Consistency, Robustness and Sparsity for Learning Algorithms

Konsistenz, Robustheit und Dünnbesetztheit von Lern-Algorithmen

zur Erlangung des Doktorgrades

Dr. rer. nat.

im Studiengang Mathematik

am Department Mathematik der
Friedrich-Alexander-Universität Erlangen-Nürnberg

vorgelegt am **4.Juli 2023**

von **Tim Roith**

Prüfer:	Martin Burger
Betreuer:	M.Sc. C

Für eine besondere Person,
ohne sie hätte ich es nie geschafft.

Acknowledgement

I would like to thank my supervisors for helping me finish this thesis and supporting me with their profound knowledge..

Contents

List of Figures

Preface

This work is structured into two main parts, Part I the presentation and explanation of the topics and results presented in Part II, the peer-reviewed articles.

Part I: Exposition	Part II: Prints
??: ??	
Chapter 1: Consistent Semi-Supervised Learning on Sparse Graphs	**????**
Chapter 2: Robust and Sparse Supervised Learning	**????**

Part I consists of three chapters, of which the first explains the paradigms, *unsupervised*, *semi-supervised* and *supervised* learning. The other chapters are the split up thematically, concerning the topics semi-supervised and supervised learning respectively. In each of these chapters a short introduction provides the necessary framework allowing us to explain the main contributions. The following publications are reprinted in Part II:

[LIP-I] T. Roith and L. Bungert. "Continuum limit of Lipschitz learning on graphs." In: *Foundations of Computational Mathematics* (2022), pp. 1–39.

[LIP-II] L. Bungert, J. Calder, and T. Roith. "Uniform convergence rates for Lipschitz learning on graphs." In: *IMA Journal of Numerical Analysis* (Sept. 2022). DOI: `10.1093/imanum/drac048`.

[CLIP] L. Bungert et al. "CLIP: Cheap Lipschitz training of neural networks." In: *Scale Space and Variational Methods in Computer Vision: 8th International Conference, SSVM 2021, Virtual Event, May 16–20, 2021, Proceedings*. Springer. 2021, pp. 307–319.

[BREG-I] L. Bungert et al. "A bregman learning framework for sparse neural networks." In: *Journal of Machine Learning Research* 23.192 (2022), pp. 1–43.

[FNO] S. Kabri et al. "Resolution-Invariant Image Classification based on Fourier Neural Operators." In: *Scale Space and Variational Methods in Computer Vision: 9th International Conference, SSVM 2023, Proceedings*. Springer. 2023, pp. 307–319.

The following two works that are not part of this thesis but provide an additional insight.

[LIP-III] L. Bungert, J. Calder, and T. Roith. *Ratio convergence rates for Euclidean first-passage percolation: Applications to the graph infinity Laplacian.* 2022. arXiv: 2210.09023 [math.PR].

[BREG-II] L. Bungert et al. "Neural Architecture Search via Bregman Iterations." In: (2021). arXiv: 2106.02479 [cs.LG].

TR's Contribution

Here we list TR's contribution to the publications included in the thesis.

[LIP-I]: This work builds upon the findings in TR's master thesis [Roi22]. It is however important to note that the results constitute a significant extension and are conceptually stronger than the ones in [Roi22], see Section 1.4.1. TR adapted the continuum limit framework to the L^∞ case, worked out most of the proofs and wrote a significant part of the paper. In collaboration with LB, he identified the crucial domain assumptions that allow to work on non-convex domains and proved convergence for approximate boundary conditions.

[LIP-II]: In collaboration with LB, TR worked on the convergence proofs building upon the ideas of JC. He contributed to both the numeric and the analysis conducted in the paper.

[CLIP]: TR worked out the main algorithm proposed in the paper together with LB, based on LB's idea. Together with LS and RR he conducted the numerical examples and also wrote most of the source code. Furthermore, he wrote large parts of the paper.

[BREG-I]: TR expanded LB's ideas of employing Bregman iteration for sparse training. Together with MB and LB he worked out the convergence analysis of stochastic Bregman iterations. Here, he also proposed a profound sparse initialization strategy. Furthermore, he conducted the numerical examples and wrote most of the source code.

[FNO]: This work is based on SK's masters thesis, employing the initial ideas of MB for resolution invariance with FNOs. In the paper TR worked out the proofs for well-definedness and Fréchet-differentiability, together with SK. He wrote large parts of the paper and the source code. Here, he conducted the numerical studies in collaboration with SK.

Part I.

Exposition

Chapter 1

Consistent Semi-Supervised Learning on Sparse Graphs

This chapter considers the consistency of graph-based semi-supervised learning methods. We contextualize the topics provided in the prints [LIP-I; LIP-II] together with additional input from the pre-print [LIP-III]. In Section 1.1 we introduce the concrete setting and task. In Section 1.2 we consider the p Laplacian both in the continuum and the graph setting. This is the cornerstone for topics connected to the ∞ Laplacian and Lipschitz extensions in Section 1.3, the main points of interest in this chapter. With the necessary tools and background we are then able to present the main contributions on consistency of Lipschitz learning in the infinite data limit. In Section 1.4.1 we first consider Γ-convergence of discrete L^∞ functionals to their continuum counterpart in [LIP-I]. After that we explain the main results of [LIP-II] in Section 1.4.2, which considers uniform convergence of graph infinity harmonic functions to their continuum counterpart. Finally, we shortly explore possible extension of the results in Section 1.4.2 to the non-deterministic percolation setting of [LIP-III].

Section 1.1: Graph-based SSL and Consistency

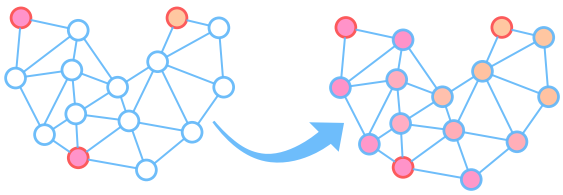
Section 1.2: The p Laplacian	Section 1.3 Lipschitz Extensions
Section 1.2.1: Continuum Setting	Section 1.3.1: Continuum Setting
Section 1.2.2: Graph Setting	Section 1.3.2: Graph Setting

Section 1.4: Main Contribution: Continuum Limits

Section 1.4.1: [LIP-I]	Section 1.4.2: [LIP-II]

link to non nameref dings

1.1. Graph-based SSL and Consistency



In the semi-supervised learning setting of **??**, we are given a finite set $\Omega_n \subset \tilde{\Omega}$ consisting of n points, where $\tilde{\Omega}$ denotes the input space. We assume that a non-empty subset $\mathcal{O}_n \subset \Omega_n$ is labeled, i.e., we are given a labeling function $\mathbf{g} : \mathcal{O}_n \to \mathbb{R}$. From now on we denote functions acting on a discrete set Ω_n using bold symbols to distinguish them from functions acting onn the continuum $\tilde{\Omega}$. In particular, the space of functions on Ω_n can simply be identified with $\mathbb{R}^n$, since $\{\mathbf{u}_n : \Omega \to \mathbb{R}\} \sim \mathbb{R}^n$. The semi-supervised learning problem consists in finding an extension of $\mathbf{g}$ from $\mathcal{O}_n$ to the whole set Ω_n, i.e.

$$\begin{aligned} &\text{find } \mathbf{u} : \Omega_n \to \mathbb{R}, \\ &\text{such that } \mathbf{u}(x) = \mathbf{g}(x) \text{ for all } x \in \mathcal{O}_n. \end{aligned} \tag{SSL}$$

In order to obtain meaningful solutions, one usually incorporates the *smoothness assumption* [ST14] which can be informally stated as follows:

"*Points that are close together are more likely to share a similar label.*"

1.1.1. Weighted Graphs

In order to employ this assumption we need a notion of "closeness" on the set Ω_n, for which we introduce weighted graphs. Namely we define a weighting function w that allows to compare points in Ω_n and therefore induces the desired notion.

Definition 1.1 (Weighted Graphs). For a finite set Ω_n and a weight function $w : V \times V \to \mathbb{R}$, the tuple (V, w) is called a *weighted graph*.

Remark 1.2. Typically, a graph is defined as a pair (Ω_n, E) where E denotes the set of edges. Here, one has two cases:

- *Undirected graph:* $E = \{\{x, y\} : \text{there is an edge between } x \in \Omega_n \text{ and } y \in \Omega_n\}$, i.e., $E \subset 2^{\Omega}_n$ and each edge is undirected.
- *Directed graph:* $E = \{(x, y) : \text{there is an edge from } x \text{ to } y\}$, i.e., $E \subset \Omega_n \times \Omega_n$ and each edge is directed.

Additionally, one then considers a weight function $W : E \to \mathbb{R}$ assigning a weight to each edge, and then defines the triple (Ω_n, E, W) as a weighted graph. However, we note that all this information can be represented much more elegantly by a weight function $w : \Omega_n \times \Omega_n \to \mathbb{R}$. A directed edge set $E \subset \Omega_n \times \Omega_n$ can be equivalently expressed by a weight function $w : \Omega_n \times \Omega_n \to \mathbb{R}$ where $w(x, y) > 0$ if and only if $(x, y) \in E$, a weighting $W : E \to \mathbb{R}$ naturally transfers to w. In the case of an undirected graph, one simply requires the weight function to be symmetric, i.e., $w(x, y) = w(y, x)$ for all

$x, y \in \Omega_n$. Furthermore, in the above definition the set Ω_n does not enter the definition up to ordering of its $n \in \mathbb{N}$ elements. Therefore, a graph could be entirely represented by a weight function $w : \{1, \dots, n\} \times \{1, \dots, n\} \to \mathbb{R}$. However, since the definition of w will incorporate information about points $\Omega_n \subset \mathbb{R}^d$ we use the notation (Ω_n, w_n) for graphs in the following. △

1.1.2. The Graph Scale

In most of our applications the data Ω_n is given as a subset of $\mathbb{R}^n$ and in fact we are interested in the limit $n \to \infty$, where Ω_n fills out a domain $\tilde{\Omega} \subset \mathbb{R}^d$. Furthermore, in the continuum we are interested in *local* operators incorporating changes of functions $u : \tilde{\Omega} \to \mathbb{R}$ at an infinitesimal small scale. Since interactions on a graph are inherently non-local in the Euclidean sense, we need to localize the interaction on the graph in the limit $n \to \infty$. A popular choice that guarantees this behavior is to set

$$w(x, y) = \eta_\varepsilon(|x - y|)$$

where $\eta_\varepsilon : [0, \infty) \to [0, \infty]$ is a kernel function depending on a scaling parameter $\varepsilon \in \mathbb{R}^+$. The parameter ε is also referred to as the *graph scale* and informally speaking determines the scale of the graph interactions. I.e., the smaller ε the smaller the interaction radius of points in Ω_n should be. In our specific setting of L^∞ problems it is typical to choose

$$\eta_\varepsilon(\cdot) = \frac{1}{c_\eta\, \varepsilon} \eta\left(\frac{\cdot}{\varepsilon}\right)$$

where η is a non-increasing kernel and c_η is a constant depending on the kernel. Typical examples of kernels include

- (constant weights) $\eta(t) = 1_{[0,1]}(t)$,
- (exponential weights) $\eta(t) = \exp(-t^2/(2\sigma^2))1_{[0,1]}(t)$,
- (non-integrable weights) $\eta(t) = \frac{1}{t^p} 1_{[0,1]}(t)$ with $p \in (0, 1]$.

Remark 1.3. In the continuum limit one needs to consider the value

$$\sigma_\eta = \sup_{t \in \mathbb{R}^+} t\, \eta(t),$$

which is assumed to be finite. Considering graph problems in L^p for $p < \infty$ the corresponding value is given as

$$\sigma_\eta^{(p)} = \int_{\mathbb{R}^+} \eta(t)\, t^{d+p} dt$$

which was first employed in [GS15] for $p = 1$ and then in [ST19] for general $p < \infty$. We see that $\sigma_\eta^{(p)}$ degenerates to σ_η in the limit $p \to \infty$.
In [LIP-I] we choose $c_\eta = 1$ and therefore σ_η then appears in the limit functional. In [LIP-II] we rescale the kernel, i.e., $c_\eta = \sigma_\eta$ which allows to work with unrescaled operators in the limit. △

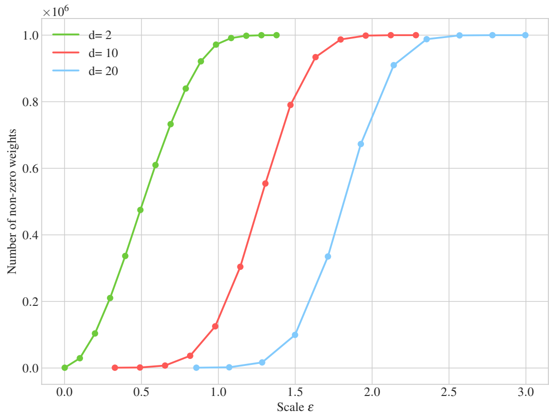


Figure 1.1.: This that, Gecko hat

Remark 1.4. In order to obtain continuum limits in the case $p < \infty$ on usually employs a factor of $1/\varepsilon^{p+d}$ in front of the graph weights [GS15; ST19]. In Section 1.3.2 we see that this factor degenerates to $1/\varepsilon$ in the case $p \to \infty$. The intuition here, is that problems in L^∞ do not respect mass in a quantitative but rather a qualitative manner. Therefore, factors like ε^d which appear because of integrals in $\mathbb{R}^d$ do not contribute for $p = \infty$. △

An important practical aspect connected to the graph scale is the sparsity of the graph, which is given by the numbers of non-zero elements in the weight matrix $W \in \mathbb{R}^{n\times n}$

$$W_{ij} := w(x_i, x_j) \quad x_i, x_j \in \Omega_n$$

where we assume an ordering $\Omega_n = \{x_1, \dots, x_n\}$. In order to have a computationally feasible problem this matrix should have very few non-zero elements. Assuming that the kernel η has compact support, w.l.o.g. $\operatorname{supp}(\eta) \subset [0, 1]$ we observe that the sparsity is directly influenced by the graph scale ε. Namely, $|x_i - x_j| > \varepsilon \Rightarrow W_{ij} = 0$. Therefore, from a practical point of view ε should be chosen relatively small. However, showing consistency of graph-based SSl algorithms often requires scaling assumptions that permit the desired length scales [GS15; ST19; Cal19]. In Section 1.4.1 we give concrete examples of such scaling assumptions. One of the main goals of [LIP-I; LIP-II] was to show convergence and rates at the smallest possible length scale, which was indeed achieved.

Remark 1.5. In many practical applications graph weights connecting all points within ε-ball of Euclidean distance are inferior to knn graphs [Cal+20; FCL19; CT22]. While most consistency results employ the ε-ball setting, recently, the authors in [CT22] were able to show convergence rates in the knn setting. In this thesis we focus on the ε-ball setting, however it is an interesting open question to transfer the results of [LIP-I; LIP-II; LIP-III] to the knn setting. △

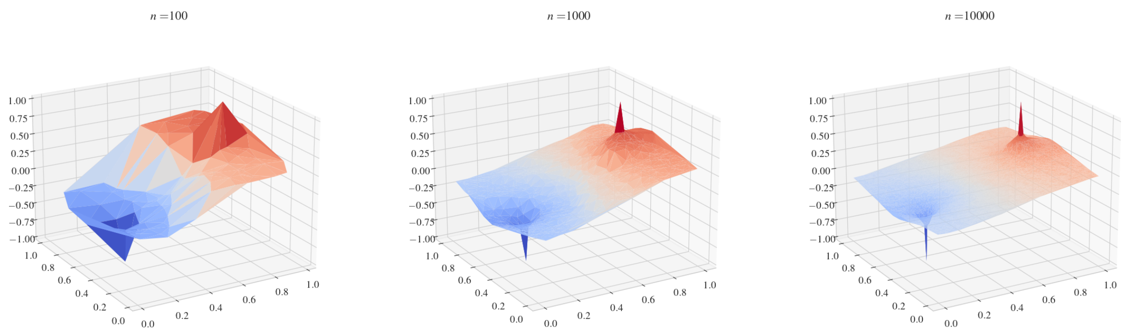


Figure 1.2.: Solution to the Laplacian Learning problem ($p = 2$) for different number of data points $n \in \{100, 1000, 10000\}$.

1.1.3. Consistency for Graph-based SSL

The archetype of the methods we consider in the following is so-called *Laplacian learning*, which had one of its first appearances in [ZGL03]. The associated problem was given as

$$\begin{aligned} &\min_{\mathbf{u}:\Omega_n\to\mathbb{R}} \sum_{x,y\in\Omega_n} w_n(x,y)^2 \left(\mathbf{u}(y) - \mathbf{u}(x)\right)^2 \\ &\text{subject to } \mathbf{u}(x) = \mathbf{g}(x) \text{ for all } x \in \mathcal{O}_n. \end{aligned} \tag{1.1}$$

The intuitive idea behind this method is that it minimizes a discrete approximation of the Dirichlet energy and should therefore enforce some a certain kind of "smoothness" of the solution $\mathbf{u}$. While there are many examples where this simple concept does in fact yield good solutions [], it has been observed that solutions tend to degenerate for large amounts of data, whenever $d \geq 2$ [NSZ09; AL11; El +16]. We illustrate this phenomenon in Example 1.6.

> **Example 1.6.** We visualize this behavior in Fig. 1.2, where we sample different number of points n in $(0,1)^2$ and keep a fixed constraint on $\mathcal{O}_n = \{o_1, o_2\} = \{(0.2, 0.5), (0.8, 0.5)\}$ with $\mathbf{g}(o_1) = -1, \mathbf{g}(o_2) = 1$. We observe that for increasing n the solution $\mathbf{u}$ of Eq. (1.1) tends to be a constant equal to the average of the given constraints and spike at the given constraints.

This observation motivates the driving question of this chapter.

Are Semi-Supervised Learning algorithms consistent in the infinite data limit?

In our case "consistency" roughly ask the discrete solutions to converge to the continuum counterparts they are motivated by, in the data limit $n \to \infty$.

Consistency for the p-Laplacian In [ZS05] it was proposed to generalize the idea of the graph Dirichlet energy in Eq. (1.1) to arbitrary $1 \leq p < \infty$ motivated by the continuum analogous. See Section 1.2.2 for a formal problem setup. As observed in [NSZ09; AL11; El +16] the question of consistency is now connected to relation between

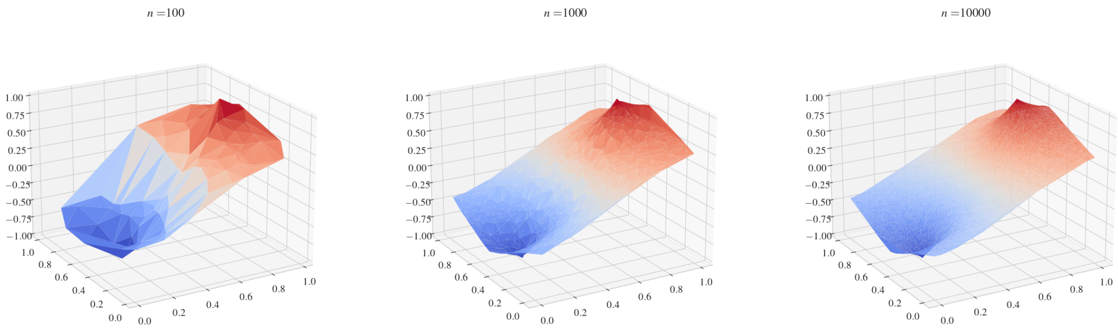


Figure 1.3.: Solution to the Laplacian Learning problem ($p = \infty$) for different number of data points $n \in \{100, 1000, 10000\}$. The setup is otherwise copied from **??**

p and d, where one has the regimes whether $p < d$ or $p = d$ and probably the most relevant case $p > d$.

The p-Dirichlet problem in the continuum can be formulated as a variational problem in $W^{1,p}$. However, it is important to remember that in our setting we most likely consider pointwise constraints on a discrete set $\mathcal{O}$ even in the continuum. In order to impose pointwise constraints, one requires that functions in $W^{1,p}$ exhibit some continuity properties. By the Sobolev embedding theorem, this is the case when $p > d$, [AF03]. This leads to a first qualitative intuition that consistency can only be achieved in the case $p > d$. The situation it is in fact more subtle, which is reviewed in Section 1.4.1.

It is important to note that p-harmonic functions are actually more regular also in the case $p \leq d$, beyond the implication of the Sobolev embedding theorem. However, this regularity does not help for the continuum limits [NSZ09; AL11; El +16].

Sending p to infinity Since the assumption that $p > d$ is crucial for asymptotic consistency, a natural idea is to send p to ∞ and analyze the corresponding limit problem. This yields the so-called *Lipschitz learning* task [vLB04; Kyn+15], which is the main point of interest in this chapter. We formalize this problem in Section 1.3. In Fig. 1.3 we observe that for $p = \infty$ we indeed obtain smoother solutions, compared to Fig. 1.2. In this regard Lipschitz learning seems promising to overcome the consistency issue.

However, as already noticed in [El +16] being a pure L^∞ problem, Lipschitz learning does not respect the density of data in any way. Namely, the method is exclusively distance based. This behavior is a drawback in many machine learning application. However, by carefully rescaling the graph weights one obtains a data sensitive problem, see [Cal19] and Section 1.4.1.

The scaling parameter

1.2. The p-Laplacian: continuum and graph

In **??** we explore concepts that borrow ideas from the theory of partial differential equations, and in particular the p-Laplace equation. In this section we briefly review important ideas and results and provide necessary preliminaries for the following sections.

1.2.1. The p-Laplacian: continuum setting

We follow the exposition in [Lin17]. Let $\Omega \subset \mathbb{R}^d$ be a bounded domain, then we consider the p-Dirichlet energy for functions $u \in W^{1,p}(\Omega)$,

$$\mathcal{E}_p(u) := \int_\Omega |\Delta_p u|^p \, dx, \tag{1.2}$$

and the associated variational problem.

Problem 1.7 (Variational Formulation). For $p \in (1, \infty)$ and $V \subset W^{1,p}(\Omega)$ find $u \in W^{1,p}(\Omega)$ such that

$$\mathcal{E}_p(u) \leq \mathcal{E}_p(v)$$

for all v, such that $(u - v) \in W_0^{1,p}$.

If $u \in V$ is a minimizer of the above problem, then its first variation must vanish, i.e., for all $\phi \in C_0^\infty(\Omega)$ one has

$$\int_\Omega \langle |\Delta_p u|^p \, \nabla u, \nabla \phi \rangle \, dx = 0. \tag{1.3}$$

A function $u \in V$ satisfying Eq. (1.3) is called a *weak solution* of the p-Laplace equation. In fact, if u is smooth enough one has that

$$\Delta_p u := \operatorname{div}(|\Delta_p u|^{p-2} \, \nabla u) = 0 \tag{1.4}$$

where Δ_p is called the p-Laplacian. For most of our applications we want to prescribe boundary conditions on $\partial\Omega$. For a given function $g \in W^{1,p}(\Omega)$ we therefore consider the set $V_g := \{u \in W^{1,p}(\Omega) : u - g \in W_0^{1,p}(\Omega)\}$ for which we have the following result.

Theorem 1.8 (Existence and Uniqueness). For $p \in (1, \infty)$ and $g \in W^{1,p}(\Omega)$ there exists a unique minimizer $u \in V_g$ of the p-Dirichlet energy, i.e.,

$$\operatorname{argmin}_{u \in V_g} \mathcal{E}_p(u) = u.$$

Moreover, u is a weak solution of the p-Laplace equation and there exists a function $\tilde{u} \in C(\Omega)$ such that $u = \tilde{u}$ a.e. in Ω. If $g \in C(\Omega)$ and Ω is sufficiently smooth, then $\tilde{u}|_{\partial\Omega} = g|_{\partial\Omega}$.

Proof. The proof can be found in [Lin17, Thm. 2.16]. □

Local minimization property

1.2.2. Laplacian Learning

A natural extension of this problem is obtained by substituting the target functional by

$$\mathbf{E}_p^{w_n}(\mathbf{u}) := \sum_{x,y\in\Omega_n} w_n(x,y)^p \left|\mathbf{u}(y) - \mathbf{u}(x)\right|^p,$$

which we refer to as the graph p-Dirichlet energy. Indeed, we notice structural similarities to the p-Dirichlet energy $\mathcal{E}_p$ in Eq. (1.2), replacing the integral by a finite sum and derivatives by weighted finite differences

$$w_n(x,y)^p \left|\mathbf{u}(y) - \mathbf{u}(x)\right|^p.$$

This naturally leads to the following minimization problem.

Problem 1.9 (Graph Energy Minimization). Given a weighted graph (Ω_n, w_n) and a labeling function $\mathbf{g} : \mathcal{O}_n \to \mathbb{R}$, for $\mathcal{O}_n \subset \Omega_n$ we consider the problem

$$\min_{\mathbf{u}:\Omega_n\to\mathbb{R}} J^{w_n,p}(\mathbf{u}) \text{ subject to } \mathbf{u}(x) = \mathbf{g}(x) \text{ for all } x \in \mathcal{O}_n.$$

Since every function $\mathbf{u} : \Omega_n \to \mathbb{R}$ can be identified with a vector $\mathbf{u} \in \mathbb{R}^n$, the above problem is in fact an optimization problem in $\mathbb{R}^n$. Therefore one can prove unique existence of solutions via standard methods.

Theorem 1.10 (Existence and Uniqueness). Problem 1.9 admits a unique solution $\mathbf{u} : \Omega_n \to \mathbb{R}$.

Proof. This and that, a gecko with a hat. □

While Laplacian learning

Graph Laplacian In the continuum case one considers the Euler–Lagrange equation for the functional $\mathcal{E}_p$, which yields the p-Laplacian, see Section 1.2. Analogously, the optimality conditions for the graph p-Dirichlet energy $\mathbf{E}_p^{w_n}$ yield

$$\Delta_p^{w_n}\mathbf{u}(x) := \sum_{y\in\Omega_n} w_n(x,y)^p \left|\mathbf{u}(y) - \mathbf{u}(x)\right|^{p-2} (\mathbf{u}(y) - \mathbf{u}(x)) = 0, \text{ for all } x \in \Omega_n,$$

where $\Delta_p^{w_n}$ is referred to as thegraph p-Laplacian operator. This yields the graph p-Laplacian problem.

Problem 1.11 (Graph p-Laplacian). Given a weighted graph (Ω_n, w_n) and a labeling function $\mathbf{g} : \mathcal{O}_n \to \mathbb{R}$ with $\mathcal{O}_n \subset \Omega_n$, find a function $\mathbf{u} : \Omega_n \to \mathbb{R}$ such that

$$\begin{aligned} \Delta_p^{w_n} \mathbf{u} = 0, \text{ in } \Omega_n \setminus \mathcal{O}_n, \\ \mathbf{u} = \mathbf{g} \text{ on } \mathcal{O}_n. \end{aligned}$$

Since the functional $\mathbf{E}_p^{w_n}$ has a unique minimizer subject to the constraints given by $\mathbf{g}$ and the graph p-Laplacian is derived via optimality conditions, one expects that the Problem 1.9 and Problem 1.11 are equivalent. This is formulated in the following theorem.

Theorem 1.12 (Existence and Uniqueness). There exists a unique solution $\mathbf{u} : \Omega_n \to \mathbb{R}$ to Problem 1.11, which also is the unique minimizer of Problem 1.9.

Proof. This and that a gecko with a hat. □

1.3. Lipschitz extensions and the infinity Laplacian: continuum and graph

1.3.1. The continuum setting

This chapter studies the limit $p \to \infty$ of the p-Laplace equation. We first recall, that for $u \in W^{1,\infty}(\Omega)$ we have that

$$\lim_{p\to\infty} \mathcal{E}_p(u)^{1/p} = \operatorname*{ess\,sup}_{x\in\Omega} |\nabla u(x)| =: \mathcal{E}_\infty(u),$$

see [Jen93]. The functional $\mathcal{E}_\infty$ is weak*-lower semicontinuous over $W^{1,\infty}(\Omega)$, (see e.g. [BJW01, Thm. 2.6]). In the classical theory developed by Jensen in [Jen93] one considers the following problem, which is tries to "minimize the *sup-norm* of the gradient" as described by Jensen.

Problem 1.13 (Variational gradient-sup problem). For an open domain $\Omega \subset \mathbb{R}^d$ find a function $u \in W^{1,\infty}(\Omega)$ such that

$$\|\nabla u\|_\infty \leq \|\nabla v\|_\infty \text{ for every } v, \text{s.t. } (u - v) \in W_0^{1,\infty}.$$

The above problem becomes more relevant when we additionally impose boundary values $g : \partial\Omega \to \mathbb{R}$. Here, [Jen93] draws the connection to so-called *Lipschitz extensions*, which are the driving concept in this section. We introduce a more general viewpoint—that does not require the notion of a gradient—later on, but first introduce the variational problem.

The intrinsic metric and the Lipschitz constant. As noticed in [Jen93] working with the Lipschitz constant and the sup-norm of the gradient requires a careful treatment of the distance measurement. Let $\tilde{\Omega}$ be a set and let d be a semi-metric on $\tilde{\Omega}$, that is d fulfills the requirement of a metric up to triangle inequality. Then we define the Lipschitz constant of a function $u : V \to \mathbb{R}$ on a subset $V \subset \tilde{\Omega}$ as

$$\operatorname{Lip}_d(u; V) := \sup_{x,y \in V, x \neq y} \frac{|u(x) - u(y)|}{d(x,y)}.$$

If d denotes the Euclidean distance we use omit the subscript. i.e. $\operatorname{Lip}_d = \operatorname{Lip}$. Additionally, we can introduce the space of Lipschitz functions $\operatorname{Lip}_d(V)$ on V via $u \in \operatorname{Lip}_d(V) \Leftrightarrow \operatorname{Lip}_d(u; V) < \infty$.

Remark 1.14 (Lipschitz and Sobolev functions). If $\Omega \subset \mathbb{R}^d$ is sufficiently regular, e.g., it has Lipschitz boundary then we have that

$$\operatorname{Lip}(\Omega) = W^{1,\infty}(\Omega),$$

where this identity is of course to be understood in the sense of equivalence classes in Ł^p spaces. We refer to [**evansgariepy**] for a proof of this result. △

The above remark already relates Lipschitz with $W^{1,\infty}$ functions. Often however, we need a quantitative comparison between the Lipschitz constant and the sup-norm of the gradient of a function. Here, it is essential which distance measure is chosen for the Lipschitz constant. For an open domain $\Omega \subset \mathbb{R}^d$ we have the inequality

$$\|\nabla u\|_\infty \leq \sup_{x \neq y} \frac{|u(x) - u(y)|}{|x - y|}$$

which can be proven via the definition of the gradient. For the reverse inquality, one has to respect the geometry of the domain, namely for $x, y \in \Omega$ we have that

$$|u(x) - u(y)| \leq \|\nabla u\|_\infty \, d_\Omega(x, y) \tag{1.5}$$

see, e.g., [BB11, Prop9.3, Rem. 7], where

$$d_\Omega(x, y) = \inf \left\{ \int_0^1 |\dot{\gamma}(t)| \, dt : \gamma \in C^1([0,1], \Omega) \text{ with } \gamma(0) = x, \gamma(1) = y \right\}$$

denotes the *geodesic distance* on Ω. If Ω is convex, we have that $d_\Omega(x, y) = |x - y|$ for every $x, y \in \Omega$ and therefore Eq. (1.5) yields $\operatorname{Lip}(u) = \|\nabla u\|_\infty$. However, this situation changes for non-convex domains, see Example 1.15. Additionally it is often necessary to define a distance measure on the closure of $\Omega \subset \mathbb{R}^d$. In order to have a geodesic on $\overline{\Omega}$ one can simply consider $d_{\overline{\Omega}}$, see e.g. [**unif**], which then yields the length space $(\overline{\Omega}, d_{\overline{\Omega}})$. In the classical theory developed in [Jen93] one alternatively considers

$$\tilde{d}_{\overline{\Omega}}(x, y) := \liminf_{(\tilde{x}, \tilde{y}) \to (x,y)} d_\Omega(\tilde{x}, \tilde{y}).$$

The differences between these notation are demonstrated in the following example. Also note, that $\tilde{d}_{\overline{\Omega}}$ is only a semi-metric on $\overline{\Omega}$ since it is lacking a triangle inequality.

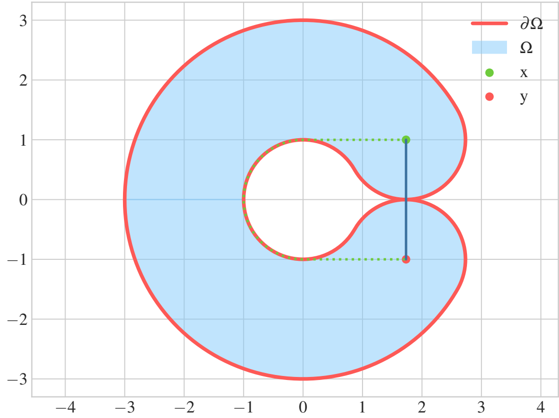


Figure 1.4.: The domain in Example 1.15.

Example 1.15. For $I = [-\pi, c] \cup [c, \pi]$ with $c = \pi/6$ we consider the domain

$$\bigcup_{\theta \in I} B_1\left((\cos(\theta), \sin(\theta)\right)$$

which is visualized in Fig. 1.4 and the points $x = (2\,c, 1), y = (2\,c, -1)$. The line segment between x and y contains the point $z = (2\,c, 0)$, however $z \notin \Omega$. One can show that the geodesic has the length $d_\Omega(x, y) = 4\,\cos(\pi/6) + \pi \approx 6.606$ which is the length of the dotted path in Fig. 1.4. However, we observe that

$$\overline{\Omega} = \bigcup_{\theta \in I} \overline{B_1\left((\cos(\theta), \sin(\theta)\right)}$$

and in particular $z \in \overline{B_1\left((c, c)\right)}$, therefore $d_{\overline{\Omega}}(x, y) = 2$.

Solutions to the gradient-sup problem Before generalizing the theory of Lipschitz extension to arbitrary metric spaces, we first note, that one can explicitly construct solutions. Namely, for given $g \in \mathrm{Lip}(\partial\Omega)$ the functions

$$\begin{aligned} \overline{g}(x) &:= \inf_{y \in \partial\Omega} g(y) + \mathrm{Lip}_{\tilde{d}_{\overline{\Omega}}}(g; \partial\Omega) \cdot \tilde{d}_{\overline{\Omega}}(x, y) \\ \underline{g}(x) &:= \sup_{y \in \partial\Omega} g(y) - \mathrm{Lip}_{\tilde{d}_{\overline{\Omega}}}(g; \partial\Omega) \cdot \tilde{d}_{\overline{\Omega}}(x, y) \end{aligned} \tag{1.6}$$

are solutions to the gradient-sup problem Problem 1.13 that coincide with g on $\partial\Omega$, see [Jen93, Th. 1.8].

Remark 1.16. The same concept of constructing solutions is applied in the following sections in a more abstract setting. These solutions are then called Whitney and McShane or respectively maximal and minimal extensions, see Lemma 1.19. △

One easily observes that there are cases where $\overline{g} \neq \underline{g}$ and therefore the sup-gradient problem does not admit a unique solution. A concrete example, to showcase this phenomena is given in [Jen93, p. 53].

Lipschitz extensions in metric spaces The problem considered in the last section was motivated by a variational problem for $\mathcal{E}_\infty(u) = \|\nabla u\|_\infty$. However, the theory of Lipschitz extensions provides a more general framework. Namely, here we do not assume that Ω is a subset of $\mathbb{R}^d$ and rather consider a metric space $(\tilde{\Omega}, d)$ with $\Omega \subset \tilde{\Omega}$.

Remark 1.17. For applications within this thesis we have that $\Omega \subset \mathbb{R}^d$ is an open bounded domain and then consider $\tilde{\Omega} := \overline{\Omega}$, i.e., the closure of Ω within the topology induced by the Euclidean distance. In this abstract setting however, we use an abstract space $\tilde{\Omega}$ while still being close notation wise. △

A result originally due to Kierszbraun [Kir34] states that for two Hilbert spaces H_1, H_2, a subset $\mathcal{O} \subset H_1$ and a function $g : \mathcal{O} \to H_1$ there exists a function $u : H_1 \to H_2$ such that

$$\begin{aligned} u &= g \text{ on } U, \\ \operatorname{Lip}(u; H_1) &= \operatorname{Lip}(g; \mathcal{O}). \end{aligned}$$

Here, the metrics for the respective Lipschitz constants are induced by the inner products of the Hilbert spaces. We refer to [Kir34] for the original proof and to [Sch69, Th. 1.31] for a proof of the version as stated above. In this work we only consider the case $H_2 = \mathbb{R}$ which allows for more general assumption on the space H_1. We now formulate the Lipschitz extension problem in our setting.

Problem 1.18 (Lipschitz Extensions). Let $(\tilde{\Omega}, d)$ be a metric space and $\mathcal{O} \subset \tilde{\Omega}$ be a bounded subset. For a given Lipschitz function $g : \mathcal{O} \to \mathbb{R}$ find a Lipschitz function $u : \tilde{\Omega} \to \mathbb{R}$ such that

$$\operatorname*{Lip}_d(u; \tilde{\Omega}) = \operatorname*{Lip}_d(g; \mathcal{O}).$$

A function $u : \tilde{\Omega}$ with this property is called *Lipschitz extension* of g to $\tilde{\Omega}$.

In this setting one can explicitly construct solutions of the Lipschitz extension task. They are not unique, however one has an upper and a lower bound. In fact, conceptually these solutions are very similar to the functions in Eq. (1.6) and even coincide, whenever the sup-norm of the gradient is given as the Lipschitz constant.

Lemma 1.19. In the setting of Problem 1.18 we have that the

- **Whitney (or maximal) extension:** $\overline{g}(x) := \inf_{y \in \mathcal{O}} g(y) + \operatorname{Lip}_d(g; \mathcal{O}) \cdot d(x, y)$ and the
- **McShane (or minimal) extension:** $\underline{g}(x) := \sup_{y \in \mathcal{O}} g(y) - \operatorname{Lip}_d(g; \mathcal{O}) \cdot d(x, y)$

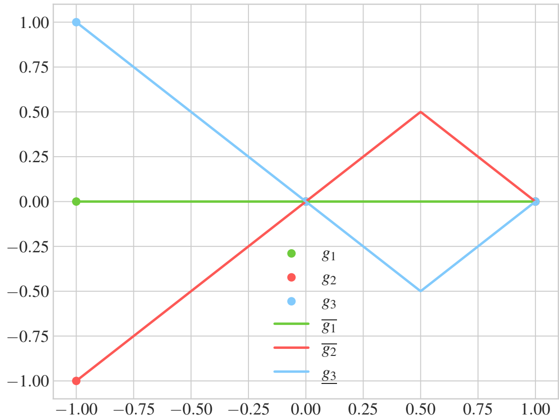


Figure 1.5.: The maximal extension does not admit a comparison principle, as demonstrated in Example 1.20.

defined for $x \in \tilde{\Omega}$ are Lipschitz extensions of g to $\tilde{\Omega}$. Moreover, let $u : \tilde{\Omega} \to \mathbb{R}$ be any Lipschitz extension of g, the we have that

$$\underline{g} \leq u \leq \overline{g}.$$

Proof. We refer to [Whi92] and [McS34] for the proofs of the respective result. □

As demonstrated in Example 1.20, there are cases where $\overline{g} \neq \underline{g}$ and therefore, Lipschitz extensions are not unique in general. Furthermore, [ACJ04] points out that the Whitney and McShane extension do not allow for a comparison principle, which can also be observed in Example 1.20.

Example 1.20. Consider the set $\tilde{\Omega} = [-1, 1]$ and $\mathcal{O} = \{-1, 0, 1\}$ with

$$\begin{aligned} g_1(x) &:= 0, \\ g_2(x) &:= 1/2(x - \operatorname{sign}(x) \cdot x), \\ g_3(x) &:= -g_2. \end{aligned}$$

Then we have that $g_2 \leq g_1$ on $\mathcal{O}$ but

$$\overline{g_2} > \overline{g_1} \text{ in } (0, 1),$$

see Fig. 1.5 for a visualization. Analogously, we have that $g_3 \geq g_1$ on $\mathcal{O}$ but

$$\underline{g_3} < \underline{g_1} \text{ in } (0, 1).$$

Absolutely Minimizing Extension Sending $p \to \infty$ in the variational formulation of the p-Laplace equation yields the Lipschitz extension task, which however does not admit for unique solutions. So the question arises, which property is lost in the limit case. For $p < \infty$ one has the crucial local minimization property. Let $\Omega \subset \mathbb{R}^d$ be an open domain and denote by u_p the solution of the p-Laplace problem with boundary values $g \in W^{1,p}(\Omega)$. Then we know that u_p is also a minimizer of $\mathcal{E}_p$ on every subset $V \subset \Omega$, i.e.,

$$\int_V |\nabla u_p|^p \, dx \leq \int_V |\nabla v|^p \, dx$$

for any function v such that $(u_p - v) \in W^{1,p}_0(V)$, see e.g. []. This lead Aronsson to introduce the concept of *absolutely minimizing Lipschitz extension* in [Aro67]. A function $u_\infty \in W^{1,\infty}$ is called absolutely minimal, if

$$\operatorname*{ess\,sup}_{x\in V} |\nabla u| \leq \operatorname*{ess\,sup}_{x \in V} |\nabla v| \quad \text{for every open } V \subset \Omega \tag{1.7}$$

and every function v such that $(u-v) \in W^{1,\infty}_0$. I fact one can also show, that $u_p \xrightarrow{p\to\infty} u_\infty$ ([Aro67]), which seems to validate the notion of absolute minimizers. In [**tour**] it was shown, that one has an equivalent formulation involving the Lipschitz constant. For a given Lipschitz function $f : \overline{\Omega} \to \mathbb{R}$ we have that u_∞ with $(u_\infty - f) \in W^{1,\infty}_0(\Omega)$ fulfills Eq. (1.7) iff

$$\operatorname{Lip}(u_\infty; V) \leq \operatorname{Lip}(v; V) \text{ for every } V \subset \Omega$$

and every function v such that $(u - v) \in W^{1,\infty}_0(V)$, see [**tour**].

Maybe not Sobolev here

In this thesis we work with a notion of absolute minimizers, which is equivalent to the above formulation for convex domains in $\mathbb{R}^d$. However, ...

Problem 1.21 (AMLEs). Let $(\tilde{\Omega}, d)$ be a length space, $\mathcal{O} \subset \tilde{\Omega}$ a closed subset and $g : \mathcal{O} \to \mathbb{R}$ a Lipschitz function. Find an extension $u \in C(\tilde{\Omega})$ such that $u = g$ on $\mathcal{O}$ and

$$\operatorname*{Lip}_d(u; \overline{V}) = \operatorname*{Lip}_d(u, \partial V) \text{ for all open and connected sets } V \subset \tilde{\Omega} \setminus \mathcal{O}.$$

A function u fulfilling this property is called absolutely minimizing Lipschitz extension of g.

Remark 1.22. In our application Ω is an open subset of $\mathbb{R}^d$ and we then choose $\tilde{\Omega} = \overline{\Omega}$. Here, it is important to note that the topological notions like boundary and interior are to be understood relative to $\overline{\Omega}$. A visualization odf this concept can be found in Fig. 1.6. $\triangle$

Comparison with Cones

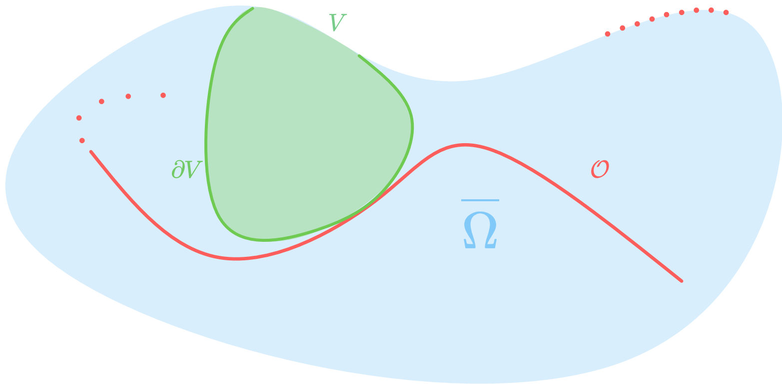


Figure 1.6.: A set $V \subset \overline{\Omega}$ can be relatively open w.r.t. the metric space $\overline{\Omega}$ although, $V \cap \partial\Omega \neq \emptyset$, where $\partial\Omega$ is the boundary within the standard topology on $\mathbb{R}^d$. The relative boundary of $\partial_{\overline{\Omega}} V$ does not include any parts of $\partial\Omega$.

The infinity Laplacian

1.3.2. Graph Lipschitz Extensions

We now consider the limit $p \to \infty$ of Problem 1.9 in the graph case. Analogously to **??** we derive

$$\lim_{p\to\infty} \left(\mathbf{E}_p^{w_n}(\mathbf{u})\right)^{1/p} = \max_{x,y\in\Omega_n} w_n(x,y)\,|\mathbf{u}(y) - \mathbf{u}(x)| =: \mathbf{E}_\infty^{w_n}(\mathbf{u})$$

which extends the graph p-Laplacian energy to the case $p = \infty$. Again we notice structural similarities to the continuum version $\mathcal{E}_\infty$.

Remark 1.23. Informally speaking the functional $\mathbf{E}$ combines elements of a gradient and Lipschitz constant. Assuming that $w_n(x,y)$ relates to $|x-y|$ we see that the finite difference approximation resembles a Lipschitz constant. However, typically $w_n(x,y)$ has also some localizing property which fits the interpretation of a gradient better. △

This functional now leads to the graph Lipschitz extension problem.

Problem 1.24 (Graph Energy Minimization). Given a weighted graph (Ω_n, w_n) and a labeling function $\mathbf{g} : \mathcal{O}_n \to \mathbb{R}$, for $\mathcal{O}_n \subset \Omega_n$ we consider the problem

$$\min_{\mathbf{u}:\Omega_n\to\mathbb{R}} \mathbf{E}_\infty^{w_n}(\mathbf{u}) \text{ subject to } \mathbf{u}(x) = \mathbf{g}(x) \text{ for all } x \in \mathcal{O}_n.$$

Since the weighting function $w_n : \Omega_n \times \Omega_n \to \mathbb{R}_0^+$ does not induce a metric, Problem 1.24 does not directly fit the framework of the abstract Lipschitz Extension in Problem 1.18. However, we can consider paths in (Ω_n, w_n), connecting arbitrary $x, y \in \Omega_n$ i.e. vectors

$\gamma \in \Omega_n^{\times k}$ such that

$$\begin{aligned} w_n(\gamma_i, \gamma_{i+1}) > 0 \quad &\text{for all } i = 1, \dots, k-1, \\ \gamma_1 = x&, \\ \gamma_k = y& \end{aligned}$$

for which we define the length as

$$|\gamma| = \sum_{i=1}^{k-1} w(\gamma_i, \gamma_{i+1})^{-1}.$$

This yields the metric space (Ω_n, d_{w_n}), where $d_{w_n} : \Omega_n \times \Omega_n \to \mathbb{R}$ is defined as

$$d_{w_n}(x, y) := \min \{|\gamma| : \gamma \text{ is a path in } (\Omega_n, w_n) \text{ from } x \text{ to } y\}. \tag{1.8}$$

Remark 1.25. We note that it is important to only consider non-negative weights, otherwise any loop with a negative "length" would decrease the length of the whole path arbitrarily. However, restricting ourselves to non-negative weights we can easily see, that the minimum in Eq. (1.8) is indeed attained. △

With this definition we can consider the Lipschitz extension task of $g : \mathcal{O}_n \to \mathbb{R}$ to Ω_n within the metric space (Ω_n, d_{w_n}), i.e. within the setting of Problem 1.18. Therefore the question arises, whether the minimization problem in Problem 1.24 is equivalent to the metric Lipschitz extension problem for which we have the following lemma.

Lemma 1.26. For a graph (Ω_n, w_n) with non-negative weights and a function $\mathbf{u} : \Omega_n \to \mathbb{R}$ we have that

$$\mathbf{E}_\infty^{w_n}(\mathbf{u}) = \operatorname*{Lip}_{d_{w_n}}(\mathbf{u}).$$

Furthermore, for $\mathcal{O}_n \subset \Omega_n$ and a function $\mathbf{g} : \mathcal{O}_n \to \mathbb{R}$ and we have that

$$\mathbf{g} = \mathbf{u} \text{ on } \mathcal{O}_n \Rightarrow \operatorname*{Lip}_{d_{w_n}}(\mathbf{g}; \mathcal{O}_n) \leq \operatorname*{Lip}_{d_{w_n}}(\mathbf{u}).$$

Proof. **Step 1:** We show that $\mathrm{Lip}_{d_{w_n}}(\mathbf{u}) \leq \mathbf{E}_\infty^{w_n}(\mathbf{u})$.
We can choose a path $\gamma \in \Omega_n^{\times k}$ such that

$$\operatorname*{Lip}_{d_{w_n}}(\mathbf{u}) = \frac{\mathbf{u}(\gamma_1) - \mathbf{u}(\gamma_k)}{|\gamma|}.$$

The path γ allows to compare vertices $\gamma_1, \gamma_k \in \Omega_n$ that aren't necessarily neighbors in the graph. However, each consecutive vertices in the path are neighbors in the graph and therefore we have

$$w_n(\gamma_i, \gamma_{i+1}) \, |\mathbf{u}(\gamma_{i+1}) - \mathbf{u}(\gamma_i)| \leq \mathbf{E}_\infty^{w_n}(\mathbf{u}) \quad \text{for all } i = 1, \dots, k-1. \tag{1.9}$$

We now employ an elementary result for numbers $a_i \in \mathbb{R}_0^+, b_i \in \mathbb{R}^+, i = 1, \dots m \in \mathbb{N}$, namely

$$[a_i \cdot b_i \le c \in \mathbb{R} \quad \text{for } i = 1, \dots, m] \Rightarrow \frac{\sum_{i=1}^m a_i}{\sum_{i=1}^m b_i^{-1}} \le c \tag{1.10}$$

which can be seen as follows

$$\begin{aligned}
a_i \cdot b_i &\le c \quad \text{for } i = 1, \dots, m\\
\Rightarrow a_i &\le b_i^{-1} \cdot c \quad \text{for } i = 1, \dots, m\\
\Rightarrow \sum_{i=1}^m a_i &\le \left(\sum_{i=1}^m b_i^{-1}\right) \cdot c\\
\Rightarrow \frac{\sum_{i=1}^m a_i}{\sum_{i=1}^m b_i^{-1}} &\le c.
\end{aligned}$$

This then yields

$$\frac{|\mathbf{u}(\gamma_1) - \mathbf{u}(\gamma_k)|}{|\gamma|} \le \frac{\sum_{i=1}^{k-1} |\mathbf{u}(\gamma_i) - \mathbf{u}(\gamma_{i+1})|}{|\gamma|} = \frac{\sum_{i=1}^{k-1} |\mathbf{u}(\gamma_i) - \mathbf{u}(\gamma_{i+1})|}{\sum_{i=1}^{k-1} w_n(\gamma_i, \gamma_{i+1)^{-1}}} \le \mathbf{E}_\infty^{w_n}(\mathbf{u})$$

where in the last inequality we employed Eq. (1.10) together with Eq. (1.9).
Step 2: We show that $\mathrm{Lip}_{d_{w_n}}(\mathbf{u}) \ge \mathbf{E}_\infty^{w_n}(\mathbf{u})$.
Let $x, y \in \Omega_n$, then we know that $d_w(x, y) \le w_n(x, y)^{-1}$ and therefore

$$|\mathbf{u}(x) - \mathbf{u}(y)|\, w_n(x, y) \le \frac{|\mathbf{u}(x) - \mathbf{u}(y)|}{d_w(x, y)} \le \max_{\bar{x}, \bar{y} \in \Omega_n} \frac{|\mathbf{u}(\bar{x}) - \mathbf{u}(\bar{y})|}{d_w(\bar{x}, \bar{y})} = \mathrm{Lip}_{d_{w_n}}(\mathbf{u}).$$

Since this holds for arbitrary $x, y \in \Omega_n$ we have that

$$\mathbf{E}_\infty^{w_n}(\mathbf{u}) = \max_{x, y \in \Omega_n} |\mathbf{u}(x) - \mathbf{u}(y)|\, w_n(x, y) \le \mathrm{Lip}_{d_{w_n}}(\mathbf{u}).$$

Step 3: We show that $\mathrm{Lip}_{d_{w_n}}(\mathbf{g}; \mathcal{O}_n) \le \mathrm{Lip}_{d_{w_n}}(\mathbf{u})$.
If $\mathbf{g} = \mathbf{u}$ on $\mathcal{O}$ this simply follows since the maximum for the Lipschitz constant of $\mathbf{u}$ is taken over a larger set. Indeed,

$$\begin{aligned}
\mathrm{Lip}_{d_{w_n}}(\mathbf{u}) &= \max_{x, y \in \Omega_n} \frac{|\mathbf{u}(x) - \mathbf{u}(y)|}{d_w(x, y)} \ge \max_{x, y \in \mathcal{O}_n} \frac{|\mathbf{u}(x) - \mathbf{u}(y)|}{d_w(x, y)} = \max_{x, y \in \mathcal{O}_n} \frac{|\mathbf{g}(x) - \mathbf{g}(y)|}{d_w(x, y)}\\
&= \mathrm{Lip}_{d_{w_n}}(\mathbf{u}; \mathcal{O}_n).
\end{aligned}$$

□

This lemma shows that the abstract Lipschitz extension task considered on the metric space (Ω_n, d_{w_n}) and the Graph ∞-Dirichlet minimization task are indeed equivalent. Therefore, we also have the Whitney and McShane extensions

$$\overline{\mathbf{g}}(x) = \inf_{y \in \mathcal{O}_n} \mathbf{g}(y) + d_{w_n}(x,y)$$
$$\underline{\mathbf{g}}(x) = \sup_{y \in \mathcal{O}_n} \mathbf{g}(y) - d_{w_n}(x,y)$$

as solutions on the graph. Analogously, the problem does not admit for unique solutions.

Absolutely Minimizing Graph Extensions Similarly to Section 1.3.1 we can now consider absolutely minimizing extensions. However, the problem in Problem 1.21 uses a notion of a boundary and it is not directly clear how to infer this concept to the discrete set Ω_n. Therefore, we define the following what we mean by "boundary" on a graph.

Definition 1.27. Let (Ω_n, w_n) be a weight graph and let $V \subset \Omega_n$ be a subset, then we define

- the **exterior** boundary as $\partial^{\text{ext}} := \{x \in \Omega_n \setminus V : w_n(x,y) > 0 \text{ for some } y \in V\}$,
- the **interior** boundary as $\partial^{\text{int}} := \{x \in V : w_n(x,y) > 0 \text{ for some } y \in \Omega_n \setminus V\}$.

The closure of V is then defined as $\overline{V}^{\text{ext}} V := V \cup \partial^{\text{ext}}$ and the interior as $\overset{\circ}{V}^{\text{int}} := V \setminus \partial^{\text{int}} V$.

We note that it is not possible to define a topology on Ω_n that would yield the above notions. Namely, the only admissible topology in our case would be the discrete topology, i.e., 2^{Ω_n}. However, in this topology the only closed sets are $\emptyset$ and Ω_n which is not useful for the applications in the following. Using the Kuratowski closure axioms [Kur22] we remark the following.

Lemma 1.28. The exterior closure on a weighted graph (Ω_n, w_n) is a preclosure or Čech closure.

Proof. Here, we use the notion of a preclosure in [ČFK66]. We first see that $\overline{\emptyset}^{\text{ext}} = \emptyset$ and that $V \subset \overline{V}^{\text{ext}}$ for every subset $V \subset \Omega_n$, i.e. the above defined closure preserves the empty set and is extensive. Furthermore, for two sets $V_1, V_2 \subset \Omega_n$ we have that

$$\begin{aligned}
&x \in \partial^{\text{ext}}(V_1 \cup V_2)\\
\Leftrightarrow\ &[x \notin V_1 \cup V_2] \wedge [\exists y \in V_1 \cup V_2 : w_n(x,y)] \neq 0\\
\Leftrightarrow\ &[x \notin V_1 \cup V_2] \wedge \Big([\exists y \in V_1 : w_n(x,y) \neq 0] \vee [\exists y \in V_2 : w_n(x,y) \neq 0] \Big)\\
\Leftrightarrow\ &\Big[x \in \partial^{\text{ext}} V_1 \setminus V_2\Big] \vee \Big[x \in \partial^{\text{ext}} V_2 \setminus V_1\Big]\\
\Leftrightarrow\ &x \in (\partial^{\text{ext}} V_1 \cup \partial^{\text{ext}} V_2) \setminus (V_1 \cup V_2).
\end{aligned}$$

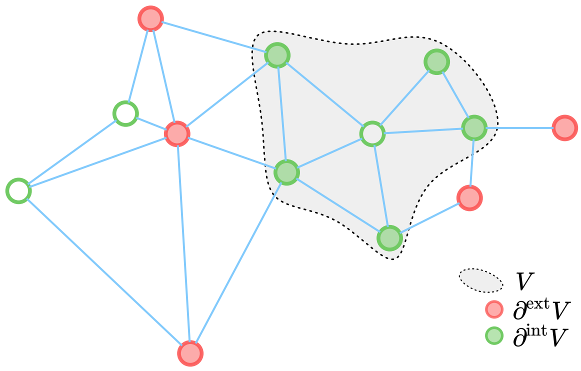


Figure 1.7.: Visualization of exterior and interior boundary on a graph.

We have shown that $\partial^{\text{ext}}(V_1 \cup V_2) = (\partial^{\text{ext}} V_1 \cup \partial^{\text{ext}} V_2) \setminus (V_1 \cup V_2)$. Therefore, we have that

$$\begin{aligned}
\overline{V_1 \cup V_2}^{\text{ext}} &= V_1 \cup V_2 \cup \partial^{\text{ext}}(V_1 \cup V_2)\\
&= V_1 \cup V_2 \cup ((\partial^{\text{ext}} V_1 \cup \partial^{\text{ext}} V_2) \setminus (V_1 \cup V_2))\\
&= V_1 \cup \partial^{\text{ext}} V_1 \cup V_2 \cup \partial^{\text{ext}} V_2\\
&= \overline{V_1}^{\text{ext}} \cup \overline{V_2}^{\text{ext}}.
\end{aligned}$$

This shows that the closure preserves binary unions and therefore we have shown, that it is indeed a Čech closure. $\square$

The missing property, that inhibits the closure to induce a topology is the so-called idempotence. Namely, there are sets $V \subset \Omega_n$ such that

$$\overline{V}^{\text{ext}} \neq \overline{\overline{V}^{\text{ext}}}^{\text{ext}}.$$

E.g. in the example visualized in Fig. 1.7 we see that $\overline{\overline{V}^{\text{ext}}}^{\text{ext}} = \Omega_n \neq \overline{V}^{\text{ext}}$. Since the closure we employ here does not induce a topology, we have a slightly modified notion of absolutely minimizers.

Problem 1.29 (Graph AMLEs). Given a connected weighted graph (Ω_n, w_n), $\mathcal{O}_n \subset \Omega_n)$ and a function $\mathbf{g} : \mathcal{O}_n \to \mathbb{R}$ find a function $\mathbf{u} : \Omega_n \to \mathbb{R}$ such that

$$\begin{aligned}
\operatorname*{Lip}_{d_{w_n}}(\mathbf{u}; \overline{V}^{\text{ext}}) = \operatorname*{Lip}_{d_{w_n}}(\mathbf{u}; \partial^{\text{ext}} V) &\text{ for all connected } V \subset \Omega_n \setminus \mathcal{O}_n,\\
\mathbf{u} = \mathbf{g} &\text{ on } \mathcal{O}_n.
\end{aligned}$$

Comparison with graph Distance functions Analogously to the continuum case **??** we can also consider comparison with distance functions on graphs. The main ingredients here, are the graph distance function d_{w_n} and the notion of closure on a graph as developed in the last section.

Definition 1.30. For a weighted graph (Ω_n, w_n) we say a function $\mathbf{u} : \Omega_n \to \mathbb{R}$ fulfills comparison with distance function from above (CDFA) on a subset $U \subset \Omega_n$ if for every $V \subset U$ we have

$$\max_{\overline{V}^{\text{ext}}} (u + a\ d_{w_n}(\cdot, z)) = \max_{\partial^{\text{ext}} V} (u + a\ d_{w_n}(\cdot, z)) \tag{CDFA}$$

for every $z \in \Omega_n \setminus V$ and every $a \in \mathbb{R}$. We say that $\mathbf{u}$ fulfills comparison with distance function from below (CDFB) on a subset $U \subset \Omega_n$ if for every $V \subset U$ we have

$$\min_{\overline{V}^{\text{ext}}} (u - a\ d_{w_n}(\cdot, z)) = \min_{\partial^{\text{ext}} V} (u - a\ d_{w_n}(\cdot, z)) \tag{CDFB}$$

for every $z \in \Omega_n \setminus V$ and every $a \in \mathbb{R}$.

Analogously to the continuum case, we say that a function fulfills comparison with distance functions, if it fulfills both, Eq. (CDFA) and Eq. (CDFB). Existence of such functions is established later, we are first interested in the question of uniqueness. Since the notion of graph boundaries is not directly compatible with the usual definitions on metric spaces, we prove it separately. Here, we adapt arguments from [smart] and [LeGruyer]. To do so we first consider the operators

$$\mathbf{S}^\varepsilon \mathbf{u}(x) := \max_{y \in \Omega_n : d_{w_n}(x,y) \leq \varepsilon} \mathbf{u}(y) \quad \mathbf{S}_\varepsilon \mathbf{u}(x) := \min_{y \in \Omega_n : d_{w_n}(x,y) \leq \varepsilon} \mathbf{u}(y)$$

and proof the following lemma, which is the analogue of

The Graph infinity Laplacian We can also obtain the limit of the Graph p-Laplace operator via the following formal calculation,

$$\begin{aligned}
&\Delta_p^{w_n} \mathbf{u}(x) = 0 \\
\Leftrightarrow &\sum_{y \in \Omega_n} w_n(x,y)^p \left|\mathbf{u}(y) - \mathbf{u}(x)\right|^{p-2} (\mathbf{u}(y) - \mathbf{u}(x)) = 0 \\
\Leftrightarrow &\sum_{y : \mathbf{u}(x) \leq \mathbf{u}(y)} w_n(x,y)^p (\mathbf{u}(y) - \mathbf{u}(x))^{p-1} = \sum_{y : \mathbf{u}(x) > \mathbf{u}(y)} w_n(x,y)^p (\mathbf{u}(x) - \mathbf{u}(y))^{p-1}.
\end{aligned}$$

Taking the terms on the left and right hand side to the power of $1/p$ and then formally sending $p \to \infty$ then yields

$$\begin{aligned}
&\max_{y : \mathbf{u}(x) \leq \mathbf{u}(y)} w_n(x,y)(\mathbf{u}(y) - \mathbf{u}(x)) = \max_{y : \mathbf{u}(x) > \mathbf{u}(y)} w_n(x,y)(\mathbf{u}(x) - \mathbf{u}(y)) \\
\Leftrightarrow &\max_{y \in \Omega_n} w_n(x,y)(\mathbf{u}(y) - \mathbf{u}(x)) = -\min_{y \in \Omega_n} w_n(x,y)(\mathbf{u}(y) - \mathbf{u}(x)).
\end{aligned}$$

This calculation motivates the definition of the graph infinity Laplacian

$$\Delta_\infty^{w_n} \mathbf{u}(x) := \max_{y \in \Omega_n} w_n(x,y)(\mathbf{u}(y) - \mathbf{u}(x)) + \min_{y \in \Omega_n} w_n(x,y)(\mathbf{u}(y) - \mathbf{u}(x)),$$

which then allows to formulate the associated problem as an extension of Problem 1.11.

Problem 1.31 (Graph ∞-Laplacian). Given a weighted graph (Ω_n, w_n) and a labeling function $\mathbf{g} : \mathcal{O}_n \to \mathbb{R}$ with $\mathcal{O}_n \subset \Omega_n$, find a function $\mathbf{u} : \Omega_n \to \mathbb{R}$ such that

$$\begin{aligned} \Delta_\infty^{w_n} \mathbf{u} &= 0, \text{ in } \Omega_n \setminus \mathcal{O}_n, \\ \mathbf{u} &= \mathbf{g} \text{ on } \mathcal{O}_n. \end{aligned}$$

This problem is again well-posed, which is formulated in the following lemma.

Lemma 1.32. There exists a unique solution for Problem 1.31.

Relation between the Graph Lipschitz Extensions We now establish the connection between the different notions of Lipschitz extensions. Compared to the continuum case we do not establish the full equivalences but only the necessary implications required for the convergence proofs in [LIP-II].

First we see, that graph AMLEs are indeed special solutions of the basic Lipschitz extension problem on the graph.

Lemma 1.33. A graph AMLE is also a Lipschitz extension.

Proof. □

We now state the main result concerning the relation between the graph infinity Laplacian, graph AMLEs and comparison cones.

Lemma 1.34. Let (Ω_n, w_n) be a weighted connected graph and $\mathbf{g} : \mathcal{O}_n \to \mathbb{R}$ be a given function for $\mathcal{O}_n \subset \Omega_n$. Furthermore, let $\mathbf{u} : \Omega_n \to \mathbb{R}$ be graph infinity harmonic on $\Omega_n \setminus \mathcal{O}_n$ with boundary conditions given by $\mathbf{g}$, i.e., $\mathbf{u}$ solves Problem 1.31 then we have that

- $\mathbf{u}$ is an graph AMLE, i.e., $\mathbf{u}$ solves Problem 1.29,
- $\mathbf{u}$ fulfills comparison with cones.

Proof. Both of the stament are proven in [LIP-II]. From [LIP-II, Prop. 3.8] we have that $\mathbf{u}$ is an graph AMLE. Furthermore, form [LIP-II, Th. 3.2] we have that $\mathbf{u}$ fulfills comparison with cones. In fact, [LIP-II, Th. 3.2], shows a more refined statement, namely that

$$\begin{aligned} -\Delta_\infty^{w_n} \mathbf{u} \leq 0 &\Rightarrow \mathbf{u} \text{ fulfills CDFA}, \\ -\Delta_\infty^{w_n} \mathbf{u} \geq 0 &\Rightarrow \mathbf{u} \text{ fulfills CDFB}. \end{aligned}$$

□

1.4. Main Contribution: Continuum Limits

We are now in the situation to present the main results of [LIP-I; LIP-III].

The kernel

(K1) η is positive and continuous at 0,

(K2) η is non-increasing,

(K3) $\mathrm{supp}(\eta) \subset [0, t_\eta]$ for some $t_\eta > 0$.

1.4.1. Gamma Convergence: [LIP-I]

Originally, the concept of Γ-convergence dates back to De Giorgi [DF75] as a type of variational convergence. We refer to [Bra02; Dal12] for a detailed overview on this notion and related topics. While Γ-convergence was successfully employed in a pure continuum setting for a longer time (see e.g. [Mod77]), it was more recently used to prove convergence form a discrete to a continuum functional [CGL10; BY12; VB+12]. The most relevant reference for this thesis was disruptive work presented by García Trillos and Slepčev in [GS15]. Here, the considered object was a graph total variation or generalized parameter, where the functional corresponds to $\mathbf{E}_p^{w_n}$ for $p = 1$ from Problem 1.9. Among other important ideas and notions, we want to highlight two ingredients that directly influenced [LIP-I]:

1) Γ-convergence on a common metric space, that allows to compare graph functions with continuum functions.

2) The proof strategy, discrete to non-local, non-local to continuum.

We review how theses concepts influence [LIP-I] in the following sections. The results in [GS15] were later transferred to the case $1 < p < \infty$ in [ST19] and in this sense [LIP-I] constitutes the generalization to $p = \infty$. G Convergence in L infty

Γ-convergence We start with the basic definition of Γ-convergence.

Definition 1.35 (Γ-convergence). Let X be a metric space and let $F_n : X \to [-\infty, \infty]$ be a sequence of functionals. We say that F_n Γ-converges to the functional $F : X \to [-\infty, \infty]$ if

(i) **(liminf inequality)** for every sequence $(x)_{n\in\mathbb{N}} \subset X$ converging to $x \in X$ we have that

$$\liminf_{n\to\infty} F_n(x_n) \geq F(x);$$

(ii) **(limsup inequality)** for every $x \in X$ there exists a sequence $(x)_{n\in\mathbb{N}} \subset X$ converging to x and

$$\limsup_{n\to\infty} F_n(x_n) \leq F(x).$$

In order to show convergence of discrete functions defined on Ω_n to continuum functions acting on $\overline{\Omega}$, one needs to define a common metric space. [GS15] introduced the space TL^p

$$TL^p := \{(\mu, u) : \mu \in \mathcal{P}(\Omega), u \in L^p(\mu)\}$$

together with the transport distance

$$d_{TL^p}((\mu, u), (\nu, v)) = \inf_{\pi\in\Gamma(\mu,\nu)} \left(\int_{\Omega\times\Omega} |x-y|^p + |u(x) - v(y)|^p \, d\pi(x, y) \right)^{1/p}$$

where $\Gamma(\mu, \nu)$ is the set of coupling between μ and ν. A sequence (mu_n, u_n) converges to (μ, u) in TL^p iff there exists a sequence of transportation maps $T_n : \Omega \to \Omega$ with $T_n\#\mu = \mu_n$ and

$$\int_\Omega |x - T_n(x)| \, d\mu(x) \to 0$$

such that $u_n \circ T_n \xrightarrow{L^p} u$, [GS15, Prop. 3.12]. Therefore, the maps T_n allow to employ standard convergence in L^p. In order to transfer this situation to L^∞ one could try to employ a ∞-Wasserstein distance. However, as argued in [Roi22] the arguments do not transfer directly, since convergence in W^∞ does not metrize weak convergence of measures [San15, Thm. 5.10]. However, as seen in [LIP-I] one can employ a more direct argument. For problems in L^p the conservation of mass was important for the maps T_n (i.e. $T_n\#\mu = \mu_n$) such that the integrals could be transformed. This condition is irrelevant in L^∞, namely we have the following analogous transformation rule in L^∞.

Lemma 1.36 ([LIP-I, Lem. 2]). For two probability measures $\mu, \nu \in \mathcal{P}(\overline{\Omega})$, a measurable map $T : \Omega \to \Omega$ which fulfills

(i) $\nu << T\#\mu$,

(ii) $T\#\mu << \nu$,

and for a measurable function $u : \Omega \to \mathbb{R}$ we have that

$$\nu\text{-}\operatorname*{ess\,sup}_{x\in\Omega} u(x) = \mu\text{-}\operatorname*{ess\,sup}_{y\in\Omega} u(T(y)).$$

We want to compare the discrete measure $\mu_n = \frac{1}{n}\sum_{x\in\Omega_n} \delta_x$ to the target measure μ. In this setting a closest point projection $p_n : \Omega \to \Omega_n$

$$p_n(x) \in \operatorname{argmin}_{y\in\Omega_n} |x - y|$$

fulfills the assumption of Lemma 1.36. This allows us to extend the functional $\mathbf{E}_\infty^{w_n}$ in Problem 1.24 to L^∞ via

$$\mathbf{E}_\infty^{w_n}(u) = \begin{cases} \mathbf{E}_\infty^{w_n}(\mathbf{u}) & \text{if } u = \mathbf{u} \circ p_n, \text{ for some } \mathbf{u} : \Omega_n \to \mathbb{R}, \\ \infty & \text{else },\end{cases} \tag{1.11}$$

which was similarly done in [GS15; ST19]. Additionally, we incorporate the constraint on $\mathcal{O}_n$ in Problem 1.24 via

$$\mathbf{E}_\infty^{w_n,\text{cons}}(\mathbf{u}) := \begin{cases} \mathbf{E}(\mathbf{u}) & \text{if } \mathbf{u} = g \text{ on } \mathcal{O}_n, \\ \infty & \text{else,}\end{cases}$$

with the analogous extension to Ł$^\infty$ as in Eq. (1.11). This now allows us to state the first main result of [LIP-I].

Theorem 1.37 (Discrete to continuum Γ-convergence). Let $\Omega \subset \mathbb{R}^d$ be a domain satisfying **??**, let the kernel fulfill (K1)-(K3), and let the constraint sets $\mathcal{O}_n, \mathcal{O}$ satisfy **??**, then for any null sequence $(\varepsilon_n)_{n\in\mathbb{N}} \subset (0,\infty)$ which satisfies the scaling condition **??** we have

$$\mathbf{E}_\infty^{n,\text{cons}} \xrightarrow{\Gamma} \sigma_\eta\, \mathcal{E}^{\text{cons}}. \tag{1.12}$$

The main proof strategy here is similar to the one in [GS15; ST19]. Namely one defines the non-local functional

$$\mathcal{E}_\infty^\varepsilon(u) := \frac{1}{\varepsilon} \operatorname*{ess\,sup}_{x,y\in\Omega} \{\eta_\varepsilon(|x-y|)\,|u(x)-u(y)|\}, \quad \varepsilon > 0$$

for which we show that for any sequence $\varepsilon_n \to 0$ we have [LIP-I, Thm. 4]

$$\mathcal{E}_\infty^{\varepsilon_n} \xrightarrow{\Gamma} \sigma_\eta\, \mathcal{E}_\infty.$$

For the liminf inequality of the discrete functionals, one has to take special care of points $x, y \in \Omega_n$ where $\eta_{\varepsilon_n}(|p_n(x) - p_n(y)|) = 0$. We want to bound $\mathbf{E}_\infty^{w_n}$ from below by $\mathcal{E}_\infty^{\varepsilon_n}$ for which we have to permit significant communication of x and y whenever $p_n(x)$ and $p_n(y)$ do not communicate. This can be done, (temporarily assuming η is constant on $[0,t]$) by introducing a smaller length scale $\tilde{\varepsilon}$ such that

(i) $|p_n(x) - p_n(y)| > t\varepsilon_n \Rightarrow |p_n(x) - p_n(y)|/\varepsilon_n < |x-y|/\tilde{\varepsilon}$,

(ii) $\lim_{n\to\infty} \tilde{\varepsilon}/\varepsilon = 1$.

As shown in [LIP-I] the choice $\tilde{\varepsilon} = \varepsilon - 2\delta/t$ fulfills (i). So in order to fulfill (ii) we obtain the scaling condition

$$\lim_{n\to\infty} \frac{\delta_n}{\varepsilon_n} = 0 \Rightarrow \lim_{n\to\infty} \frac{\varepsilon_n - 2\delta_n/t}{\varepsilon_n} = 1 - \frac{2}{t} \lim_{n\to\infty} \frac{\delta_n}{\varepsilon_n} = 1.$$

This then allows to prove the liminf inequality since $\mathbf{E}_\infty^{w_n}(u_n) \geq \frac{\tilde{\varepsilon}_n}{\varepsilon_n} \mathcal{E}_\infty^{\tilde{\varepsilon}_n}(u_n)$ holds. The limsup inequality can then be shown by choosing the constant sequence, with some additional care for the changing constraint set $\mathcal{O}_n$.

Convergence of Minimizers The convenient aspect of Γ-convergence is, that under additional compactness properties it directly shows convergence of minimizers, [Bra02, Thm. 8]. This yields the second main result in [LIP-I].

Theorem 1.38 ([LIP-I, Thm. 2]). Let $\Omega \subset R^d$ be a domain satisfying **??**, let the kernel fulfil (K1)-(K3), let the constraint sets $\mathcal{O}_n, \mathcal{O}$ satisfy **??**, and $(\varepsilon_n)_{n\in\mathbb{N}} \subset (0,\infty)$ be a null sequence which satisfies the scaling condition **??**. Then any sequence $(u)_{n\in\mathbb{N}} \subset L^\infty(\Omega)$ such that

$$\lim_{n\to\infty}\left(\mathbf{E}_\infty^{n,\mathrm{cons}}(u_n) - \inf_{u\in L^\infty(\Omega)} \mathbf{E}_\infty^{n,\mathrm{cons}}(u)\right) = 0$$

is relatively compact in $L^\infty(\Omega)$ and

$$\lim_{n\to\infty} \mathbf{E}_\infty^{n,\mathrm{cons}}(u_n) = \min_{u\in L^\infty(\Omega)} \sigma_\eta\, \mathcal{E}_\infty^{\mathrm{cons}}(u).$$

Furthermore, every cluster point of $(u_n)_{n\in\mathbb{N}}$ is a minimizer of $\mathcal{E}_{\mathrm{cons}}$.

In order to show the compactness in the above theorem one employs the following lemma.

Lemma 1.39 ([LIP-I, Lem. 4]). Let (Ω, μ) be a finite measure space and $K \subset L^\infty(\Omega;\mu)$ be a bounded set w.r.t. $\|\cdot\|_{L^\infty(\Omega;\mu)}$ such that for every $\varepsilon > 0$ there exists a finite partition $\{V_i\}_{i=1}^n$ of Ω into subsets V_i with positive and finite measure such that

$$\mu\text{-}\operatorname*{ess\,sup}_{x,y\in V_i} |u(x) - u(y)| < \varepsilon \;\forall u \in K, i = 1, \dots, n, \tag{1.13}$$

then K is relatively compact.

Remark 1.40. The proof of this statement employs ideas from [DS88, Lem. IV.5.4] and appeared similarly in TR's master thesis. However, therein the statement was slightly wrong, which was corrected in [LIP-I]. △

This lemma is the used to show that if a sequence u_n fulfills

$$\sup_{n\in\mathbb{N}} \mathbf{E}_\infty^{w_n,\mathrm{cons}}(u_n) < \infty$$

then it is relatively compact.

Application to ground states Ground states, no rates.

1.4.2. Ratio Convergence

Chapter 2
Robust and Sparse Supervised Learning

In this chapter we now focus on supervised learning as described in **??**. [CLIP; BREG-II; FNO; BREG-I]

Remark on Notation In the following we deviate from the notation of the previous part. Most notably the input space is now denoted by $\mathcal{X}$ instead of Ω. Furthermore, we employ $\mathcal{Y}$ to denote an abstract output space.

2.1. Setting

We are given a finite training set $\mathcal{T} \subset \mathcal{X} \times \mathcal{Y}$. For a family of functions $f_\theta : \mathcal{X} \to \mathcal{Y}$ parameterized by $\theta \in \theta$ we consider the empirical minimization

$$\min_{\theta \in \theta} \mathcal{L}(\theta)$$

where for a function $\ell : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$ we define

$$\mathcal{L}(\theta) := \frac{1}{|\mathcal{T}|} \sum_{(x,y) \in \mathcal{T}} \ell(f_\theta(x), y). \tag{2.1}$$

Assuming that $\mathcal{T}$ is sampled from a joint distribution π on $\mathcal{X} \times \mathcal{Y}$ this approximates the infeasible population risk minimization

$$\int_{\mathcal{X} \times \mathcal{Y}} \ell(f_\theta(x), y) d\pi(x, y).$$

In this thesis we focus on feed-forward neural networks, i.e., we consider layers of the form

$$\Phi(w, W, b)(z) := wz + \sigma(Wz + b)$$

where $w \in \mathbb{R}$ models a residual connection, $W \in \mathbb{R}^{n \times n}$ is a weight matrix, $b \in \mathbb{R}^n$ a bias vector and $z \in \mathbb{R}^m$. We consider a concatenation of $L \in \mathbb{N}$ such layers, which then forms a neural network

$$f_\theta = \Phi^L \circ \ldots \circ \Phi^1$$

with parameters $\theta = (w_1, \ldots, w_L, W_1, \ldots, W_L, b_1, \ldots, b_L) \in \theta$ and layers $\Phi^i := \Phi(w_i, W_i, b_i)$.

MLP

Convolutions

ResNets

2.1.1. Gradient Computation and Stochastic Gradient Descent

Training a neural network requires to solve a optimization problem w.r.t. to the parameters $\theta \in \Theta$. In this work we only focus on first order methods, however both zero [**Riedl**] and second order methods [**Hessian**] have been successfully applied in this context. Employing first order methods, requires to evaluate the gradient $\nabla_\theta \mathcal{L}$, however in this scenario it is not common to compute the full gradient but rather to have a gradient estimator. This estimator is usually obtained by randomly dividing the train set $\mathcal{T}$ into disjoint minibatches $B_1 \cup \ldots \cup B_b = \mathcal{T}$ and then successively computing the gradient of the minibatch loss

$$\frac{1}{|B_i|} \sum_{(x,y) \in B_i} \ell(f_\theta(x), y).$$

Iterating over all batches $i = 1, \ldots, b$ is referred to as one epoch. From a mathematical point of view this yields stochastic optimization methods, since in each step the true gradient is replaced by an estimator. In the abstract setting we let $(\Omega, F, \mathbb{P})$ be a probability space and consider a function $g : \Theta \times \Omega \to \Theta$ as an unbiased estimator of $\nabla \mathcal{L}$, i.e.

$$\mathbb{E}\left[g(\theta; \omega)\right] = \nabla \mathcal{L}(\theta) \text{ for all } \theta \in \Theta.$$

Most notably this method transforms the standard gradient descent update [Cau+47]

$$\theta^{(k+1)} = \theta^{(k)} - \tau^{(k)} \nabla \mathcal{L}(\theta^{(k)})$$

to *stochastic* gradient descent [RM51]

$$\begin{aligned}
&\text{draw } \omega^{(k)} \text{ from } \Omega \text{ using the law of } \mathbb{P}, \\
&g^{(k)} := g(\theta^{(k)}; \omega^{(k)}), \\
&\theta^{(k+1)} := \theta^{(k)} - \tau^{(k)} g^{(k)}.
\end{aligned}$$

2.2. Adversarial Stability

2.3. Sparsity via Bregman Iterations: [BREG-I]

Intro sparsity blah blah

1. efficency
2. robustness
3. generalization

2.3.1. Preliminaries on Convex Analysis and Bregman Iterations

We first review some necessary concepts from convex analysis that allow us to introduce the framework in [BREG-I]. We refer to [BB18; Roc97; BC11] for a more exhaustive introduction to the topics. The functional J is called lower semicontinuous if $J(u) \leq \liminf_{n\to\infty} J(u_n)$ holds for all sequences $(u_n)_{n\in\mathbb{N}} \subset \Theta$ converging to u.

Definition 2.1. Given a Hilbert space Θ and a functional $J : \Theta \to (-\infty, \infty]$.

1. The functional J is called convex, if

$$J(\lambda\overline{\theta} + (1-\lambda)\theta) \leq \lambda J(\overline{\theta}) + (1-\lambda)J(\theta), \quad \forall \lambda \in [0,1], \overline{\theta}, \theta \in \Theta. \tag{2.2}$$

2. The effective domain of J is defined as $\mathrm{dom}(J) := \{\theta \in \Theta : J(\theta) \neq \infty\}$ and J is called proper if $\mathrm{dom}(J) \neq \emptyset$.

In the following we want to consider functionals J that are convex, but not necessarily differentiable. Therefor, we define the subdifferential.

Definition 2.2. of a convex and proper functional $J : \Theta \to (-\infty, \infty]$ at a point $\theta \in \Theta$ as

$$\partial J(\theta) := \left\{ p \in \Theta : J(\theta) + \langle p, \overline{\theta} - \theta\rangle \leq J(\overline{\theta}), \ \forall \overline{\theta} \in \Theta \right\}. \tag{2.3}$$

If J is differentiable, then the subdifferntial coincides with the classical gradient (or Fréchet derivative). We denote $\mathrm{dom}(\partial J) := \{\theta \in \Theta : \partial J(\theta) \neq \emptyset\}$ and observe that $\mathrm{dom}(\partial J) \subset \mathrm{dom}(J)$.

The main algorithm in this section are so-called Bregman iterations, for which we first define the Bregman distance.

Definition 2.3 (Bregman Distance). Let $J : \Theta \to (-\infty, \infty]$ be a proper, convex functional. Then we define for $\theta \in \mathrm{dom}(\partial J), \overline{\theta} \in \Theta$

$$D_J^p(\overline{\theta}, \theta) := J(\overline{\theta}) - J(\theta) - \langle p, \overline{\theta} - \theta\rangle, \quad p \in \partial J(\theta). \tag{2.4}$$

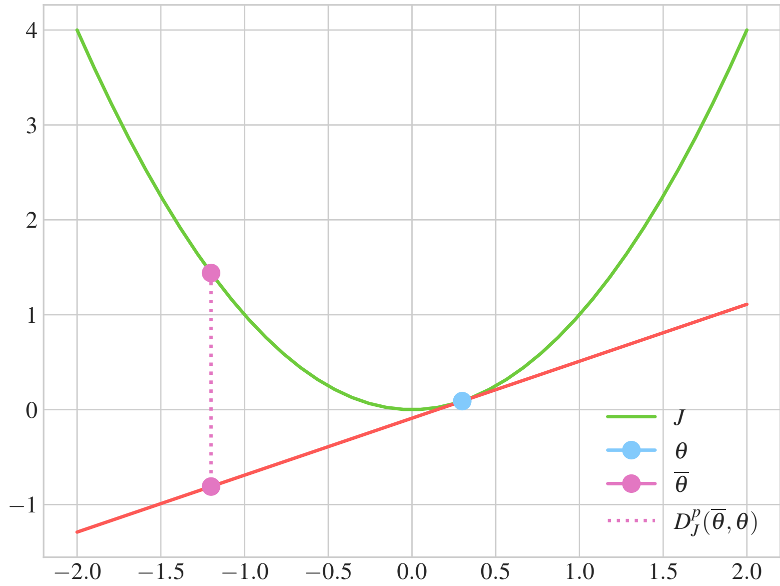


Figure 2.1.: Visualization of the Bregman distance.

For $p \in \partial J(\theta)$ and $\overline{p} \in \partial J(\overline{\theta})$ we define the *symmetric* Bregman distance as

$$D_J^{\mathrm{sym}}(\overline{\theta},\theta) := D_J^p(\overline{\theta},\theta) + D_J^{\overline{p}}(\theta,\overline{\theta}). \tag{2.5}$$

Intuitively, the Bregman distance $D_J^p(\overline{\theta},\theta)$, measures the distance of J to its linearization around θ, see Fig. 2.1. If J is differentiable, then the subdifferential is single valued—we can suppress the sup script p—and we have

$$D_J(\overline{\theta},\theta) = J(\overline{\theta}) - J(\theta) - \langle \nabla J(\theta), \overline{\theta} - \theta\rangle.$$

Example 2.4. For $\Theta = \mathbb{R}^n$ and $J = \frac{1}{2}\|\cdot\|_2^2$ we see that $\partial J(\theta) = \{\theta\}$ and therefore

$$\begin{aligned} D_J^p(\overline{\theta},\theta) &= \frac{1}{2}\langle\overline{\theta},\overline{\theta}\rangle - \frac{1}{2}\langle\theta,\theta\rangle - \langle\theta,\overline{\theta}-\theta\rangle \\ &= \frac{1}{2}\langle\overline{\theta},\overline{\theta}\rangle + \frac{1}{2}\langle\theta,\theta\rangle - \langle\theta,\overline{\theta}\rangle \\ &= \frac{1}{2}\|\overline{\theta}-\theta\|_2^2 = J(\overline{\theta}-\theta). \end{aligned}$$

We can easily see that in general it is neither definite, symmetric nor fulfills the triangle inequality, hence it is not a metric. However, it fulfills the two distance axioms

$$D_J^p(\overline{\theta},\theta) \geq 0, \quad D_J^p(\theta,\theta) = 0, \quad \forall\overline{\theta}\in\Theta, \theta\in\mathrm{dom}(\partial J). \tag{2.6}$$

The same holds for the symmetric Bregman distance, where additionally—as the name suggests—the symmetry property is fulfilled. The last concept that is crucial in [BREG-I] is the so-called proximal operator.

Definition 2.5. Let $J : \Theta \to (-\infty, \infty]$ be convex, proper and lower semicontinuous functional, then we define the *proximal operator* as

$$\mathrm{prox}_J(\overline{\theta}) := \mathrm{argmin}_{\theta \in \Theta} \frac{1}{2} \|\theta - \overline{\theta}\|^2 + J(\theta).$$

If J is additionally a closed function, i.e., its sublevel sets

$$N_\alpha = \{\theta \in \mathrm{dom}\, J : J(\theta) \leq \theta\}$$

are closed for every $\alpha \in \mathbb{R}$ then we have that the function $\tilde{J} = \frac{1}{2}\|\theta - \cdot\|^2 + J(\theta)$ is closed, proper and *strongly* convex and therefore has a unique minimizer, see [Roc97, Thm. 27.1].

Example 2.6. If $J = \|\cdot\|$ is a norm and $\lambda > 0$ then we have that (see e.g. [**parikh2014proximal**])

$$\mathrm{prox}_{\lambda J}(\overline{\theta}) = \overline{\theta} - \mathrm{Proj}_{\|\cdot\|^*}(\overline{\theta}/\lambda)$$

where $\mathrm{Proj}_{\|\cdot\|^*}$ denotes the projection operator w.r.t. the dual norm $\|\theta\|^* = \sup\{|\langle f, \theta \rangle| : f \in \Theta^*\}$. In the case of ℓ^p norms on $\mathbb{R}^n$ we know that that

$$\|\theta\|_p^* = \|\theta\|_q$$

with $1/p + 1/q = 1$ with the notational convention of $1/\infty = 0$. Especially relevant are the cases $p \in \{1, 2\}$. Here, we then have that

$$\mathrm{prox}_{\lambda\|\cdot\|_2}(\overline{\theta}) = \overline{\theta}\left(1 - \min\left\{\frac{\lambda}{\|\overline{\theta}\|_2}, 1\right\}\right) = \begin{cases} \overline{\theta}\,(1 - \lambda/\|\overline{\theta}\|_2) & \text{if } \|\overline{\theta}\|_2 \geq \lambda \\ 0 & \text{else} \end{cases}$$

and for $i = 1, \dots, n$

$$\mathrm{prox}_{\lambda\|\cdot\|_1}(\overline{\theta})_i = \mathrm{sign}(\overline{\theta}_i)\, \max\left\{\left|\overline{\theta}_i\right| - \lambda, 0\right\} = \begin{cases} \overline{\theta}_i - \lambda & \text{if } \overline{\theta} > \lambda \\ 0 & \text{if } \left|\overline{\theta}_i\right| \leq \lambda \\ \overline{\theta}_i + \lambda & \text{if } \overline{\theta} < -\lambda \end{cases}$$

the so called *soft thresholding operator*.

Example 2.7 (Group Norms). Another relevant functional J is the group norm $\ell_{1,2}$ that—in the context of sparse neural networks—was first employed by [**scardapane2017group**]. Here, we assume that the parameters in Θ can be grouped in a collection of parameters $\mathcal{G}$, for which we the choose

$$J(\theta) = \sum_{g \in \mathcal{G}} \sqrt{\#\mathcal{G}} \|g\|_2$$

In this case the proximal operator is given as

$$\operatorname{prox}_{\lambda J}(\overline{\theta})_g = g \max \left\{ 1 - \min \left\{ \frac{\lambda \sqrt{\#\mathcal{G}}}{\|g\|_2}, 1 \right\}, 0 \right\}$$

Example 2.8 (Elastic Net). For a convex functional J we also consider the elastic net version $J_\delta = J + \frac{1}{2\delta}\|\cdot\|^2$ in the following. Here we see that

$$\begin{aligned}
\operatorname{prox}_{\lambda J_\delta}(\overline{\theta}) &= \operatorname{argmin}_{\theta \in \Theta} \frac{1}{2}\|\theta - \overline{\theta}\|^2 + \lambda J(\theta) + \frac{1}{2\delta}\|\theta\|^2 \\
&= \operatorname{argmin}_{\theta \in \Theta} \frac{1+\delta}{2\delta}\|\theta\|^2 - \frac{1}{2}\langle \theta, \overline{\theta} \rangle + \lambda J(\theta) \\
&= \operatorname{argmin}_{\theta \in \Theta} \frac{1}{2}\|\theta\|^2 - \frac{1}{2}\left\langle \theta, \frac{\delta}{1+\delta}\overline{\theta} \right\rangle + (1+\delta)^{-1}\lambda J(\theta) \\
&= \operatorname{prox}_{\frac{\delta}{1+\delta}\lambda J}\left(\frac{\delta}{1+\delta}\overline{\theta} \right).
\end{aligned}$$

Setting $\tilde{\delta} = \delta(1+\delta)^{-1}$ we see that for $J = \|\cdot\|_1$ we have that

$$\operatorname{prox}_{\lambda J_\delta}(\overline{\theta})_i = \operatorname{sign}(\tilde{\delta}\,\overline{\theta}_i) \max\left\{ \left|\tilde{\delta}\,\overline{\theta}_i\right| - \lambda, 0 \right\} = \operatorname{sign}(\overline{\theta}_i)$$

The the optimality conditions yield for $\theta = \operatorname{prox}_J(\overline{\theta})$

$$\theta - \overline{\theta} \in \partial J(\theta).$$

If J is differentiable, we then have

$$\theta - \nabla J(\theta) = \overline{\theta} \Leftrightarrow \theta = (I + \nabla J)^{-1}(\overline{\theta}).$$

We first consider the following implicit Euler scheme, for a step size $\tau > 0$

check this!

$$\theta^{(k+1)} = \operatorname{argmin}_{\theta \in \Theta} D_J^{p^{(k)}}\left(\theta, \theta^{(k)}\right) + \tau \mathcal{L}(\theta), \tag{2.7a}$$

$$p^{(k+1)} = p^{(k)} - \tau \nabla \mathcal{L}(\theta^{(k+1)}) \in \partial J(\theta^{(k+1)}) \tag{2.7b}$$

Minimizing Movment proximal point

which is known as the *Bregman iteration* [Osh+05]. The intuitive interpretation here is, that in each step we want to minimize $\mathcal{L}$ while also being close to the previous iterate in terms of the Bregman distance induced by J. Therefore, the very nature of Bregman iterations means starting with a iterate $\theta^{(0)}$ that has a low value in J—preferably $J(\theta^{(0)}) = 0$—and only increase $J(\theta^{(k)})$ gradually as k increases.

Remark 2.9. Originally, the iterations were employed for solving inverse problems. Here, we are given a forward operator $A : \Theta \to \tilde{\Theta}$ and a noisy measurement $f = A\theta + \delta$ where $\delta \in \tilde{\Theta}$ is additive noise. The loss function is then of the form

$$\mathcal{L} = \frac{1}{2}\|A \cdot -f\|_2^2$$

for which one can show that the Bregman iterations converge to a solution of

$$\min \{J(\theta) : A\theta = f\}, \tag{2.8}$$

see e.g. [Osh+05]. In comparison the concept of adding a regularizing term with parameter $\lambda > 0$, i.e. considering the problem

$$\min_{\theta} \mathcal{L}(\theta) + \lambda J(\theta)$$

actually modifies the minimizers. In this sense Bregman iterations do not introduce a bias. △

Example 2.10. In order to get an intuition about the behavior of Bregman iterations, we consider an image denoising task. I.e. we are given a noisy image $\mathbb{R}^{n\times m} \ni f = u+\delta$ where $\delta \in \mathbb{R}^{n\times m}$ is additive noise. In order to obtain $u \in \mathbb{R}^{n\times n}$ from f we employ the TV functional

$$J(u) = TV(u) := MISSING,$$

together with the loss function $\mathcal{L}(u) := \frac{1}{2}\|u - f\|_2^2$. We start with an image $u^{(0)}$ such that $TV(u^{(0)}) = 0$, i.e. a constant image. In Fig. 2.2 we visualize the iteration. At lower iterations $u^{(k)}$ only displays features on a larger scale, while at the end, the iteration converges back to smallest possible scale, the noisy data. In order to obtain an appropriate denoising, one needs to employ a early stopping here. This fits well to the insight from Eq. (2.8) since here the forward operator is the identity. I.e.

$$\{u : \frac{1}{2}\|u - f\|^2 = 0\} = \{f\}.$$

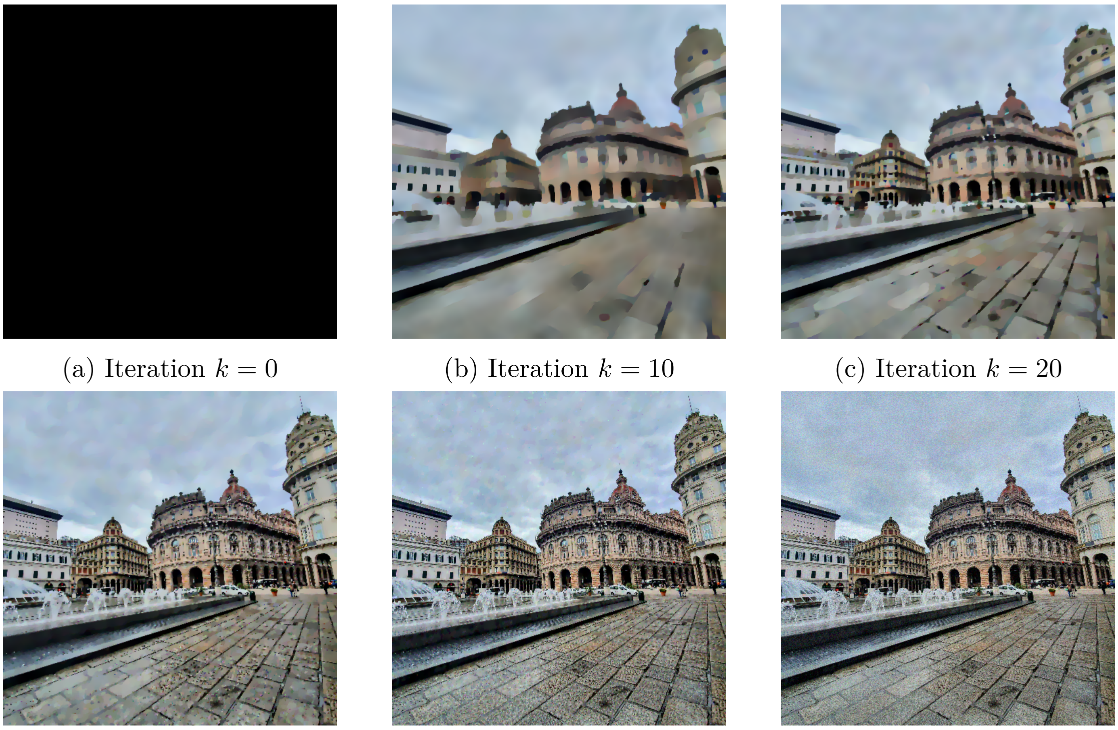
It should also be noted that this example only serves a explanatory purpose. In practice directly applying Eq. (2.7) for $J = TV$ can become infeasible since the first minimization problem is expensive.

If $J = \frac{1}{2}\|\cdot\|_2^2$ as in Example 2.4 then this amounts to the step

$$\theta^{(k+1)} = \text{argmin}_{\theta\in\Theta} \frac{1}{2}\|\theta - \theta^{(k)}\|_2^2 + \tau\mathcal{L}(\theta),$$

where the optimality conditions then yield

$$\theta^{(k+1)} - \theta^{(k)} + \tau\nabla\mathcal{L}(\theta^{(k+1)}) = 0 \Leftrightarrow \theta^{(k+1)} = \theta^{(k)} - \tau\nabla\mathcal{L}(\theta^{(k+1)})$$



(a) Iteration $k = 0$ (b) Iteration $k = 10$ (c) Iteration $k = 20$

(d) Iteration $k = 50$ (e) Iteration $k = 100$ (f) Iteration $k = 500$

Figure 2.2.: Bregman iterations for image denoising in Example 2.10

which is a standard implicit Euler iteration. The time continuous flow for $\tau \to 0$ is known as the *inverse scale space* flow [Bur+06; Bur+07],

$$\begin{cases} \dot{p}_t = -\nabla\mathcal{L}(\theta_t), \\ p_t \in \partial J(\theta_t), \end{cases}$$

where again for $J = \frac{1}{2}\|\cdot\|_2^2$ we obtain that $\partial J(\theta_t) = \theta_t$ and therefore obtain the standard gradient flow. Hence we see, that the inverse scale space flow is a generalization of the standard gradient flow.

2.3.2. Linearized Bregman Iterations and Mirror Descent

The minimization step in Eq. (2.7) is infeasible for large scale applications, especially in our setting of neural networks. Therefore, we employ the idea introduced in [Yin+08; COS09]. Here, we first linearize the loss function around the previous iterate,

$$\mathcal{L}(\theta) \approx \mathcal{L}(\theta^{(k)}) + \left\langle \nabla\mathcal{L}(\theta^{(k)}), \theta - \theta^{(k)} \right\rangle.$$

The next step is to replace J with the strongly convex elastic net regularization

$$J_\delta := J + \frac{1}{2\delta}\|\cdot\|_2^2. \tag{2.9}$$

The minimization step then transforms to

$$\begin{aligned} &\mathrm{argmin}_{\theta\in\Theta} D_{J_\delta}^{p^{(k)}}\left(\theta, \theta^{(k)}\right) + \tau\left\langle \nabla\mathcal{L}(\theta^{(k)}), \theta \right\rangle \qquad (2.10)\\ &\quad= \mathrm{argmin}_{\theta\in\Theta}\, J(\theta) + \frac{1}{2\delta}\|\theta\|_2^2 - \langle p^k, \theta\rangle + \tau\left\langle \nabla\mathcal{L}(\theta^{(k)}), \theta \right\rangle \\ &\quad= \mathrm{argmin}_{\theta\in\Theta}\, J(\theta) + \frac{1}{2\delta}\|\theta - \delta(p^{(k)} - \tau\nabla\mathcal{L}(\theta^{(k)})\|_2^2 - \underbrace{\|p^{(k)} - \tau\nabla\mathcal{L}(\theta^{(k)})\|_2^2}_{\text{constant in }\theta} \\ &\quad= \mathrm{prox}_{\delta J}\left(\delta\left(p^{(k)} - \tau\nabla\mathcal{L}(\theta^{(k)})\right)\right). \end{aligned}$$

Note that here $p^{(k)}$ is a subgradient of J_δ at θ therefore we derive the subgradient update rule

$$p^{(k+1)} := p^{(k)} - \tau\mathcal{L}(\theta^{(k)}).$$

This finally yields the linearized Bregman iterations

$$p^{(k+1)} = p^{(k)} - \tau\nabla\mathcal{L}(\theta^{(k)}), \tag{2.11a}$$
$$\theta^{(k+1)} = \mathrm{prox}_{\delta J}(\delta p^{(k+1)}). \tag{2.11b}$$

The last line is equivalent to $p^{(k+1)} \in \partial J_\delta(\theta^{(k+1)})$ for which we obtain the continuous linearized flow

$$\begin{aligned} \dot{p}_t &= -\nabla\mathcal{L}(\theta_t), \\ p_t &\in \partial J_\delta(\theta_t). \end{aligned}$$

As already noticed by [**??**] linearized Bregman iteration are equivalent to mirror descent in some situations. We show the equivalence in the following, where we employ similar arguments as in [**Duchi**]. One assumes to be given a differentiable and strongly convex function $h : \Theta \to \mathbb{R}$, i.e.,

A lot of citations missing here

$$h(\overline{\theta}) - h(\theta) - \langle \nabla h(\theta), \overline{\theta} - \theta \rangle \geq \frac{1}{2} \|\overline{\theta} - \theta\|_2^2$$

for all $\theta, \overline{\theta} \in \Theta$. The mirror descent update then reads ([NY83; BT03])

$$\theta^{(k+1)} = \nabla h^* \left(\nabla h \left(\theta^{(k)} \right) - \tau \mathcal{L}(\theta^{(k)}) \right) \tag{2.12}$$

where h^* denotes the Fenchel conjugate

$$h^*(p) = \sup_{\theta} \langle p, \theta \rangle - h(\theta)$$

with the gradient

$$\nabla h^*(p) = \operatorname{argmax}_\theta \left\{ \langle p, \theta \rangle - h(\theta) \rangle \right\}.$$

Therefore, we see that Eq. (2.12) can be written as

$$\begin{aligned} \theta^{(k+1)} &= \operatorname{argmax}_\theta \left\{ \left\langle \nabla h \left(\theta^{(k)} \right) - \tau \mathcal{L}(\theta^{(k)}), \theta \right\rangle - h(\theta) \right\} \\ &= \operatorname{argmax}_\theta \left\{ -D_h(\theta, \theta^{(k)}) - \tau \left\langle \mathcal{L}(\theta^{(k)}), \theta \right\rangle \right\} \\ &= \operatorname{argmin}_\theta \left\{ D_h(\theta, \theta^{(k)}) + \tau \left\langle \mathcal{L}(\theta^{(k)}), \theta \right\rangle \right\} \end{aligned}$$

which was our starting point to derive linearized Bregman iterations for $h = J_\delta$ in Eq. (2.10). In fact, we can always find a convex functional $J : \Theta \to \mathbb{R}$ such that $h = J + \frac{1}{2}\| \cdot \|_2^2$ for which we see, that Eq. (2.11) is a more general formulation of Eq. (2.12).

2.3.3. Stochastic and Momentum Variants

We want to employ linearized Bregman iterations to train a neural network. As mentioned in Section 2.1.1 we therefore do not compute the full gradient of $\mathcal{L}$ but rather a minibatched variant. This yields stochastic Bregman Iterations

$$\begin{aligned} &\text{draw } \omega^{(k)} \text{ from } \Omega \text{ using the law of } \mathbb{P}, \\ g^{(k)} &:= g(\theta^{(k)}; \omega^{(k)}), \\ v^{(k+1)} &:= v^{(k)} - \tau^{(k)} g^{(k)}, \\ \theta^{(k+1)} &:= \operatorname{prox}_{\delta J}(\delta v^{(k+1)}). \end{aligned} \tag{2.13}$$

The basic update scheme is given as the linearized Bregman iterations from[Osh+05], however the presence of a stochastic gradient estimator significantly complicates the convergence analysis, as observed in Section 2.3.4. However, this algorithm can now be efficiently employed to train a neural network. For the analogous stochastic mirror descent algorithm we refer to [Nem+09].

Momentum Variant Typically, the learning process of a neural network can be improved by introducing a momentum term (see e.g. [Nes83; Qia99]) in the optimizer. In our case this can be achieved, by replacing the gradient update on the subgradient variable. In [BREG-I] we first consider the inertia version of the gradient flow as

$$\begin{cases}\gamma \ddot{v}_t + \dot{v}_t = -\nabla\mathcal{L}(\theta_t),\\ v_t \in \partial J_\delta(\theta_t).\end{cases}$$

for which the discretization then reads

$$\begin{aligned} m^{(k+1)} &= \beta^{(k)} m^{(k)} + (1-\beta^{(k)})\tau^{(k)} g^{(k)},\\ v^{(k+1)} &= v^{(k)} - m^{(k+1)},\\ \theta^{(k+1)} &= \operatorname{prox}_{\delta J}(\delta v^{(k+1)}). \end{aligned} \tag{2.14}$$

Adamized Bregman Iteration We shortly remark that one can replace the momentum update in Eq. (2.14) with a Adam update [KB14]. This then yields an Adamized version of linearized Bregman iterations as employed in [BREG-I].

2.3.4. Convergence of Stochastic Bregman Iterations

While various previous works prove convergence of linearized Bregman iterations (see e.g. [Osh+05; COS09]), the stochastic setting requires special treatment. In [BREG-I] the first guarantees for the algorithm in Eq. (2.13) were proven. Other work on convergence of stochastic Bregman iterations [DEH21; HR21; ZH18; DOr+21] requires a differentiable functional J. However, since our main motivation is to a functional in the flavor of the ℓ^1 norm this is not applicable. Therefore, we present the novel convergence analysis of [BREG-I].

Assumptions on the Gradient Estimator In order to obtain convergence guarantees, we need to assume mainly two properties on the gradient estimator $g(\cdot,\cdot)$. First we assume unbiasedness, which means

$$\mathbb{E}\left[g(\theta;\omega)\right] = \nabla\mathcal{L}(\theta) \text{ for all } \theta\in\Theta.$$

The second assumption we need in the following is referred to as *bounded variance* of the estimator.

Assumption 2.11 (Bounded variance). There exists a constant $\sigma > 0$ such that for any $\theta\in\Theta$ it holds

$$\mathbb{E}\left[\|g(\theta;\omega) - \nabla\mathcal{L}(\theta)\|^2\right] \leq \sigma^2. \tag{2.15}$$

Remark 2.12. We want to remark, that this property is weaker than the bounded gradient assumption

$$\mathbb{E}\left[\|g(\theta;\omega)\|^2\right] \leq C$$

for some constant $C > 0$. In fact this condition can not be enforced together with a strong convexity assumption—which we employ in **??**—as shown in [Ngu+18]. △

Assumptions on the Regularizer and on the Loss Function The assumptions on the regularization functional J are mild and merely ensure the well-definedness of the proximal mapping,

Assumption 2.13 (Regularizer). We assume that $J : \Theta \to (-\infty, \infty]$ is a convex, proper, and lower semicontinuous functional on the Hilbert space Θ.

Our assumptions on the loss function $\mathcal{L}$ are more restrictive. We require it to be bounded from below and differentiable, which are both standard assumptions. Additionally, we require Lipschitz continuity of the gradient, which also commonly employed in optimization literature.

Assumption 2.14 (Loss function). We assume the following conditions on the loss function:

- The loss function $\mathcal{L}$ is bounded from below and without loss of generality we assume $\mathcal{L} \geq 0$.
- The function $\mathcal{L}$ is continuously differentiable.
- The gradient of the loss function $\theta \mapsto \nabla\mathcal{L}(\theta)$ is L-Lipschitz for $L \in (0, \infty)$:

$$\|\nabla\mathcal{L}(\tilde{\theta}) - \nabla\mathcal{L}(\theta)\| \leq L\|\tilde{\theta} - \theta\|, \quad \forall \theta, \tilde{\theta} \in \Theta. \tag{2.16}$$

If the loss function $\mathcal{L}$ fulfills the previous assumptions we are able to prove loss decay of the iterates in Theorem 2.18. However, in order to show convergence of the iterates we additionally need a convexity assumption. For a differentiable functional J, the authors in [DEH21] assumed

$$\nu\ D_J(\overline{\theta}, \theta) \leq D_{\mathcal{L}}(\overline{\theta}, \theta)$$

which for twice differentiable $J, \mathcal{L}$ transfers to

$$\nu\ \nabla^2 J \lesssim \nabla^2 \mathcal{L}, \qquad \forall \overline{\theta}, \theta \in \Theta.$$

Plugging in the definition of the Bregman dist $D_{\mathcal{L}}$ we obtain

$$\nu\ D_J(\overline{\theta}, \theta) \leq \mathcal{L}(\overline{\theta}) - \mathcal{L}(\theta) - \langle \nabla\mathcal{L}(\theta), \overline{\theta} - \theta \rangle.$$

In this form one observes that this is in fact a convexity assumption on $\mathcal{L}$ in a J dependent distance, as employed in [BREG-I].

Assumption 2.15 (Strong convexity). For a proper convex function $H : \Theta \to \mathbb{R}$ and $\nu \in (0, \infty)$, we say that the loss function $\theta \mapsto \mathcal{L}(\theta)$ is ν-strongly convex w.r.t. H, if

$$\mathcal{L}(\overline{\theta}) \geq \mathcal{L}(\theta) + \langle \nabla \mathcal{L}(\theta), \overline{\theta} - \theta \rangle + \nu D_J^p(\overline{\theta}, \theta), \quad \forall \theta, \overline{\theta} \in \Theta, p \in \partial H(\theta). \tag{2.17}$$

Remark 2.16. We have two relevant cases for the choice of H. For $H = \frac{1}{2}\|\cdot\|^2$ Assumption 2.15 reduces to standard strong ν-convexity. The other relevant case, is $H = J_\delta$, i.e. we consider convexity w.r.t. to the functional J_δ. △

Remark 2.17. In the setting of training a neural network, where we employ the empirical loss Eq. (2.1), this convexity assumption usually fails. While it is possible to enforce this conditions only locally around the minimum, this does not significantly improve the applicability. For future work, it would be desirable to enforce a Kurdyka–Łojasiewicz inequality, as in [Ben+21] for the deterministic case. △

Loss Decay The first convergence result considers the loss decay of the iterates. Here, we do not assume convexity of the loss function. Under this assumptions [Ben+21; BB18] were able to show the inequality

$$\mathbb{E}\left[\mathcal{L}(\theta^{(k+1)})\right] + \frac{1}{\tau^{(k)}}\mathbb{E}\left[D_J^{\mathrm{sym}}(\theta^{(k+1)}, \theta^{(k)})\right] + \frac{C}{2\delta\tau^{(k)}}\mathbb{E}\left[\|\theta^{(k+1)} - \theta^{(k)}\|^2\right] \\ \leq \mathbb{E}\left[\mathcal{L}(\theta^{(k)})\right].$$

In our setting, employing a stochastic gradient estimator, we are able to prove a similar estimate. Here, we obtain an additional term scaled σ which controls the expected squared difference between the gradient estimator and the actual gradient. It should however be noted, that the proof is not only a trivial extension.

Theorem 2.18 ([BREG-I, Th. 2]: Loss decay). Assume that Assumptions 2.11, 2.13 and 2.14 hold true, let $\delta > 0$, and let the step sizes satisfy $\tau^{(k)} \leq \frac{2}{\delta L}$. Then there exist constants $c, C > 0$ such that for every $k \in \mathbb{N}$ the iterates of (2.13) satisfy

$$\mathbb{E}\left[\mathcal{L}(\theta^{(k+1)})\right] + \frac{1}{\tau^{(k)}}\mathbb{E}\left[D_J^{\mathrm{sym}}(\theta^{(k+1)}, \theta^{(k)})\right] + \frac{C}{2\delta\tau^{(k)}}\mathbb{E}\left[\|\theta^{(k+1)} - \theta^{(k)}\|^2\right] \\ \leq \mathbb{E}\left[\mathcal{L}(\theta^{(k)})\right] + \tau^{(k)}\delta\frac{\sigma^2}{2c}, \tag{2.18}$$

Convergence of the Iterates Here, we have two cases respectively proving convergence w.r.t. the L^2 distance and the Bregman distance of J_delta. The first assumes strong convexity with $H = \frac{1}{2}\|\cdot\|^2$ in Assumption 2.15.

Theorem 2.19 ([BREG-I, Th. 6]: Convergence in norm). Assume that Assumptions 2.11, 2.13 and 2.14 and Assumption 2.15 for $H = \frac{1}{2}\|\cdot\|^2$ hold true and let $\delta > 0$. Furthermore, assume that the step sizes $\tau^{(k)}$ are such that for all $k \in \mathbb{N}$:

$$\tau^{(k)} \leq \frac{\mu}{2\delta L^2}, \qquad \tau^{(k+1)} \leq \tau^{(k)}, \qquad \sum_{k=0}^{\infty} (\tau^{(k)})^2 < \infty, \qquad \sum_{k=0}^{\infty} \tau^{(k)} = \infty.$$

The function $\mathcal{L}$ has a unique minimizer θ^* and if $J(\theta^*) < \infty$ the stochastic linearized Bregman iterations (2.13) satisfy the following:

- Letting $d_k := \mathbb{E}\left[D_{J_\delta}^{v^{(k)}}(\theta^*, \theta^{(k)})\right]$ it holds

$$d_{k+1} - d_k + \frac{\mu}{4}\tau^{(k)}\mathbb{E}\left[\|\theta^* - \theta^{(k+1)}\|^2\right] \leq \frac{\sigma}{2}\left((\tau^{(k)})^2 + \mathbb{E}\left[\|\theta^{(k)} - \theta^{(k+1)}\|^2\right]\right). \tag{2.19}$$

- The iterates possess a subsequence converging in the L^2-sense of random variables:

$$\lim_{j \to \infty} \mathbb{E}\left[\|\theta^* - \theta^{(k_j)}\|^2\right] = 0. \tag{2.20}$$

Here, J_δ is defined as in (2.9).

For the second result we assume convexity w.r.t. the Bregman distance, i.e. we choose $H = J_\delta$ in Assumption 2.15. This induces a relation between the Bregman distance of J and the loss function $\mathcal{L}$, which has been similarly employed in [DEH21].

Theorem 2.20 ([BREG-I, Th. 11]: Convergence in the Bregman distance). Assume that Assumptions 2.11, 2.13 and 2.14 and Assumption 2.15 for $H = J_\delta$ hold true and let $\delta > 0$. The function $\mathcal{L}$ has a unique minimizer θ^* and if $J(\theta^*) < \infty$ the stochastic linearized Bregman iterations (2.13) satisfy the following:

- Letting $d_k := \mathbb{E}\left[D_{J_\delta}^{v^{(k)}}(\theta^*, \theta^{(k)})\right]$ it holds

$$d_{k+1} \leq \left[1 - \tau^{(k)}\nu\left(1 - \tau^{(k)}\frac{2\delta^2 L^2}{\nu}\right)\right] d_k + \delta(\tau^{(k)})^2\sigma^2. \tag{2.21}$$

- For any $\varepsilon > 0$ there exists $\tau > 0$ such that if $\tau^{(k)} = \tau$ for all $k \in \mathbb{N}$ then

$$\limsup_{k \to \infty} d_k \leq \varepsilon. \tag{2.22}$$

- If $\tau^{(k)}$ is such that

$$\lim_{k \to \infty} \tau^{(k)} = 0 \quad \text{and} \quad \sum_{k=0}^{\infty} \tau^{(k)} = \infty \tag{2.23}$$

then it holds

$$\lim_{k\to\infty} d_k = 0. \tag{2.24}$$

Here, J_δ is defined as in (2.9).

2.3.5. Numerical Results and Practical Considerations

Before briefly reviewing the numerical results in [BREG-I, Sec. 4], we remark on some practical considerations. In particular we comment on the parameter initialization strategy.

Parameter Initialization As already noticed in [GB10] parameter initialization has a significant impact on the training of the neural network. Here, in contrast to standard Bregman methods, we are not able to initialize the parameters of the neural network as $\theta = 0$. This is due to that fact, that a zero initialization induces symmetries in the network weights, for which one cannot utilize the full expressivity of the architecture [GBC16, Ch. 6]. Therefore, we rather employ the approach from [Liu+21; DZ19; Mar10] of sparsifying weight matrices $\tilde{W}^l \in \mathbb{R}^{n_{l+1}\times n_l}$ up to a certain level, by a pointwise multiplication with a binary mask $M^l \in 0,1^{n_{l+1}\times n_l}$

$$W^l := \tilde{W}^l \odot M^l.$$

Each entry in M^l is i.i.d. sampled from a Bernoulli distribution

$$M^l_{i,j} \sim \mathcal{B}(r).$$

where the parameter r determines the sparsity

$$\mathrm{N}(W^l) := \frac{\|W^l\|_0}{n_l \cdot n_{l-1}} = 1 - \mathrm{S}(W^l).$$

In [GB10] the authors advise to especially control to the variance of the parameter initialization distribution, for which in [BREG-I] we derive

$$\mathrm{Var}\left[\tilde{W}^l\right] = \frac{1}{r}\,\mathrm{Var}\left[\tilde{W}^l \odot M^l\right] \tag{2.25}$$

and therefore scale the weights with the sparsity parameter r at initialization.

Choice of Regularizers In all our experiments we choose a L^1 type sparsity promoting regularization function functional J. We do not employ any coupling between weight matrices of different layers, and therefore for $\theta = ((W^1, b^1), \dots, (W^L, b^L))$ we have

$$J(\theta) = \sum_{l=1}^{L} J^l(W^l)$$

where J^l is choosen according to the layer type. In the easiest case of a fully connected layer, we can choose

$$J^l(W^l) := \|W^l\|_1.$$

In the case of a convolutional layer we have that W^l is determined by convolutional kernels $K_{i,j} \in \mathbb{R}^{k\times k}$, see Section 2.1. Here, we typically employ a group sparsity term in the form

$$J^l(W^l) = \|W^l\|_{2,1} = \sum_{i,j} \|K_{i,j}\|_2.$$

The outer sum acts as a L^1 regularizer on the instances $\|K_{i,j}\|$. Sparsity in this sense, then amounts to having indices (i,j) for which $\|K_{i,j}\|_2 = 0 \Leftrightarrow K_{i,j} = 0$, i.e., we prune away whole convolutional filters. This effect is displayed in [BREG-I, Fig. 1]. We can also employ group sparsity on fully connected layers, by considering row sparsity of $W^l \in \mathbb{R}^{n_{l+1}, n_l}$s

$$J^l(W^l) = \sum_{i=1}^{n_{l+1}} \|W_{i,:}\|_2 = \sum_{i=1}^{n_{l+1}} \sqrt{\sum_{js=1}^{n_l} W_{i,j}^2}.$$

In this setting we have a L^1 penalty on the row norms $\|W_{i,:}\|_2$ which therefore enforces whole rows to be zero. This is relevant, if we employ a layer architecture with $\Psi^l(0) = 0$, e.g., using no bias vectors and the ReLU activation function. In this setting, if the ith row of W^l is zero this effectively means, that the ith neuron in layer $l+1$ is inactive. This observation allows the neural architecture search in one of the following paragraphs.

Comments on the Numerical Results We briefly remark on the numerical results as displayed in [BREG-I, Sec. 4]. In the experiments we employed feed-forward networks with simple linear, convolutional and residual layers and tested on the three datasets [Kri09; XRV17; LC10].

The basic comparison between the algorithms SGD, ProxGD and LinBreg shows the qualitative behaviour of each iterations. Infantilizing sparse does not have any effect on SGD, since it does not preserve the sparsity in any way. ProxGD rather starts with many active parameters and reduces the this number during the iteration. Only the discretization of the inverse scale space flow—via Bregman iterations—shows the desired behaviour of gradually adding active parameters. Furthermore, in [BREG-I, Fig. 2] we can see, that the choice of λ in the regularizer $J = \lambda \|\cdot\|_1$ changes the results significantly. In the light of Eq. (2.8) this is not expected for the standard Bregman iterations with a convex loss. It is therefore interesting to see, that in our non-convex and stochastic situation this effect changes.

The momentum variants, as discussed in Section 2.3.3 yield the desired effect of enhancing the validation accuracy, and respectively converging faster. However in each of the experiments, one can also observe that adding a momentum term has the effect that more parameters are added faster. On the one hand this could mean that the network actually requires more parameters to have a higher accuracy, for which a momentum variant is more likely to increase the number of needed parameters. However, the quantitative evaluation on the CIFAR10 dataset [Kri09] shows, that especially the Adamized version tends to increase the number of used parameters rather aggressively, while only slightly increasing the performance of the net. The performance here is very similar to the one of proximal gradient descent. However, the training of a residual network seems to be slightly better with a standard Lasso implementation. Here, one neglects the non-differentiability of the L^1 norm and computes a derivate via automatic differentiation [Ral81; MDA15] (we employed the `autograd` library of the `PyTorch` package [Pas+19]). In order to obtain true zeros in the weight matrix one then has to employ a thresholding operation after the training. In some sense this method is not a proper sparse training approach, but rather a regularization method with an added pruning step at the end.

Comments on Efficiency One of the major advantages of the Bregman approach, is that the network is sparse already during the training time. As with all sparse-to-sparse training approaches this yields a very small number of active parameters over all training step. This sparsity can be easily exploited in each forward pass. However, it is not directly possible to achieve performance gains during the backward pass of the network, since in general

$$W_{ij}^l = 0 \not\Rightarrow \partial_{W_{ij}^l} \mathcal{L}(\theta) = 0.$$

In **????** there are no evaluation on the training time and memory consumption of the Bregman algorithm. Since the complexity does not increase in comparison to the standard SGD, one hopes to obtain a faster training time here. This is an interesting open question for future work.

Neural Architecture Search

2.4. Resolution Stability

Part II.

Prints

Bibliography

Books

[AF03] R. A. Adams and J. J. Fournier. *Sobolev spaces.* Elsevier, 2003.

[Lin17] P. Lindqvist. *Notes on the p-Laplace equation.* 161. University of Jyväskylä, 2017.

[BB11] H. Brezis and H. Brézis. *Functional analysis, Sobolev spaces and partial differential equations.* Vol. 2. 3. Springer, 2011.

[Sch69] J. T. Schwartz. *Nonlinear Functional Analysis -.* Boca Raton, Fla: CRC Press, 1969.

[Bra02] A. Braides. *Gamma-convergence for Beginners.* Vol. 22. Oxford University Press, Oxford, 2002.

[Dal12] G. Dal Maso. *An introduction to Γ-convergence.* Vol. 8. Springer Science & Business Media, 2012.

[DS88] N. Dunford and J. T. Schwartz. *Linear operators, part 1: general theory.* Vol. 10. John Wiley & Sons, 1988.

[Roc97] R. Rockafellar. *Convex analysis.* Princeton, N.J: Princeton University Press, 1997.

[BC11] H. Bauschke and P. Combettes. *Convex analysis and monotone operator theory in Hilbert spaces.* New York: Springer, 2011.

[GBC16] I. Goodfellow, Y. Bengio, and A. Courville. *Deep Learning.* `http://www.deeplearningbook.org`. MIT Press, 2016.

[Ral81] L. B. Rall. *Automatic differentiation: Techniques and applications.* Springer, 1981.

Articles

[LIP-I] T. Roith and L. Bungert. “Continuum limit of Lipschitz learning on graphs.” In: *Foundations of Computational Mathematics* (2022), pp. 1–39.

[LIP-II] L. Bungert, J. Calder, and T. Roith. “Uniform convergence rates for Lipschitz learning on graphs.” In: *IMA Journal of Numerical Analysis* (Sept. 2022). DOI: `10.1093/imanum/drac048`.

[BREG-I] L. Bungert et al. “A bregman learning framework for sparse neural networks.” In: *Journal of Machine Learning Research* 23.192 (2022), pp. 1–43.

[ST14] A. Subramanya and P. P. Talukdar. “Graph-based semi-supervised learning.” In: *Synthesis Lectures on Artificial Intelligence and Machine Learning* 8.4 (2014), pp. 1–125.

[GS15] N. García Trillos and D. Slepčev. “Continuum Limit of Total Variation on Point Clouds.” In: *Archive for Rational Mechanics and Analysis* 220.1 (2015), pp. 193–241. DOI: 10.1007/s00205-015-0929-z.

[ST19] D. Slepcev and M. Thorpe. “Analysis of p-Laplacian regularization in semisupervised learning.” In: *SIAM Journal on Mathematical Analysis* 51.3 (2019), pp. 2085–2120.

[Cal19] J. Calder. “Consistency of Lipschitz learning with infinite unlabeled data and finite labeled data.” In: *SIAM Journal on Mathematics of Data Science* 1.4 (2019), pp. 780–812.

[FCL19] M. Flores, J. Calder, and G. Lerman. “Algorithms for Lp-based semi-supervised learning on graphs.” In: *arXiv preprint arXiv:1901.05031* (2019).

[CT22] J. Calder and N. G. Trillos. “Improved spectral convergence rates for graph Laplacians on ε-graphs and k-NN graphs.” In: *Applied and Computational Harmonic Analysis* 60 (2022), pp. 123–175.

[NSZ09] B. Nadler, N. Srebro, and X. Zhou. “Statistical analysis of semi-supervised learning: The limit of infinite unlabelled data.” In: *Advances in neural information processing systems* 22 (2009).

[AL11] M. Alamgir and U. Luxburg. “Phase transition in the family of p-resistances.” In: *Advances in neural information processing systems* 24 (2011).

[vLB04] U. von Luxburg and O. Bousquet. “Distance-Based Classification with Lipschitz Functions.” In: *J. Mach. Learn. Res.* 5.Jun (2004), pp. 669–695.

[Jen93] R. Jensen. “Uniqueness of Lipschitz extensions: minimizing the sup norm of the gradient.” In: *Archive for Rational Mechanics and Analysis* 123 (1993), pp. 51–74.

[Kir34] M. Kirszbraun. “Über die zusammenziehende und Lipschitzsche Transformationen.” In: *Fundamenta Mathematicae* 22 (1934), pp. 77–108. DOI: 10.4064/fm-22-1-77-108.

[Whi92] H. Whitney. “Analytic extensions of differentiable functions defined in closed sets.” In: *Hassler Whitney Collected Papers* (1992), pp. 228–254.

[McS34] E. J. McShane. “Extension of range of functions.” In: (1934).

[ACJ04] G. Aronsson, M. Crandall, and P. Juutinen. “A tour of the theory of absolutely minimizing functions.” In: *Bulletin of the American mathematical society* 41.4 (2004), pp. 439–505.

[Aro67] G. Aronsson. "Extension of functions satisfying Lipschitz conditions." In: *Arkiv för Matematik* 6.6 (1967), pp. 551–561.

[Kur22] C. Kuratowski. "Sur l'opération A de l'analysis situs." In: *Fundamenta Mathematicae* 3.1 (1922), pp. 182–199.

[ČFK66] E. Čech, Z. Frolík, and M. Katětov. "Topological spaces." In: (1966).

[DF75] E. De Giorgi and T. Franzoni. "Su un tipo di convergenza variazionale." In: *Atti della Accademia Nazionale dei Lincei. Classe di Scienze Fisiche, Matematiche e Naturali. Rendiconti* 58.6 (1975), pp. 842–850.

[Mod77] L. Modica. "Un esempio di Γ-convergenza." In: *Boll. Un. Mat. Ital. B* 14 (1977), pp. 285–299.

[CGL10] A. Chambolle, A. Giacomini, and L. Lussardi. "Continuous limits of discrete perimeters." In: *ESAIM: Mathematical Modelling and Numerical Analysis* 44.2 (2010), pp. 207–230.

[BY12] A. Braides and N. K. Yip. "A quantitative description of mesh dependence for the discretization of singularly perturbed nonconvex problems." In: *SIAM Journal on Numerical Analysis* 50.4 (2012), pp. 1883–1898.

[VB+12] Y. Van Gennip, A. L. Bertozzi, et al. "Γ-convergence of graph Ginzburg-Landau functionals." In: *Adv. Differential Equations* 17.11-12 (2012), pp. 1115–1180.

[San15] F. Santambrogio. "Optimal transport for applied mathematicians." In: *Birkäuser, NY* 55.58-63 (2015), p. 94.

[BREG-II] L. Bungert et al. "Neural Architecture Search via Bregman Iterations." In: (2021). arXiv: 2106.02479 [cs.LG].

[Cau+47] A. Cauchy et al. "Méthode générale pour la résolution des systemes d'équations simultanées." In: *Comp. Rend. Sci. Paris* 25.1847 (1847), pp. 536–538.

[RM51] H. Robbins and S. Monro. "A stochastic approximation method." In: *The annals of mathematical statistics* (1951), pp. 400–407.

[BB18] M. Benning and M. Burger. "Modern regularization methods for inverse problems." In: *Acta Numerica* 27 (2018), pp. 1–111.

[Osh+05] S. Osher et al. "An iterative regularization method for total variation-based image restoration." In: *Multiscale Modeling & Simulation* 4.2 (2005), pp. 460–489.

[Bur+06] M. Burger et al. "Nonlinear inverse scale space methods." In: *Communications in Mathematical Sciences* 4.1 (2006), pp. 179–212.

[Bur+07] M. Burger et al. "Inverse total variation flow." In: *Multiscale Modeling & Simulation* 6.2 (2007), pp. 366–395.

[Yin+08] W. Yin et al. "Bregman iterative algorithms for ℓ_1-minimization with applications to compressed sensing." In: *SIAM Journal on Imaging sciences* 1.1 (2008), pp. 143–168.

[COS09] J.-F. Cai, S. Osher, and Z. Shen. "Convergence of the linearized Bregman iteration for ℓ_1-norm minimization." In: *Mathematics of Computation* 78.268 (2009), pp. 2127–2136.

[NY83] A. S. Nemirovskij and D. B. Yudin. "Problem complexity and method efficiency in optimization." In: (1983).

[BT03] A. Beck and M. Teboulle. "Mirror descent and nonlinear projected subgradient methods for convex optimization." In: *Operations Research Letters* 31.3 (2003), pp. 167–175.

[Nem+09] A. Nemirovski et al. "Robust stochastic approximation approach to stochastic programming." In: *SIAM Journal on optimization* 19.4 (2009), pp. 1574–1609.

[Nes83] Y. Nesterov. "A method for unconstrained convex minimization problem with the rate of convergence o (1/k2)." In: *Doklady ANSSSR* 269.3 (1983), pp. 543–547.

[Qia99] N. Qian. "On the momentum term in gradient descent learning algorithms." In: *Neural networks* 12.1 (1999), pp. 145–151.

[KB14] D. Kingma and J. Ba. "Adam: A Method for Stochastic Optimization." In: *arXiv preprint arXiv:1412.6980* (2014).

[HR21] F. Hanzely and P. Richtárik. "Fastest rates for stochastic mirror descent methods." In: *Computational Optimization and Applications* 79 (2021), pp. 717–766.

[ZH18] S. Zhang and N. He. "On the convergence rate of stochastic mirror descent for nonsmooth nonconvex optimization." In: *arXiv preprint arXiv:1806.04781* (2018).

[DOr+21] R. D'Orazio et al. "Stochastic mirror descent: Convergence analysis and adaptive variants via the mirror stochastic Polyak stepsize." In: *arXiv preprint arXiv:2110.15412* (2021).

[Ben+21] M. Benning et al. "Choose your path wisely: gradient descent in a Bregman distance framework." In: *SIAM Journal on Imaging Sciences* 14.2 (2021), pp. 814–843.

[Liu+21] S. Liu et al. "Sparse evolutionary deep learning with over one million artificial neurons on commodity hardware." In: *Neural Computing and Applications* 33.7 (2021), pp. 2589–2604.

[DZ19] T. Dettmers and L. Zettlemoyer. "Sparse networks from scratch: Faster training without losing performance." In: *arXiv preprint arXiv:1907.04840* (2019).

[XRV17] H. Xiao, K. Rasul, and R. Vollgraf. "Fashion-MNIST: a Novel Image Dataset for Benchmarking Machine Learning Algorithms." In: *CoRR* abs/1708.07747 (2017).

[LC10] Y. LeCun and C. Cortes. "MNIST handwritten digit database." In: (2010).

Erklärung

Hiermit versichere ich, dass ich die vorliegende Arbeit selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe, dass alle Stellen der Arbeit, die wörtlich oder sinngemäß aus anderen Quellen übernommen wurden, als solche kenntlich gemacht sind und dass die Arbeit in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegt wurde.

Erlangen, den 4.Juli 2023 ....

Tim Roith