CPSC 532D — 13. IMPLICIT REGULARIZATION AND MARGINS

Danica J. Sutherland

University of British Columbia, Vancouver

Fall 2025

For deep learning, as we've seen, the fact that something is an ERM may not be enough to see that it generalizes: there are "good" ERMs that generalize, but also bad ones that don't, for the same architecture and even for reasonably similar parameter norms/etc. So, we'll need to specify something more than just that it's an ERM.

One way to study this is to ask *which* solution gradient descent will converge to, then try to prove properties of that. The idea that our optimization algorithm or other such "implementation details" can actually choose for us which of the "equally valid solutions" we end up with is called the *implicit regularization* of the algorithm: we don't explicitly write down a regularizer, but the choice of algorithm has a similar effect.

It's also sometimes called the implicit bias *of the algorithm, in the sense that the algorithm has a certain inductive bias towards certain kinds of solutions. That can sometimes cause confusion with the concept of the same name from social science, though, and just generally kind of imply that it's "bad" when actually often the presence of this implicit regularization is "good."*

13.1 GRADIENT DESCENT FOR LINEAR REGRESSION

Let's start with the function

$$f(w) = \mathrm{L}_{\mathrm{S}}^{\mathrm{sq}}(x \mapsto w \cdot x) = \frac{1}{m}\|\mathrm{X}w - y\|^2,$$

where $\mathrm{X} \in \mathbb{R}^{m \times d}$ is the matrix stacking up S_x and $y \in \mathbb{R}^m$ is the vector form of S_y. We have

$$\nabla f(w) = \frac{2}{m}\mathrm{X}^\mathsf{T}(\mathrm{X}w - y),$$

which notice is $\frac{2}{m}\|\mathrm{X}^\mathsf{T}\mathrm{X}\|$-smooth, so f is convex and β-smooth, thus small-learning-rate gradient descent finds a global optimum (Theorem 11.5). In the traditional $m > d$ case when X is full-rank, there's a unique solution to this problem, typically with $\mathrm{X}w \neq y$ but always having $\mathrm{X}^\mathsf{T}(\mathrm{X}w - y) = 0$. In high-dimensional settings $d > m$, though, it's possible to achieve $\mathrm{X}w = y$ (interpolation) in infinitely many ways. Which one does gradient descent find?

There's a more explicit (but longer) analysis for least squares, which gives some more details without relying on any general gradient descent analyses, in the 2023 notes.

Proposition 13.1. *Let $\mathrm{X} \in \mathbb{R}^{m \times d}$ be of rank m (implying $d \geq m$), and $y \in \mathbb{R}^m$. Suppose that $l(h, (x, y)) = l_y(h(x))$ for a differentiable function l_y such that $l_y(\hat{y}) \to 0$ implies $\hat{y} \to y$.*

Consider any iterative optimization method which begins at a point w_0 and then has updates of the form $w_{t+1} - w_t \in \operatorname{span}\{\nabla \mathrm{L}_{\mathrm{S}}(x \mapsto w_k \cdot x) : 0 \leq k \leq t\}$. If this method converges to a global minimizer w_∞ of $\mathrm{L}_{\mathrm{S}}(x \mapsto w \cdot x)$, then

$$w_\infty = \mathrm{X}^\mathsf{T}(\mathrm{X}\mathrm{X}^\mathsf{T})^{-1}y + (\mathrm{I} - \mathrm{X}^\mathsf{T}(\mathrm{X}\mathrm{X}^\mathsf{T})^{-1}\mathrm{X})w_0 = \operatorname*{arg\,min}_{w : \mathrm{X}w = y} \|w - w_0\|.$$

Proof. This was Assignment 1, Question 3. □

For more, visit https://cs.ubc.ca/~dsuth/532D/25w1/.

13.2 SEPARABLE LOGISTIC REGRESSION

There's another major class of loss functions not satisfying the requirement of Proposition 13.1: for instance, with logistic loss $l_y(\hat{y}) = \log(1 + \exp(-y\hat{y}))$, $l_y(\hat{y}) \to 0$ implies $\hat{y} \to y\infty$, not y.

So, let's consider logistic regression in particular: for $y_i \in \{-1, 1\}$,

$$f(w) = \frac{1}{m} \sum_{i=1}^{m} \log(1 + \exp(-y_i \langle x_i, w \rangle)).$$

We're also going to assume that the data is *linearly separable*: there is some w^* such that $y_i \langle x_i, w^* \rangle > 0$ for all i. Then, it's possible to drive $f(w)$ arbitrarily close to zero, but never to actually reach it: we only get $\log(1 + \exp(-t)) \to 0$ for $t \to \infty$, so we need $\|w\| \to \infty$. A solution of the form cw^* for $c \to \infty$ would work, but potentially so would many other solutions, since there are probably many possible perfect linear separators on this dataset. Which one does gradient descent find?

We're going to approach this informally, for time and simplicity. Soudry et al. [Sou+18] and Gunasekar et al. [GLSS18] handle it in full, and Ji and Telgarsky [JT19] approach the non-separable case; Bach [Bach25, Section 11.1.2] gives an overview including a few things we aren't covering here.

Notice that

$$\nabla f(w) = -\frac{1}{m} \sum_{i=1}^{m} \frac{\exp(-y_i \langle x_i, w \rangle)}{1 + \exp(-y_i \langle x_i, w \rangle)} y_i x_i.$$

We know that we'll get $\|w_t\| \to \infty$ from the argument above; it's reasonable to expect, then, that we'll have $\frac{w_t}{\|w_t\|} \to v$ for some $\|v\| = 1$, and $y_i \langle x_i, v \rangle > 0$ for all i since otherwise we wouldn't approach a minimizer. This gives us, roughly speaking,

$$\nabla f(\|w_t\| v) \sim -\frac{1}{m} \sum_{i=1}^{m} \frac{\exp(-y_i \|w_t\| \langle x_i, v \rangle)}{1 + \exp(-y_i \|w_t\| \langle x_i, v \rangle)} y_i x_i \sim -\frac{1}{m} \sum_{i=1}^{m} \exp(-y_i \|w_t\| \langle x_i, v \rangle) y_i x_i,$$

since $\frac{t}{1+t} = t + \mathcal{O}(t^2)$ and we'll eventually have $\exp(-y_i \|w_t\| \langle x_i, v \rangle) \ll 1$.

So, eventually each gradient term gets small. Which ones are bigger than the others? The asymptotic ratio between the size of the gradient contributions from x_i and x_j is

$$\frac{\exp(-y_i \|w_t\| \langle x_i, v \rangle) |y_i| \|x_i\|}{\exp(-y_j \|w_t\| \langle x_j, v \rangle) |y_j| \|x_j\|} = \frac{\|x_i\|}{\|x_j\|} \exp\left(-\|w_t\| (y_i \langle x_i, v \rangle - y_j \langle x_j, v \rangle)\right).$$

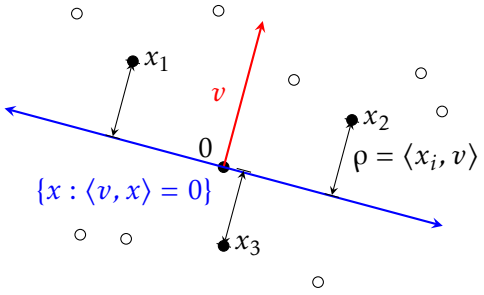
As $\|w_t\| \to \infty$, this ratio goes to 0 if $y_i \langle x_i, v \rangle > y_j \langle x_j, v \rangle$, or ∞ if the order is reversed; it is $\|x_i\| / \|x_j\| \in (0, \infty)$ if and only if $y_i \langle x_i, v \rangle = y_j \langle x_j, v \rangle$. So, for whatever v we have, let $\mathcal{I}_v$ be the set of indices such that $y_i \langle x_i, v \rangle$ is minimized. Only these terms really matter:

$$\nabla f(\|w_t\| v) \sim -\frac{1}{m} \sum_{i \in \mathcal{I}_v} \exp(-y_i \|w_t\| \langle x_i, v \rangle) y_i x_i.$$

So, if gradient descent diverges in a direction v, the dominant direction in which w_t moves is a (positive) linear combination of the points $\{x_i : i \in \mathcal{I}_v\}$. Let's define $\rho = \min_i y_i \langle x_i, v \rangle$; then, summarizing,

$$v = \sum_{i=1}^{m} \alpha_i y_i x_i \text{ with } \forall i,\ (\alpha_i \geq 0 \text{ and } y_i \langle x_i, v \rangle = \rho) \text{ or } (\alpha_i = 0 \text{ and } y_i \langle x_i, v \rangle > \rho). \quad (13.1)$$

In fact, ρ is a quantity known as the *geometric margin* of the linear separator v; it is exactly the smallest distance from any of the x_i to the hyperplane $\{x : v^\top x = 0\}$, the decision boundary of the linear classifier with unit-norm weights v. (Claim 15.1 of [SSBD14] proves this, if you're skeptical.)



13.2.1 *Margin maximization*

The equations (13.1) turn out to be equivalent to the KKT conditions of the problem of finding the *max-margin separator*, also known as a hard support vector machine (SVM). This problem is given by

$$\arg\max_{v:\|v\|=1} \min_{i\in[m]} y_i\langle x_i, v\rangle \quad \text{s.t. } \forall i \in [m],\ y_i\langle x_i, v\rangle > 0$$

Change so that $v = w/\|w\|$ for any w:

$$= \arg\max_{w\in\mathbb{R}^d} \min_{i\in[m]} \frac{y_i\langle x_i, w\rangle}{\|w\|} \quad \text{s.t. } \forall i \in [m],\ y_i \frac{\langle x_i, w\rangle}{\|w\|} > 0$$

$$= \arg\max_{w\in\mathbb{R}^d} \frac{1}{\|w\|} \min_{i\in[m]} y_i\langle x_i, w\rangle \quad \text{s.t. } \forall i \in [m],\ y_i\langle x_i, w\rangle > 0$$

The objective is the same for any $w' = cw$ for $c > 0$, so we might as well limit ourselves to solutions where $\min_i y_i\langle x_i, w\rangle = 1$:

$$\supseteq \arg\max_{w\in\mathbb{R}^d} \frac{1}{\|w\|} \quad \text{s.t. } \forall i \in [m],\ y_i\langle x_i, w\rangle \geq 1$$

$$= \arg\min_{w\in\mathbb{R}^d} \frac{1}{2}\|w\|^2 \quad \text{s.t. } \forall i \in [m],\ y_i\langle x_i, w\rangle \geq 1. \tag{13.2}$$

Proposition 13.2. *Let $x_1, \ldots, x_m \in \mathcal{X}$ for a real Hilbert space $\mathcal{X}$. Then*

$$\arg\min_{w\in\mathbb{R}^d} \frac{1}{2}\|w\|^2 \quad \textit{s.t. } \forall i \in [m],\ y_i\langle x_i, w\rangle \geq 1$$

$$= \left\{ w = \sum_i \alpha_i y_i x_i : \forall i \in [m],\ (\alpha_i \geq 0 \textit{ and } y_i\langle x_i, w\rangle = 1) \textit{ or } (\alpha_i = 0 \textit{ and } y_i\langle x_i, w\rangle > 1) \right\}.$$

Proof. This is a direct application of the KKT conditions; the problem is convex and strictly feasible, hence the conditions are both necessary and sufficient.

Alternatively, we also give a direct argument without appealing to the KKT conditions. We first show that the given conditions ensure margin maximization.

Let $w = \sum_i \alpha_i y_i x_i$ such that for each $i \in [m]$, $y_i\langle x_i, w\rangle \geq 1$ and either $\alpha_i = 0$ and/or

$y_i\langle x_i, w\rangle = 1$. Let v be any vector such that for all i, $y_i\langle x_i, v\rangle \geq 1$. Then

$$\langle w, v\rangle = \left\langle \sum_i \alpha_i y_i x_i, v \right\rangle = \sum_i \alpha_i y_i \langle x_i, v\rangle \geq \sum_i \alpha_i,$$

while

$$\langle w, w\rangle = \sum_i \alpha_i y_i \langle x_i, w\rangle = \sum_i \begin{cases} 0 & \text{if } \alpha_i = 0 \\ \alpha_i & \text{if } y_i\langle x_i, w\rangle = 1 \end{cases} = \sum_i \alpha_i.$$

Thus we have that for any feasible v,

$$\|w\|\|v\| \geq \langle w, v\rangle \geq \langle w, w\rangle = \|w\|^2,$$

and hence since the problem is trivial when $w = 0$, $\|v\| \geq \|w\|$. Therefore w has minimal norm among the feasible solutions, and hence solves (13.2).

The other direction is shown in the following section. □

Aside: Margin maximization ensures given conditions

This direction is less directly relevant for us, but is here for completeness.

First, the solution w to (13.2) is unique: the objective is strictly convex, the intersection of a bunch of affine constraints is always convex, and we assumed that the constraints are feasible so it's not an empty set.

Suppose that $w + t\delta$ is feasible for small $t > 0$ and $\|\delta\| = 1$; since w is a strict minimizer, we know

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{1}{2}\|w + t\delta\|^2\right)\Big|_{t=0} = (\langle w, \delta\rangle + t)\Big|_{t=0} = \langle w, \delta\rangle$$

must be positive.

When is $w + t\delta$ feasible? This happens exactly when

$$y_i\langle x_i, w + t\delta\rangle = y_i\langle x_i, w\rangle + t y_i \langle x_i, \delta\rangle \geq 1.$$

Let $\mathcal{I} = \mathcal{I}_{w/\|w\|} = \{i \in [m] : y_i\langle x_i, w\rangle = 1\}$. If $i \in \mathcal{I}$, feasibility with any $t > 0$ requires $y_i\langle x_i, \delta\rangle \geq 0$. For $i \notin \mathcal{I}$, since w is feasible we have $y_i\langle x_i, w\rangle > 1$; we can thus always move some small amount t in the direction δ.

So, we know that the solution w must satisfy that $\langle w, \delta\rangle > 0$ for all δ with $y_i\langle x_i, \delta\rangle \geq 0$, $i \in \mathcal{I}$. It's a geometric fact that the only w for which this is possible are the $\sum_{i\in\mathcal{I}} \alpha_i y_i x_i$, shown in the following lemma; the proof is then complete.

K *is the "polar cone" of* C.

Lemma 13.3. *Let $x_1, \ldots, x_m \in \mathcal{X}$ for a finite m and real Hilbert space $\mathcal{X}$. Define* $\mathrm{C} = \{w : \forall i \in [m],\ \langle x_i, w\rangle \geq 0\}$. *The set* $\mathrm{K} = \{w : \langle w, x\rangle \geq 0 : w \in \mathrm{C}\}$ *can be written* $\mathrm{K} = \{\sum_i \alpha_i x_i : \alpha_i \geq 0\}$.

Proof. Let $\mathrm{L} = \{\sum_i \alpha_i x_i : \alpha_i \geq 0\}$; we will show $\mathrm{L} = \mathrm{K}$.

First, $\mathrm{L} \subseteq \mathrm{K}$: for any $w \in \mathrm{C}$ and $\alpha_i \geq 0$, we have that

$$\left\langle \sum_i \alpha_i x_i, w \right\rangle = \sum_i \underbrace{\alpha_i}_{\geq 0} \underbrace{\langle x_i, w\rangle}_{\geq 0} \geq 0.$$

It remains to show $K \subseteq L$. Suppose that $x \notin L$; then we will produce a $w \in C$ for which $\langle w, z \rangle < 0$, implying that $x \notin K$. Thus any element of K must also be in L.

It is easy to see that L is a closed, convex set. Thus x has a unique projection in L, call it y; since $x \notin L$, $\|x - y\| > 0$.

Since y is the closest point to x and L is convex, $\langle y - x, y - z \rangle \leq 0$ for any $z \in L$: if the inner product were positive, then $y + \varepsilon(z - y) \in L$ would be closer to x, contradicting that y is the closest point.

Take any $z \in L$; then $tz \in L$ for any $t > 0$ as well, and so $\langle y - x, y - tz \rangle \leq 0$. Dividing by t, $\left\langle y - x, \frac{1}{t} y - z \right\rangle \leq 0$; letting $t \to \infty$, this means that $\langle y - x, z \rangle \geq 0$.

Noting that $x_i \in L$ for each i, this means that $\langle y - x, x_i \rangle \geq 0$, so that $y - x \in C$. But we also have that

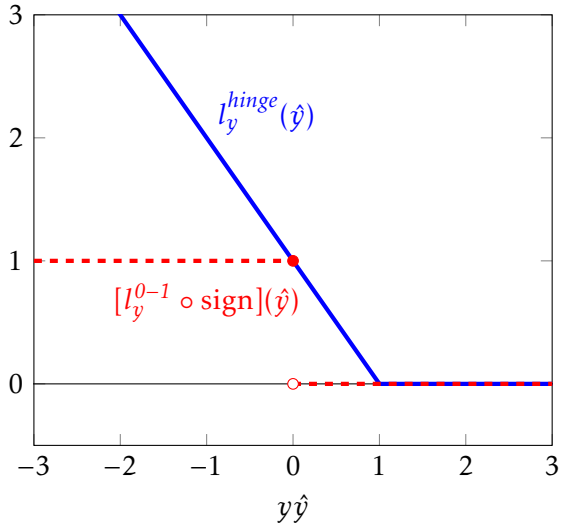
$$\langle y - x, x \rangle = \langle y - x, y \rangle + \langle y - x, x - y \rangle \leq 0 - \|x - y\|^2 < 0,$$

since $y \in L$ and $x \neq y$. Thus we have found $y - x \in C$ for which $\langle y - x, x \rangle < 0$, and hence $x \notin K$. □

13.2.2 *Hinge loss interpolation*

The *hinge loss* is given by

$$l_y^{hinge}(\hat{y}) = \begin{cases} 1 - y\hat{y} & \text{if } y\hat{y} \leq 1 \\ 0 & \text{if } y\hat{y} \geq 1. \end{cases}$$



Notice that if $L_S^{hinge}(x \mapsto w \cdot x) = 0$, then for all $i \in [m]$, $y_i x_i \cdot w \geq 1$. Thus (13.2) is equivalent to

$$\underset{w : L_S^{hinge}(x \mapsto w \cdot x) = 0}{\arg\min} \|w\|, \tag{13.3}$$

the *minimum-norm hinge loss interpolator*. This is kind of a nice analogy to how gradient descent for least squares or similar losses (starting at $w_0 = 0$) finds the minimum-norm interpolator for that loss! But, interestingly, explicitly minimizing logistic loss (with gradient descent) implicitly minimizes hinge loss.

Transforming the hard constraint into a soft one gives us a soft support vector

machine,

$$\arg\min_h \mathrm{L}_S^{hinge}(h) + \lambda\|h\|^2.$$

13.2.3 *Margin analysis*

How can we think about the 0-1 generalization error of the max-margin predictor?

We know that in dimension d, the VC dimension is either d or $d+1$, depending on if we put an intercept in. But when d is high, e.g. $d > m$, this doesn't really tell us anything, and in particular this doesn't use the norm at all.

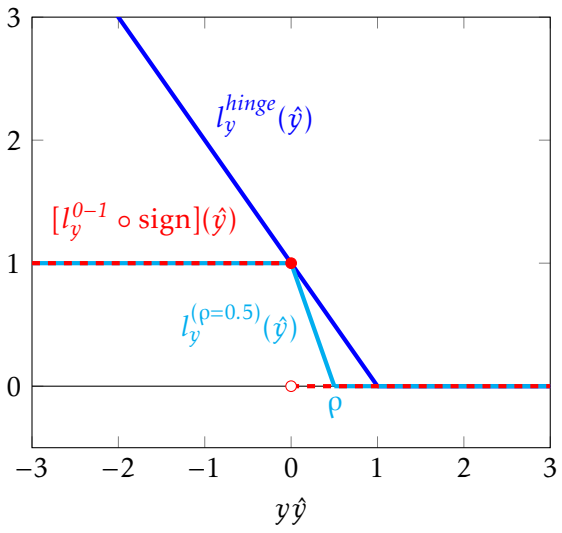
We're finding the minimum-norm interpolator, though, so maybe we can use a Rademacher bound that exploits that the norm isn't too big. So, let's think about $\mathcal{H}_B = \{h : \|h\| \le B\}$ with the norm $\|x \mapsto \langle w, x\rangle\| = \|w\|$. We know that $\mathbb{E}_S \mathrm{Rad}(\mathcal{H}_B|_{S_x}) \le \frac{B}{\sqrt{m}}\sqrt{\mathbb{E}\|x\|^2}$. To use this for a generalization bound on the 0-1 loss, though, we need to convert these soft predictions into hard ones with the sign function, so that the estimation error is bounded in terms of $\mathrm{Rad}((\ell_{0-1} \circ \mathrm{sign} \circ \mathcal{H}_B)|_S)$. But $\ell_{0-1} \circ \mathrm{sign}$ *isn't Lipschitz*; it jumps suddenly from 0 to 1 as the sign of the predictor changes. So we can't use Talagrand's lemma to peel it off at all.

(When deriving VC dimension, we pretended the 0-1 loss was Lipschitz, but that only worked because we were working with a hypothesis class mapping to ± 1. There's no similar trick we can play with continuous-output $\mathcal{H}$.)

We can work around this problem with *surrogate losses*. The hinge loss, above, is one example: $\ell_{0-1}(h,z) \le \ell_{hinge}(h,z)$ for any inputs, so necessarily $\mathrm{L}_{\mathcal{D}}^{0-1}(h) \le \mathrm{L}_{\mathcal{D}}^{hinge}(h)$, and so a bound on $\mathrm{L}_{\mathcal{D}}^{hinge}(h)$ (based on the empirical hinge loss plus some complexity term) will also apply to $\mathrm{L}_{\mathcal{D}}^{0-1}(h)$.

We can also use a tighter surrogate, though. One choice is *margin loss*:

$$l_y^\rho(\hat{y}) = \begin{cases} 1 & \text{if } y\hat{y} \le 0 \\ 1 - \frac{1}{\rho} y\hat{y} & \text{if } 0 \le y\hat{y} \le \rho \\ 0 & \text{if } y\hat{y} \ge \rho. \end{cases}$$



This is $1/\rho$-Lipschitz, bounded in $[0,1]$, and always an upper bound to the 0-1 loss.

If $\min_i y_i h(x_i) \geq \rho$, then $L_S^\rho(h) = 0$. We get an immediate result:

$$L_{\mathcal{D}}^{0-1}(\operatorname{sign} \circ h) \leq L_{\mathcal{D}}^{\rho}(h) \leq L_S^{\rho}(h) + \frac{2}{\rho} \mathop{\mathbb{E}}_{S} \operatorname{Rad}(\mathcal{H}|_{S_x}) + \sqrt{\frac{1}{2m} \log \frac{1}{\delta}} \tag{13.4}$$

if $h \in \mathcal{H}$ and we picked ρ independently of S and h.

Separable case

One interesting case: suppose that not only the training data S but actually the *distribution* $\mathcal{D}$ is separable with a geometric margin ρ.

Equivalently, this means there's an h^* with $\|h^*\| = 1/\rho$ such that $L_{\mathcal{D}}^1(h^*) = 0$; here the 1-margin loss is also sometimes called *ramp loss*. This means that almost surely in S we have that $L_S^\rho(h^*) = 0$ as well.

Consider ERM with $\mathcal{H} = \{h : \|h\| \leq \|h^*\| = 1/\rho\}$; its sample error will thus also be zero, and (13.4) becomes that

$$L_{\mathcal{D}}^{0-1}(\operatorname{sign} \circ \operatorname{ERM}_{\mathcal{H}}(S)) \leq 2\frac{\|h^*\|}{\sqrt{m}} \sqrt{\mathbb{E}\|x\|^2} + \sqrt{\frac{1}{2m} \log \frac{1}{\delta}}.$$

But notice that if the hinge loss is zero, then the ramp loss is as well. So, in this realizable-with-a-margin setting, the h^* that achieves zero ramp loss will also achieve zero hinge loss, and so $\mathcal{H}$ contains at least one zero-error predictor. Thus (13.2) or equivalently (13.3) will be guaranteed to find a zero-training-loss solution in $\mathcal{H}$, meaning that they find an ERM in this $\mathcal{H}$, and so the bound above applies.

This shows nonuniform learning over bounded domains, but not over $\mathbb{R}^d$ because of the $\mathbb{E}\|x\|^2$.

The only problem is that we probably don't actually know the value of $\|h^*\|$! The following approach will handle this, as well as more general settings.

General case

But... it's weird to pick ρ independently of S and h! If we obtain a margin of ρ, then $L_S^{\rho'}(h)$ will be zero for any $\rho' \leq \rho$ and start growing for $\rho' > \rho$; the optimal choice of ρ will need to trade this off with the $1/\rho'$ term. Why would we know how big a margin we're going to reasonably get before we look at the data?

We can do a nonuniform analysis to avoid committing in advance to a particular margin ρ, exactly like what we did for SRM:

Proposition 13.4. *Let $\mathcal{H}$ contain functions mapping to $\mathbb{R}$, and fix some $r > 0$. Then for any $\delta \in (0,1)$, we have with probability at least $1 - \delta$ over the choice of $S \sim \mathcal{D}^m$ that it holds for all $h \in \mathcal{H}$ and $\rho \in (0, r]$ that*

$$L_{\mathcal{D}}^{0-1}(\operatorname{sign} \circ h) \leq L_S^{\rho}(h) + \frac{4}{\rho} \mathop{\mathbb{E}}_{S' \sim \mathcal{D}^m} \operatorname{Rad}(\mathcal{H}|_{S'_x}) + \sqrt{\frac{1}{m} \log \log_2 \frac{2r}{\rho}} + \sqrt{\frac{1}{2m} \log \frac{2}{\delta}}.$$

Proof. Let $\rho_k = r2^{-k}$ for all $k \geq 0$, and $\delta_k = \frac{6\delta}{\pi^2 k^2}$ for $k \geq 1$; note that $\sum_{k=1}^{\infty} \delta_k = \delta$. By (13.4), it holds with probability at least $1 - \delta_k$ for each ρ_k that

$$\forall h \in \mathcal{H}, \quad L_{\mathcal{D}}^{0-1}(\operatorname{sign} \circ h) \leq L_S^{\rho_k}(h) + \frac{2}{\rho_k} \mathop{\mathbb{E}}_{S' \sim \mathcal{D}^m} \operatorname{Rad}(\mathcal{H}|_{S'_x}) + \sqrt{\frac{1}{2m} \log \frac{1}{\delta_k}}.$$

For any $\rho \in (0, r]$, the smallest k such that $\rho_k \leq \rho$ is given by $k = \left\lceil \log_2 \frac{r}{\rho} \right\rceil$.

We have $\ell_{\rho'} \le \ell_\rho$ for any $\rho' \le \rho$, so $L_S^{\rho_k}(h) \le L_S^{\rho}(h)$.

We also know that $\rho \le \rho_{k-1} = 2\rho_k$, so $\frac{1}{\rho_k} \le \frac{2}{\rho}$.

Finally, from $\log \frac{1}{\delta_k} = \log \frac{\pi^2}{6\delta} + 2 \log \log_2 \left\lceil \log_2 \frac{r}{\rho} \right\rceil$ we use that $\pi^2/6 < 2$ and $\lceil \log_2 a \rceil < \log_2(a) + 1 = \log_2(2a)$, then $\sqrt{a+b} \le \sqrt{a} + \sqrt{b}$. □

Then, we can run whatever learning algorithm and look at what ρ gives us the best tradeoff in the upper bound above. For instance, if the data is separable and h maximized the margin, then maybe we pick ρ to be the margin we got; then $L_S^{\rho}(h) = 0$, and hopefully the $1/\rho$ term isn't too big since we were trying to maximize the margin. We might get a better tradeoff by using a bigger ρ, though, if it still keeps our margin loss reasonably small.

We do have to commit to some predefined upper bound r on the biggest margin we can handle, but the resulting bound only depends on it through $\sqrt{\log \log_2(r/\rho)}$, so we can pick something big.

13.3 OTHER MODELS/ALGORITHMS

Lyu and Li [LL20] and Ji and Telgarsky [JT20] study small-learning-rate gradient descent on L-*homogeneous* networks, those satisfying $h(x; \alpha w) = \alpha^L h(x; w)$ for $\alpha > 0$; this is true e.g. for (leaky)-ReLU networks. (We'll describe the [LL20] results.) Their analysis is in terms of the *normalized margin*

$$\bar{\gamma}(w) = \frac{\min_{i \in [m]} y_i h(x_i; w)}{\|w\|_2^L}.$$

This normalization is exactly the one that makes $\bar{\gamma}(\alpha w) = \bar{\gamma}(w)$. They show, using an approach like that of Section 13.2, that gradient flow or small-learning-rate gradient descent (under some additional regularity conditions) monotonically increase the log-sum-exp version of normalized margin, which means they approximately monotonically increase the normalized margin, which roughly means that it finds a local maximum (ish) of the normalized margin. Unfortunately, in this case the KKT conditions aren't actually enough to get a global minimizer, and in fact gradient descent doesn't even always converge to even a *local* maximizer of the margin [VSS22], but we can generally expect that it "usually" does.

This is a kind of margin maximization, and Proposition 13.4 applies. Knowing these results, you can ask questions like what this margin maximization actually does on particular models [e.g. Fre+23].

There's been a bunch of recent work trying to figure out the implicit regularization of Adam, rather than SGD, on homogeneous networks; some recent papers are [WMCL21; Wan+22; CKS23; XL24].

There's also a *ton* more work in this area; Vardi [Var22] gives a (now kind of outdated) survey.

REFERENCES

[Bach25] Francis Bach. *Learning Theory from First Principles*. May 2025.

[CKS23] Matias D. Cattaneo, Jason M. Klusowski, and Boris Shigida. *On the Implicit Bias of Adam*. 2023. arXiv: 2309.00079.

[Fre+23] Spencer Frei, Gal Vardi, Peter L. Bartlett, Nathan Srebro, and Wei Hu. "Implicit Bias in Leaky ReLU Networks Trained on High-Dimensional Data". *ICLR*. 2023. arXiv: 2210.07082.

[GLSS18] Suriya Gunasekar, Jason Lee, Daniel Soudry, and Nathan Srebro. "Characterizing Implicit Bias in Terms of Optimization Geometry". *ICML*. 2018. arXiv: 1802.08246.

[JT19] Ziwei Ji and Matus Telgarsky. "The implicit bias of gradient descent on nonseparable data". *COLT*. 2019. arXiv: 1803.07300.

[JT20] Ziwei Ji and Matus Telgarsky. "Directional convergence and alignment in deep learning". *NeurIPS*. 2020. arXiv: 2006.06657.

[LL20] Kaifeng Lyu and Jian Li. "Gradient Descent Maximizes the Margin of Homogeneous Neural Networks". *ICLR*. 2020. arXiv: 1906.05890.

[Sou+18] Daniel Soudry, Elad Hoffer, Mor Shpigel Nacson, Suriya Gunasekar, and Nathan Srebro. The Implicit Bias of Gradient Descent on Separable Data. *JMLR* (2018). arXiv: 1710.10345.

[SSBD14] Shai Shalev-Shwartz and Shai Ben-David. *Understanding Machine Learning: From Theory to Algorithms*. Cambridge University Press, 2014.

[Var22] Gal Vardi. *On the Implicit Bias in Deep-Learning Algorithms*. 2022. arXiv: 2208.12591.

[VSS22] Gal Vardi, Ohad Shamir, and Nathan Srebro. "On Margin Maximization in Linear and ReLU Networks". *NeurIPS*. 2022. arXiv: 2110.02732.

[Wan+22] Bohan Wang, Qi Meng, Huishuai Zhang, Ruoyu Sun, Wei Chen, Zhi-Ming Ma, and Tie-Yan Liu. "Does Momentum Change the Implicit Regularization on Separable Data?" *NeurIPS*. 2022. arXiv: 2110.03891 [cs.LG].

[WMCL21] Bohan Wang, Qi Meng, Wei Chen, and Tie-Yan Liu. "The Implicit Bias for Adaptive Optimization Algorithms on Homogeneous Neural Networks". *ICML*. 2021. arXiv: 2012.06244.

[XL24] Shuo Xie and Zhiyuan Li. "Implicit Bias of AdamW: ℓ_∞ Norm Constrained Optimization". *ICML*. 2024. arXiv: 2404.04454.