S}\right)\right)\right) \leq \underbrace{\min_{w\in\mathcal{H}} R_g\left(w\right) + \lambda B^2}_{\text{(I)}} + \underbrace{2\frac{\rho^2}{\lambda m}}_{\text{(II)}} \tag{4.12}$$

Note 47. Furthermore, if we choose a shrinkage parameter $\lambda_m = \sqrt{\frac{2}{m}}\frac{\rho}{B}$ in (4.2) the upper bound in (4.11) becomes

$$\mathrm{E}_{\mathcal{S}\sim g}\left(R_g\left(\mathfrak{A}\left(\mathcal{S}\right)\right)\right) \leq \min_{w\in\mathcal{H}} R_g\left(w\right) + \lambda_m B\sqrt{\frac{8}{m}} \tag{4.13}$$

4.4. **Deriving PAC guarantees.**

Note 48. In particular, if the size m of the training sample $\mathcal{S}$ is

$$m \geq \frac{\lambda_m^2 B^2 8}{\epsilon^2} = \frac{8\rho^2 B^2}{\epsilon^2}$$

for some $\epsilon > 0$ then for every distribution g, the upper bound in (4.11) becomes

$$\mathrm{E}_{\mathcal{S}\sim g}\left(R_g\left(\mathfrak{A}\left(\mathcal{S}\right)\right)\right) \leq \min_{w\in\mathcal{H}} R_g\left(w\right) + \epsilon \tag{4.14}$$

and hence we can have a PAC guarantee that is summarized in the following.

Summary 49. Let $(\mathcal{H}, \mathcal{Z}, \ell)$ be a learning problem convex-Lipschitz-bounded with parameters ρ and B, and let m be the training data size. The associated Ridge Regularized Loss Minimization rule with regularization function $\lambda_m = \sqrt{\frac{2}{m}}\frac{\rho}{B}$ satisfies

$$\mathrm{E}_{\mathcal{S}\sim g}\left(R_g\left(\mathfrak{A}\left(\mathcal{S}\right)\right)\right) \leq \min_{w\in\mathcal{H}} R_g\left(w\right) + \lambda_m B\sqrt{\frac{8}{m}}$$

Moreover, if $m \geq \frac{8\rho^2 B^2}{\epsilon^2}$, $\epsilon > 0$ then for every distribution g we can get a PAC -guarantee

$$\mathrm{E}_{\mathcal{S}\sim g}\left(R_g\left(\mathfrak{A}\left(\mathcal{S}\right)\right)\right) \leq \min_{w\in\mathcal{H}} R_g\left(w\right) + \epsilon$$