Advanced Machine Learning - Hilary Term 2017
8 : Learning Real-valued Functions

Lecturer: Varun Kanade

So far our focus has been on learning boolean functions. Boolean functions are suitable for modelling binary classification problems; in fact, even multi-class classification can be viewed as a sequence of binary classification problems. Many commonly used approaches for multi-class classification, such as one-vs-rest or one-vs-one, solve several binary classification problems as a means to perform multi-class classification. However, sometimes we may need to learn functions whose output is real-valued (or vector-valued). In this lecture, we will study linear models and generalised linear models. In order to give bounds on the generalisation error, we'll need to introduce some new concepts that play a role analogous to the VC dimension. We will also study some basic convex optimisation techniques.

1 Learning Real-Valued Functions

Let us start with the general question of learning real-valued functions. Suppose our instance space is $X_n = \mathbb{R}^n$. Let $g : \mathbb{R}^n \to \mathbb{R}$ be the target function and D be the target distribution over $\mathbb{R}^n$. Let us define a notion of an example oracle for real-valued functions. We consider the oracle $\mathsf{EX}(g, D)$ that when queried does the following: Draw $x \sim D$, draw $y \sim D_x$, where D_x is a distribution over $\mathbb{R}$ such that $\mathop{\mathbb{E}}_{y \sim D_x}[y] = g(x)$, and return (x, y). One may consider a more restricted oracle, which actually returns $(x, g(x))$ directly; but assuming that we get the exact function value without any noise is even less realistic in the context of real-valued functions than in the case of boolean functions.

The goal of a learning algorithm is to output some hypothesis, $\widehat{g} : \mathbb{R}^n \to \mathbb{R}$, such that the expected squared error (or loss), $\varepsilon(\widehat{g}) := \mathop{\mathbb{E}}_{x \sim D}\left[(g(x) - \widehat{g}(x))^2\right] \leq \epsilon$.[1] As we don't observe $g(x)$ at all, but only y, such that $\mathbb{E}\left[y \mid x\right] = g(x)$, the learning algorithm cannot directly aim to minimise the empirical squared error,

$$\widehat{\varepsilon}_S(\widehat{g}) = \frac{1}{m}\sum_{i=1}^{m}(g(x_i) - \widehat{g}(x_i))^2 \tag{1}$$

where $S = \{(x_1, y_1), \ldots, (x_m, y_m)\}$ is the training data. Instead, we will consider learning algorithms that attempt to minimise, the empirical *risk* with respect to the *observed data*,

$$\widehat{R}_S(\widehat{g}) = \frac{1}{m}\sum_{i=1}^{m}(\widehat{g}(x_i) - y_i)^2 \tag{2}$$

This approach is referred to as *empirical risk minimisation* (ERM). Let $\widehat{g} : \mathbb{R}^n \to \mathbb{R}$ be any function.

Let us first argue that under the assumption that $\mathbb{E}\left[y \mid x\right] = g(x)$, minimising $\widehat{\varepsilon}_G(\widehat{g})$ and $\widehat{R}_S(\widehat{g})$ are not that different. We will argue that this is the case, when considering the respective quantities with respect to the actual data distribution. In Section 4, we will show how the empirical estimates of these quantites on a sample relate to the true values under the distribution.

[1] It is certainly possible, and often even desirable, to consider other loss functions. Due to lack of time, we will focus only on the case where the goal is to minimise the expected squared error.

Consider the following:

$$\mathop{\mathbb{E}}_{x\sim D}\mathop{\mathbb{E}}_{y\sim D_x}\left[(\widehat{g}(x)-y)^2\right] = \mathop{\mathbb{E}}_{x\sim D}\left[(\widehat{g}(x)-g(x))^2\right] + \mathop{\mathbb{E}}_{x\sim D}\mathop{\mathbb{E}}_{y\sim D_x}\left[(g(x)-y)^2\right] \tag{3}$$

$$+\ 2\cdot\mathop{\mathbb{E}}_{x\sim D}\left[(\widehat{g}(x)-g(x))\mathop{\mathbb{E}}_{y\sim D_x}\left[g(x)-y\right]\right] \tag{4}$$

The second term in (3) does not depend on $\widehat{g}$ at all. The inner expectation in (4) is 0 as $\mathop{\mathbb{E}}_{y\sim D_x}[y] = g(x)$ by assumption on the data distribution. Thus, we have

$$\mathop{\mathbb{E}}_{x\sim D}\mathop{\mathbb{E}}_{y\sim D_x}\left[(\widehat{g}(x)-y)^2\right] = \mathop{\mathbb{E}}_{x\sim D}\left[(\widehat{g}(x)-g(x))^2\right] + \mathop{\mathbb{E}}_{x\sim D}\mathop{\mathbb{E}}_{y\sim D_x}\left[(g(x)-y)^2\right] \tag{5}$$

Thus, we see that $\widehat{g}$ that minimises $\mathop{\mathbb{E}}_{x\sim D}\mathop{\mathbb{E}}_{y\sim D_x}\left[(\widehat{g}(x)-y)^2\right]$ also minimises $\mathop{\mathbb{E}}_{x\sim D}\left[(\widehat{g}(x)-g(x))^2\right]$.

2 Linear Regression

Let us look at the most well-studied regression problem—linear regression. In linear regression, we assume that the target function, g, is of the form $g(x) = w\cdot x$ for $w\in\mathbb{R}^n$. The goal is to estimate, $\widehat{g}$, represented by parameters $\widehat{w}$, such that $\varepsilon(\widehat{w}) = \mathop{\mathbb{E}}_{x\sim D}\left[(w\cdot x-\widehat{w}\cdot x)^2\right]\leq\epsilon$. In order to bound the generalisation error, we will typically require that the distribution D has support over a bounded set and that the target function, g, is also represented using parameters, w, with bounded ℓ_2 norm. Furthermore, we'll also assume that the observations y lie in some interval $[-M, M]$.

For radius R, let $\mathbb{B}_n(0,R) = \{z\in\ \mathbb{R}^n\mid \|z\|_2\leq R\}$ denote the ℓ_2 ball of radius R in $\mathbb{R}^n$. Let D be a distribution that has support over the unit ball, $\mathbb{B}_n(0,1)$ and for some $W>0$, define the set of linear functions,

$$\mathcal{G}_W = \{x\mapsto w\cdot x\mid \|w\|_2\leq W\}.$$

Let $M\geq W$ and for each $x\in\mathbb{B}_n(0,1)$, suppose that D_x has support contained in $[-M,M]$. Note that as,

$$\sup_{\substack{x\in\mathbb{B}_n(0,1)\\ w\in\mathbb{B}_n(0,W)}}|w\cdot x| = W,$$

for any $x\in\mathbb{B}_n(0,1)$, $w\in\mathbb{B}_n(0,W)$, there always exists a distribution D_x with support contained in $[-M,M]$, such that $\mathop{\mathbb{E}}_{y\sim D_x}[y] = w\cdot x$.

Least Squares Method

Let $S = \{(x_1,y_1),\ldots,(x_m,y_m)\}$ denote the observed training data sample. The ERM approach suggests that we should obtain $\widehat{w}$ by minimising the *empirical risk*, defined as

$$\widehat{R}_S(\widehat{w}) = \frac{1}{m}\sum_{i=1}^{m}(\widehat{w}\cdot x_i - y_i)^2$$

In the case of least squares regression, we can obtain $\widehat{w}$ in closed form by setting the gradient to 0. Observe that $\widehat{R}_S(\widehat{w})$ is a convex function of $\widehat{w}$; thus, alternatively we may use standard convex optimisation techniques to obtain a solution. In certain cases, this may even be desirable as the closed form solution involves constructing and inverting matrices—if the data-dimension is large this can be much more expensive than performing gradient descent.

Appendix A describes a projected gradient descent algorithm for constrained convex optimisation and give a proof that the algorithm finds a close-to-optimal solution, provided the

feasible set has finite diamater and the gradient remains bounded on the feasible set. This is by no means the most general result. A detailed study of convex optimisation is beyond the scope of this course; the student is referred to the following books (Bubeck, 2015; Boyd and Vandenberghe, 2004; Nemirovski and Yudin, 1983).

Instead of minimising $\widehat{R}_S(\widehat{w})$ as an unconstrainted over $\mathbb{R}^n$, we insstead minimise $\widehat{R}_S(\widehat{w})$, under the constraint that $\widehat{w} \in \mathbb{B}_n(0, W)$. After all, we are promised that the target lies in the set $\mathbb{B}_n(0, W)$. Note that the diameter of $\mathbb{B}_n(0, W) = 2W$ and it is easy to see that the ℓ_2 norm of the gradient of $\widehat{R}_S(\widehat{w})$ is bounded by $W + M$ over the set $\mathbb{B}_n(0, W)$. Thus, Theorem 5 shows that if we run the projected gradient descent algorithm for $\Theta(\frac{WM}{\epsilon^2})$ iterations, we are guaranteed to obtain $\widehat{w}$, such that,

$$\widehat{R}_S(\widehat{w}) \leq \min_{w \in \mathbb{B}_n(0,W)} \widehat{R}_S(w) + \epsilon$$

This only shows that the empirical risk of the obtained $\widehat{w}$ is at most ϵ larger than minimum possible empirical risk. In Section 4 we show how to bound the expected risk in terms of the empirical risk as a function of the sample size and confidence parameter δ.

3 Generalised Linear Models

Let us now look at a more expressive class of functions. In the statistics literature, these are referred to as generalised linear models. These are models of the form, $g(x) = u(w \cdot x)$, where $x \in \mathbb{R}^n$, $w \in \mathbb{R}^n$ and $u : \mathbb{R} \to \mathbb{R}$. It is common to assume that u is monotone and Lipschitz, however these models can be defined more broadly. Supose that u is strictly monotone, so that u^{-1} is well-defined. Then although g is no longer a linear function of x, $u^{-1}(g(x))$ is linear. The function u^{-1} is referred to as the link function.

Generalised linear models are widely used in machine learning and also form the basis of a single unit in most neural networks. Note that units with a sigmoid, hyperbolic tangent, or rectifier activation functions are all generalised linear models.

In what follows we'll assume that u is monotonically increasing (not necessarily strict) and 1-Lipschitz, *i.e.*, $|u(z) - u(z')| \leq |z - z'|$ for all $z, z' \in \mathbb{R}$. We will also assume that u is known to the learning algorithm. As in the case of linear regression, let us suppose that D has support over $\mathbb{B}_n(0, 1)$ in $\mathbb{R}^n$ and for $W > 0$ consider the class of generalised linear models:

$$\mathcal{G}_{W,u} = \{x \mapsto u(w \cdot x) \mid w \in \mathbb{B}_n(0, W)\}$$

Note that as we are allowing W to be an arbitrary parameter, the requirements that for all x in the support of D, $\|x\|_2 \leq 1$ and that u is 1-Lipschitz are not stringent restrictions. For example, if our data is such that the norm of x in the support is bounded by some $B > 1$, we can scale the data to get all norms bounded by 1 and allow $\|w\|_2$ to be as large as WB. Similarly, if u were l-Lipschitz, we can use some $\widetilde{u}$, where $\widetilde{u}(z) = u(\frac{z}{l})$ is 1-Lipschitz, and instead allow $\|w\|_2$ to be as large as Wl.

For simplicity we'll assume that $u(0) = 0$ (although, this is not the case for some functions, we can easily centre u for the purpose of the algorithm and then undo the centering at the time of prediction). Thus, as in the case of linear regression, we'll assume that for all $x \in \mathbb{B}_n(0, 1)$, the distribution D_x has support contained in $[-M, M]$ for some $M \geq W$. Again, observe that as $|w \cdot x| \leq W$ for all $w \in \mathbb{B}_n(0, W)$ and $x \in \mathbb{B}_n(0, 1)$, and since u is 1-Lipschitz, there exists a distribution with support contained in $[-M, M]$, such that $\mathop{\mathbb{E}}_{y \sim D_x}[y] = u(w \cdot x)$.

3.1 Empirical Risk Minimisation

As in the case of linear regression, we can attempt to find a minimiser of the empirical risk, defined on a sample $S = \{(x_1, y_1), \ldots, (x, m)\}$, as,

$$\widehat{R}_S(w) = \frac{1}{m}\sum_{i=1}^{m}(u(w \cdot x_i) - y_i)^2.$$

The trouble is that unlike in the case of linear regression, the empirical risk is no longer a convex function of w. In fact, it has been shown by Auer et al. (1996) that for even relatively simple inverse link functions, such as the sigmoid, $u(z) = \frac{1}{1+e^{-z}}$, the empirical risk may have exponentially many local minima as a function of the dimension.

Surrogate Loss Function

In order to avoid optimising a non-convex function (for which there are no general purpose algorithms), we'll use a strategy often employed in machine learning—using a surrogate convex loss function. Remarkably, in the case of generalised linear models, there exists a surrogate loss function, such that the expected risk minimiser for this surrogate loss function and that for the squared loss is exactly the same! In addition, we can also show that an approximate minimiser of the risk for the surrogate loss function is also an approximate minimiser of the risk for squared loss.

For $x \in \mathbb{B}_n(0,1), y \in [-M, M]$, define the (surrogate) loss, ℓ, for $w \in \mathbb{B}_n(0, W)$ as follows:

$$\ell(w; x, y) = \int_0^{w\cdot x}(u(z) - y)dz.$$

We will define the empirical risk of the surrogate loss as,

$$\widehat{R}^{\ell}_S(w) = \frac{1}{m}\sum_{i=1}^{m}\ell(w; x_i, y_i)$$

Let us also compute the gradient of this empirical risk:

$$\nabla_w \widehat{R}^{\ell}_S(w) = \frac{1}{m}\sum_{i=1}^{m}\nabla_w \ell(w; x_i, y_i) = \frac{1}{m}\sum_{i=1}^{m}(u(w \cdot x) - y_i)x_i$$

For comparison, let us also write the gradient of the empirical risk for the squared loss, $\widehat{R}_S(w)$,

$$\nabla_w \widehat{R}_S(w) = \frac{2}{m}\sum_{i=1}^{m}(u(w \cdot x_i) - y_i)u'(w \cdot x_i)x_i$$

Notice that apart from the factor 2, the main difference is that the i^{th} example has $u'(w \cdot x_i)$ as a multiplicative factor in the gradient. Although, $u' \geq 0$ as u is monotonically increasing, it may at times be very small.[2] Let us also show that $\widehat{R}^{\ell}_S(w)$ is indeed convex—it is sufficient to show that $\ell(w; x, y)$ is convex. Notice that the Hessian of $\ell(w; x, y)$ is simply $u'(w\cdot x)xx^{\mathsf{T}}$, where $x \in \mathbb{R}^n$ is treated as a column vector. Since u is monotonically increasing, u' is non-negative, and hence $u'(w \cdot x)xx^{\mathsf{T}}$ is positive semi-definite—thus, $\ell(w; x, y)$ is convex.

Next, let us also show that w^*, the expected risk minimiser for the squared loss, is also the expected risk minimiser for this surrogate loss function ℓ. Let $w^* \in \mathbb{R}^n$ be used to define the target function; then Eq. (5) shows that w^* is a minimiser of the expected risk using the squared

[2] For instance, when u is the sigmoid function $u'(z) \approx 0$ when $|z|$ is somewhat large. This is also the reason why cross-entropy loss is better than squared loss for avoiding the vanishing gradient problem.

loss function. In what follows, we will also make use of the assumption that $\mathop{\mathbb{E}}_{y\sim D_x}[y] = u(w^* \cdot x)$. Consider the following for any $x \in \mathbb{B}_n(0,1)$:

$$\begin{aligned}
\mathop{\mathbb{E}}_{y\sim D_x}\left[\ell(w;x,y)\right] - \mathop{\mathbb{E}}_{y\sim D_x}\left[\ell(w^*;x,y)\right] &= \mathop{\mathbb{E}}_{y\sim D}\left[\int_{w^*\cdot x}^{w\cdot x}(u(z)-y)dz\right] \\
&= \int_{w^*\cdot x}^{w\cdot x}\left(u(z) - \mathop{\mathbb{E}}_{y\sim D_x}[y]\right)dz \\
&= \int_{w^*\cdot x}^{w\cdot x}(u(z)-u(w^*\cdot x))dz \geq \frac{1}{2}(u(w\cdot x)-u(w^*\cdot x))^2
\end{aligned}$$

The last inequality int the calculations above follows from the fact that u is monotonically increasing and 1-Lipschitz. This shows that the expected risk, $R^\ell(w) \geq R^\ell(w^*)$, *i.e.,* w^* is a minimiser of the risk with respect to the surrogate loss function ℓ. Furthermore, it also establishes that,

$$\varepsilon(w) := \mathop{\mathbb{E}}_{x\sim D}\left[(u(w\cdot x)-u(w^*\cdot x))^2\right] \leq 2\cdot(R^\ell(w)-R^\ell(w^*))$$

This shows that it is sufficient to identify a $\widehat{w}$, whose expected risk is at most $\frac{\epsilon}{2}$ larger than that of w^* with respect to the surrogate loss function. In order to find a good enough empirical risk minimiser, is easy to see that performing roughly $\Theta\left(\frac{WM}{\epsilon^2}\right)$ projected gradient steps suffices. In the next section, we'll show how to relate empirical risk to expected risk by introducing a new complexity measure called Rademacher complexity.

4 Rademacher Complexity

Let us now address the question of bounding the generalisation error. We will introduce a new concept called Rademacher complexity. Let us first define *empirical Rademacher complexity* for a family of functions.

Definition 1 (Empirical Rademacher Complexity)**.** *Let $\mathcal{G}$ be a family of functions mapping some space $X \to [a,b]$ and let $S = \{x_1, x_2, \ldots, x_m\} \subseteq X$ be a fixed sample of size m. Then the* empirical Rademacher complexity *of $\mathcal{G}$ with respect to the sample S is defined as,*

$$\widehat{\mathsf{RAD}}_S(G) = \mathop{\mathbb{E}}_{\boldsymbol{\sigma}}\left[\sup_{g\in\mathcal{G}}\frac{1}{m}\sum_{i=1}^{m}\sigma_i g(x_i)\right],$$

where $\boldsymbol{\sigma} = (\sigma_1, \sigma_2, \ldots, \sigma_m)$ are i.i.d. Rademacher random variables, i.e., σ_i takes value in $\{-1,1\}$ uniformly at random.

In words, the empirical Rademacher complexity measures how well functions from a class correlate with random noise. This corresponds to our notion that the more complex the class $\mathcal{G}$, the more easily it can fit noise. In particular, let us suppose that $\mathcal{G}$ is a class of boolean functions (with range $\{-1,1\}$) with $\mathsf{VCD}(\mathcal{G}) \geq m$ and let S be a set that is shattered by $\mathcal{G}$, then $\widehat{\mathsf{RAD}}_S(G) = 1$. However, Rademacher complexity can be defined for any class of real-valued functions.[3] Let us now define Rademacher complexity, which is defined as the expected empirical Rademacher complexity of sets of size m drawn from a distribution D over X.

Definition 2 (Rademacher Complexity)**.** *Let D be a distribution over X and let $\mathcal{G}$ be a family of functions mapping $X \to [a,b]$. For any integer, $m \geq 1$, the Rademacher complexity of $\mathcal{G}$,*

[3]We will ignore issues of measurability; this will not be a matter of concern for the function classes we study in this course.

is the expectation of the empirical Rademacher complexity of $\mathcal{G}$ over samples of size m drawn independently from D, i.e.,

$$\mathsf{RAD}_m(\mathcal{G}) = \mathop{\mathbb{E}}_{S\sim D^m}\left[\widehat{\mathsf{RAD}}_S(\mathcal{G})\right].$$

For a function, $g \in \mathcal{G}$, and a sample $S = \{x_1, \ldots, x_m\}$ drawn according to D, let us use the notation $\widehat{\mathbb{E}}_S[g] = \frac{1}{m}\sum_{i=1}^m g(x_i)$. We are interested in understanding the behaviour of the difference, $\left|\widehat{\mathbb{E}}_S[g] - \mathop{\mathbb{E}}_{x\sim D}\left[g(x)\right]\right|$ as a function of the sample size m and the Rademacher complexity $\mathsf{RAD}_m(\mathcal{G})$. We will prove the following theorem, which is analogous to a theorem we proved using the VC dimension.

Theorem 3. *Let $\mathcal{G}$ be a family of functions mapping $X \to [0,1]$. Suppose that a sample $S = \{x_1, \ldots, x_m\}$ of size m is drawn according to distribution D over X. Then for any $\delta > 0$, with probability at least $1-\delta$, the following holds for all $g \in \mathcal{G}$,*

$$\mathop{\mathbb{E}}_{x\sim D}\left[g(x)\right] \le \widehat{\mathbb{E}}_S[g] + 2\mathsf{RAD}_m(\mathcal{G}) + \sqrt{\frac{\log\frac{1}{\delta}}{2m}}$$

We will defer the proof of this theorem until later in the section. Let us first see how we may utilise this theorem to give bounds on the generalisation error when learning real-valued functions (or for that matter boolean functions). Let H be some hypothesis class and suppose our learning algorithm finds $h \in H$ that approximately minimises the empirical risk with respect to some loss function ℓ. Say H is a family of functions from $X \to Y$ and $\ell : Y \times Y \to [0,1]$. Define $\mathcal{G}$ to be a family of functions from $X \times Y \to [0,1]$ as:

$$\mathcal{G} = \{(x,y) \mapsto \ell(h(x), y) \mid h \in H\}$$

Suppose we get a dataset $S = \{(x_1, y_1), \ldots, (x_m, y_m)\}$ drawn from some distribution over $X \times Y$ (viewed as drawing $x \sim D$ and then $y \sim D_x$). Then, for any $h \in H$ and if $g \in \mathcal{G}$ is the corresponding function derived from h, the empirical risk of h is given by,

$$\widehat{R}^{\ell}_S(h) = \frac{1}{m}\sum_{i=1}^m \ell(h(x_i), y_i) = \frac{1}{m}\sum_{i=1}^m g(x_i, y_i),$$

and the expected risk of h is $\mathop{\mathbb{E}}_{x\sim D}\mathop{\mathbb{E}}_{y\sim D_x}\left[\ell(h(x), y)\right] = \mathop{\mathbb{E}}_{x\sim D}\mathop{\mathbb{E}}_{y\sim D_x}\left[g(x,y)\right]$. Thus, if we can bound the Rademacher complexity of $\mathcal{G}$, we will be able to bound the expected risk of h in terms of the empirical risk of h. The following composition lemma due to Talagrand often proves to be a very useful tool.

Lemma 1 (Talagrand's Lemma)**.** *Let $\mathcal{G}$ be a family of functions from $X \to \mathbb{R}$ and $\phi : \mathbb{R} \to [a,b]$ be l-Lipschitz. Let $\phi \circ \mathcal{G} = \{\phi \circ g \mid g \in \mathcal{G}\}$, then,*

$$\widehat{\mathsf{RAD}}_S(\phi \circ \mathcal{G}) \le l \cdot \widehat{\mathsf{RAD}}_S(\mathcal{G})$$

Proof. Fix some sample $S = \{x_1, \ldots, x_m\}$. Then, we have the following:

$$\begin{aligned}\widehat{\mathsf{RAD}}_S(\phi \circ \mathcal{G}) &= \frac{1}{m}\cdot\mathop{\mathbb{E}}_{\boldsymbol{\sigma}}\left[\sup_{g\in\mathcal{G}}\sum_{i=1}^m \sigma_i(\phi\circ g)(x_i)\right] \\ &= \frac{1}{m}\cdot\mathop{\mathbb{E}}_{\sigma_1,\ldots,\sigma_{m-1}}\mathop{\mathbb{E}}_{\sigma_m}\left[\sup_{g\in\mathcal{G}} u_{m-1}(g) + \sigma_m(\phi\circ g)(x_i)\right],\end{aligned}$$

where $u_{m-1}(g) = \sum_{i=1}^{m-1} \sigma_i(\phi \circ g)(x_i)$. Let us concentrate on just the inner expectation:

$$\mathop{\mathbb{E}}_{\sigma_m}\left[\sup_{g\in\mathcal{G}} u_{m-1}(g) + \sigma_m(\phi\circ g)(x_m)\right].$$

Note that by definition of supremum, we have the existence of $g_1, g_2 \in \mathcal{G}$ satisfying the following for every $\epsilon > 0$:

$$u_{m-1}(g_1) + (\phi\circ g_1)(x_m) \geq \sup_{g\in\mathcal{G}} u_{m-1}(g) + (\phi\circ g)(x_m) - \epsilon \tag{6}$$

$$u_{m-1}(g_2) - (\phi\circ g_2)(x_m) \geq \sup_{g\in\mathcal{G}} u_{m-1}(g) - (\phi\circ g)(x_m) - \epsilon \tag{7}$$

Then, we have the following for every $\epsilon > 0$:

$$\begin{aligned}\mathop{\mathbb{E}}_{\sigma_m}\left[\sup_{g\in\mathcal{G}} u_{m-1}(g) + \sigma_m(\phi\circ g)(x_m)\right] - \epsilon \leq \frac{1}{2}\big[&u_{m-1}(g_1) + (\phi\circ g_1)(x_m)\\ &+ u_{m-1}(g_2) - (\phi\circ g_2)(x_m)\big]\end{aligned}$$

As ϕ is l-Lipschitz, we have $|\phi(g_1(x_m)) - \phi(g_2(x_m))| \leq l\cdot|g_1(x_m) - g_2(x_m)| = ls(g_1(x_m) - g_2(x_m))$, where $s = \text{sign}(g_1(x_m) - g_2(x_m))$. Thus, we have

$$\begin{aligned}\mathop{\mathbb{E}}_{\sigma_m}\left[\sup_{g\in\mathcal{G}} u_{m-1}(g) + \sigma_m(\phi\circ g)(x_m)\right] - \epsilon &\leq \frac{1}{2}\big[u_{m-1}(g_1) + u_{m-1}(g_2) + ls(g_1(x_m) - g_2(x_m))\big]\\ &= \frac{1}{2}\big[u_{m-1}(g_1) + lsg_1(x_m) + u_{m-1}(g_2) - lsg_2(x_m)\big]\end{aligned}$$

As $\{s, -s\} = \{-1, 1\}$, we can rewrite the above as:

$$\mathop{\mathbb{E}}_{\sigma_m}\left[\sup_{g\in\mathcal{G}} u_{m-1}(g) + \sigma_m(\phi\circ g)(x_m)\right] - \epsilon \leq \mathop{\mathbb{E}}_{\sigma_m}\left[\sup_{g\in\mathcal{G}} u_{m-1}(g) + lg(x_m)\right]$$

As this inequality holds for every $\epsilon > 0$, we can in fact write,

$$\mathop{\mathbb{E}}_{\sigma_m}\left[\sup_{g\in\mathcal{G}} u_{m-1}(g) + \sigma_m(\phi\circ g)(x_m)\right] \leq \mathop{\mathbb{E}}_{\sigma_m}\left[\sup_{g\in\mathcal{G}} u_{m-1}(g) + lg(x_m)\right]$$

We can repeat the above for $i = m-1, m-2, \ldots, 1$, to show that,

$$\mathop{\mathbb{E}}_{\boldsymbol{\sigma}}\left[\sup_{g\in\mathcal{G}} \frac{1}{m}\sum_{i=1}^{m} \sigma_i(\phi\circ g)(x_i)\right] \leq l\cdot\mathop{\mathbb{E}}_{\boldsymbol{\sigma}}\left[\sup_{g\in\mathcal{G}} \frac{1}{m}\sum_{i=1}^{m} \sigma_i g(x_i)\right] = l\cdot\widehat{\mathsf{RAD}}_S(\mathcal{G})$$

□

4.1 Application to ℓ_p loss functions

Let $S = \{(x_1, y_1), \ldots, (x_m, y_m)\} \subseteq X \times [-M, M]$. Let H be a family of functions mapping $X \to [-W, W]$. Let $\overline{S} = \{x_1, \ldots, x_m\}$. Let $\phi(z) = |z|^p$ for $p \geq 1$; $|\phi'(z)| = p|z|^{p-1}$ (we can also consider $p = 1$, though it is not differentiable at 0). Thus, ϕ is pa^{p-1}-Lipschitz on the interval $[-a, a]$. Let $\mathcal{G} = \{(x, y) \mapsto |h(x) - y|^p \mid h \in H\}$ be a family of functions from

$X \times [-M, M] \to [-a, a]$, where $a \leq (M+W)^p$. Suppose, $\widetilde{H} = \{h(x) - y \mid h \in H\}$ be a family of functions from $X \times [-M, M] \to [-(M+W), (M+W)]$. Let us observe that,

$$\begin{aligned}\widehat{\mathsf{RAD}}_S(\widetilde{H}) &= \mathop{\mathbb{E}}_{\boldsymbol{\sigma}}\left[\sup_{h\in H}\frac{1}{m}\sum_{i=1}^{m}\sigma_i(h(x_i)-y_i)\right] \\ &= \mathop{\mathbb{E}}_{\boldsymbol{\sigma}}\left[\sup_{h\in H}\frac{1}{m}\sum_{i=1}^{m}\sigma_i h(x_i)\right] + \mathop{\mathbb{E}}_{\boldsymbol{\sigma}}\left[\frac{1}{m}\sum_{i=1}^{m}\sigma_i y_i\right] = \widehat{\mathsf{RAD}}_{\overline{S}}(H)\end{aligned}$$

Then, using Talagrand's lemma, we have that:

$$\widehat{\mathsf{RAD}}_S(\phi \circ \widetilde{H}) \leq p \cdot (W+M)^{p-1} \cdot \widehat{\mathsf{RAD}}_S(\widetilde{H}) = p \cdot (W+M)^{p-1} \cdot \widehat{\mathsf{RAD}}_{\overline{S}}(H)$$

For instance, when using the squared loss, we use $\phi(z) = |z|^2$, and thus, we get $\widehat{\mathsf{RAD}}_S(\phi \circ \widetilde{H}) \leq 2(W+M)\widehat{\mathsf{RAD}}_{\overline{S}}(H)$.

Remark: Since Theorem 3 assumes that the range of functions is $[0, 1]$, we need to rescale the loss function appropriately, *e.g.,* use $\ell(h(x), y) = \frac{1}{a^p}|h(x) - y|^p$, where $a = \sup_{h,x,y}\{|h(x) - y|\}$.

4.2 Rademacher Complexity for Linear Functions and GLMs

Let us consider the set of linear functions, $\mathcal{G}_W = \{x \mapsto w \cdot x \mid w \in \mathbb{R}^n, \|w\|_2 \leq W\}$. Let $S = \{x_1, \ldots, x_m\} \subseteq \mathbb{R}^n$. Let $R_X = \sup_{x\in S}\|x\|_2$. We can compute the empirical Rademacher complexity of $\mathcal{G}_W$ on the set S as follows:

$$\begin{aligned}\widehat{\mathsf{RAD}}_S(\mathcal{G}_W) &= \mathop{\mathbb{E}}_{\boldsymbol{\sigma}}\left[\sup_{w\in\mathbb{B}_n(0,W)}\frac{1}{m}\sum_{i=1}^{m}\sigma_i(w\cdot x_i)\right] \\ &= \mathop{\mathbb{E}}_{\boldsymbol{\sigma}}\left[\sup_{w\in\mathbb{B}_n(0,W)}\left(w\cdot\frac{1}{m}\sum_{i=1}^{m}\sigma_i x_i\right)\right]\end{aligned}$$

Using the Cauchy-Schwartz Inequality (the equality case),

$$= W \cdot \mathop{\mathbb{E}}_{\boldsymbol{\sigma}}\left[\left\|\frac{1}{m}\sum_{i=1}^{m}\sigma_i x_i\right\|_2\right]$$

Using Jensen's Inequality,

$$\begin{aligned}&\leq W \cdot \left(\mathop{\mathbb{E}}_{\boldsymbol{\sigma}}\left[\left\|\frac{1}{m}\sum_{i=1}^{m}\sigma_i x_i\right\|_2^2\right]\right)^{\frac{1}{2}} \\ &= W \cdot \left(\mathop{\mathbb{E}}_{\boldsymbol{\sigma}}\left[\frac{1}{m^2}\sum_{i=1}^{m}\|x_i\|_2^2 + \frac{2}{m^2}\sum_{i<j}\sigma_i\sigma_j(x_i\cdot x_j)\right]\right)^{\frac{1}{2}}\end{aligned}$$

As σ_i are i.i.d. and have mean 0, and using the bound $\|x_i\|_2 \leq R_X$, we get

$$\widehat{\mathsf{RAD}}_S(\mathcal{G}_W) \leq \frac{W \cdot R_X}{\sqrt{m}}.$$

Bounding the Generalisation Error for learning GLMs

Let us now consider the surrogate loss function, $\ell(w; x, y)$, used for learning GLMs. We can write $\ell(w; x, y)$ as follows:

$$\ell(w; x, y) = \int_0^{w \cdot x} (u(z) - y) dz = \left(\int_0^{w \cdot x} u(z) dz \right) - y(w \cdot x)$$

Thus, we can write $\ell(w; x, y) = \phi_1(w \cdot x) - \phi_2(y(w \cdot x))$, where ϕ_1 is a W-Lipschitz function and ϕ_2 is the identity function.

Consider the class of functions defined as follows:

$$\mathcal{G}_{\ell,W} = \{(x, y) \mapsto \ell(w; x, y) \mid w \in \mathbb{B}(0, W)\}$$

Let $S = \{(x_1, y_1), \ldots, (x_m, y_m)\} \subseteq \mathbb{B}_n(0, 1) \times [-M, M]$ and let $\overline{S} = \{x_1, \ldots, x_m\}$. Let us consider two classes defined as follows:

$$\mathcal{G}^1_{\ell,W} = \{x \mapsto \phi_1(w \cdot x) \mid w \in \mathbb{B}_n(0, W)\}$$
$$\mathcal{G}^2_{\ell,W} = \{(x, y) \mapsto y(w \cdot x) \mid w \in \mathbb{B}_n(0, W)\}$$

It is left as a straightforward exercise to show the following:

$$\widehat{\mathsf{RAD}}_S(\mathcal{G}_{\ell,W}) \leq \widehat{\mathsf{RAD}}_{\overline{S}}(\mathcal{G}^1_{\ell,W}) + \widehat{\mathsf{RAD}}_S(\mathcal{G}^2_{\ell,W})$$

It follows easily that $\widehat{\mathsf{RAD}}_{\overline{S}}(\mathcal{G}^1_{\ell,W}) \leq W \cdot \frac{W}{\sqrt{m}}$ using Talagrand's lemma and the bound on the Rademacher complexity for linear functions. Similarly, it can be shown that $\widehat{\mathsf{RAD}}_S(\mathcal{G}^2_{\ell,W}) \leq \frac{WM}{\sqrt{m}}$—the functions in $\mathcal{G}^2_{\ell,W}$ can be viewed as linear functions with the vectors x_i replaced by $y_i x_i$, and observing that $\|y_i x_i\|_2 \leq |y_i| \|x_i\|_2 \leq M$. Using Theorem 3 and Theorem 5, this shows that the class of generalised linear models $\mathcal{G}_{W,u}$ can be learnt with running time polynomial in W, M, n and $\frac{1}{\epsilon}$ and with sample complexity polynomial in W, M and $\frac{1}{\epsilon}$. Notice that the sample complexity does not depend on the dimension n at all! Thus, these models and algorithms can be kernelised.

4.3 Proof of Theorem 3

Let us now complete the proof of Theorem 3. In order to prove the result we will use McDiarmid's inequality, which we state below without proof. A proof can be found in the lecture notes by Bartlett.[4]

Theorem 4 (McDiarmid's Inequality)**.** *Let $\mathcal{X}$ be some set and let $f : \mathcal{X}^m \to \mathbb{R}$ be a function such that for all i, there exists $c_i > 0$, such that for all $x_1, x_2, \ldots, x_m, x_i'$, the following holds:*

$$\left| f(x_1, \ldots, x_{i-1}, x_i, x_{i+1}, \ldots, x_m) - f(x_1, \ldots, x_{i-1}, x_i', x_{i+1}, \ldots, x_m) \right| \leq c_i$$

Let $X_1, X_2, \ldots, X_m$ be independent random variables taking values in $\mathcal{X}$. Then, for every $\epsilon > 0$, the following holds:

$$\mathbb{P}\left[f(X_1, \ldots, X_m) \geq \mathbb{E}\left[f(X_1, \ldots, X_m) \right] + \epsilon \right] \leq \exp\left(-\frac{2\epsilon^2}{\sum_{i=1}^m c_i^2} \right)$$

[4] Lecture notes available at `https://people.eecs.berkeley.edu/~bartlett/courses/281b-sp08/13.pdf`.

McDiarmid's inequality is a generalisation of the Chernoff-Hoeffding bound. For instance, if $\mathcal{X} = [a, b]$, using $f(x_1, \ldots, x_m) = \frac{1}{m}\sum_{i=1}^{m} x_i$ and $c_i = (b-a)/m$ gives the Chernoff-Hoeffding bound. McDiarmid's inequality shows that as long as no single variable has significant influence over the function f, then the random variable $f(X_1, \ldots, X_m)$ is strongly concentrated around its expectation.

Let $S = \{x_1, \ldots, x_m\}$ be a subset of X and $\mathcal{G}$ a set of functions from $X \to [0, 1]$. Let us now complete the proof of Theorem 3 by applying McDiarmid's inequality to the function,

$$\Phi(S) = \sup_{g \in \mathcal{G}} \left(\mathop{\mathbb{E}}_{x \sim D} \left[g(x) \right] - \widehat{\mathbb{E}}_S[g] \right)$$

Above we've used $\Phi(S)$ instead of $\Phi(x_1, \ldots, x_m)$ to keep the notation tidy, as Φ is symmetric. Let $S' = (S \setminus \{x_i\}) \cup \{x_i'\}$. Consider the following:

$$\begin{aligned}
\Phi(S) - \Phi(S') &= \sup_{g \in \mathcal{G}} \left(\mathop{\mathbb{E}}_{x \sim D} \left[g(x) \right] - \widehat{E}_S[g] \right) - \sup_{g \in \mathcal{G}} \left(\mathop{\mathbb{E}}_{x \sim D} \left[g(x) \right] - \widehat{\mathbb{E}}_{S'}[g] \right) \\
&\leq \frac{1}{m} \sup_{g \in G} (g(x_i) - g(x_i')) \leq \frac{1}{m}
\end{aligned}$$

Above, we used the fact that the difference between suprema is at most the supremum of the difference. As S and S' are completely symmetric, this shows that $|\Phi(S) - \Phi(S')| \leq \frac{1}{m}$. Thus, we may apply McDiarmid's inequality with all $c_i = \frac{1}{m}$ to obtain,

$$\mathbb{P}\left[\Phi(S) \geq \mathbb{E}\left[\Phi(S) \right] + \epsilon \right] \leq \exp\left(-2\epsilon^2 m \right) \tag{8}$$

Let us now compute $\mathbb{E}\left[\Phi(S)\right]$. Here, we'll use a trick similar to the one we used when proving the analogous result for VC dimension of introducing an independent draw $\overline{S} = \{\overline{x}_1, \ldots, \overline{x}_m\}$ from D and using the symmetric nature of S and $\overline{S}$.

$$\mathop{\mathbb{E}}_{S \sim D^m} \left[\Phi(S) \right] = \mathop{\mathbb{E}}_{S \sim D^m} \left[\sup_{g \in \mathcal{G}} \mathop{\mathbb{E}}_{x \sim D} \left[g(x) \right] - \widehat{\mathbb{E}}_S[g] \right] \tag{9}$$

We use the fact that $\mathop{\mathbb{E}}_{\overline{S} \sim D^m} \left[\widehat{\mathbb{E}}_{\overline{S}}[g] \right] = \mathop{\mathbb{E}}_{x \sim D} \left[g(x) \right]$ to obtain the following:

$$\mathop{\mathbb{E}}_{S \sim D^m} \left[\Phi(S) \right] = \mathop{\mathbb{E}}_{S \sim D^m} \left[\sup_{g \in \mathcal{G}} \mathop{\mathbb{E}}_{\overline{S} \sim D^m} \left[\widehat{\mathbb{E}}_{\overline{S}}[g] \right] - \widehat{E}_S[g] \right] \tag{10}$$

Pushing the supremum inside the expectation, we obtain,

$$\begin{aligned}
\mathop{\mathbb{E}}_{S \sim D^m} \left[\Phi(S) \right] &\leq \mathop{\mathbb{E}}_{S \sim D^m} \mathop{\mathbb{E}}_{S' \sim D^m} \left[\sup_{g \in \mathcal{G}} \left(\widehat{\mathbb{E}}_{\overline{S}}[g] - \widehat{\mathbb{E}}_S[g] \right) \right] && (11) \\
&= \mathop{\mathbb{E}}_{S \sim D^m} \mathop{\mathbb{E}}_{\overline{S} \sim D^m} \left[\sup_{g \in \mathcal{G}} \frac{1}{m} \sum_{i=1}^{m} (g(x_i) - g(\overline{x}_i)) \right] && (12)
\end{aligned}$$

Using the symmetric nature of S and $\overline{S}$, we may introduce Rademacher random variables σ_i and obtain the following,

$$\begin{aligned}
\mathop{\mathbb{E}}_{S \sim D^m} \left[\Phi(S) \right] &\leq \mathop{\mathbb{E}}_{S \sim D^m} \mathop{\mathbb{E}}_{\overline{S} \sim D^m} \mathop{\mathbb{E}}_{\boldsymbol{\sigma}} \left[\sup_{g \in \mathcal{G}} \frac{1}{m} \sum_{i-1}^{m} \sigma_i (g(x_i) - g(\overline{x}_i)) \right] && (13) \\
&\leq \mathop{\mathbb{E}}_{S \sim D^m} \mathop{\mathbb{E}}_{\boldsymbol{\sigma}} \left[\sup_{g \in \mathcal{G}} \sum_{i=1}^{m} \sigma_i g(x_i) \right] + \mathop{\mathbb{E}}_{\overline{S} \sim D^m} \mathop{\mathbb{E}}_{\boldsymbol{\sigma}} \left[\sup_{g \in \mathcal{G}} \frac{1}{m} \sum_{i=1}^{m} (-\sigma_i) g(\overline{x}_i) \right] && (14)
\end{aligned}$$

As $-\sigma_i$ is distributed identically to the σ_i, we conclude that,

$$\mathop{\mathbb{E}}_{S\sim D^m}\left[\Phi(S)\right] \leq 2\mathsf{RAD}_m(\mathcal{G}) \tag{15}$$

Using the above and setting $\epsilon = \sqrt{\frac{\log\frac{1}{\delta}}{2m}}$ completes the proof of Theorem 3.

References

Peter Auer, Mark Herbster, and Manfred K Warmuth. Exponentially many local minima for single neurons. *Advances in neural information processing systems*, pages 316–322, 1996.

Stephen Boyd and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

Sébastien Bubeck. *Convex Optimization: Algorithms and Complexity*. Foundations and Trends in Machine Learning. Now, 2015.

Mehryar Mohri, Afshin Rostamizadeh, and Ameet Talwalkar. *Foundations of machine learning*. MIT Press, 2012.

A. Nemirovski and D. Yudin. *Problem Complexity and Method Efficiency in Optimization*. Wiley Interscience, 1983.

A Projected Gradient Descent for Lipschitz functions

We will briefly describe an algorithm for minimising Lipschitz convex functions. Our treatment of convex optimisation is at best cursory and for further details the student may refer to any of the following references (Bubeck, 2015; Boyd and Vandenberghe, 2004; Nemirovski and Yudin, 1983).

Consider a function $f : \mathbb{R}^n \to \mathbb{R}$; f is convex if $f(\lambda x + (1-\lambda)x') \leq \lambda f(x) + (1-\lambda)f(x')$ for all $x, x' \in \mathbb{R}^n$ and for all $\lambda \in [0,1]$. Let $K \subseteq \mathbb{R}^n$ be a closed, bounded, convex set. We are interested in solving the following constrained optimisation problem: minimise $f(x)$ subject to $x \in K$.

We will see a proof that projected gradient descent (approximately) minimises f. In order to do so, let us define the projection operator, $\Pi_K(x) = \mathrm{argmin}_{y\in K} \|y - x\|_2$, *i.e.,* the projection operation finds a point in K closest to the point x (such a point always exists as K is closed). In general, projection is itself a convex optimisation problem, but for some common cases in machine learning such as projecting onto the ℓ_2- or ℓ_1-ball, this operation is very easy to perform.

Alg. 1 shows the iterative projected gradient descent procedure.

Algorithm 1 Projected Gradient Descent

Inputs: η, T
Pick $x_1 \in K$
for $t = 1, \ldots, T$ **do**
  $x'_{t+1} = x_t - \eta \nabla f(x_t)$
  $x_{t+1} = \Pi_K(x'_{t+1})$
end for
Output: $\frac{1}{T}\sum_{t=1}^{T} x_t$

We will prove the following result.

Theorem 5. *Suppose K is such that $\sup_{x,x' \in K} \left\|x - x'\right\|_2 \leq R$ and that $\sup_{x \in K} \left\|\nabla f(x)\right\|_2 \leq L$, then Alg. 1 run with $\eta = \frac{R}{L\sqrt{T}}$, outputs $\bar{x}$, such that,*

$$f(\bar{x}) \leq \min_{x \in K} f(x) + \frac{RL}{\sqrt{T}}$$

Proof. Let x_t denote the point at the t^{th} iteration of the gradient descent procedure. Let $x^* \in K$ be such that $x^* \in \operatorname{argmin}_{x \in K} f(x)$. Then consider, the following:

$$\begin{aligned} f(x_t) - f(x^*) &\leq \nabla f(x_t) \cdot (x_t - x^*) && \text{By convexity of } f \\ &= \frac{1}{\eta}(x_t - x'_{t+1}) \cdot (x_t - x^*) \\ &= \frac{1}{2\eta}\left(\left\|x_t - x^*\right\|_2^2 + \left\|x_t - x'_{t+1}\right\|_2^2 - \left\|x'_{t+1} - x_*\right\|_2^2\right) \\ &= \frac{1}{2\eta}\left(\left\|x_t - x^*\right\|_2^2 - \left\|x'_{t+1} - x^*\right\|_2^2\right) + \frac{\eta}{2}\left\|\nabla f(x_t)\right\|_2^2 \end{aligned}$$

We use the bound $\left\|\nabla f(x_t)\right\|_2 \leq L$ and the fact that $\left\|x'_{t+1} - x^*\right\|_2 \geq \left\|x_{t+1} - x^*\right\|_2$ (we will prove this fact later), to obtain,

$$f(x_t) - f(x^*) \leq \frac{1}{2\eta}\left(\left\|x_t - x^*\right\|_2^2 - \left\|x_{t+1} - x^*\right\|_2^2\right) + \frac{\eta}{2}L^2 \tag{16}$$

Let us first complete the proof before proving the claim that $\left\|x'_{t+1} - x^*\right\|_2 \geq \left\|x_{t+1} - x^*\right\|_2$. By convexity of f, it follows that $f\left(\frac{1}{T}\sum_{t=1}^T x_t\right) \leq \frac{1}{T}\sum_{t=1}^T f(x_t)$. We can average Eq. (16) over $t = 1, \ldots, T$ to obtain,

$$\begin{aligned} f\left(\frac{1}{T}\sum_{t=1}^T x_t\right) - f(x^*) &\leq \frac{1}{T}\sum_{t=1}^T (f(x_t) - f(x^*)) \\ &\leq \frac{1}{2T\eta}\left\|x_1 - x^*\right\|_2^2 + \frac{\eta}{2}L^2 \quad \text{Dropping the negative term } -\frac{1}{2T\eta}\left\|x_{T+1} - x^*\right\|_2^2. \\ &\leq \frac{R^2}{2T\eta} + \frac{\eta L^2}{2} \end{aligned}$$

Setting $\eta = \frac{R}{L\sqrt{T}}$ completes the proof.

Proof of claim that projecting decreases distance: Let us now prove our claim that projecting a point onto K only reduces the distance to any point in K. Let $x' \in \mathbb{R}^n$, $x = \Pi_K(x')$ and let $z \in K$. We will show that for K closed and convex, it must be the case that $\left\|z - x\right\|_2 \leq \left\|z - x'\right\|_2$. Consider the following:

$$\begin{aligned} \left\|z - x'\right\|_2^2 &= \left\|z - x + x - x'\right\|_2^2 \\ &= \left\|z - x\right\|_2^2 + \left\|x - x'\right\|_2^2 - 2(z - x) \cdot (x' - x) \end{aligned}$$

Now if, $(z - x) \cdot (x' - x) \leq 0$ we are done. Suppose for the sake of contradiction that $(z - x) \cdot (x' - x) > 0$. We will establish that x cannot be the projection of x' onto K. Consider the following:

$$\left\|\lambda z + (1 - \lambda)x - x'\right\|_2^2 = \left\|x - x'\right\|_2^2 + \lambda^2\left\|z - x\right\|_2^2 - 2\lambda(z - x) \cdot (x' - x)$$

Notice that this implies for $\lambda \in \left(0, \min\left\{1, \frac{(z-x)\cdot(x'-x)}{\left\|z-x\right\|_2^2}\right\}\right)$, $\left\|\lambda z + (1 - \lambda)x - x'\right\|_2 < \left\|x - x'\right\|_2$. As K is a convex set and $x, z \in K$, $\lambda z + (1 - \lambda)x \in K$, contradicting the claim that $\Pi_K(x') = x$. Thus, it must be the case that $\left\|z - x'\right\|_2 \geq \left\|z - x\right\|_2$. □