D I P L O M A R B E I T

Implicit Regularization for Artificial Neural Networks

zur Erlangung des akademischen Grades

Diplom-Ingenieur

im Rahmen des Studiums

Technische Mathematik

eingereicht von

Jakob M. Heiss

Matrikelnummer: 01128223

ausgeführt am

Institut für Financial and Actuarial Mathematics
TU Wien
in Zusammenarbeit mit der ETH Zürich

unter der Betreuung von

Professor Josef Teichmann

Wien, am

Jakob Heiss

Josef Teichmann

Abstract

The main result is a rigorous proof that artificial neural networks without explicit regularization implicitly regularize the strain energy $\int (\hat{f}'')^2 dx$ when trained by gradient descent by solving very precisely the smoothing spline regression problem

$$\hat{f} := \arg\min_{f \in \mathcal{C}^2} \left(\sum_{i=1}^{N} (f(x_i^{\text{train}}) - y_i^{\text{train}})^2 + \lambda \int (f'')^2 \, dx \right) \tag{1}$$

under certain conditions[1].[2] Artificial neural networks are often used in Machine Learning to estimate an unknown function[3] f_{True} by only observing *finitely* many data points. There are many methods that guarantee the convergence of the estimated $\hat{f}$ to the true function f_{True} as the number of samples tends to infinity. But in practice there is almost always only a finite number N of samples available. Given a finite number of data points there are infinitely many functions that fit perfectly through the N data points but generalize arbitrary bad. Therefore one needs some regularization to find a suitable[4] function. With the help of Theorem 3.1.4 one can solve the paradox why training neural networks without explicit regularization works surprisingly well under certain conditions[1].

[1] The main Theorem 3.1.4 only considers *1-dimensional wide ReLU randomized shallow neural networks* (2.2) *using squared loss* (i.e. one hidden layer with $n \to \infty$ many hidden hidden nodes; randomly chosen weights and biases in the first layer that are not trained—only the last layer is trained with (stochaistic) gradient descend; $d = 1$-dimensional input; ReLU activation functions; squared loss $L\left(\hat{f}\right) := \sum_{i=1}^{N} \left(\hat{f}(x_i^{\text{train}}) - y_i^{\text{train}}\right)^2$ is used as training loss). Some popular engineer's rules of thumb how to choose meta-parameters can be better understood with the help of the main theorem, since some of this rules appear as important necessary conditions in the main theorem: It's crucial that the weights in the last layer are initialized close to zero ($w^0 = 0$). The learning rate shouldn't be too large ($\gamma \to 0$). Depending on the choice of randomness (probability distributions of the random weights and biases) the network will converge to a (slightly) adapted version of the regression spline. If one uses the *Keras*-default distributions the adapted regression spline does not exactly equal the regression spline, but if one follows the rule to scale the training data to fit inside the $[-1, 1]$-cube, one can see intuitively that in this case the adapted regression spline is typically quite close to the classical regression spline inside $[-1, 1]$. Then there are more technical assumptions: If one uses plain (stochastic) gradient descend without any explicit regularization the main result is only *exactly* provable for the limit of the training algorithm $T \to \infty$. But the thesis motivates theoretically and empirically what approximately happens if early stopping at $T \in \mathbb{R}_{\geq 0}$ of the (stochastic) gradient descend is applied. Precise results for early stopping can be applied if a ridge-penalty is applied on the weights (also known as weight decay, L2-penalty or or Tikhonov regularization). Assumption 2 is probably not necessary and might be weakened in future work, but makes the proof easier without being very restrictive in real world computer implementations. Assumption 3 allows the formulation of the easier readable Theorem 3.1.4 instead of the more general Corollary 3.1.7. This footnote covers all the assumptions made in this thesis. For most of these assumptions there will be a discussion what happens if they do not hold.

[2] Equation (1) should be interpreted such that $\hat{f}$ is the unique minimizer of

$$\hat{f} :\in \arg\min_{f \in C^2(R)} \left(\sum_{i=1}^{N} \left(f(x_i^{\text{train}}) - y_i^{\text{train}} \right)^2 + \lambda \int_{\mathbb{R}} \left(f''(x) \right)^2 dx \right),$$

if $\exists (i,j) \in \{1, \ldots, N\}^2 : x_i^{\text{train}} \neq x_j^{\text{train}}$.

[3] Usually $f_{\text{True}}(x) = \mathbb{E}\left[Y | X = x\right]$.

[4] From a Bayesian point of view regularization can be connected to prior information. For example one can typically assume apriori that the unknown function f_{True} is more likely to be smooth than rough.

Eidesstattliche Erklärung

Ich erkläre an Eides statt, dass ich die vorliegende Diplomarbeit selbstständig und ohne fremde Hilfe verfasst, andere als die angegebenen Quellen und Hilfsmittel nicht benutzt bzw. die wörtlich oder sinngemäß entnommenen Stellen als solche kenntlich gemacht habe.

Zürich, am 12. August 2019

Jakob Heiss

Jakob M. Heiss

Contents

Chapter 1

Introduction

Even though neural networks are becoming more and more popular, their theoretical understanding is still very limited. Today Neural Networks are mainly used as black box methods that often work surprisingly well in applications without being fully understood. Today's most important open questions in the mathematical theory of neural networks include the following:[1]

I. **Generalisation**: Why can neural networks make good output predictions for new unseen input data even though they have only seen finitely many training data points. How can one get control of overfitting? How does the trained function behave in between the training data?

II. **Gradient Descend**: When training Neural Networks, a typically very high dimensional non-convex optimization problem is claimed to be solved by (stochastic) gradient descend quite fast. But what does this algorithm actually do? What does it converge to? What happens if you stop it after a realistic number of steps?

III. **Expressiveness:** How expressive are Neural Networks with a finite number of nodes? [27, 4, 16]

IV. **Summary**: What are the advantages and disadvantages of different architectures? What are the advantages and disadvantages compared to other methods like Random Forest or Kernel-based Gaussian process based methods? Answering I to III would basically solve IV.

The goal of this thesis is to contribute in answering these questions by rigorously proofing Theorems 3.1.4 and 3.2.5 that answer question II almost completely (cp. eqs. (5.1) and (5.2)) for the restricted class of wide Randomized Shallow Neural Networks with ReLU activation. These answers together with the intuition acquired from sections 1.1 and 1.2 give quite extensive insights to question I and thus question IV.

The result of this thesis can be seen in analogy to the breakthrough in thermodynamics theory: Like we are understanding the collision behavior of each particle, we understand the training behavior of each neuron[2]. However due to large number of interactions between particles/neurons the complexity increases in a way that the individual behavior does no longer give a direct insight into the overall system behavior. In both cases taking the limit to infinity allows

[1] The literature agrees with questions I–III too be the central questions [25]. Question IV motivates the important of questions I–III by summarizing them and concluding their implications.

[2] In this thesis only *artificial* neural networks are considered. Therefore terms like neurons and neural networks always refer to their artificial counterparts not to actual biological neurons.

to statistically derive precisely the overall system behavior in terms of interpretable macroscopic laws/theorems (see Theorem 3.1.4[3]).

1.1 The Regression Problem as Basis for Machine Learning

The setting of supervised machine learning is typically introduced as: Let $\mathcal{X}$ be the input space and $\mathcal{Y}$ be the output space. Assume we observe a finite number N of i.i.d. samples $(x_i^{\text{train}}, y_i^{\text{train}}) \in \mathcal{X} \times \mathcal{Y}$ with $i \in \{1, \ldots, N\}$ from an *unknown* probability distribution on $\mathcal{X} \times \mathcal{Y}$. When we get a new realization of (X, Y) from the same unknown distribution, but for the new realization we can only observe $X(\omega)$ but not $Y(\omega)$, we want to make a prediction $\hat{f}(X(\omega))$ of $Y(\omega)$. For a given cost function $C : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$ we are interested in an estimator $\hat{f} : \mathcal{X} \to \mathcal{Y}$ with low expected costs $\mathbb{E}\left[C\left(Y, \hat{f}(X)\right)\middle|X\right]$. As the distribution of (X, Y) is unknown we cannot calculate the expected costs. In supervised machine learning one tries to learn an estimator $\hat{f}$ from the given training data $(x_i^{\text{train}}, y_i^{\text{train}})_{i \in \{1,\ldots,N\}}$.

The goal in regression analysis is to get an approximation $\hat{f} : \mathcal{X} \to \mathcal{Y}$ of an unknown function $f_{\text{True}} : \mathcal{X} \to \mathcal{Y}$. Assume we observe a finite number N of samples $(x_i^{\text{train}}, y_i^{\text{train}}) \in \mathcal{X} \times \mathcal{Y}$ with $i \in \{1, \ldots, N\}$ where y_i^{train} is generated as $y_i^{\text{train}} := \hat{f}\left(x_i^{\text{train}}\right) + \varepsilon_i$, where ε_i is the noise.

If $\mathbb{E}[\varepsilon_i] = 0$ and $C(y, y') = (y - y')^2$, then the unknown true function f_{True} corresponds to $f_{\text{True}}(x) = \mathbb{E}[Y|X = x]$, which connects the two different points of view.

In Chapter 3 these points of view do not matter, because the main theorems there only tell what function $\hat{f}$ is learned by a given training algorithm for given training data (to answer question II). The unknown true distribution of (X, Y) in one point of view or the unknown true function f_{True} in the other point of view only matter for questions I and IV which are more connected to Chapter 1.

For simplicity in the rest of this thesis $\mathcal{X} = \mathbb{R}^d$ with input dimension $d \in \mathbb{N}$ and $\mathcal{Y} = \mathbb{R}$ will be assumed.

Historically one of the first regression analysis was the linear regression [10, 11, 20], where we restrict ourselves to a tiny subspace of all functions: the space of linear functions. If the number of samples N is larger than the input dimension d there exists a unique[4] function that fits through the training data the best by minimizing the training loss

$$L\left(\hat{f}\right) := \sum_{i=1}^{N} \left(\hat{f}(x_i^{\text{train}}) - y_i^{\text{train}}\right)^2. \tag{1.1}$$

In real world applications the space of linear functions is often not sufficient. Therefore with the philosophy of machine learning the restriction to a small subspace of functions is not appropriate. The new challenge is to choose the "most desirable" function $\hat{f}$ out of the infinitely many functions with equal training loss $L\left(\hat{f}\right)$. This opens the question what "most desirable" means mathematically. At least intuitively engineers have quite specific convictions (also known as *inductive bias*) which functions are not desirable (see Figures 1.1 and 1.2). This intuition

[3]Theorem 3.1.4 results from letting the number of neurons n tend to infinity. In thermodynamics Brownian motion particle movements or heat equations result from taking the limit of the number of particles to infinity.

[4]The solution of linear regression is unique if there are d training data input points x_i^{train} which are linearly independent. If the training data points are drawn as i.i.d. samples from a distribution which is absolutely continuous with respect to the d-dimensional Lebesgue-measure, this is almost surely the case, if $d \leq N$.

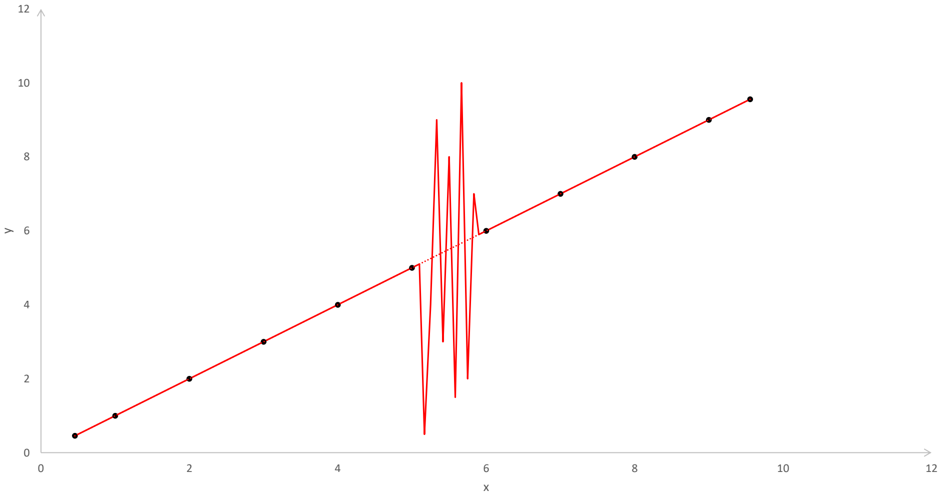

Figure 1.1: Example: Given these $N = 11$ training data points $(x_i^{\text{train}}, y_i^{\text{train}})$ (black dots) there are infinitely many functions f that perfectly fit through the training data and therefore have training loss $L(f) = 0$. Our intuition tells us that we should prefer the straight dotted line over the oscillating solid line, even though both functions have zero training loss $L(f) = 0$.

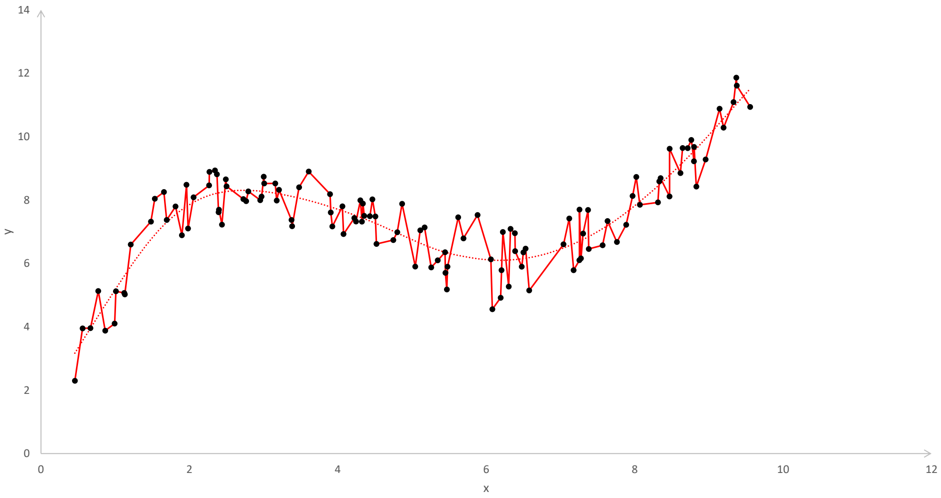

Figure 1.2: Example: Given these $N = 120$ training data points $(x_i^{\text{train}}, y_i^{\text{train}})$ (black dots) there are infinitely many functions f that perfectly fit through the training data and therefore have training loss $L(f) = 0$. For many applications our intuition tells us that we should prefer the smooth dotted line $f^{*,\lambda}$ over the oscillating solid line, even though the smooth function $f^{*,\lambda}$ has training loss $L\left(f^{*,\lambda}\right) > 0$.

could be formalized mathematically as a Bayesian prior knowledge[5] [6, e.g. page 22].

One approach to capture the engineer's intuition about the prior knowledge is to directly regularize the second derivative of $\hat{f}$. Therefore in the $d = 1$-dimensional case the the widely used spline regression [26, 7, 17] is considered in order to choose the function $\hat{f}$ with minimizes a weighted combination of the integrated square of the second derivative and the training loss L.

Definition 1.1.1 (spline regression)**.** Let $\forall i \in \{1, \ldots, N\} : x_i^{\text{train}}, y_i^{\text{train}} \in \mathbb{R}$ and $\lambda \in \mathbb{R}_{>0}$. Then the *(smoothing[6]) regression spline* $f^{*,\lambda} : \mathbb{R} \to \mathbb{R}$ is defined[7] as:

$$f^{*,\lambda} \overset{7}{:\in} \underset{f \in \mathcal{C}^2(\mathbb{R})}{\arg\min} \underbrace{\left(\overbrace{\sum_{i=1}^{N} \left(f(x_i^{\text{train}}) - y_i^{\text{train}}\right)^2}^{L(f)=} + \lambda \int_{-\infty}^{\infty} \left(f''(x)\right)^2 dx \right)}_{=:F^{\lambda}(f)} \tag{1.2}$$

and for a given function $g : \mathbb{R} \to \mathbb{R}_{\geq 0}$ the *weighted regression spline* $f_g^{*,\lambda}$ is defined[7] as

$$f_g^{*,\lambda} \overset{7}{:\in} \underset{\substack{f \in \mathcal{C}^2(\mathbb{R}) \\ \text{supp}(f) \subseteq \text{supp}(g)}}{\arg\min} \underbrace{\left(\overbrace{\sum_{i=1}^{N} \left(f(x_i^{\text{train}}) - y_i^{\text{train}}\right)^2}^{L(f)=} + \lambda g(0) \int_{\text{supp}(g)} \frac{\left(f''(x)\right)^2}{g(x)} dx \right)}_{=:F^{\lambda,g}(f)} . \tag{1.3}$$

The meta parameter λ controls the trade-off between low training loss and low squared second derivative. For an example of spline regression (with $g(x) = 1 \quad \forall x \in \mathbb{R}$) see $f^{*,\lambda}$ in Figure 1.2.

[5]From the machine learning point of view one could theoretically formulate this prior knowledge regarding the unknown distribution of (X, Y) on $\mathcal{X} \times \mathcal{Y}$ as a (probability)-measure on the space of all probability measures on $\mathcal{X} \times \mathcal{Y}$. From a regression point of view the prior regarding the unknown function f_{True} would be be a (probability)-measure on the set of all functions from $\mathcal{X}$ to $\mathcal{Y}$. If the prior measure is a probability measure one can work perfectly rigorous in the framework of classical Bayes law. If the prior measure is not a probability measure it is called an improper prior which can also lead to good results in applications. Consider for example the very restrictive prior measure that assigns measure 0 to the huge set of all nonlinear functions and weights all linear functions the same. Since this measure assigns ∞ to the subspace of all linear functions, it is an improper prior. This improper prior leads to the standard linear regression in the case of i.i.d. normally distributed noise ε_i. The simple intuitive prior knowledge "I am absolutely sure that f_{True} is linear, but I consider all linear functions as equally likely." is captured quite well by this improper prior and the solution of the corresponding Bayesian problem can be computed quite fast (linear regression). But for most real world applications a more realistic intuitive prior knowledge like "I cannot exclude any function for sure, but I have some vague feeling that f_{True} is more likely to be a 'simpler', 'smoother' function than a 'heavily oscillating' function.", it is harder to formalize it mathematically and calculating the solution of such Bayesian problems is often not traceable (with today's computational power). Still Bayesian theory can be considered as a very powerful and general abstract theoretical framework without explicitly solving Bayesian problems and even without explicitly writing down priors. (If anyone could write down mathematically precisely a prior measure that captures all available prior knowledge (for each domain) and then develop a fast algorithm to solve the corresponding Bayesian problem, the field of supervised machine learning would be solved.)

[6]In the literature the spline regression is often called *(cubic) smoothing spline*, but in this text $f^{*,\lambda}$ will simply be called regression spline.

[7]The (weighted) regression spline $f_g^{*,\lambda}$ is uniquely defined if $\exists (i,j) \in \{1, \ldots, N\}^2 : x_i^{\text{train}} \neq x_j^{\text{train}}$.

Definition 1.1.2 (spline interpolation)**.** Let $\forall i \in \{1, \ldots, N\} : x_i^{\text{train}}, y_i^{\text{train}} \in \mathbb{R}$ and $\lambda \in \mathbb{R}_{>0}$. Then the *(smooth) spline interpolation* $f^{*,0+} : \mathbb{R} \to \mathbb{R}$ is defined[8] as:

$$f^{*,0+} := \lim_{\lambda \to 0+} f^{*,\lambda} \overset{8}{\in} \underset{\substack{f \in \mathcal{C}^2(\mathbb{R}), \\ f(x_i^{\text{train}}) = y_i^{\text{train}} \ \forall i \in \{1,\ldots,N\}}}{\arg\min} \left(\int_{-\infty}^{\infty} \left(f''(x)\right)^2 dx \right). \tag{1.4}$$

The Definitions 1.1.1 and 1.1.2 can also be seen as solutions to mathematically defined Bayesian problems [17][9].

1.2 The Paradox of Neural Networks

This section discusses the paradox why standard neural networks training algorithms find "desirable" functions $\hat{f}$ without explicit regularization. Within this paradox we will demonstrate two severe misassumptions in the classical approach to explain neural networks.

The paradox holds for deep neural networks [13] as well as for shallow[10] neural networks. This thesis resolves the paradox only rigorously in the context of shallow neural networks[10] (cp. Chapter 3). Further work is required to extend the results to deep neural networks.[10]

Definition 1.2.1 (Shallow neural network[10])**.** Let the activation function $\sigma : \mathbb{R} \to \mathbb{R}$ be Lipschitz continuous and non-constant. Then a *shallow neural network* is defined as $\mathcal{NN}_\theta : \mathbb{R}^d \to \mathbb{R}$ s.t.

$$\mathcal{NN}_\theta(x) := \sum_{k=1}^{n} w_k \, \sigma \left(b_k + \sum_{j=1}^{d} v_{k,j} x_j \right) + c \quad \forall x \in \mathbb{R}^d \tag{1.5}$$

- number of neurons $n \in \mathbb{N}$ and input dimension $d \in \mathbb{N}$
- weights $w_k \in \mathbb{R}$, $k = 1, \ldots, n$
- biases $b_k \in \mathbb{R}$, $k = 1, \ldots, n$
- weights $v_k \in \mathbb{R}^d$, $k = 1, \ldots, n$
- bias $c \in \mathbb{R}$
- all the weights and biases are summarized in $\theta := (w, b, v, c) \in \Theta := \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}^{n \times d} \times \mathbb{R}$.

The paradox (summarized in Figure 1.3) consists of two parts:

1. In the literature it is often claimed that the goal of training a neural network is to find parameters

$$\theta^* \in \underset{\theta \in \Theta}{\arg\min}\, L\left(\mathcal{NN}_\theta\right) \tag{1.6}$$

[8] Analogous to footnote 7 the spline interpolation $f^{*,0+}$ is uniquely defined if $\exists (i,j) \in \{1, \ldots, N\}^2 : x_i^{\text{train}} \neq x_j^{\text{train}}$.

[9] More precisely speaking the Definitions 1.1.1 and 1.1.2 can be seen as limits of Bayesian problems [17, p. 502]. The Definitions 1.1.1 and 1.1.2 can not be the solution fo an classical Bayesian problem with a *proper* proior (cp. footnote 5 on page 4 and [17, eq. (4.1) on p. 501]).

[10] In very recent literature it became fashionable to call *shallow neural networks* "simple deep neural networks" or "two-layer (deep) neural network" [12, Section 1.1 p. 3]. All three notations make sense since a shallow neural network has three layers of neurons (input→hidden→output) therefore it has two layers of weights and biases $((v, b) \to (w, c))$ and thus one hidden layer of neurons. In this thesis we are using the classical notation of "shallow neural networks" to describe them. When we discuss here or in Chapter 5 that we want to extend our theory to deep neural networks this can also be read as "even deeper neural networks".

such that the corresponding neural network $\hat{f} := \mathcal{NN}_{\theta^*}$ fits through the training data as good as possible.

But such a $\mathcal{NN}_{\theta^*}$ can have bad generalization properties: If $n \geq N-1$, there are infinitely many (1.6)-optimizing shallow neural networks $\mathcal{NN}_{\theta^*}$ that generalize arbitrary bad[11], even if there were only zero noise $\varepsilon_i = 0$ on the training data. If $n \leq N-2$, then $\mathcal{NN}_{\theta^*}$ can be unique, but $\mathcal{NN}_{\theta^*}$ might still overfit to the noise on the training data (see Figure 1.4). The universal approximation theorem [8, 15] tells already that large neural networks $\mathcal{NN}_{\theta^*}$ (or any other universal approximating class of functions) can behave arbitrary bad (like in Figure 1.1 for example) in between the training data x_i^{train} while having a arbitrary low training loss $L(\mathcal{NN}_{\theta^*}) \leq \epsilon$, exactly because of their universal approximation properties. (If a very small number of neurons $n << \frac{N}{d}$ were chosen, overfitting of $\mathcal{NN}_{\theta^*}$ would not be such a problem, but then neural networks would loose their universal approximation property (which is one of their main selling points) and therefore $\mathcal{NN}_{\theta^*}$ could not achieve a low loss $L(\mathcal{NN}_{\theta^*})$.)

The paradox is that in practice extremely large neural networks $\mathcal{NN}_{\theta^T}$ typically generalize very well. Actually the main Theorems 3.1.4 and 3.2.5 of this thesis will show how well neural networks $\mathcal{NN}_{\theta^T}$ with an infinite number of neurons behave in between the data.

2. As the optimization problem (1.6) optimizes (in the case of typical activation functions like ReLUs) an Lebesgue-almost everywhere differentiable function on a finite dimensional $\mathbb{R}$-vector space Θ the optimization algorithm that first comes to the mind of probably most engineers is a gradient descend algorithm (which is called backpropagation algorithm in the case of neural networks). In the case of the training loss L one can use stochastic gradient descend as well.[12]

   But there are no guarantees that this algorithm converges to global optimum for a general typically non-convex optimization problem. And numerical experiments show that if one runs the algorithm for a reasonable time, one is still by far not optimal (w.r.t. the target function L, that the algorithm claims to try to optimize.) (e.g. Figure 1.4).

1.3 Resolving the Paradox of Neural Networks: Implicit Regularization

In this section the paradox will be resolved and at the end of this section a short overview will be given how this thesis contributes to a better understanding of this phenomena.

1, 2 and the observation that Neural Networks are very useful in practice can be true at the same time:

Even though an “optimal” network $\mathcal{NN}_{\theta^*}$ would typically perform quite poorly in practice (cp. 1), we never find $\mathcal{NN}_{\theta^*}$ in practice, as one is almost always using a gradient descend based algorithm to search for $\mathcal{NN}_{\theta^*}$. Because fortunately the back-propagation algorithm that was designed to find something close to $\mathcal{NN}_{\theta^*}$ by minimizing the training loss L does not

[11] For ReLU activation functions one can easily proof that for every training data $\left(x_i^{\text{train}}, y_i^{\text{train}}\right)_{i\in\{1,\dots,N\}}$ there exist infinity many $\mathcal{NN}_{\theta^*}$ such that the d-dimensional Lebesgue-measure of the set $\left\{ x \in [0,1]^d \;\middle|\; |\mathcal{NN}_{\theta^*}(x)| > 9999 \right\}$ is larger than 99% and $L(\mathcal{NN}_{\theta^*}) = 0$.

[12] The stochastic gradient descend has huge computational advantages in the case of a very large number N of training observations. In future work we will go more into detail on stochastic gradient descend (cp. item 2 on page 35), but in this thesis stochastic gradient descend can be treated equivalent to ordinary gradient descend as we are always taking the limit of the learning rate $\gamma \to 0$.

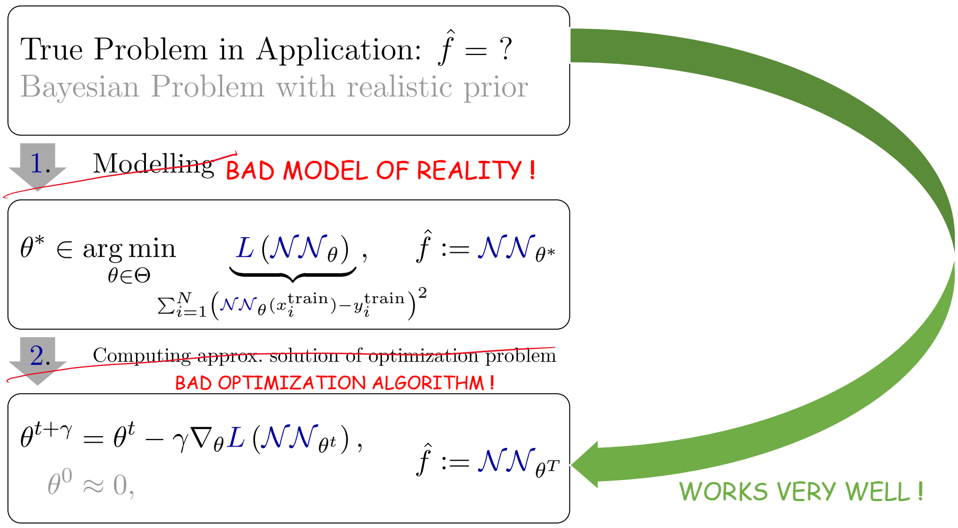

Figure 1.3: The paradox of neural networks: 1. It would not be a desirable goal for neural networks to minimize the training loss L solely. 2. The (stochastic) gradient descend algorithm (also known as back-propagation algorithm) does typically not find the global optimum. Nevertheless the algorithm result in surprisingly useful functions $\hat{f} = \mathcal{NN}_{\theta^T}$ for a wide range of practical applications.

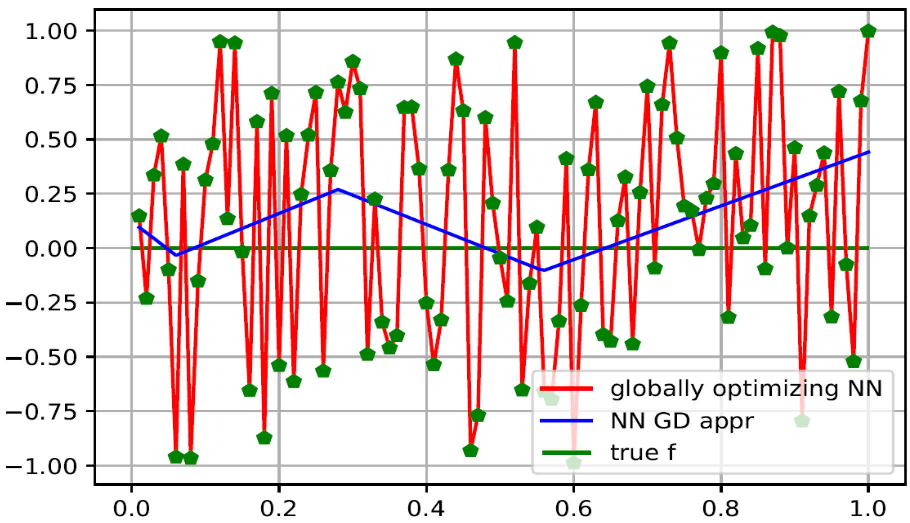

Figure 1.4: Example: Let $N = 100$ training samples $(x_i^{\text{train}}, y_i^{\text{train}})$ be scattered uniformly around the true function $f_{\text{True}} = 0$ and consider a shallow neural network $\mathcal{NN}$ with $n = N = 100$ hidden nodes. After 10000 training epochs of Adam SGD [18] the neural network does not converge to the global optimum $\mathcal{NN}_{\theta^*}$ (red line) with $L(\mathcal{NN}_{\theta^*}) = 0$, but to a more regular function $\mathcal{NN}_{\theta^T}$ (blue line) which is closer to the true function f_{True}.

achieve[13] the goal it was destined to (cp. 2. $L(\mathcal{NN}_{\theta T}) >> L(\mathcal{NN}_{\theta^*})$)—it surprisingly achieves a much more desirable goal by not only minimizing the training loss L but somehow *implicitly*[14] regularizing the problem. So the bad property 1 of $\mathcal{NN}_{\theta^*}$ is not a contradiction to the great performance of the much more regular $\mathcal{NN}_{\theta T}$. This phenomena is known in the literature as "implicit regularization" [24, 23, 21, 19, 28, 25, 12] (also known as "implicit bias"[28]).

Hence the phenomena of implicit regularization demonstrates that the question I about generalization and question II about the gradient descend algorithm are strongly linked to each other in practice.

The phenomena of implicit regularization is highly observable in practice [14, 22, 24, 23, 21, 19, 25], but the theory behind it is still mainly open[21, 19, 25, 22].

The contribution of this thesis is to proof very precisely how the implicit regularization works for a special type of neural networks (see footnote 1 from the abstract)—it regularizes the second derivative of the network (seen as a function from $\mathcal{X}$ to Y). For the considered type of network we can prove mathematically to which function the network converges (cp. Definition 3.1.1 and Theorems 3.1.4 and 3.2.5). In a typical setting this is very close to a regression spline $f^{*,\lambda}$, whose theory is highly understood [26, 7, 17].

In this thesis we will state two main theorems:

- Theorem 3.2.5 connects the ordinary gradient descend without any explicit regularization to an implicit ridge regularization of the weights. (Very similar theorems are already well known [5, 9, 25, 12].)
- Theorem 3.1.4 shows how the weight's ridge regularization from Theorem 3.2.5 results in the (slightly adopted) spline regularization of the learned network function if the number of neurons $n \to \infty$. This theorem is the main contribution of this thesis.

Known theorems in that field are:

- There are many theorems that help to explain how implicit regularization could work on the weight space (similar to Theorem 3.2.5) [5, 28, 25, 12]. But they do not precisely explain how this translates to implicit regularization on the function space—only in the case of classification[15] these results give insight about the margins between the classes, which is a property of the learned function. These papers provide a precise and quite complete mathematical understanding of linear neural networks without any hidden layers. The theorems in these papers that deal with neural networks with one (ore more) hidden layers serve as basis for arguments why an implicit regularization effect can exist on a qualitative level, but not on a precise quantitative level (especially when non-linear activation functions σ are considered). So there are still many open questions.
- Since this thesis' main contribution Theorem 3.1.4 explains the implicit regularization on the function space, the more closely related literature is [22, 19, 21].

[13] In the limit training time $T \to \infty$ it can find a global optimum, but not any arbitrary golbal optimum out of the typically infinite many global optima, but a very special global optimum (c.p. Definitions 2.0.5 and 3.1.3, Theorems 3.1.4 and 3.2.5 and eq. (5.1)). But typically training is stopped after a few epochs ($T << \infty$), where $L(\mathcal{NN}_{\theta T}) >> L(\mathcal{NN}_{\theta^*})$ holds (which is the much more desirable solution—cp. Definition 3.1.1 and eq. (5.2)).

[14] "*Implicitly*" means that one uses exactly the same algorithm (gradient descend on the training loss L cp. Figure 1.3) that one would use, if one did not care about regularization, but running the algorithm results surprisingly in a very regular $\mathcal{NN}_{\theta T}$.

[15] [28, 25] focus more on classification (exponential loss) and [5, 12] focus more on regression (least square training loss L).

- [21] studies the implicit regularization for a fully trained shallow neural network $\mathcal{NN}$ with nonlinear ReLU activation functions $\sigma = \max(0, \cdot)$ in the context of classification (cross entropy loss over the softmax as training loss) on a qualitative level. They use already the notion “pseudo-smooth” [21, e.g. p. 4], but a quantitative mathematical analysis of the pseudo-smoothness is missing.
- [22] (by Google Brain) also studies the implicit regularization for a fully trained shallow neural network $\mathcal{NN}$ with nonlinear ReLU activation functions $\sigma = \max(0, \cdot)$, but also in the context of regression (differentiable loss function). This paper is closest to this thesis as its main goal is to explain how the learned neural network function $\mathcal{NN}_{\theta^T}$ behaves macroscopically in-between the training data. They provide a very rich qualitative understanding of $\mathcal{NN}_{\theta^T}$ and provide very helpful visualizations, but they cannot provide a precise quantitative formula—they cannot completely characterize how the learned function behaves macroscopically. Whereas this thesis can provide the precise quantitative macroscopic formula (Definition 3.1.1) in the case of randomized neural networks $\mathcal{RN}$with the help of Theorem 3.1.4 and eq. (5.2), which provides a quite complete understanding of $\mathcal{RN}$. (I have already an analogous theorem in mind for future work that characterize in which sense a fully trained network $\mathcal{NN}_{\theta^T}$ is macroscopically optimal[16], which would answer some of the open questions posed in [22])
- [19] studies implicit regularization of deep neural network with nonlinear ReLU activation functions $\sigma = \max(0, \cdot)$ by trying to explain that the learned function interpolates “almost linearly” between samples, which is related to a low (in the case of ReLUs distributional) second derivative which corresponds to their notion of “gradient gaps”. Furthermore they try to establish some connection to Brownian bridges.[17]

In Chapter 2 the considered type of neural networks $\mathcal{RN}$ are defined: *1-dimensional wide ReLU randomized*[18] *shallow neural networks* (2.2). The definitions in chapter 2 are important to understand the main Theorems 3.1.4 and 3.2.5.

[16] The main difference of $\mathcal{NN}$ compared to $\mathcal{RN}$ is that in Definition 3.1.1 the squared second derivative is replaced by the absolute value of the distributional second derivative (the L^1-norm has a very natural extension to distributions). This explains many of the phenomena described by [22]. The proof of this conjecture will be similar to the proof of Theorem 3.1.4, but the details will be figured out in future work.

[17] The theorems proven in [19] rely on unrealistic assumptions (i.i.d. gradient gaps), but, based on their thoughts, in the case of shallow neural networks $\mathcal{NN}$ with random initialization without any training, one could easily derive an precise mathematical theorem under realistic assumptions: This shallow network $\mathcal{NN}_{\theta^0}$ would converge ($n \to \infty$) to an adapted Brownian bridge with a variable volatility analogous to g introduces later in this thesis in Theorem 3.1.4 or Definition 3.1.1. If a typical choice of randomness is made, the adapted Brownian bridge is quite close to an ordinary Brownian bridge (constant volatility) inside the $[-1, 1]$-cube by similar arguments as in item 5 on page 36, where we will argument that the adapted regression spline $f^{*,\lambda}_{g,\pm}$ is close to the ordinary regression spline $f^{*,\lambda}$ inside the $[-1, 1]$-cube. This adapted theorem would be a more precise version of [2, Proposition A1] cited in [19]. Similar results for deeper networks would be plausible, still in the case of fully random weights without any training. But their idea to model *trained* networks $\mathcal{NN}_{\theta^T}$ as Brownian bridges in the limit of infinitely many neurons $n \to \infty$ does not really fit to the much smoother limits suggested by [22] or Theorem 3.1.4, Lemma 4.1.11 and eq. (5.2) in this thesis ($f^{*,\lambda}_{g,\pm}$ is not only smoother but also deterministic in contrast to a Brownian bridge). Still their observation that [19, Figure 1(b)] looks similar to a Brownian motion is interesting. Maybe this cannot explained by the limit of neurons $n \to \infty$, but by the limit of training data $N \to \infty$ to infinity, since they are using the ImageNet dataset which contains millions of samples. But in any case there are still open questions.

[18] The most special property of this type of networks is that their first layer is chosen randomly and not trained—after random initialization only the last layer is trained. One migth expect that this randomness decreases the regularity of the leanred function, but actually the learned function will be especially smooth this way (in the sence of integrated squared derivative; cp. Theorem 3.1.4)

In the two Sections 3.1 and 3.2 the two main Theorems 3.1.4 and 3.2.5 an their respective definitions will be formulated.

The proofs are in Chapter 4. The rest of the thesis is still understandable, if Chapter 4 is skipped.

In Chapter 5 the implications of the main Theorems 3.1.4 and 3.2.5 will be summarized in eqs. (5.1) and (5.2) and planned future work will be discussed.

Chapter 2

Randomized Shallow Neural Networks

Definition 2.0.1 (Randomized shallow neural network)**.** Let $(\Omega, \Sigma, \mathbb{P})$ be a probability space, and the activation function $\sigma : \mathbb{R} \to \mathbb{R}$ Lipschitz continuous and non-constant. Then a *randomized shallow neural network* is defined as $\mathcal{RN}_{w,\omega} : \mathbb{R}^d \to \mathbb{R}$ s.t.

$$\mathcal{RN}_{w,\omega}(x) := \sum_{k=1}^{n} w_k \, \sigma \left(b_k(\omega) + \sum_{j=1}^{d} v_{k,j}(\omega) x_j \right) \quad \forall \omega \in \Omega \; \forall x \in \mathbb{R}^d \tag{2.1}$$

- number of neurons $n \in \mathbb{N}$ and input dimension $d \in \mathbb{N}$
- trainable weights $w_k \in \mathbb{R}$, $k = 1, \dots, n$
- random biases $b_k : (\Omega, \Sigma) \to (\mathbb{R}, \mathfrak{B})$ i.i.d. real valued random variables k=1,. . . ,n
- random weights $v_k : (\Omega, \Sigma) \to (\mathbb{R}^d, \mathfrak{B}^d)$ i.i.d. $\mathbb{R}^d$-valued random variables k=1,. . . ,n

Assumption 1. Using the notation from Definition 2.0.1:

a) The activation function $\sigma = \max(0, \cdot)$ is ReLU.

b) the distribution of the quotient $\xi_k := \frac{-b_k}{v_k}$ has a probability density function g_ξ with respect to the Lebesgue-measure.[1]

c) The input dimension $d = 1$.

Under this assumptions eq. (2.1) simplifies to:

$$\mathcal{RN}_{w}(x) = \sum_{k=1}^{n} w_k \, \max(0, b_k + v_k x_j) \quad \forall x \in \mathbb{R} \tag{2.2}$$

Definition 2.0.2 (kink positions ξ)**.** The *kink positions* $\xi_k := \frac{-b_k}{v_k}$ are defined using the notation of Definition 2.0.1 under the Assumption 1.

Definition 2.0.3 (kink position density g_ξ)**.** The *probability density function* $g_\xi : \mathbb{R} \to \mathbb{R}_{\geq 0}$ *of the kink position* $\xi_k := \frac{-b_k}{v_k}$ is defined in the setting of Definition 2.0.2.

[1] Assumption 1b) holds for all the distributions that are typically used in practice. Assumption 1b) implies that $\mathbb{P}[v_k = 0] = 0 \quad \forall k \in \{1, \dots, n\}$. Assumption 1b) could be weakened.

Definition 2.0.4 (ridge penalized network)**.**

$$w^{*,\tilde{\lambda}}(\omega) :\in \arg\min_{w\in\mathbb{R}^n} \underbrace{\overbrace{\sum_{i=1}^{N}(\mathcal{RN}_{w,\omega}(x_i^{\text{train}}) - y_i^{\text{train}})^2}^{L(\mathcal{RN}_{w,\omega})} + \tilde{\lambda}||w||_2^2}_{F_n^{\tilde{\lambda}}(\mathcal{RN}_{w,\omega})} \quad \forall\omega\in\Omega \tag{2.3}$$

$$\mathcal{RN}_{\omega}^{*,\tilde{\lambda}} := \mathcal{RN}_{w^{*,\tilde{\lambda}}(\omega),\omega} \quad \forall\omega\in\Omega \tag{2.4}$$

The ridge-penalization is also known as weight decay, L^2 (parameter) regularization or Tikhonov regularization (or ridge regression, ℓ_2 penalty, ...)[13, section 7.1.1 on p. 227].

Definition 2.0.5 (minimum norm network)**.** Let $\forall i \in \{1,\dots,N\} : (x_i^{\text{train}}, y_i^{\text{train}}) \in \mathbb{R}^{d+1}$ for some $N, d \in \mathbb{N}$. Furthermore, $\mathcal{RN}_{w,\omega}$ be a randomized shallow network with $\omega \in \Omega$ and $n \in \mathbb{N}$ hidden nodes such that $n \geq N$. For any $\omega \in \Omega$, the *minimum norm network* is then defined as $\mathcal{RN}_{w^\dagger(\omega),\omega}$ with weights $w^\dagger(\omega)$ solving

$$\min_{w\in\mathbb{R}^n} \|w\|_2 , \quad \text{s.t. } \mathcal{RN}_{w,\omega}(x_i^{\text{train}}) = y_i^{\text{train}}, \quad \forall i \in \{1,\dots,N\} . \tag{2.5}$$

Chapter 3

Main Theorems

3.1 Ridge Regularized RSN $\to$ Spline Regularization $(d = 1, \lambda \in \mathbb{R}_{>0})$

Depending on the distribution of the random weights w_k and biases w_b the random network $\mathcal{RN}^{*,\tilde{\lambda}}$ will converge to a (slightly) adapted version $f^{*,\lambda}_{g,\pm}$ of the classical regression spline $f^{*,\lambda}$.

Definition 3.1.1 (adapted spline regression)**.** Let $\forall i \in \{1, \dots, N\} : x_i^{\text{train}}, y_i^{\text{train}} \in \mathbb{R}$ and $\lambda \in \mathbb{R}_{>0}$. Then for a given function $g : \mathbb{R} \to \mathbb{R}_{\geq 0}$ the *adapted regression spline* $f^{*,\lambda}_{g,\pm} := f^{*,\lambda}_{g,+} + f^{*,\lambda}_{g,-}$ is defined[1] with

$$\left(f^{*,\lambda}_{g,+}, f^{*,\lambda}_{g,-}\right) \overset{1}{:\in} \underset{(f_+,f_-)\in\mathcal{T}}{\arg\min} \underbrace{\left(L\left(f_+ + f_-\right) + 2\lambda g(0)\left(\int_{\operatorname{supp}(g)} \frac{\left(f_+{}''(x)\right)^2}{g(x)} dx + \int_{\operatorname{supp}(g)} \frac{\left(f_-{}''(x)\right)^2}{g(x)} dx\right)\right)}_{=:F^{\lambda,g}_{+-}(f_+,f_-)}, \tag{3.1}$$

with

$$\begin{aligned}\mathcal{T} := \Big\{(f_+, f_-) \in \mathcal{C}^2(\mathbb{R}) \times \mathcal{C}^2(\mathbb{R}) \,\Big|\, & \operatorname{supp}(f_+'') \subseteq \operatorname{supp}(g), \operatorname{supp}(f_-'') \subseteq \operatorname{supp}(g), \\ & \lim_{x\to-\infty} f_+(x) = 0, \lim_{x\to-\infty} f_+'(x) = 0, \\ & \lim_{x\to+\infty} f_-(x) = 0, \lim_{x\to+\infty} f_-'(x) = 0\Big\}.\end{aligned}$$

Remark 3.1.2. If for the weighting function g it holds that $\operatorname{supp}(g)$ is compact (cp. Assumption 2a)), we define

$$C_g^\ell := \min(\operatorname{supp}(g)) \quad \text{and} \quad C_g^u := \max(\operatorname{supp}(g)). \tag{3.2}$$

Furthermore in that case, the set $\mathcal{T}$ of function tuples considered in the minimization of Definition 3.1.1 can be rewritten: From $\operatorname{supp}(f_+'') \subseteq \operatorname{supp}(g)$ it follows that $f_+' \in \mathcal{C}^1(\mathbb{R})$ is constant on $(-\infty, C_g^\ell]$. With $\lim_{x\to-\infty} f_+'(x) = 0$ we obtain that $f_+'(x) = 0\ \forall x \leq C_g^\ell$. By the same argument we obtain $f_+(x) = 0\ \forall x \leq C_g^\ell$. Moreover, we have that $\exists c_+ \in \mathbb{R} : f_+'(x) \equiv c_+$ on $[C_g^u, \infty)$.

[1] The tuple $\left(f^{*,\lambda}_{g,+}, f^{*,\lambda}_{g,-}\right)$ and thus the adapted regression spline $f^{*,\lambda}_{g,\pm}$ is uniquely defined if g is the probability density function of a distribution with finite first and second moment and if $\exists (i,j) \in \{1, \dots, N\}^2 : x_i^{\text{train}} \neq x_j^{\text{train}}$.

Analogous derivations lead to $f'_-(x) \equiv c_- \ \forall x \leq C_g^\ell$ with $c_- \in \mathbb{R}$ and $f_-(x) = f'_-(x) = 0$ on $[C_g^u, \infty)$. Hence altogether we have

$$\mathcal{T} = \Big\{ (f_+, f_-) \in \mathcal{C}^2(\mathbb{R}) \times \mathcal{C}^2(\mathbb{R}) \Big| \operatorname{supp}(f''_+) \subseteq \operatorname{supp}(g), \operatorname{supp}(f''_-) \subseteq \operatorname{supp}(g),$$
$$\forall x \leq C_g^\ell : f_+(x) = 0 = f'_+(x),$$
$$\forall x \geq C_g^u : f_-(x) = 0 = f'_-(x) \qquad \Big\}.$$

If we assume *supp*$(g) = [C_g^\ell, C_g^u]$ we get:

$$\mathcal{T} = \Big\{ (f_+, f_-) \in \mathcal{C}^2(\mathbb{R}) \times \mathcal{C}^2(\mathbb{R}) \Big| \exists c_-, c_+ \in \mathbb{R} :$$
$$\forall x \leq C_g^\ell : \Big(f_+(x) = 0 = f'_+(x) \ \wedge \ f'_-(x) = c_- \Big),$$
$$\forall x \geq C_g^u : \Big(f_-(x) = 0 = f'_-(x) \ \wedge \ f'_+(x) = c_+ \Big) \qquad \Big\}.$$

Definition 3.1.3 (adapted spline interpolation)**.** Let $\forall i \in \{1, \dots, N\} : x_i^{\text{train}}, y_i^{\text{train}} \in \mathbb{R}$ and $\lambda \in \mathbb{R}_{>0}$. Then the *adapted spline interpolation* $f_{g,\pm}^{*,0+} : \mathbb{R} \to \mathbb{R}$ is defined as:

$$f_{g,\pm}^{*,0+} := \lim_{\lambda \to 0+} f_{g,\pm}^{*,\lambda}. \tag{3.3}$$

The following technical assumption makes the proof of Theorem 3.1.4 easier, even though it could be weakened (see footnotes 2–5).

Assumption 2. Using the notation from Definitions 2.0.1 and 2.0.3 the following assumptions extend Assumption 1:

a) The probability density function g_ξ of the kinks ξ_k has compact support $\operatorname{supp}(g_\xi)$.[2]

b) The density $g_\xi|_{\operatorname{supp}(g_\xi)}$ is uniformly continuous on $\operatorname{supp}(g_\xi)$.[3]

c) The reciprocal density $\frac{1}{g_\xi}\Big|_{\operatorname{supp}(g_\xi)}$ is uniformly continuous on $\operatorname{supp}(g_\xi)$.[4]

d) The conditioned distribution $\mathcal{L}(v_k|\xi_k = x)$ of v_k is uniformly continuous in x on $\operatorname{supp}(g_\xi)$.[5]

[2] Assumption 2a) can probably be weakened a lot, but it is not that restricting because real world computers only cover a compact range of numbers anyway. This assumption makes proofs much easier and it assures that a minimum of (3.1) exists. If one skipped Assumption 2a) completely, it could happen that (3.1) does not have a classical minimum (e.g. $\mathbb{P}[v_k = -1] = \frac{1}{2} = \mathbb{P}[v_k = 1]$ and $b_k \sim$ Cauchy), but one could easily define another weaker minimum concept as the limit of minimizing sequences which converge to a unique function on every compact set. This also corresponds to the unique point-wise limit of minimizing sequences, which is not a classical minimum, because it doesn't satisfy all the boundary conditions $\lim_{x\to-\infty} f_+(x) = 0 = \lim_{x\to+\infty} f_-(x)$ anymore. Because of this weaker minimum concept, the Theorem 3.1.4 would have to be reformulated a bit at least, if Assumption 2a) were skipped completely. This weaker minimum concept can also be seen as the limit of adapted regression splines $f_{g,\pm}^{*,\lambda}$ for truncated g as the range of the truncation tends to $(-\infty, \infty)$. This footnote won't be proven in this thesis.

[3] Assumption 2b) could maybe be replaced by the weaker assumption that g_ξ is (improper) Riemann-integrable, but almost all the distributions that are typically used in practice satisfy Assumption 2b) anyway.

[4] Assumption 2c) implies that $\min_{x\in\operatorname{supp}(g_\xi)} g_\xi > 0$. Similarly to footnote 3, this assumption can probably be weakened in a way that g_ξ could have finitely many jumps and that $\min_{x\in\operatorname{supp}(g_\xi)} g_\xi$ could be zero.

[5] Assumption 2d) can probably be weakened similarly to footnote 3.

e) $\mathbb{E}\left[v_k^2\right] < \infty$.[6]

The following technical Assumption 3 makes the result of Theorem 3.1.4 more readable by referring to the easier Definition 3.1.1. Without Assumption 3, the Corollary 3.1.7 would still hold, which is more general than Theorem 3.1.4, but uses the heavier notation of Definition 3.1.5.

Assumption 3. Using the notation from Definitions 2.0.1 and 2.0.3 the following assumptions extend Assumption 1:

a) $g_\xi(0) \neq 0$.[7]

b) the the distributions of the random weights v_k and the random biases b_k are symmetric w.r.t the sign—i.e.:

   i) $\mathbb{P}\left[v_k \in E\right] = \mathbb{P}\left[v_k \in -E\right] \quad \forall E \in \mathfrak{B}$ and

   ii) $\mathbb{P}\left[b_k \in E\right] = \mathbb{P}\left[b_k \in -E\right] \quad \forall E \in \mathfrak{B}$.

Theorem 3.1.4 (ridge weight penalty corresponds to adapted spline)**.** *Let $N \in \mathbb{N}$ be a finite number of arbitrary training data $(x_i^{train}, y_i^{train})$. Using the notation from Definitions 2.0.1, 2.0.3, 2.0.4 and 3.1.1 and let*[8] *$\forall x \in \mathbb{R} : g(x) := g_\xi(x)\mathbb{E}\left[v_k^2 | \xi_k = x\right]$ and $\tilde{\lambda} := \lambda n g(0)$ then under the Assumptions 1–3 the following statement holds for every compact set $K \subset \mathbb{R}$:*

$$\mathbb{P}\text{-}\lim_{n\to\infty} \left\| \mathcal{RN}^{*,\tilde{\lambda}} - f_{g,\pm}^{*,\lambda} \right\|_{W^{1,\infty}(K)} = 0.^{9} \tag{3.4}$$

Proof. The proof of Theorem 3.1.4 is formulated in Section 4.1. □

Without Assumption 3 the Theorem 3.1.4 has to be reformulated to Corollary 3.1.7. This is done in the rest of this section.

Definition 3.1.5 (asymmetric adapted spline regression)**.** Let $\forall i \in \{1, \ldots, N\} : x_i^{\text{train}}, y_i^{\text{train}} \in \mathbb{R}$ and $\lambda \in \mathbb{R}_{>0}$. Then for given functions $g_+ : \mathbb{R} \to \mathbb{R}_{\geq 0}$, $g_- : \mathbb{R} \to \mathbb{R}_{\geq 0}$ the *asymmetric adapted regression spline* $f_{g_+,g_-,\pm}^{*,\lambda} := f_{g_+,g_-,+}^{*,\lambda} + f_{g_+,g_-,-}^{*,\lambda} + \gamma_{g_+,g_-}^{*,\lambda}$ is defined[10] with $\left(f_{g_+,g_-,+}^{*,\lambda}, f_{g_+,g_-,-}^{*,\lambda}, \gamma_{g_+,g_-}^{*,\lambda}\right) \overset{10}{:\in}$

$$\underset{(f_+,f_-,\gamma)\in\mathcal{T}_{g_+,g_-}}{\arg\min} \underbrace{\left(L\left(f_+ + f_- + \gamma\right) + \lambda \left(\frac{\int_{\text{supp}(g_+)} \frac{\left(f_+{''}(x)\right)^2}{g_+(x)} dx}{\mathbb{P}\left[v > 0\right]} + \frac{\int_{\text{supp}(g_-)} \frac{\left(f_-{''}(x)\right)^2}{g_-(x)} dx}{\mathbb{P}\left[v < 0\right]} + \frac{\gamma^2}{\mathbb{P}\left[v = 0\right]\mathbb{E}\left[\max\left(0, b\right)^2\right]} \right)\right)}_{=:F_{+-}^{\lambda,g_+,g_-}(f_+,f_-,\gamma)}, \tag{3.5}$$

[6] Assumption 2e) is in typical scenarios always satisfied. Assumption 2e) together with Assumption 2a) and d) implies that $\mathbb{E}\left[v_k^2 | \xi_k = x\right]$ is bounded on $\text{supp}(g_\xi)$.

[7] Assumption 3a) has to be satisfied in the way Definition 3.1.1 and Theorem 3.1.4 are formulated in this thesis, but all the theory of this thesis could be easily reformulated (see Corollary 3.1.7 for example) if Assumption 3a) were not satisfied. All the theorems of this thesis would hold as well if one replaces $g(0)$ by a fixed value $g(x_{\text{mid}})$ or for example by $\int_{-1}^{1} g(x)dx$, but the results are better interpretable if x_{mid} lies somewhere "in the middle" of the training data. Theorem 3.1.4 would even hold true if one skips $g(0)$ completely by replacing it by 1 (see Corollary 3.1.7 and Definition 3.1.5).

[8] Since all v_k are identically distributed and all ξ_k are identically distributed as well, the conditioned expectation $\mathbb{E}\left[v_k^2 | \xi_k = x\right]$ that obviously only corresponds on their *distribution* does not depend on the choice of $k \in \{1, \ldots, n\}$. Therefor we will sometimes use notations like $\mathbb{E}\left[v | \xi = x\right] := \mathbb{E}\left[v_k | \xi_k = x\right]$

[9] Using the definition of the $\mathbb{P}$-lim, equation (3.4) reads as: $\forall \epsilon \in \mathbb{R}_{>0} : \forall P \in (0,1) : \exists n_0 \in \mathbb{N} : \forall n \geq n_0 :$ $\mathbb{P}\left[\left\| \mathcal{RN}^{*,\tilde{\lambda}} - f_{g,\pm}^{*,\lambda} \right\|_{W^{1,\infty}(K)} < \epsilon\right] > P$.

[10] The optimization problem (3.5) should be interpreted such that $\frac{0}{0}$ is replaced by zero (For example, if $\mathbb{P}\left[v = 0\right] = 0$ the last fraction should be ignored.). The triple $\left(f_{g_+,g_-,+}^{*,\lambda}, f_{g_+,g_-,-}^{*,\lambda}, \gamma_{g_+,g_-}^{*,\lambda}\right)$ and thus the adapted regression spline $f_{g,\pm}^{*,\lambda}$ is uniquely defined if g_+, g_- are probability density functions of distributions with finite first and second moment and if $\exists (i,j) \in \{1, \ldots, N\}^2 : x_i^{\text{train}} \neq x_j^{\text{train}}$.

with

$$\mathcal{T}_{g_+,g_-} := \Big\{(f_+, f_-, \gamma) \in \mathcal{C}^2(\mathbb{R}) \times \mathcal{C}^2(\mathbb{R}) \times \mathbb{R} \Big| \operatorname{supp}(f_+'') \subseteq \operatorname{supp}(g_+), \operatorname{supp}(f_-'') \subseteq \operatorname{supp}(g_-),$$
$$\lim_{x\to-\infty} f_+(x) = 0, \lim_{x\to-\infty} f_+'(x) = 0,$$
$$\lim_{x\to+\infty} f_-(x) = 0, \lim_{x\to+\infty} f_-'(x) = 0,$$
$$\mathbb{P}[v > 0] = 0 \Rightarrow f_+ \equiv 0,$$
$$\mathbb{P}[v < 0] = 0 \Rightarrow f_- \equiv 0,$$
$$\mathbb{P}[v = 0] = 0 \Rightarrow \gamma = 0\Big\}.$$

Definition 3.1.6 (conditioned kink position density g_ξ^+, g_ξ^-)**.** The *conditioned kink position density* $g_\xi^+ : \mathbb{R} \to \mathbb{R}$ of ξ_k conditioned on $v_k > 0$ is defined such that $\int_E g_\xi^+(x)dx = \mathbb{P}[\xi_k \in E | v_k > 0] \quad \forall E \in \mathfrak{B}$. Analogous $\int_E g_\xi^-(x)dx = \mathbb{P}[\xi_k \in E | v_k < 0] \quad \forall E \in \mathfrak{B}$

Corollary 3.1.7 (generalized Theorem 3.1.4)**.** *Let $N \in \mathbb{N}$ be a finite number of arbitrary training data $(x_i^{train}, y_i^{train})$. Using the notation from Definitions 2.0.1, 2.0.4, 3.1.5 and 3.1.6 and let*[11] $\forall x \in \mathbb{R} : g_+(x) := g_\xi^+(x)\mathbb{E}\left[v_k^2 | \xi_k = x, v_k > 0\right]$, $g_-(x) := g_\xi^-(x)\mathbb{E}\left[v_k^2 | \xi_k = x, v_k < 0\right]$ *and* $\tilde{\tilde{\lambda}} := \lambda n$ *then under the Assumptions 1 and 2 the following statement holds for every compact set* $K \subset \mathbb{R}$*:*

$$\mathbb{P}\text{-}\lim_{n\to\infty} \left\| \mathcal{RN}^{*,\tilde{\tilde{\lambda}}} - f^{*,\lambda}_{g_+,g_-,\pm} \right\|_{W^{1,\infty}(K)} = 0.^{12} \tag{3.6}$$

Proof. The proof of Corollary 3.1.7 is analagous to the proof of Theorem 3.1.4 in Section 4.1. (The footnotes 1, 2 and 6 on pages 18, 19 and 22 in Section 4.1 help to understand this analogy.) □

3.2 RSN and Gradient Descent → Implicit Ridge Regularization ($d \in \mathbb{N}$)

The following results in Section 3.2 are analogous to the results presented in [5, 9, 25, 12], but we are going to formulate them here in the context of random shallow networks $\mathcal{RN}$.

Definition 3.2.1 (time-T solution)**.** Let $\forall i \in \{1, \ldots, N\} : (x_i^{\text{train}}, y_i^{\text{train}}) \in \mathbb{R}^{d+1}$ for some $N, d \in \mathbb{N}$ and $\mathcal{RN}_w$ be a randomized shallow network with $n \in \mathbb{N}$ hidden nodes. For any $\omega \in \Omega$ and $T > 0$, the time-T solution to the problem

$$\min_{w\in\mathbb{R}^n} \underbrace{\sum_{i=1}^{N} \left(\mathcal{RN}_{w,\omega}(x_i^{\text{train}}) - y_i^{\text{train}}\right)^2}_{L(\mathcal{RN}_{w,\omega})} \tag{3.7}$$

is defined as $\mathcal{RN}_{w^T(\omega),\omega}$, with weights $w^T(\omega) \in \mathbb{R}^n$ obtained by taking the gradient flow

$$\begin{aligned} dw^t &= -\nabla_w L\left(\mathcal{RN}_{w^t}\right) dt, \\ w^0 &= 0, \end{aligned} \tag{GD}$$

corresponding to (3.7) up to time T.

[11] Since all v_k are identically distributed and all ξ_k are identically distributed as well, the conditioned expectation $\mathbb{E}\left[v_k^2 | \xi_k = x\right]$ that obviously only corresponds on their *distribution* does not depend on the choice of $k \in \{1, \ldots, n\}$.

[12] Using the definition of the $\mathbb{P}$-lim, equation (3.6) reads as: $\forall \epsilon \in \mathbb{R}_{>0} : \forall P \in (0,1) : \exists n_0 \in \mathbb{N} : \forall n \geq n_0 :$ $\mathbb{P}\left[\left\| \mathcal{RN}^{*,\tilde{\tilde{\lambda}}} - f^{*,\lambda}_{g_+,g_-,\pm} \right\|_{W^{1,\infty}(K)} < \epsilon\right] > P$.

Remark 3.2.2. In practice, the weights w^T of the time-T solution as introduced in Definition 3.2.1 are approximated by taking $\tau := T/\gamma$ steps of size $\gamma > 0$ according to the Euler discretization

$$\begin{aligned}\hat{w}^{t+\gamma} &= \hat{w}^t - \gamma \nabla_w L(\mathcal{RN}_{\hat{w}^t}),\\ \hat{w}^0 &= 0,\end{aligned}$$

corresponding to (GD).

Lemma 3.2.3. *Let* $\forall i \in \{1, \dots, N\} : (x_i^{train}, y_i^{train}) \in \mathbb{R}^{d+1}$ *for some* $N, d \in \mathbb{N}$ *and for any* $\omega \in \Omega$*, let* $\mathcal{RN}_{w,\omega}$ *be a randomized shallow network with* $n \geq N$ *hidden nodes. Define further* $X(\omega) \in \mathbb{R}^{N \times n}$ *via*

$$X_{i,k}(\omega) := \sigma\left(b_k(\omega) + \sum_{j=1}^{d} v_{k,j}(\omega) x_{i,j}^{train}\right) \quad \forall i \in \{1, \dots, N\}\, \forall k \in \{1, \dots, n\},$$

where $x_{i,j}^{train}$ *denotes the* j^{th} *component of* x_i^{train}*. For any* $T \geq 0$*, the weights* $w^T(\omega)$ *corresponding to the time-*T *solution* $\mathcal{RN}_{w^T(\omega),\omega}$ *satisfy*

$$w^T(\omega) = -\exp\left(-2TX^\top(\omega)X(\omega)\right) w^\dagger(\omega) + w^\dagger(\omega), \tag{3.8}$$

with weights $w^\dagger(\omega)$ *corresponding to the minimum norm network (see Definition 2.0.5).*

Proof. The proof of Lemma 3.2.3 is formulated in Section 4.2. □

Remark 3.2.4 (limiting solution of gradient descent). By Lemma 3.2.3, the weights w^T corresponding to the time-T solution converge to the minimum norm solution $w^\dagger$ as time tends to infinity—i.e. taking the limit $T \to \infty$ in (3.8), we have $\lim_{T\to\infty} w^T(\omega) = w^\dagger(\omega)\ \forall \omega \in \Omega$.

Proof. The proof of Remark 3.2.4 is formulated in Section 4.2. □

Theorem 3.2.5. *Let* $\mathcal{RN}_{w^T}$ *be the* T*-step solution and consider for* $\tilde{\lambda} = \frac{1}{T}$ *the corresponding ridge solution* $\mathcal{RN}^{*,\frac{1}{T}}$ *(cp. Definitions 2.0.4 and 3.2.1). We then have that*

$$\forall \omega \in \Omega : \quad \lim_{T\to\infty} \left\| \mathcal{RN}_\omega^{*,\frac{1}{T}} - \mathcal{RN}_{w^T(\omega),\omega} \right\|_{W^{1,\infty}(K)} = 0. \tag{3.9}$$

Proof. The proof of Theorem 3.2.5 is formulated in Section 4.2. □

Chapter 4

Proofs

In this chapter, we rigorously prove the results presented throughout this thesis.

4.1 Proof of Theorem 3.1.4 ($\mathcal{RN}^{*,\tilde{\lambda}} \to f_{g,\pm}^{*,\lambda}$)

All the lemmas necessary for the proof of Theorem 3.1.4 will be derived in this section. We start by defining the objects that are central to the subsequent derivations.

Throughout this section, we henceforth require Assumptions 1–3 to be in place.

Definition 4.1.1 (estimated kink distance $\bar{h}$ w.r.t. $\operatorname{sgn}(v)$)**.** Let $\mathcal{RN}$ be a randomized shallow neural network with n hidden nodes as introduced in Definition 2.0.1. The *estimated kink distance w.r.t.* $\operatorname{sgn}(v)$ at the k^{th} kink position ξ_k corresponding to $\mathcal{RN}$ is defined as[1]

$$\bar{h}_k := \frac{2}{n\, g_\xi(\xi_k)}. \tag{4.2}$$

Definition 4.1.2 (spline approximating RSN)**.** Let $\mathcal{RN}$ be a real-valued randomized shallow neural network with n hidden nodes (cp. Definition 2.0.1) and $f_{g,\pm}^{*,\lambda} = f_{g,+}^{*,\lambda} + f_{g,-}^{*,\lambda} \in \mathcal{C}^2(\mathbb{R})$ be the adapted regression spline as introduced in Definition 3.1.1. The *spline approximating RSN* $\mathcal{RN}_{\tilde{w}}$ w.r.t. $f_{g,\pm}^{*,\lambda}$ is given by

$$\mathcal{RN}_{\tilde{w}(\omega),\omega}(x) = \sum_{k=1}^{n} \tilde{w}_k(\omega)\, \sigma\left(b_k(\omega) + v_k(\omega)x\right) \quad \forall \omega \in \Omega\ \forall x \in \mathbb{R} \tag{4.3}$$

[1] Without Assumption 3b) one would define:

$$\bar{h}^+{}_k := \frac{1}{n\mathbb{P}\left[v_k > 0\right]\, g_\xi^+(\xi_k)} \tag{4.1a}$$

$$\bar{h}^-{}_k := \frac{1}{n\mathbb{P}\left[v_k < 0\right]\, g_\xi^-(\xi_k)}. \tag{4.1b}$$

Under Assumption 3b) we have the equality:

$$\bar{h}_k = \bar{h}^+{}_k = \bar{h}^-{}_k. \tag{4.1c}$$

with weights $\tilde{w}(\omega)$ defined as[2]

$$\tilde{w}_k(\omega) := w_k^{f_{g,\pm}^{*,\lambda},n}(\omega) := \begin{cases} \frac{\bar{h}_k(\omega)v_k(\omega)}{\mathbb{E}[v^2|\xi=\xi_k(\omega)]} {f_{g,+}^{*,\lambda}}''(\xi_k(\omega)), & v_k(\omega) > 0 \\ \frac{\bar{h}_k(\omega)v_k(\omega)}{\mathbb{E}[v^2|\xi=\xi_k(\omega)]} {f_{g,-}^{*,\lambda}}''(\xi_k(\omega)), & v_k(\omega) < 0 \end{cases} \qquad \forall k \in \{1,\ldots,n\} \ \forall \omega \in \Omega.$$

Further we define $\forall \omega \in \Omega$:

$$\mathfrak{K}^+(\omega) := \{\, k \in \{1,\ldots,n\} \mid v_k(\omega) > 0 \,\}, \tag{4.4a}$$

$$\mathfrak{K}^-(\omega) := \{\, k \in \{1,\ldots,n\} \mid v_k(\omega) < 0 \,\} \tag{4.4b}$$

and $\tilde{w}^+ := (\tilde{w}_k)_{k\in\mathfrak{K}^+}$ respectively $\tilde{w}^- := (\tilde{w}_k)_{k\in\mathfrak{K}^-}$. With the above, spline approximating RSNs can be alternatively represented as

$$\mathcal{RN}_{\tilde{w}(\omega),\omega}(x) = \underbrace{\sum_{k\in\mathfrak{K}^+(\omega)} \tilde{w}_k(\omega)\,\sigma\left(b_k(\omega) + v_k(\omega)x\right)}_{=:\mathcal{RN}^+_{\tilde{w}^+(\omega),\omega}} + \underbrace{\sum_{k\in\mathfrak{K}^-(\omega)} \tilde{w}_k(\omega)\,\sigma\left(b_k(\omega) + v_k(\omega)x\right)}_{=:\mathcal{RN}^-_{\tilde{w}^-(\omega),\omega}}. \tag{4.5}$$

Remark 4.1.3. The spline approximating RSN introduced in Definition 4.1.2 is a particular randomized shallow neural network designed to be "close" to the adapted regression spline $f_{g,\pm}^{*,\lambda}$ in the sense that its curvature in between kinks is approximately captured by the size of corresponding weights $\tilde{w}$.

Definition 4.1.4 (smooth RSN approximation)**.** For $w^{*,\tilde{\lambda}}$ and $\mathcal{RN}^{*,\tilde{\lambda}}$ as in Definition 2.0.4 with corresponding kink density g_ξ consider for every $x \in \mathbb{R}$ the kernel

$$\kappa_x : \mathbb{R} \to \mathbb{R}, \quad \kappa_x(s) := \mathbb{1}_{B_{\frac{1}{2\sqrt{n}g_\xi(x)}}}(s)\sqrt{n}g_\xi(x) \quad \forall s \in \mathbb{R},$$

where $B_{\frac{1}{2\sqrt{n}g_\xi(x)}} := \{\tau \in \mathbb{R} : |\tau| \leq \frac{1}{2\sqrt{n}g_\xi(x)}\}$. The *smooth RSN approximation* $f^{w^{*,\tilde{\lambda}}}$ then is defined as the convolution[3]

$$f^{w^{*,\tilde{\lambda}}(\omega)}(x) := \left(\mathcal{RN}^{*,\tilde{\lambda}}_\omega * \kappa_x\right)(x) \qquad \forall \omega \in \Omega \ \forall x \in \mathbb{R}. \tag{4.6}$$

Moreover, with the notation

$$\mathcal{RN}^{*,\tilde{\lambda}}(x) = \underbrace{\sum_{k\in\mathfrak{K}^+} w_k^{*,\tilde{\lambda}}\,\sigma\left(b_k + v_k x\right)}_{=:\mathcal{RN}^{*,\tilde{\lambda}}_+} + \underbrace{\sum_{k\in\mathfrak{K}^-} w_k^{*,\tilde{\lambda}}\,\sigma\left(b_k + v_k x\right)}_{=:\mathcal{RN}^{*,\tilde{\lambda}}_-} \quad \forall x \in \mathbb{R}. \tag{4.7}$$

[2]Note that under Assumption 1b), the set $\{v_k = 0\}$ is of zero measure for any $k \in \{1,\ldots,n\}$ and hence is not included in the definition of the weights $\tilde{w}(\omega)$. Without Assumption 3b) (and with a weakened form of Assumption 1b)), $\tilde{w}$ would need to be reformulated:

$$\tilde{w}_k(\omega) := w_k^{f_{g_+,g_-,\pm}^{*,\lambda},n}(\omega) := \begin{cases} \frac{\bar{h}^+{}_k(\omega)v_k(\omega)}{\mathbb{E}\left[v^2|\xi=\xi_k(\omega),v>0\right]} {f_{g_+,g_-,+}^{*,\lambda}}''(\xi_k(\omega)), & v_k(\omega) > 0 \\ \frac{\bar{h}^-{}_k(\omega)v_k(\omega)}{\mathbb{E}\left[v^2|\xi=\xi_k(\omega),v<0\right]} {f_{g_+,g_-,-}^{*,\lambda}}''(\xi_k(\omega)), & v_k(\omega) < 0 \\ \frac{\max(0,b_k(\omega))}{n\mathbb{P}[v=0]\mathbb{E}\left[\max(0,b)^2\right]}\gamma^{*,\lambda}_{g_+,g_-}, & v_k(\omega) = 0 \end{cases} \qquad \forall k \in \{1,\ldots,n\} \ \forall \omega \in \Omega.$$

[3]This "convolution" is a bit special, because the kernel κ_x changes with $x \in \mathbb{R}$. Therefore, the notation $\mathcal{RN}^{*,\tilde{\lambda}} * \kappa$ would not be properly defined, but we could define $\mathcal{RN}^{*,\tilde{\lambda}} \circledast \kappa$ as: $\left(\mathcal{RN}^{*,\tilde{\lambda}}_\omega \circledast \kappa\right)(x) := \left(\mathcal{RN}^{*,\tilde{\lambda}}_\omega * \kappa_x\right)(x) = \int_{\mathbb{R}} \mathcal{RN}^{*,\tilde{\lambda}}_\omega(x-s)\kappa_x(s)ds \quad \forall \omega \in \Omega \ \forall x \in \mathbb{R}$. Hence, $f^{w^{*,\tilde{\lambda}}} := \mathcal{RN}^{*,\tilde{\lambda}} \circledast \kappa$ would be another correct way to define $f^{w^{*,\tilde{\lambda}}}$.

with $w^{*+,\tilde{\lambda}} := \left(w_k^{*,\tilde{\lambda}}\right)_{k\in\mathfrak{K}^+}$ and $w^{*-,\tilde{\lambda}}$ analogously defined as $\tilde{w}^+$ and $\tilde{w}^-$, we have

$$f^{w^{*,\tilde{\lambda}}}(x) = \underbrace{\left(\mathcal{RN}_+^{*,\tilde{\lambda}} * \kappa_x\right)(x)}_{=:f_+^{w^{*,\tilde{\lambda}}}(x)} + \underbrace{\left(\mathcal{RN}_-^{*,\tilde{\lambda}} * \kappa_x\right)(x)}_{=:f_-^{w^{*,\tilde{\lambda}}}(x)} \qquad \forall x \in \mathbb{R}. \tag{4.8}$$

Remark 4.1.5. For any $x \in \mathbb{R}$ the kernel κ_x introduced in Definition 4.1.4 satisfies

1. $\int_{\mathbb{R}} \kappa_x(s)\, ds = 1$ and
2. $\lim_{n\to\infty} \kappa_x = \delta_0$, where δ_0 denotes the Dirac distribution at zero.

Proof of Theorem 3.1.4. The two auxiliary functions $\mathcal{RN}_{\tilde{w}}$ and $f^{w^{*,\tilde{\lambda}}}$ defined above in Definitions 4.1.2 and 4.1.4 will play an important role in this proof.[4]

In the end we want to show the convergence of $\mathcal{RN}^{*,\tilde{\lambda}}$ to $f_{g,\pm}^{*,\lambda}$. Our strategy to achieve this goal is to proof that both these functions $\mathcal{RN}^{*,\tilde{\lambda}}$ and $f_{g,\pm}^{*,\lambda}$ get closer to the same function $f^{w^{*,\tilde{\lambda}}}$ in the limit $n \to \infty$. The first first convergence will be shown in Lemma 4.1.13. The proof of the second convergence $f^{w^{*,\tilde{\lambda}}} \to f_{g,\pm}^{*,\lambda}$ will need more steps—first we will show the convergence $F_{+-}^{\lambda,g}\left(f_+^{w^{*,\tilde{\lambda}}}, f_-^{w^{*,\tilde{\lambda}}}\right) \to F_{+-}^{\lambda,g}\left(f_{g,+}^{*,\lambda}, f_{g,-}^{*,\lambda}\right)$ (in multiple steps based on Lemmas 4.1.10 and 4.1.14) to further imply with the help of Lemma 4.1.17 the convergence $f^{w^{*,\tilde{\lambda}}} \to f_{g,\pm}^{*,\lambda}$.

Following this strategy we proof Theorem 3.1.4 step by step:

step -0.5 Before starting with the proof, we need the auxiliary Lemmas 4.1.6 and 4.1.7

step 0 Lemma 4.1.8 shows

$$\mathbb{P}\text{-}\lim_{n\to\infty} \left\| \mathcal{RN}_{\tilde{w}} - f_{g,\pm}^{*,\lambda} \right\|_{W^{1,\infty}(K)} = 0.$$

step 1 It is directly clear that

$$F_n^{\tilde{\lambda}}\left(\mathcal{RN}^{*,\tilde{\lambda}}\right) \le F_n^{\tilde{\lambda}}\left(\mathcal{RN}_{\tilde{w}}\right),$$

because of the optimality of $\mathcal{RN}^{*,\tilde{\lambda}}$ (see Definition 2.0.4).

step 1.5 The auxiliary Lemma 4.1.9 will be needed for step 2 and step 4

step 2 Lemma 4.1.10 shows

$$\mathbb{P}\text{-}\lim_{n\to\infty} F_n^{\tilde{\lambda}}\left(\mathcal{RN}_{\tilde{w}}\right) = F_{+-}^{\lambda,g}\left(f_{g,+}^{*,\lambda}, f_{g,-}^{*,\lambda}\right).$$

step 2.5 The auxiliary Lemmas 4.1.11 and 4.1.12 will be needed for step 3 and step 4

step 3 Lemma 4.1.13 shows

$$\mathbb{P}\text{-}\lim_{n\to\infty} \left\| \mathcal{RN}^{*,\tilde{\lambda}} - f^{w^{*,\tilde{\lambda}}} \right\|_{W^{1,\infty}(K)} = 0.$$

step 4 Lemma 4.1.14 shows

$$\mathbb{P}\text{-}\lim_{n\to\infty} \left| F_n^{\tilde{\lambda}}\left(\mathcal{RN}^{*,\tilde{\lambda}}\right) - F_{+-}^{\lambda,g}\left(f_+^{w^{*,\tilde{\lambda}}}, f_-^{w^{*,\tilde{\lambda}}}\right) \right| = 0.$$

[4] At the end of the proof we will see that the functions $\mathcal{RN}^{*,\tilde{\lambda}}$, $f^{w^{*,\tilde{\lambda}}}$ and $\mathcal{RN}_{\tilde{w}}$ will converge to the same function $f_{g,\pm}^{*,\lambda}$ in probability with respect to the Sobolev-norm [1] $\|\cdot\|_{W^{1,\infty}(K)}$.

step 5 After defining $\tilde{\mathcal{T}}$ (see Definition 4.1.15) it follows directly (with help of Remark 4.1.16) that

$$F_{+-}^{\lambda,g}\left(f_{g,+}^{*,\lambda}, f_{g,-}^{*,\lambda}\right) \leq F_{+-}^{\lambda,g}\left(f_{+}^{w^{*,\tilde{\lambda}}}, f_{-}^{w^{*,\tilde{\lambda}}}\right)$$

holds, because of the optimality of $\left(f_{g,+}^{*,\lambda}, f_{g,-}^{*,\lambda}\right) \in \tilde{\mathcal{T}}$.

step 6 Combining step 1, step 2, step 4 and step 5 we directly get:[5] and sometimes

$$\begin{aligned} F_{+-}^{\lambda,g}\left(f_{+}^{w^{*,\tilde{\lambda}}}, f_{-}^{w^{*,\tilde{\lambda}}}\right) &\overset{\text{step 4}}{\approx} F_{n}^{\tilde{\lambda}}\left(\mathcal{RN}^{*,\tilde{\lambda}}\right) \overset{\mathbb{P}}{\pm} \epsilon_1 \leq \\ &\overset{\text{step 1}}{\leq} F_{n}^{\tilde{\lambda}}\left(\mathcal{RN}_{\tilde{w}}\right) \overset{\mathbb{P}}{\pm} \epsilon_1 \approx \\ &\overset{\text{step 2}}{\approx} F_{+-}^{\lambda,g}\left(f_{g,+}^{*,\lambda}, f_{g,-}^{*,\lambda}\right) \overset{\mathbb{P}}{\pm} \epsilon_1 \overset{\mathbb{P}}{\pm} \epsilon_2 \overset{\text{step 5}}{\leq} F_{+-}^{\lambda,g}\left(f_{+}^{w^{*,\tilde{\lambda}}}, f_{-}^{w^{*,\tilde{\lambda}}}\right) \overset{\mathbb{P}}{\pm} \epsilon_1 \overset{\mathbb{P}}{\pm} \epsilon_2, \end{aligned}$$

and thus:

$$F_{+-}^{\lambda,g}\left(f_{+}^{w^{*,\tilde{\lambda}}}, f_{-}^{w^{*,\tilde{\lambda}}}\right) \overset{\substack{\text{step 4}\\ \text{step 2}\\ \text{step 1}}}{\lessapprox} F_{+-}^{\lambda,g}\left(f_{g,+}^{*,\lambda}, f_{g,-}^{*,\lambda}\right) \overset{\mathbb{P}}{\pm} \epsilon_3 \qquad \overset{\text{step 5}}{\leq} F_{+-}^{\lambda,g}\left(f_{+}^{w^{*,\tilde{\lambda}}}, f_{-}^{w^{*,\tilde{\lambda}}}\right) \overset{\mathbb{P}}{\pm} \epsilon_3,$$

which directly implies

$$\mathbb{P}\text{-}\lim_{n\to\infty} F_{+-}^{\lambda,g}\left(f_{+}^{w^{*,\tilde{\lambda}}}, f_{-}^{w^{*,\tilde{\lambda}}}\right) = F_{+-}^{\lambda,g}\left(f_{g,+}^{*,\lambda}, f_{g,-}^{*,\lambda}\right). \tag{4.9}$$

step 7 Lemma 4.1.17 shows

$$\mathbb{P}\text{-}\lim_{n\to\infty} \left\| f^{w^{*,\tilde{\lambda}}} - f_{g,\pm}^{*,\lambda} \right\|_{W^{1,\infty}(K)} = 0,$$

if one applies it on the result (4.9) of step 6.

step 8 Combining step 4 and step 7 with the triangle inequality directly results in the statement (3.4) we want show.

$\square$

Lemma 4.1.6 (Poincaré typed inequality). *Let $f : \mathbb{R} \to \mathbb{R}$ differentiable with $f' : \mathbb{R} \to \mathbb{R}$ Lebesgue integrable. Then, for any interval $K = [a,b] \subset \mathbb{R}$ such that $f(a) = 0$ there exists a $C_K^{\infty} \in \mathbb{R}_{>0}$ such that*

$$\|f\|_{W^{1,\infty}(K)} \leq C_K^{\infty} \left\|f'\right\|_{L^{\infty}(K)}. \tag{4.10}$$

If additionally f is twice differentiable with $f'' : \mathbb{R} \to \mathbb{R}$ Lebesgue integrable, there exists a $C_K^2 \in \mathbb{R}_{>0}$ such that

$$\|f\|_{W^{1,\infty}(K)} \leq C_K^{2} \left\|f''\right\|_{L^{2}(K)}. \tag{4.11}$$

[5] We are using the following notation:

$$a_n \approx b_n \overset{\mathbb{P}}{\pm} \epsilon_1 :\Leftrightarrow \forall \epsilon_1 \in \mathbb{R}_{>0} : \forall P_1 \in (0,1) : \exists n_0 \in \mathbb{N} : \forall n \in \mathbb{N}_{>n_0} : \mathbb{P}\left[a_n \in b_n + [-\epsilon_1, \epsilon_1]\right] > P,$$

but a complete formalization of this notation would be quite long. This notation needs to be interpreted depending on the context—e.g.:

$$b_n \overset{\mathbb{P}}{\pm} \epsilon_1 \approx b_n \overset{\mathbb{P}}{\pm} \epsilon_1 \overset{\mathbb{P}}{\pm} \epsilon_2 :\Leftrightarrow \forall \epsilon_2 \in \mathbb{R}_{>0} : \forall P_2 \in (0,1) : \exists n_0 \in \mathbb{N} : \forall n \in \mathbb{N}_{>n_0} : \mathbb{P}\left[b_n \in c_n + [-\epsilon_2, \epsilon_2]\right] > P_2,$$

or sometimes it makes sense to replace "$\in$" by "$\subseteq$" in a reasonable way. And in the proofs of some later lemmas $\overset{\mathbb{P}}{\pm}\epsilon_2$ can have the meaning of $\overset{\mathbb{P}}{\underset{\delta,\epsilon_1\to 0}{\pm}} \epsilon_2$ instead of $\overset{\mathbb{P}}{\underset{n\to 0}{\pm}} \epsilon_2$ depending on the context.

Proof. By the fundamental theorem of calculus, if $\|f'\|_{L^\infty(K)} < \infty$, then

$$\|f\|_{L^\infty(K)} = \sup_{x\in K}\left|\int_a^x f'(y)\,dy\right| \le |b-a| \sup_{y\in K}|f'(y)|.$$

Hence it follows that

$$\|f\|_{W^{1,\infty}(K)} = \max\left\{\|f\|_{L^\infty(K)}, \|f'\|_{L^\infty(K)}\right\} \le \max\{|b-a|,1\}\,\|f'\|_{L^\infty(K)} = C_k^\infty \|f'\|_{L^\infty(K)}.$$

Similarly, by the Hölder inequality we have

$$\|f'\|_{L^\infty(K)} = \sup_{x\in K}\left|\int_a^b f''(y)\mathbb{1}_{[a,x]}(y)\,dy\right| \le \sup_{y\in K}\|f'\|_{L^2(K)}\left\|\mathbb{1}_{[a,y]}\right\|_{L^2(K)} = |b-a|\,\|f''\|_{L^2(K)}.$$

Thus (4.11) follows from

$$\|f\|_{W^{1,\infty}(K)} \le C_K^\infty \|f'\|_{L^\infty(K)} \le C_K^\infty |b-a|\,\|f''\|_{L^2(K)} = C_K^2 \|f''\|_{L^2(K)}.$$

$\square$

Lemma 4.1.7. *Let $\mathcal{RN}$ be a real-valued randomized shallow network. For $\varphi : \mathbb{R}^2 \to \mathbb{R}$ uniformly continuous such that for all $x \in \operatorname{supp}(g_\xi)$, $\mathbb{E}\left[\varphi(\xi,v)\frac{1}{n g_\xi(\xi)}|\xi = x\right] < \infty$, it then holds that*[6]

$$\mathbb{P}\text{-}\lim_{n\to\infty} \sum_{k\in\mathfrak{K}^+:\xi_k<T} \varphi(\xi_k, v_k)\bar{h}_k = \int_{C^\ell_{g_\xi}\wedge T}^{C^u_{g_\xi}\wedge T} \mathbb{E}\left[\varphi(\xi,v)|\xi = x\right]\,dx \tag{4.12}$$

uniformly in $T \in K$.

Proof. For $T \le C^\ell_{g_\xi}$ both sides of (4.12) are zero, thus we restrict ourselves to $T > C^\ell_{g_\xi}$. By uniform continuity of φ and $\frac{1}{g_\xi}$ in ξ, for any $\epsilon > 0$ there exists a $\delta(\epsilon)$ such that for every $|\xi' - \xi| < \delta(\epsilon)$ we have $|\varphi(\xi,v)\frac{1}{g_\xi(\xi)} - \varphi(\xi',v)\frac{1}{g_\xi(\xi')}| < \epsilon$ uniformly in v. W.l.o.g. assume $\operatorname{supp}(g_\xi)$ is an interval. Thus, by splitting the interval $[C^\ell_{g_\xi}, C^u_{g_\xi}\wedge T]$ into disjoint strips[7] of equal length

[6] The same statement as (4.12) is true analogous if one replaces $\mathfrak{K}^+$ by $\mathfrak{K}^-$ of course. Also

$$\mathbb{P}\text{-}\lim_{n\to\infty} \sum_{k:\xi_k<T} \varphi(\xi_k, v_k)\frac{\bar{h}_k}{2} = \int_{C^\ell_{g_\xi}\wedge T}^{C^u_{g_\xi}\wedge T} \mathbb{E}\left[\varphi(\xi,v)|\xi = x\right]\,dx$$

holds analogously. Without Assumption 3b) the statement (4.12) needed to be reformulated as:

$$\mathbb{P}\text{-}\lim_{n\to\infty} \sum_{k\in\mathfrak{K}^+:\xi_k<T} \varphi(\xi_k, v_k)\bar{h}^+{}_k = \int_{C^\ell_{g^+_\xi}\wedge T}^{C^u_{g^+_\xi}\wedge T} \mathbb{E}\left[\varphi(\xi,v)|\xi = x, v>0\right]\,dx$$

$$\mathbb{P}\text{-}\lim_{n\to\infty} \sum_{k\in\mathfrak{K}^-:\xi_k<T} \varphi(\xi_k, v_k)\bar{h}^-{}_k = \int_{C^\ell_{g^-_\xi}\wedge T}^{C^u_{g^-_\xi}\wedge T} \mathbb{E}\left[\varphi(\xi,v)|\xi = x, v<0\right]\,dx$$

[7] Assume $\exists \ell_1, \ell_2 \in \mathbb{Z} : C^\ell_{g_\xi} = \delta\ell_1, C^u_{g_\xi} = \delta\ell_2$ to make the notation simpler. For a cleaner proof, one should choose a suitable partition of $\operatorname{supp}(g_\xi)$.

$\delta \le \delta(\epsilon)$, we have[8]

$$\sum_{\substack{k\in\mathfrak{K}^+\\ \xi_k<T}} \varphi(\xi_k, v_k)\bar{h}_k =$$

$$\stackrel{7}{=} \sum_{\substack{\ell\in\mathbb{Z}\\ [\delta\ell,\delta(\ell+1))\subseteq[C^\ell_{g_\xi},C^u_{g_\xi}\wedge T]}} \left(\sum_{\substack{k\in\mathfrak{K}^+\\ \xi_k\in[\delta\ell,\delta(\ell+1))}} \varphi(\xi_k, v_k)\bar{h}_k \right)$$

$$\approx \sum_{\substack{\ell\in\mathbb{Z}\\ [\delta\ell,\delta(\ell+1))\subseteq[C^\ell_{g_\xi},C^u_{g_\xi}\wedge T]}} \left(\sum_{\substack{k\in\mathfrak{K}^+\\ \xi_k\in[\delta\ell,\delta(\ell+1))}} \left(\varphi(\ell\delta, v_k)\frac{2}{ng_\xi(\ell\delta)} \pm \frac{\epsilon}{n}\right) \frac{|\{m\in\mathfrak{K}^+ : \xi_m\in[\delta\ell,\delta(\ell+1))\}|}{|\{m\in\mathfrak{K}^+ : \xi_m\in[\delta\ell,\delta(\ell+1))\}|} \right)$$

$$\approx \sum_{\substack{\ell\in\mathbb{Z}\\ [\delta\ell,\delta(\ell+1))\subseteq[C^\ell_{g_\xi},C^u_{g_\xi}\wedge T]}} \left(\frac{\sum_{\substack{k\in\mathfrak{K}^+\\ \xi_k\in[\delta\ell,\delta(\ell+1))}} \varphi(\ell\delta, v_k)}{|\{m\in\mathfrak{K}^+ : \xi_m\in[\delta\ell,\delta(\ell+1))\}|} \frac{2|\{m\in\mathfrak{K}^+ : \xi_m\in[\delta\ell,\delta(\ell+1))\}|}{ng_\xi(\ell\delta)} \right) \pm \epsilon.$$

The number of nodes within a δ-strip follows a binomial distribution with

$$\mathbb{E}\left[|\{m\in\mathfrak{K}^+ : \xi_m\in[\delta\ell,\delta(\ell+1))\}|\right] = \mathbb{P}[v_k>0]\, n\int_{[\delta\ell,\delta(\ell+1))} g_\xi(x)\,dx \approx \frac{1}{2}n(\delta g_\xi(\ell\delta)\pm\delta\tilde{\epsilon}),$$

for any $\delta \le \delta(\epsilon,\tilde{\epsilon})$, since g_ξ is uniformly continuous on $\operatorname{supp}(g_\xi)$ by Assumption 2b). For $\delta \le \delta(\epsilon,\tilde{\epsilon})$ small enough we have $\mathcal{L}(v_k) \approx \mathcal{L}(v|\xi=\ell\delta)\ \forall k\in\mathfrak{K}^+ : \xi_k\in[\delta\ell,\delta(\ell+1))$ and we may apply the law of large numbers to further obtain

$$\sum_{k\in\mathfrak{K}^+:\xi_k<T} \varphi(\xi_k, v_k)\bar{h}_k \approx \sum_{\substack{\ell\in\mathbb{Z}\\ [\delta\ell,\delta(\ell+1))\subseteq[C^\ell_{g_\xi},C^u_{g_\xi}\wedge T]}} \left(\mathbb{E}[\varphi(\xi,v)|\xi=\ell\delta] \stackrel{\mathbb{P}}{\pm} \tilde{\tilde{\epsilon}}\right)\delta\left(1\pm\frac{\tilde{\epsilon}}{g_\xi(\ell\delta)}\right)\pm\epsilon$$

$$\approx \left(\sum_{\substack{\ell\in\mathbb{Z}\\ [\delta\ell,\delta(\ell+1))\subseteq[C^\ell_{g_\xi},C^u_{g_\xi}\wedge T]}} \left(\mathbb{E}[\varphi(\xi,v)|\xi=\ell\delta]\,\delta\right) \stackrel{\mathbb{P}}{\pm} \tilde{\tilde{\epsilon}}|C^u_{g_\xi}-C^\ell_{g_\xi}| \right)\left(1\pm\frac{\tilde{\epsilon}}{g_\xi(\ell\delta)}\right)\pm\epsilon$$

Since $1/g_\xi(\cdot)$ and $\mathbb{E}[\varphi(\xi,v)|\xi=\cdot]$ are bounded on $\operatorname{supp}(g_\xi)$, and $\epsilon,\tilde{\epsilon}$ depend on δ only, we may for some $\epsilon^*, P^*\in(0,1)$ define

$$\epsilon := \frac{\epsilon^*}{3}, \tag{4.13a}$$

$$\tilde{\epsilon} := \frac{\epsilon^* \min_{x\in\operatorname{supp}(g_\xi)} g_\xi(x)}{3|C^u_{g_\xi}-C^\ell_{g_\xi}|\left(\max_{x\in\operatorname{supp}(g_\xi)}\mathbb{E}[\varphi(\xi,v)|\xi=x]+1\right)}, \tag{4.13b}$$

$$\tilde{\tilde{\epsilon}} := \frac{\epsilon^*}{3|C^u_{g_\xi}-C^\ell_{g_\xi}|}, \tag{4.13c}$$

$$\tilde{\tilde{P}} := (P^*)^{\frac{\delta}{|C^u_{g_\xi}-C^\ell_{g_\xi}|}}, \tag{4.13d}$$

$$n_0^* := \tilde{n}_0(\tilde{\tilde{\epsilon}},\tilde{\tilde{P}}). \tag{4.13e}$$

[8]The notation $\pm\epsilon$ from footnote 5 on page 21 and slight adaptions of it will be used in this proof a lot. The relations of all the epsilons will be explicitly described in (4.13)

With the above it follows, that for any $\epsilon^*, P^* \in (0,1)$ there exists a n_0^* such that $\forall n > n_0^*$:

$$\mathbb{P}\left[\left|\sum_{k\in\mathfrak{K}^+:\xi_k<T}\varphi(\xi_k,v_k)\bar{h}_k - \sum_{\substack{\ell\in\mathbb{Z}\\ [\delta\ell,\delta(\ell+1))\subseteq[C^\ell_{g_\xi},C^u_{g_\xi}\wedge T]}}\mathbb{E}\left[\varphi(\xi,v)|\xi=\ell\delta\right]\delta\right| \le \epsilon^*\right] > P^*.$$

For δ small enough, the above Riemann sum converges uniformly in T to yield the desired result. □

Lemma 4.1.8 (step 0)**.** *For any choice of penalty parameter $\lambda > 0$ and $K \subset \mathbb{R}$ compact, the spline approximating RSN $\mathcal{RN}_{\tilde{w}}$ converges to the adapted regression spline $f^{*,\lambda}_{g,\pm}$ in probability w.r.t. $\|\cdot\|_{W^{1,\infty}(K)}$ with increasing number of nodes, i.e. for any $\lambda > 0$ and $K \subset \mathbb{R}$ we have*

$$\mathbb{P}\text{-}\lim_{n\to\infty}\left\|\mathcal{RN}_{\tilde{w}} - f^{*,\lambda}_{g,\pm}\right\|_{W^{1,\infty}(K)} = 0.^{9}$$

Proof. Let $\lambda > 0$ and $K \subset \mathbb{R}$ compact with $[C^\ell_g, C^u_g] \subset K$. Directly from the definition (4.5) of $\mathcal{RN}^+_{\tilde{w}+}$ and $\mathcal{RN}^+_{\tilde{w}+}$ and the Definition 3.1.1 of $f^{*,\lambda}_{g,\pm}$ it follows that it is sufficient to show:

$$\mathbb{P}\text{-}\lim_{n\to\infty}\left\|\mathcal{RN}^+_{\tilde{w}+} - f^{*,\lambda}_{g,+}\right\|_{W^{1,\infty}(K)} = 0 \qquad \text{and} \tag{4.14}$$

$$\mathbb{P}\text{-}\lim_{n\to\infty}\left\|\mathcal{RN}^-_{\tilde{w}-} - f^{*,\lambda}_{g,-}\right\|_{W^{1,\infty}(K)} = 0 \qquad . \tag{4.15}$$

W.l.o.g. we restrict ourselves to proving (4.14), as the latter limit follows analogously. By Lemma 4.1.6 it suffices to show that

$$\mathbb{P}\text{-}\lim_{n\to\infty}\left\|{\mathcal{RN}^+_{\tilde{w}+}}' - {f^{*,\lambda}_{g,+}}'\right\|_{L^\infty(K)} = 0. \tag{4.16}$$

Since for any $x \in K$

$${\mathcal{RN}^+_{\tilde{w}+}}'(x) = \sum_{k\in\mathfrak{K}^+}\tilde{w}_k v_k = \sum_{k\in\mathfrak{K}^+}{f^{*,\lambda}_{g,+}}''(\xi_k)\frac{v_k^2}{\mathbb{E}\left[v^2|\xi=\xi_k\right]}\bar{h}_k,$$

we may employ Lemma 4.1.7[10] with $\varphi(z,y) = {f^{*,\lambda}_{g,+}}''(z)\frac{y^2}{\mathbb{E}[v^2|\xi=z]}$ to obtain

$$\mathbb{P}\text{-}\lim_{n\to\infty}{\mathcal{RN}^+_{\tilde{w}+}}'(x) = \int_{C^\ell_{g_\xi}\wedge x}^{C^u_{g_\xi}\wedge x}\mathbb{E}\left[{f^{*,\lambda}_{g,+}}''(\xi)\frac{v^2}{\mathbb{E}\left[v^2|\xi=z\right]}|\xi=z\right]dz = \int_{C^\ell_{g_\xi}\wedge x}^{C^u_{g_\xi}\wedge x}{f^{*,\lambda}_{g,+}}''(z)\,dz$$

uniformly in $x \in K$. Employing the fundamental theorem of calculus we further obtain

$$\mathbb{P}\text{-}\lim_{n\to\infty}{\mathcal{RN}^+_{\tilde{w}+}}'(x) = {f^{*,\lambda}_{g,+}}'(C^u_{g_\xi}\wedge x) - {f^{*,\lambda}_{g,+}}'(C^\ell_{g_\xi}\wedge x) \qquad \forall x \in \mathbb{R}.$$

[9] Using the definition of the $\mathbb{P}$-lim, we get:

$$\forall\epsilon\in\mathbb{R}_{>0} : \forall P\in(0,1) : \exists n_0\in\mathbb{N} : \forall n\ge n_0 : \mathbb{P}\left[\left\|\mathcal{RN}_{\tilde{w}} - f^{*,\lambda}_{g,\pm}\right\|_{W^{1,\infty}(K)} < \epsilon\right] > P.$$

[10] Note that $\varphi(x,y)$ is uniformly continuous on $\mathrm{supp}(g_\xi)$ since by definition $f^{*,\lambda}_{g,+} \in \mathcal{C}^2(\mathbb{R})$ and $\mathrm{supp}(g_\xi)$ is compact by Assumption 2.

By Remark 3.1.2 we have that ${f_{g,+}^{*,\lambda}}'(C_{g_\xi}^{\ell} \wedge x) = 0$ for any $x \in \mathbb{R}$. Since by the same remark, ${f_{g,+}^{*,\lambda}}'$ is constant on $[C_{g_\xi}^{u}, \infty)$, we finally obtain

$$\mathbb{P}\text{-}\lim_{n\to\infty} {\mathcal{RN}_{\tilde{w}^+}^{+}}'(x) = {f_{g,+}^{*,\lambda}}'(x) \qquad \text{uniformly in } x \in K.$$

Hence (4.16) follows.

□

Lemma 4.1.9 ($L(f_n) \to L(f)$)**.** *For any data $(x_i^{train}, y_i^{train}) \in \mathbb{R}^2$, $i \in \{1, \dots, N\}$, let $(f_n)_n \in \mathbb{N})$ be a sequence of functions that converges point-wise[11] in probability to a function $f : \mathbb{R} \to \mathbb{R}$, then the training loss L (c.p. eq. (1.1)) of f_n converges in probability to $L(f)$ as n tends to infinity, i.e.*

$$\mathbb{P}\text{-}\lim_{n\to\infty} L(f_n) = L(f). \tag{4.17}$$

Proof. By continuity, the result follows directly:

$$\begin{aligned}
\mathbb{P}\text{-}\lim_{n\to\infty} L(f_n) &= \mathbb{P}\text{-}\lim_{n\to\infty} \sum_{i=1}^{N} \left(f_n(x_i^{\text{train}}) - y_i^{\text{train}}\right)^2 \\
&= \sum_{i=1}^{N} \left(\mathbb{P}\text{-}\lim_{n\to\infty} f_n(x_i^{\text{train}}) - y_i^{\text{train}}\right)^2 \\
&= \sum_{i=1}^{N} \left(f(x_i^{\text{train}}) - y_i^{\text{train}}\right)^2 = L(f).
\end{aligned}$$

□

Lemma 4.1.10 (step 2)**.** *For any $\lambda > 0$ and data $(x_i^{train}, y_i^{train}) \in \mathbb{R}^2$, $i \in \{1, \dots, N\}$, we have*

$$\mathbb{P}\text{-}\lim_{n\to\infty} F_n^{\tilde{\lambda}}(\mathcal{RN}_{\tilde{w}}) = F_{+-}^{\lambda,g}\left(f_{g,+}^{*,\lambda}, f_{g,-}^{*,\lambda}\right), \tag{4.18}$$

with $\tilde{\lambda}$ and g as defined in Theorem 3.1.4.

Proof. We start by showing

$$\mathbb{P}\text{-}\lim_{n\to\infty} \tilde{\lambda}\, \|\tilde{w}\|_2^2 = 2\lambda g(0) \left(\int_{\operatorname{supp}(g)} \frac{\left({f_{g,+}^{*,\lambda}}''(x)\right)^2}{g(x)} dx + \int_{\operatorname{supp}(g)} \frac{\left({f_{g,-}^{*,\lambda}}''(x)\right)^2}{g(x)} dx \right). \tag{4.19}$$

Since $\|\tilde{w}\|_2^2 = \|\tilde{w}^+\|_2^2 + \|\tilde{w}^-\|_2^2$ we restrict ourselves to proving

$$\mathbb{P}\text{-}\lim_{n\to\infty} \tilde{\lambda}\, \left\|\tilde{w}^+\right\|_2^2 = 2\lambda g(0) \int_{\operatorname{supp}(g_\xi)} \frac{\left({f_{g,+}^{*,\lambda}}''(x)\right)^2}{g(x)} dx. \tag{4.20}$$

[11] If $\mathbb{P}\text{-}\lim_{n\to\infty} \|f_n - f\|_{W^{1,\infty}(K)} = 0$, then f_n converges point-wise in probability to f (by using Sobolev's embedding theorem [1] or by assuming f_n and f to be continuous). Hence Lemma 4.1.9 can be used together with Lemma 4.1.8 to show $\mathbb{P}\text{-}\lim_{n\to\infty} L(\mathcal{RN}_{\tilde{w}}) = L(f_{g,\pm}^{*,\lambda})$ or together with Lemma 4.1.13 to show $\mathbb{P}\text{-}\lim_{n\to\infty} L\left(\mathcal{RN}^{*,\tilde{\lambda}}\right) = L\left(f^{w^{*,\tilde{\lambda}}}\right)$.

With the definitions of $\tilde{w}^+$, $\tilde{\lambda}$ and $\bar{h}$ we have

$$\begin{aligned}
\tilde{\lambda}\left\|\tilde{w}^+\right\|_2^2 &= \tilde{\lambda}\sum_{k\in\mathfrak{K}^+}\left({f_{g,+}^{*,\lambda}}''(\xi_k)\frac{\bar{h}_k v_k}{\mathbb{E}\left[v^2|\xi=\xi_k\right]}\right)^2 \\
&= \tilde{\lambda}\sum_{k\in\mathfrak{K}^+}\left(\left({f_{g,+}^{*,\lambda}}''\right)^2(\xi_k)\frac{\bar{h}_k v_k^2}{\mathbb{E}\left[v^2|\xi=\xi_k\right]}\right)\bar{h}_k \\
&= 2\lambda g(0)\sum_{k\in\mathfrak{K}^+}\left(\left({f_{g,+}^{*,\lambda}}''\right)^2(\xi_k)\frac{v_k^2}{g_\xi(\xi_k)\mathbb{E}\left[v^2|\xi=\xi_k\right]}\right)\bar{h}_k.
\end{aligned}$$

An application of Lemma 4.1.7 with $\varphi(x,y) = \left({f_{g,+}^{*,\lambda}}''\right)^2(x)\frac{y^2}{g_\xi(x)\mathbb{E}[v^2|\xi=y]}$ further yields (4.20) via

$$\begin{aligned}
\mathbb{P}\text{-}\lim_{n\to\infty}\tilde{\lambda}\left\|\tilde{w}^+\right\|_2^2 &= 2\lambda g_\xi(0)\mathbb{E}\left[v^2|\xi=0\right]\int_{\operatorname{supp}(g_\xi)}\mathbb{E}\left[\left({f_{g,+}^{*,\lambda}}''\right)^2(\xi)\frac{v^2}{g_\xi(\xi)\mathbb{E}\left[v^2|\xi=x\right]^2}\middle|\xi=x\right]dx \\
&= 2\lambda g_\xi(0)\mathbb{E}\left[v^2|\xi=0\right]\int_{\operatorname{supp}(g_\xi)}\frac{\left({f_{g,+}^{*,\lambda}}''\right)^2(x)}{g_\xi(x)\mathbb{E}\left[v^2|\xi=x\right]}dx \\
&= 2\lambda g(0)\int_{\operatorname{supp}(g_\xi)}\frac{\left({f_{g,+}^{*,\lambda}}''(x)\right)^2}{g(x)}dx.
\end{aligned}$$

Thus we have proven the convergence of the penalization terms (4.19). Together with Lemmas 4.1.8 and 4.1.9, (4.18) follows.

$\square$

Lemma 4.1.11. *Using the notation of Definitions 2.0.2 and 2.0.4 the following statement holds:*

$$\forall\epsilon\in\mathbb{R}_{>0}:\exists\delta\in\mathbb{R}_{>0}:\forall\omega\in\Omega:\forall l,l'\in\{1,\dots,N\}:\forall n\in\mathbb{N}$$
$$\left(\left(\left|\underbrace{\xi_l(\omega)-\xi_{l'}(\omega)}_{=:\Delta\xi(\omega)}\right|<\delta\ \wedge\ \operatorname{sgn}(v_l(\omega))=\operatorname{sgn}(v_{l'}(\omega))\right)\Rightarrow\left|\frac{w_l^{*,\tilde{\lambda}}(\omega)}{v_l(\omega)}-\frac{w_{l'}^{*,\tilde{\lambda}}(\omega)}{v_{l'}(\omega)}\right|<\frac{\epsilon}{n}\right),$$

if we assume that v_k is never zero.

Proof. We will proof the even stronger statement:

$$\left|\frac{w_l^{*,\tilde{\lambda}}}{v_l}-\frac{w_{l'}^{*,\tilde{\lambda}}}{v_{l'}}\right| \underset{\substack{\text{conditioned on}\\ \operatorname{sgn}(v_l)=\operatorname{sgn}(v_{l'})}}{\overset{1}{\leq}} \frac{|\Delta\xi|}{\tilde{\lambda}}\sum_{i=1}^N\left|\mathcal{RN}^{*,\tilde{\lambda}}(x_i^{\text{train}})-y_i^{\text{train}}\right|\overset{2}{\leq} \tag{4.21a}$$

$$\overset{2}{\leq}\frac{|\Delta\xi|}{\tilde{\lambda}}\sqrt{N}\sqrt{\sum_{i=1}^N\left|\mathcal{RN}^{*,\tilde{\lambda}}(x_i^{\text{train}})-y_i^{\text{train}}\right|^2}\overset{3}{\leq}\frac{|\Delta\xi|}{\tilde{\lambda}}\sqrt{N}\sqrt{\sum_{i=1}^N|y_i^{\text{train}}|^2}, \tag{4.21b}$$

because with the help of inequality (4.21), $\delta := \frac{\epsilon\lambda g(0)}{\sqrt{N\sum_{i=1}^N|y_i^{\text{train}}|^2}}$ would be a valid choice of δ in the statement of Lemma 4.1.11.

1. Proof of (4.21a): First we define the disturbed weight vector $w^{\Delta s}$ such that

$$w_k^{\Delta s} := w_k^{*,\tilde{\lambda}} + \begin{cases} +\frac{\Delta s}{|v_l|} & k = l \\ -\frac{\Delta s}{|v_{l'}|} & k = l' \\ 0 & \text{else-wise} \end{cases}$$

by shifting a little bit of the distributional second derivative Δs from the l'th kink to the lth kink. By a case analysis (or by drawing a sketch) one can easily show conditioned on $\operatorname{sgn}(v_l) = \operatorname{sgn}(v_{l'})$:

$$\forall x \in \mathbb{R} : \left| \mathcal{RN}^{*,\tilde{\lambda}}(x) - \left(\mathcal{RN}_{w^{\Delta s}}(x)\right) \right| \leq \Delta x \Delta s. \tag{4.22}$$

As $\mathcal{RN}^{*,\tilde{\lambda}}$ is optimal the derivative

$$0 = \left. \frac{dF_n^{\tilde{\lambda}}\left(\mathcal{RN}_{w^{\Delta s}}\right)}{d\Delta s} \right|_{\Delta s=0} = \tilde{\lambda} 2 \left(\frac{w_l^{*,\tilde{\lambda}}}{v_l} - \frac{w_{l'}^{*,\tilde{\lambda}}}{v_{l'}} \right) + \left. \frac{dL\left(\mathcal{RN}_{w^{\Delta s}}\right)}{d\Delta s} \right|_{\Delta s=0} \tag{4.23}$$

has to be zero. Transforming this equation and taking absolute values on both sides gives:

$$\left| \tilde{\lambda} 2 \left(\frac{w_l^{*,\tilde{\lambda}}}{v_l} - \frac{w_{l'}^{*,\tilde{\lambda}}}{v_{l'}} \right) \right| \overset{(4.23)}{=} \left| \left. \frac{dL\left(\mathcal{RN}_{w^{\Delta s}}\right)}{d\Delta s} \right|_{\Delta s=0} \right| \overset{(4.22)}{\leq} 2 \sum_{i=1}^{N} \left| \left(\mathcal{RN}^{*,\tilde{\lambda}}(x_i^{\text{train}}) - y_i^{\text{train}} \right) \Delta\xi \right|. \tag{4.24}$$

Dividing both sides by $2\tilde{\lambda}$ results in (4.21a).

2. (4.21a)$\leq$(4.21b) holds because of the general inequality $\forall a \in \mathbb{R}^N : \|a\|_1 \leq \sqrt{N} \|a\|_2$.

3. (4.21b) holds because the optimal network $\mathcal{RN}^{*,\tilde{\lambda}}$will never be worse than the 0-function.

□

Lemma 4.1.12 **($\frac{w^{*,\tilde{\lambda}}}{v} \approx \mathcal{O}(\frac{1}{n})$).** *For any $\lambda > 0$ and data $(x_i^{train}, y_i^{train}) \in \mathbb{R}^2$, $i \in \{1, \dots, N\}$, we have*

$$\max_{k \in \{1,\dots,n\}} \frac{w_k^{*,\tilde{\lambda}}}{v_k} = \mathbb{P}\text{-}\mathcal{O}_{n\to\infty}\left(\frac{1}{n}\right).[12] \tag{4.25}$$

Proof. Let $k^* \in \arg\max_{k \in \{1,\dots,n\}} \frac{w_k^{*,\tilde{\lambda}}}{v_k}$ and thus $\frac{w_{k^*}^{*,\tilde{\lambda}}}{v_{k^*}} = \max_{k \in \{1,\dots,n\}} \frac{w_k^{*,\tilde{\lambda}}}{v_k}$. W.l.o.g. assume

[12] Using the definition of $\mathbb{P}$-$\mathcal{O}$, eq. (4.25) reads as:

$$\forall P \in (0,1) : \exists C \in \mathbb{R}_{>0} : \exists n_0 \in \mathbb{N} : \forall n > n_0 : \mathbb{P}\left[\max_{k \in \{1,\dots,n\}} < C\frac{1}{n} \right] > P.$$

$k^* \in \mathfrak{K}^+$.

$$
\begin{aligned}
\frac{F_{+-}^{\lambda,g}\left(f_{g,+}^{*,\lambda}, f_{g,-}^{*,\lambda}\right)}{\tilde{\lambda}} &\overset{\text{Lemma 4.1.10}}{\underset{}{\overset{\mathbb{P}}{\geq}}} \frac{1}{2\tilde{\lambda}} F_n^{\tilde{\lambda}}\left(\mathcal{RN}^{*,\tilde{\lambda}}\right) && (4.26a)\\
&\geq \frac{1}{2} \sum_{k \in \mathfrak{K}^+ : \xi_k \in (\xi_{k^*}, \xi_{k^*}+\delta)} {w_k^{*,\tilde{\lambda}}}^2 && (4.26b)\\
&= \frac{1}{2} \sum_{k \in \mathfrak{K}^+ : \xi_k \in (\xi_{k^*}, \xi_{k^*}+\delta)} \frac{{w_k^{*,\tilde{\lambda}}}^2}{v_k^2} v_k^2 && (4.26c)\\
&\overset{\text{Lemma 4.1.11}}{\geq} \frac{1}{4} \frac{{w_{k^*}^{*,\tilde{\lambda}}}^2}{v_{k^*}^2} \sum_{k \in \mathfrak{K}^+ : \xi_k \in (\xi_{k^*}, \xi_{k^*}+\delta)} v_k^2 && (4.26d)\\
&\overset{\mathbb{P}}{\geq} \frac{1}{8} \frac{{w_{k^*}^{*,\tilde{\lambda}}}^2}{v_{k^*}^2} \frac{n\delta g_\xi(\xi_{k^*})}{2} \mathbb{E}\left[v_k^2 \middle| \xi_k = \xi_{k^*}\right]. && (4.26e)
\end{aligned}
$$

Transforming inequality (4.26) and using the definition $\tilde{\lambda} := \lambda n g(0)$ gives:

$$
\frac{{w_{k^*}^{*,\tilde{\lambda}}}^2}{v_{k^*}^2} \overset{\mathbb{P}}{\leq} \frac{16}{n^2} \frac{F_{+-}^{\lambda,g}\left(f_{g,+}^{*,\lambda}, f_{g,-}^{*,\lambda}\right)}{\delta g_\xi(\xi_{k^*}) \lambda g(0)}. \tag{4.27}
$$

Taking the square root of both sides an using some bounds, we get:

$$
\frac{w_{k^*}^{*,\tilde{\lambda}}}{v_{k^*}} \overset{\mathbb{P}}{\leq} \frac{4}{n} \left(\frac{F_{+-}^{\lambda,g}\left(f_{g,+}^{*,\lambda}, f_{g,-}^{*,\lambda}\right)}{\delta \min_{x \in \operatorname{supp}(g)} g_\xi(x) \lambda g(0)} \right)^{\frac{1}{2}}. \tag{4.28}
$$

This proofs statement (4.25) by choosing C from footnote 12 as:

$$
C := 4 \left(\frac{F_{+-}^{\lambda,g}\left(f_{g,+}^{*,\lambda}, f_{g,-}^{*,\lambda}\right)}{\delta \min_{x \in \operatorname{supp}(g)} g_\xi(x) \lambda g(0)} \right)^{\frac{1}{2}}. \tag{4.29}
$$

□

Lemma 4.1.13 (step 3)**.** *For any $\lambda > 0$ and data $(x_i^{train}, y_i^{train}) \in \mathbb{R}^2$, $i \in \{1, \dots, N\}$, we have*

$$
\mathbb{P}\text{-}\lim_{n \to \infty} \left\| \mathcal{RN}^{*,\tilde{\lambda}} - f^{w^{*,\tilde{\lambda}}} \right\|_{W^{1,\infty}(K)} = 0, \tag{4.30}
$$

with $\tilde{\lambda}$ as defined in Theorem 3.1.4.

Proof. By Lemma 4.1.6 (as $\mathcal{RN}^{*,\tilde{\lambda}}$, $f^{w^{*,\tilde{\lambda}}}$ are zero outside of $\operatorname{supp}(g) + \operatorname{supp}(\kappa_x)$ like described in Remark 3.1.2), we only need to show that for all $\epsilon > 0$:

$$
\lim_{n \to \infty} \mathbb{P}\left[\left\| \mathcal{RN}^{*,\tilde{\lambda}'} - f^{w^{*,\tilde{\lambda}'}} \right\|_{L^\infty(K)} < \epsilon \right] = 1.
$$

W.l.o.g. it is sufficient to prove:

$$
\lim_{n \to \infty} \mathbb{P}\left[\left\| \mathcal{RN}_+^{*,\tilde{\lambda}'} - f_+^{w^{*,\tilde{\lambda}'}} \right\|_{L^\infty(K)} < \epsilon \right] = 1.
$$

For every $x \in K$ and $\omega \in \Omega$, using the Definition 4.1.4 of $f_+^{w^*,\tilde{\lambda}}$ we have

$$\mathcal{RN}_+^{*,\tilde{\lambda}\,\prime}(x) - f_+^{w^*,\tilde{\lambda}\,\prime}(x) = \mathcal{RN}_+^{*,\tilde{\lambda}\,\prime}(x) - \left(\mathcal{RN}_+^{*,\tilde{\lambda}\,\prime} * \kappa_x\right)(x)$$
$$= \int_{\mathbb{R}} \mathcal{RN}_+^{*,\tilde{\lambda}\,\prime}(x)\kappa_x(t)\,dt - \int_{\mathbb{R}} \mathcal{RN}_+^{*,\tilde{\lambda}\,\prime}(x-t)\kappa_x(t)\,dt$$
$$= \int_{\mathbb{R}} \left(\mathcal{RN}_+^{*,\tilde{\lambda}\,\prime}(x) - \mathcal{RN}_+^{*,\tilde{\lambda}\,\prime}(x-t)\right)\kappa_x(t)\,dt.$$

Using the definition of $\mathcal{RN}_+^{*,\tilde{\lambda}}$ we get:

$$\mathcal{RN}_+^{*,\tilde{\lambda}\,\prime}(x) = \sum_{k \in \mathfrak{K}^+ : \xi_k < x} w_k^{*,\tilde{\lambda}} v_k \tag{4.31}$$

and hence with $r_n := \frac{1}{2\sqrt{n}g_\xi(x)}$ we can get after some algebraic calculations:

$$\mathcal{RN}_+^{*,\tilde{\lambda}\,\prime}(x) - f_+^{w^*,\tilde{\lambda}\,\prime}(x) = \sum_{k \in \mathfrak{K}^+ : x - r_n < \xi_k < x} w_k^{*,\tilde{\lambda}} v_k \int_{x-r_n}^{\xi_k} \kappa_x(s-x)ds$$
$$- \sum_{k \in \mathfrak{K}^+ : x < \xi_k < x + r_n} w_k^{*,\tilde{\lambda}} v_k \int_{\xi_k}^{x+r_n} \kappa_x(s-x)ds =$$
$$= \sum_{k \in \mathfrak{K}^+ : x - r_n < \xi_k < x} \frac{w_k^{*,\tilde{\lambda}}}{v_k} v_k^2 \int_{x-r_n}^{\xi_k} \kappa_x(s-x)ds$$
$$- \sum_{k \in \mathfrak{K}^+ : x < \xi_k < x + r_n} \frac{w_k^{*,\tilde{\lambda}}}{v_k} v_k^2 \int_{\xi_k}^{x+r_n} \kappa_x(s-x)ds$$

Thus we can use the triangle inequality[13] and the properties of the kernel κ_x to get:

$$\left|\mathcal{RN}_+^{*,\tilde{\lambda}\,\prime}(x) - f_+^{w^*,\tilde{\lambda}\,\prime}(x)\right| \leq \frac{1}{2} \sum_{k \in \mathfrak{K}^+ : x - r_n < \xi_k < x + r_n} \left|\frac{w_k^{*,\tilde{\lambda}}}{v_k} v_k^2\right| \tag{4.32a}$$
$$\leq \frac{1}{2} \max_{k \in \mathfrak{K}^+} \left|\frac{w_k^{*,\tilde{\lambda}}}{v_k}\right| \sum_{k \in \mathfrak{K}^+ : x - r_n < \xi_k < x + r_n} v_k^2 \tag{4.32b}$$
$$\overset{\text{Lemma 4.1.12}}{\leq} \mathbb{P}\text{-}\mathcal{O}_{n\to\infty}\left(\frac{1}{n}\right) \mathbb{P}\text{-}\mathcal{O}_{n\to\infty}\left(\sqrt{n}\right) = \mathbb{P}\text{-}\mathcal{O}_{n\to\infty}\left(\frac{1}{\sqrt{n}}\right) \tag{4.32c}$$

uniformly in x on $\operatorname{supp}(g_\xi)$ and thus on K (since outside of $\operatorname{supp}(g_\xi) + (-r_n, r_n)$ both functions and there derivatives are zero). □

Lemma 4.1.14 (step 4)**.** *For any $\lambda > 0$ and data $(x_i^{train}, y_i^{train}) \in \mathbb{R}^2$, $i \in \{1, \ldots, N\}$, we have*

$$\mathbb{P}\text{-}\lim_{n\to\infty} \left| F_n^{\tilde{\lambda}}\left(\mathcal{RN}^{*,\tilde{\lambda}}\right) - F_{+-}^{\lambda,g}\left(f_+^{w^*,\tilde{\lambda}}, f_-^{w^*,\tilde{\lambda}}\right)\right| = 0, \tag{4.33}$$

with $\tilde{\lambda}$ as defined in Theorem 3.1.4.

[13] Actually one could use a much tighter bound the triangle inequality used in inequality (4.32a), because in asymptotic expectation the positive and negative summands would cancel each other instead of adding up.

Proof. Lemmas 4.1.9 and 4.1.13 show together that

$$\mathbb{P}\text{-}\lim_{n\to\infty}\left|L\left(\mathcal{RN}^{*,\tilde{\lambda}}\right)-L\left(f_+^{w^{*,\tilde{\lambda}}},f_-^{w^{*,\tilde{\lambda}}}\right)\right|=0.$$

So it is sufficient to show:

$$\mathbb{P}\text{-}\lim_{n\to\infty}\left|\tilde{\lambda}\left\|w^{*,\tilde{\lambda}}\right\|_2^2-2\lambda g(0)\left(\int_{\text{supp}(g)}\frac{\left(f_+^{w^{*,\tilde{\lambda}}\,\prime\prime}(x)\right)^2}{g(x)}dx+\int_{\text{supp}(g)}\frac{\left(f_-^{w^{*,\tilde{\lambda}}\,\prime\prime}(x)\right)^2}{g(x)}dx\right)\right|=0. \tag{4.34}$$

Since $\left\|w^{*,\tilde{\lambda}}\right\|_2^2=\sum_{k\in\mathfrak{K}^+}{w_k^{*,\tilde{\lambda}}}^2+\sum_{k\in\mathfrak{K}^-}{w_k^{*,\tilde{\lambda}}}^2$, we restrict ourselves to proving

$$\mathbb{P}\text{-}\lim_{n\to\infty}\left|\tilde{\lambda}\sum_{k\in\mathfrak{K}^+}{w_k^{*,\tilde{\lambda}}}^2-2\lambda g(0)\int_{\text{supp}(g_\xi)}\frac{\left(f_+^{w^{*,\tilde{\lambda}}\,\prime\prime}(x)\right)^2}{g(x)}dx\right|=0. \tag{4.35}$$

Using the Definition 4.1.4 of $f_+^{w^{*,\tilde{\lambda}}}$ we get:

$$f_+^{w^{*,\tilde{\lambda}}\,\prime\prime}(x)\overset{\text{Definition 4.1.4}}{=}\sum_{k\in\mathfrak{K}^+:|\xi_k-x|<\frac{1}{2\sqrt{n}g_\xi(x)}}\sqrt{n}g_\xi(x)w_k^{*,\tilde{\lambda}}v_k \tag{4.36a}$$

$$=\sum_{k\in\mathfrak{K}^+:|\xi_k-x|<\frac{1}{2\sqrt{n}g_\xi(x)}}\sqrt{n}g_\xi(x)\frac{w_k^{*,\tilde{\lambda}}}{v_k}v_k^2 \tag{4.36b}$$

$$\overset{\text{Lemma 4.1.11}}{\approx}\left(\frac{w_{l_x}^{*,\tilde{\lambda}}}{v_{l_x}}\pm\frac{\epsilon}{n}\right)\sum_{k\in\mathfrak{K}^+:|\xi_k-x|<\frac{1}{2\sqrt{n}g_\xi(x)}}\sqrt{n}g_\xi(x)v_k^2 \tag{4.36c}$$

$$\approx\left(\frac{w_{l_x}^{*,\tilde{\lambda}}}{v_{l_x}}\pm\frac{\epsilon}{n}\right)\left(1\overset{\mathbb{P}}{\pm}\epsilon_1\right)\mathbb{P}\left[v_k>0\right]ng_\xi(x)\left(\mathbb{E}\left[v_k^2|\xi_k=x\right]\overset{\mathbb{P}}{\pm}\epsilon_2\right) \tag{4.36d}$$

$$\overset{\text{Lemma 4.1.12}}{\approx}\frac{w_{l_x}^{*,\tilde{\lambda}}}{v_{l_x}}\mathbb{P}\left[v_k>0\right]ng_\xi(x)\mathbb{E}\left[v_k^2|\xi_k=x\right]\overset{\mathbb{P}}{\pm}\epsilon_3 \tag{4.36e}$$

uniformly in x on K for any l_x satisfying $l_x\in\mathfrak{K}^+:|\xi_l-x|<\frac{1}{2\sqrt{n}g_\xi(x)}\ \forall x\in\text{supp}(g_\xi)$. Therefore we can plug this into the right-hand term of eq. (4.35):

$$2\lambda g(0)\int_{\text{supp}(g_\xi)}\frac{\left(f_+^{w^{*,\tilde{\lambda}}\,\prime\prime}(x)\right)^2}{g(x)}dx\approx 2\lambda g(0)\int_{\text{supp}(g_\xi)}\frac{\left(\frac{w_{l_x}^{*,\tilde{\lambda}}}{v_{l_x}}\mathbb{P}\left[v_k>0\right]ng_\xi(x)\mathbb{E}\left[v_k^2|\xi_k=x\right]\overset{\mathbb{P}}{\pm}\epsilon_3\right)^2}{g(x)}dx$$

$$\approx\underbrace{2\lambda g(0)\int_{\text{supp}(g_\xi)}\frac{\left(\frac{w_{l_x}^{*,\tilde{\lambda}}}{v_{l_x}}\mathbb{P}\left[v_k>0\right]ng_\xi(x)\mathbb{E}\left[v_k^2|\xi_k=x\right]\right)^2}{g(x)}dx}_{=\frac{\tilde{\lambda}n}{2}\int_{\text{supp}(g_\xi)}\left(\frac{w_{l_x}^{*,\tilde{\lambda}}}{v_{l_x}}\right)^2g_\xi(x)\mathbb{E}\left[v_k^2|\xi_k=x\right]dx}\overset{\mathbb{P}}{\pm}\epsilon_4$$

by uniformity of approximation (4.36) and by using the definitions of $\tilde{\lambda} := \lambda n g(0)$ and $g(x) := g_\xi(x)\mathbb{E}\left[v_k^2 \middle| \xi_k = x\right]$. In the next steps we show that the left-hand term of eq. (4.35) converges to the same term as the right-hand side did:[14]

$$
\begin{aligned}
\tilde{\lambda} \sum_{k \in \mathfrak{K}^+} {w_k^{*,\tilde{\lambda}}}^2 \quad &\overset{14}{=} \quad \tilde{\lambda} \sum_{\substack{\ell \in \mathbb{Z} \\ [\delta\ell, \delta(\ell+1)) \subseteq [C^\ell_{g_\xi}, C^u_{g_\xi}]}} \left(\sum_{\substack{k \in \mathfrak{K}^+ \\ \xi_k \in [\delta\ell, \delta(\ell+1))}} \left(\frac{w_k^{*,\tilde{\lambda}}}{v_k} \right)^2 v_k^2 \right) \\
&\overset{\text{Lemma 4.1.11}}{\approx} \tilde{\lambda} \sum_{\substack{\ell \in \mathbb{Z} \\ [\delta\ell, \delta(\ell+1)) \subseteq [C^\ell_{g_\xi}, C^u_{g_\xi}]}} \left(\left(\frac{w_{l_{\delta\ell}}^{*,\tilde{\lambda}}}{v_{l_{\delta\ell}}} \pm \frac{\epsilon_5}{n} \right)^2 \underbrace{\sum_{\substack{k \in \mathfrak{K}^+ \\ \xi_k \in [\delta\ell, \delta(\ell+1))}} v_k^2}_{\approx \left(1 \overset{\mathbb{P}}{\pm} \epsilon_6\right) \frac{n}{2} \delta g_\xi(\delta\ell) \left(\mathbb{E}\left[v_k^2 \middle| \xi_k = \delta\ell\right] \overset{\mathbb{P}}{\pm} \epsilon_7\right)} \right) \\
&\overset{\text{Lemma 4.1.12}}{\approx} \frac{\tilde{\lambda} n}{2} \sum_{\substack{\ell \in \mathbb{Z} \\ [\delta\ell, \delta(\ell+1)) \subseteq [C^\ell_{g_\xi}, C^u_{g_\xi}]}} \left(\left(\frac{w_{l_{\delta\ell}}^{*,\tilde{\lambda}}}{v_{l_{\delta\ell}}} \right)^2 \delta g_\xi(\delta\ell) \left(\mathbb{E}\left[v_k^2 \middle| \xi_k = \delta\ell\right] \right) \overset{\mathbb{P}}{\pm} \epsilon_8 \right) \\
&\overset{\text{Riemann}}{\approx} \quad \frac{\tilde{\lambda} n}{2} \int_{\operatorname{supp}(g_\xi)} \left(\frac{w_{l_x}^{*,\tilde{\lambda}}}{v_{l_x}} \right)^2 g_\xi(x) \mathbb{E}\left[v_k^2 \middle| \xi_k = x\right] \, dx \overset{\mathbb{P}}{\pm} \epsilon_9
\end{aligned}
$$

This proves eq. (4.33). □

Definition 4.1.15 (extended feasible set $\tilde{\mathcal{T}}$)**.** The *extended feasible set* $\tilde{\mathcal{T}}$ is defined as:

$$
\begin{aligned}
\tilde{\mathcal{T}} := \Bigg\{ (f_+, f_-) \in H^2(\mathbb{R}) \times H^2(\mathbb{R}) \Bigg| \; & \operatorname{supp}(f_+'') \subseteq \operatorname{supp}(g), \operatorname{supp}(f_-'') \subseteq \operatorname{supp}(g), \\
& f_+(x) = 0 = f_+'(x) \quad \forall x \le C_g^\ell, \\
& f_-(x) = 0 = f_-'(x) \quad \forall x \ge C_g^u \Bigg\}.
\end{aligned}
$$

by replacing $\mathcal{C}^2(\mathbb{R})$ by the Sobolev space [1] $H^2(\mathbb{R}) := W^{2,2}(\mathbb{R}) \supset \mathcal{C}^2(\mathbb{R})$ in $\mathcal{T}$ from Definition 3.1.1.

Remark 4.1.16. If one replaces $\mathcal{C}^2(\mathbb{R})$ by the Sobolev space $H^2(\mathbb{R}) := W^{2,2}(\mathbb{R})$ in Definition 3.1.1 the minimizer $\left(f_{g,+}^{*,\lambda}, f_{g,-}^{*,\lambda}\right)$ does not change—i.e.:

$$
\underset{(f_+, f_-) \in \mathcal{T}}{\arg\min} \; F_{+-}^{\lambda, g}(f_+, f_-) = \underset{(f_+, f_-) \in \tilde{\mathcal{T}}}{\arg\min} \; F_{+-}^{\lambda, g}(f_+, f_-) \,.
$$

Lemma 4.1.17 (step 7)**.** *For any $\lambda > 0$ and data $(x_i^{train}, y_i^{train}) \in \mathbb{R}^2$, $i \in \{1, \dots, N\}$, for any sequence of tuples of functions $\left(f_+^n, f_-^n\right) \in H^2(\mathbb{R}) \times H^2(\mathbb{R})$ such that*

$$
\mathbb{P}\text{-}\lim_{n \to \infty} F_{+-}^{\lambda, g}\left(f_+^n, f_-^n\right) = F_{+-}^{\lambda, g}\left(f_{g,+}^{*,\lambda}, f_{g,-}^{*,\lambda}\right), \tag{4.37}
$$

then it follows that:

$$
\mathbb{P}\text{-}\lim_{n \to \infty} \left\| \left(f_+^n + f_-^n\right) - \underbrace{f_{g,\pm}^{*,\lambda}}_{f_{g,+}^{*,\lambda} + f_{g,-}^{*,\lambda}} \right\|_{W^{1,\infty}(K)} = 0. \tag{4.38}
$$

[14] Assume $\exists \ell_1, \ell_2 \in \mathbb{Z} : C^\ell_{g_\xi} = \delta\ell_1, C^u_{g_\xi} = \delta\ell_2$ to make the notation simpler. For a cleaner proof, one should choose a suitable partition of $\operatorname{supp}(g_\xi)$.

Proof. Define the tuple of $H^2(\mathbb{R})$-functions

$$\left(u_+^n, u_-^n\right) := \left(f_{g,+}^{*,\lambda}, f_{g,-}^{*,\lambda}\right) - \left(f_+^n, f_-^n\right) \tag{4.39}$$

as the difference. The difference $\left(u_+^n, u_-^n\right)$ of elements from $\mathcal{T}$ and $\tilde{\mathcal{T}}$ obviously lies in $\tilde{\mathcal{T}}$.

Define the penalty term of $F_{+-}^{\lambda,g}$ as:

$$P_{+-}^{\lambda,g}\left(f_+, f_-\right) := 2\lambda g(0) \left(\int_{\operatorname{supp}(g)} \frac{\left(f_+{}''(x)\right)^2}{g(x)} dx + \int_{\operatorname{supp}(g)} \frac{\left(f_-{}''(x)\right)^2}{g(x)} dx \right). \tag{4.40}$$

This penalty $P_{+-}^{\lambda,g}$ is obviously a quadratic form. Note that $\frac{(f_+^n, f_-^n) + \left(f_{g,+}^{*,\lambda}, f_{g,-}^{*,\lambda}\right)}{2} \in \tilde{\mathcal{T}}$. Since the training loss L is convex, we get the inequality:

$$L\left(\frac{f_+^n + f_-^n + f_{g,+}^{*,\lambda} + f_{g,-}^{*,\lambda}}{2}\right) \leq \frac{L\left(f_+^n + f_-^n\right)}{2} + \frac{L\left(f_{g,+}^{*,\lambda} + f_{g,-}^{*,\lambda}\right)}{2}. \tag{4.41}$$

Since the penalty $P_{+-}^{\lambda,g}$ is a quadratic form, we get with the help of some algebraic calculations the inequality:

$$P_{+-}^{\lambda,g}\left(\frac{(f_+^n, f_-^n) + \left(f_{g,+}^{*,\lambda}, f_{g,-}^{*,\lambda}\right)}{2}\right) \leq \frac{P_{+-}^{\lambda,g}(f_+^n, f_-^n)}{2} + \frac{P_{+-}^{\lambda,g}\left(f_{g,+}^{*,\lambda}, f_{g,-}^{*,\lambda}\right)}{2} - \frac{P_{+-}^{\lambda,g}(u_+^n, u_-^n)}{4}. \tag{4.42}$$

Adding the inequalities (4.41) and (4.42) results in:

$$F_{+-}^{\lambda,g}\left(\frac{(f_+^n, f_-^n) + \left(f_{g,+}^{*,\lambda}, f_{g,-}^{*,\lambda}\right)}{2}\right) \leq \underbrace{\frac{F_{+-}^{\lambda,g}(f_+^n, f_-^n) + F_{+-}^{\lambda,g}\left(f_{g,+}^{*,\lambda}, f_{g,-}^{*,\lambda}\right)}{2}}_{\overset{(4.37)}{\approx} F_{+-}^{\lambda,g}\left(f_{g,+}^{*,\lambda}, f_{g,-}^{*,\lambda}\right) \overset{\mathbb{P}}{\pm} \epsilon} - \frac{P_{+-}^{\lambda,g}(u_+^n, u_-^n)}{4}. \tag{4.43}$$

Together with the optimality of $\left(f_{g,+}^{*,\lambda}, f_{g,-}^{*,\lambda}\right)$ this result leads directly to:

$$F_{+-}^{\lambda,g}\left(f_{g,+}^{*,\lambda}, f_{g,-}^{*,\lambda}\right) \overset{\substack{\text{optimality} \\ \text{Remark 4.1.16}}}{\leq} F_{+-}^{\lambda,g}\left(\frac{(f_+^n, f_-^n) + \left(f_{g,+}^{*,\lambda}, f_{g,-}^{*,\lambda}\right)}{2}\right) \tag{4.44a}$$

$$\overset{(4.43)}{\lessapprox} F_{+-}^{\lambda,g}\left(f_{g,+}^{*,\lambda}, f_{g,-}^{*,\lambda}\right) \overset{\mathbb{P}}{\pm} \epsilon - \frac{P_{+-}^{\lambda,g}(u_+^n, u_-^n)}{4}. \tag{4.44b}$$

By subtracting $\left(F_{+-}^{\lambda,g}\left(f_{g,+}^{*,\lambda}, f_{g,-}^{*,\lambda}\right) - \frac{P_{+-}^{\lambda,g}(u_+^n, u_-^n)}{4}\right)$ from both sides of ineq. (4.44) and multiplying by 4 we get:

$$P_{+-}^{\lambda,g}(u_+^n, u_-^n) \overset{(4.44)}{\lessapprox} \overset{\mathbb{P}}{\pm} 4\epsilon,$$

which implies that

$$\mathbb{P}\text{-}\lim_{n\to\infty} P_{+-}^{\lambda,g}(u_+^n, u_-^n) = 0. \tag{4.45}$$

First we will show that the weak second derivative ${u_+^n}''$ converges to zero:

$$\left\| {u_+^n}'' \right\|_{L^2(K)} \leq \frac{max_{x\in\operatorname{supp}(g)} g(x)}{2\lambda g(0)} P_{+-}^{\lambda,g}\left(u_+^n, u_-^n\right) \quad \forall K \subseteq \mathbb{R}, \tag{4.46}$$

because $\left(u_+^n, u_-^n\right) \in \tilde{\mathcal{T}}$ has zero second derivative outside $\operatorname{supp}(g)$. Thus, $\mathbb{P}\text{-}\lim_{n\to\infty} \left\| {u_+^n}'' \right\|_{L^2(K)} = 0$ (by combining eqs. (4.45) and (4.46)). This can be used to apply two times the Poincaré-typed Lemma 4.1.6 (first on ${u_+^n}''$ then on ${u_+^n}'$) to get for every compact set $K \subset \mathbb{R}$:

$$\mathbb{P}\text{-}\lim_{n\to\infty} \left\| u_+^n \right\|_{W^{1,\infty}(K)} = 0, \tag{4.47}$$

as $\left(u_+^n, u_-^n\right) \in \tilde{\mathcal{T}}$ satisfies the boundary conditions at C_g^ℓ (cp. Remark 3.1.2) because of the compact support of g. Analogously, $\mathbb{P}\text{-}\lim_{n\to\infty} \left\| u_+^n \right\|_{W^{1,\infty}(K)} = 0$ for every compact set $K \subset \mathbb{R}$ and hence:

$$\mathbb{P}\text{-}\lim_{n\to\infty} \left\| u_+^n + u_-^n \right\|_{W^{1,\infty}(K)} = 0. \tag{4.48}$$

Thus, by the definition (4.39) of $\left(u_+^n, u_-^n\right)$ we get

$$\mathbb{P}\text{-}\lim_{n\to\infty} \Big\| \left(f_+^n + f_-^n\right) - \underbrace{f_{g,\pm}^{*,\lambda}}_{f_{g,+}^{*,\lambda} + f_{g,-}^{*,\lambda}} \Big\|_{W^{1,\infty}(K)} \overset{(4.39)}{=} \mathbb{P}\text{-}\lim_{n\to\infty} \left\| u_+^n + u_-^n \right\|_{W^{1,\infty}(K)} \overset{(4.48)}{=} 0,$$

which shows (4.38). $\square$

4.2 Proof of Theorem 3.2.5 ($\mathcal{RN}_{w^T,\omega} \to \mathcal{RN}_\omega^{*,\frac{1}{T}}$)

In this section we prove all the results (Lemma 3.2.3, Remark 3.2.4 and Theorem 3.2.5) presented in Section 3.2. These results are analogous to the results presented in [5, 9, 25, 12], but we will repeat the proofs briefly in this section.

Proof of Lemma 3.2.3. We need to show that for any $\omega \in \Omega$,

$$w^T(\omega) = -\exp\left(-2TX^\top(\omega)X(\omega)\right) w^\dagger(\omega) + w^\dagger(\omega), \tag{“3.8”}$$

satisfies (GD). Let $\omega \in \Omega$ be fixed and set $y := (y_1^{\text{train}}, \dots, y_N^{\text{train}})^\top$. Clearly, $w^0 = 0$. Since

$$\nabla_w L(\mathcal{RN}_w) = 2X^\top (Xw - y),$$

(GD) reads as

$$dw^t = -2(X^\top X w^t - X^\top y)\, dt. \tag{4.49}$$

Differentiating (3.8) we obtain

$$\frac{d}{dt} w^t = 2X^\top X \exp\left(-2tX^\top X\right) w^\dagger. \tag{4.50}$$

Moreover, since

$$\begin{aligned} -2(X^\top X w^t - X^\top y) &= 2X^\top X \exp\left(-2tX^\top X\right) w^\dagger - 2X^\top y w^\dagger + 2X^\top y w^\dagger \\ &= 2X^\top X \exp\left(-2tX^\top X\right) w^\dagger \end{aligned}$$

the result follows (as the solution of linear ODEs is unique, because of Picard-Lindelöf theorem). $\square$

Proof of Remark 3.2.4. Using some basic knowledge about the Moore-Penrose pseudoinverse [3] and singular value decomposition one can directly see that the minimum norm solution $w^\dagger$ does not have any singular-value-components in null-space of the matrix X. Combining this with some basic knowledge about the matrix exponential of diagonalizable matrices the result follows, since the matrix-exponential in eq. (3.8) only preserves the null-space of X—every singular-value-component outside the null-space is scaled down to zero as $T \to \infty$. $\square$

Proof of Theorem 3.2.5. First, we note that obviously

$$\lim_{T\to\infty} w^{*,\frac{1}{T}}(\omega) = w^\dagger(\omega) \quad \forall \omega \in \Omega \tag{4.51}$$

holds by Definitions 2.0.4 and 2.0.5.

Secondly, the continuity of the map $(\mathbb{R}^n, \|\cdot\|_2) \to W^{1,\infty}(K) : w \mapsto \mathcal{RN}_{w,\omega}$ implies: $\forall \omega \in \Omega$:

$$\lim_{T\to\infty} \left\| \mathcal{RN}_\omega^{*,\frac{1}{T}} - \mathcal{RN}_{w^\dagger(\omega),\omega} \right\|_{W^{1,\infty}(K)} = 0, \text{ because of eq. (4.51)} \tag{4.52a}$$

$$\lim_{T\to\infty} \left\| \mathcal{RN}_{w^T(\omega),\omega} - \mathcal{RN}_{w^\dagger(\omega),\omega} \right\|_{W^{1,\infty}(K)} = 0, \text{ because of Remark 3.2.4.} \tag{4.52b}$$

Thirdly, by applying the triangle inequality on eqs. (4.52) the result (3.9) follows. $\square$

Chapter 5

Conclusion and Future Work

Combining the main Theorems 3.1.4 and 3.2.5 tells us that that for a large number of training epochs $\tau = T/\gamma$ and a large number of neurons n, the obtained network

$$\mathcal{RN}_{\hat{w}^T,\hat{w}^0} \overset{\hat{w}^0\to 0}{\approx} \mathcal{RN}_{\hat{w}^T} \overset{\gamma\to 0}{\approx} \mathcal{RN}_{w^T} \underset{\text{Theorem 3.2.5}}{\overset{T\to\infty}{\approx}} \mathcal{RN}^{*,\frac{1}{T}} \underset{\text{Theorem 3.1.4}}{\overset{\overset{\mathbb{P}}{n\to\infty}}{\approx}} f^{*,0+}_{g,\pm} \overset{\frac{g}{g(0)}\to 1}{\approx} f^{*,0+} \tag{5.1}$$

is very close to the spline interpolation $f^{*,0+}$, where each of the $\overset{\to}{\approx}$ in eq. (5.1) corresponds to a mathematically proved exact limit in the very strong[1] Sobolev-Norm $\|\cdot\|_{W^{1,\infty}(K)}$ (in probability in the case of $\overset{\overset{\mathbb{P}}{n\to\infty}}{\approx}$). But the much more interesting statement for applications is that for *arbitrary* training time $T \in \mathbb{R}_{>0}$ (including early stopping $T << \infty$) in typical settings the following equations hold approximately:

$$\mathcal{RN}_{\hat{w}^T,\hat{w}^0} \underset{1}{\overset{\hat{w}^0\approx 0}{\approx}} \mathcal{RN}_{\hat{w}^T} \underset{2}{\overset{\gamma\approx 0}{\approx}} \mathcal{RN}_{w^T} \underset{3}{\approx} \mathcal{RN}^{*,\frac{1}{T}} \underset{4}{\overset{\overset{\mathbb{P}}{n \text{ large}}}{\approx}} f^{*,\frac{1}{Tng(0)}}_{g,\pm} \underset{5}{\overset{\substack{\text{standard distrib.}\\ \text{for } v \text{ and } b\\ \text{and } K\subseteq[-1,1]}}{\approx}} f^{*,\frac{1}{Tng(0)}}, \tag{5.2}$$

where each of the "≈" holds up to a (small) approximation error (that can be strictly larger than zero).[2] It is planned to give a better understanding of approximation (5.2) in future work:

1. The first approximation should be quite easy but is not focus of this thesis.[3] (As only the last layer of $\mathcal{RN}$ is trained, one could just start with $w^0 = 0$)

2. A small learning rate γ is more important, but not the main focus of this thesis.[4] Future work could contain a short discussion why stochastic gradient descend allows a larger total step size per epoch, which is quite intuitive (cp. footnote 12 on page 6). An interesting insight from this thesis is that for a randomized network $\mathcal{RN}$ the learning rate γ should

[1]Convergence in $\|\cdot\|_{W^{1,\infty}(K)}$ implies uniform convergence on K for example or convergence in $W^{1,p}(K)$. Even stronger Sobolev-convergenve like in $W^{2,p}$, cannot be defined, because $\mathcal{RN}_w \notin W^{2,p}(K)$

[2]By assuming $T = \frac{1}{(\lambda n g(0))} = \frac{1}{\tilde{\lambda}}$, eq. (5.2) should be read as:

$$\mathcal{RN}_{\hat{w}^{\frac{1}{\lambda ng(0)}},\hat{w}^0} \underset{1}{\overset{\hat{w}^0\approx 0}{\approx}} \mathcal{RN}_{\hat{w}^{\frac{1}{\lambda ng(0)}}} \underset{2}{\overset{\gamma\approx 0}{\approx}} \mathcal{RN}_{w^{\frac{1}{\lambda ng(0)}}} \underset{3}{\approx} \mathcal{RN}^{*,\lambda ng(0)} \underset{\text{Theorem 3.1.4}}{\overset{\overset{\mathbb{P}}{n\to\infty}}{\approx}} f^{*,\lambda}_{g,\pm} \underset{5}{\overset{\substack{\text{standard distrib.}\\ \text{for } v \text{ and } b\\ \text{and } K\subseteq[-1,1]}}{\approx}} f^{*,\lambda}$$

[3]Lemma 4.1.12 demonstrates, that with increasing n the initial weights $\hat{w}^0$ should be chosen closer to zero.

[4]For thinite values of T standard result about Euler discretization can be used. In the limit $T \to \infty$ one can formulate a direct argument that combines items 2 and 3: $\lim_{T\to\infty} \hat{w}^T = w^\dagger$, if the learning rate $\gamma < 1/r(X^\top X)$ is smaller than 1 over the spectral radius (largest eigenvalue) of $X^\top X$ [5, p. 4] [12, p. 11].

typically be chosen approximately inverse proportional to the number of neurons n. Another interesting insight that we might elaborate in more detail in future work is that the "approximation error" we get from larger values of γ has a very specific structure that allows to some extent to explain it on a macroscopic functional level.

3. Multiple papers assume that the third approximation is quite precise for arbitrary values of $T \in \mathbb{R}_{>0}$ without rigorous proof [5, 9, 25]. I have already a theory in mind that would be able to give a better understanding of the typically "rather small" but not vanishing "approximation errors", that could even have a positive effect by canceling out with the "approximation errors" from 5 to some extend. This theory could be part of close future work. 3 would be particularly interesting for real world applications by explaining *early stopping*.[5]

4. Theorem 3.1.4 is proven in this thesis' Section 4.1, but future work might show how many neurons n are actually needed to get good results.

5. The adapted regression spline $f^{*,\lambda}_{g,\pm}$ is already an easily interpretable macroscopically defined object. Intuitively it is already very plausible, that $f^{*,\lambda}_{g,\pm}$ is very close to the very desirable $f^{*,\lambda}$ on the $[-1,1]$-cube (and in its close surrounding), if one uses typical[6] distributions for v and b, if the training data is scaled and shifted to fit into the $[-1,1]$-cube. And with the same intuition one can see that if popular rules of thumb like scaling and shifting the data to the $[-1,1]$-cube are broken, one can obtain very worthless functions $f^{*,\lambda}_{g,\pm}$. This is an important contribution of Theorem 3.1.4 to answering question IV about which choices one should make to get good results with machine learning, as Theorem 3.1.4 also tells you under which conditions the algorithm would give you bad results.

The next steps in future works will probably be:

- Generalizing to multidimensional input in $\mathcal{X} = \mathbb{R}^d$. (I will publish this theorem very soon.)[7]

- With the insights won from Theorem 3.1.4, possibilities arise how to save computational time, memory and energy consumption by replacing certain groups of neurons by others algorithms (or simply by adding certain direct connections form input to the output skipping the hidden layer). This can also offer other advantages[8]. Theorem 3.1.4 and its proof inspire to choose special types of randomness for the weights and biases. It would be interesting if they provide advantages for $\mathcal{RN}$ and for other architectures.[7]

- Proofing convergence to a differently regularized function in the case of ordinary training of both layers of $\mathcal{NN}$ instead of only training the last layer (cp. footnote 16 on page 9 and the subitem about [22]).[7]

[5] Instead of $\tilde{\lambda} = \frac{1}{T}$ it would be probably better to chose $\tilde{\lambda} = \frac{se^{-2sT}}{1-e^{-2sT}}$ with an appropriate choice of s to get better approximation bounds. In this thesis we used $\tilde{\lambda} = \frac{1}{T}$, because it is suggested by the literature [5, Section 2.3 on p. 5].

[6] For example, $b_k, v_k \sim Unif(-c, c)$ i.i.d. uniformly symmetrically distributed or $b_k, v_k \sim \mathcal{N}(0, c)$ i.i.d. normally distributed with zero mean.

[7] Since we will publish these theorems very soon, it would be a waste of resources if multiple people work on it independently. If you are working on similar results, it makes more sense to collaborate—if you want to do so, please contact Hanna Wutte and me by writing to ilovemathematik-MasterThesisJakobHeiss@yahoo.com. (Other feedback, remarks and questions are very welcome as well to the same mail-adress or directly to me.)

[8] By certain modifications of the network one could also make the algorithm numerically more stable and adjust the regularization—e.g. the adapted regression spline can easily be moddified to the ordinary regression spline.

- Generalization do deep neural networks with more hidden layers (e.g. deep convolutional neural networks).[7]

Bibliography

[1] Robert A Adams and John J F Fournier. *Sobolev spaces; 2nd ed.* Pure and applied mathematics. Academic Press, New York, NY, 2003. URL http://cds.cern.ch/record/1990498. 20, 25, 31

[2] David Balduzzi, Marcus Frean, Lennox Leary, JP Lewis, Kurt Wan-Duo Ma, and Brian McWilliams. The Shattered Gradients Problem: If resnets are the answer, then what is the question? *arXiv e-prints*, art. arXiv:1702.08591, February 2017. URL https://arxiv.org/abs/1702.08591v2. 9

[3] Adi Ben-Israel and Thomas NE Greville. *Generalized inverses: theory and applications*, volume 15. Springer Science & Business Media, 2003. URL https://doi.org/10.1007%2Fb97366. 34

[4] M. Bianchini and F. Scarselli. On the complexity of neural network classifiers: A comparison between shallow and deep architectures. *IEEE Transactions on Neural Networks and Learning Systems*, 25(8):1553–1565, August 2014. ISSN 2162-237X. doi: 10.1109/TNNLS.2013.2293637. URL https://doi.org/10.1109/TNNLS.2013.2293637. 1

[5] Chris M. Bishop. Regularization and complexity control in feed-forward networks. In *Proceedings International Conference on Artificial Neural Networks ICANN'95*, volume 1, pages 141–148. EC2 et Cie, January 1995. URL https://www.microsoft.com/en-us/research/publication/regularization-and-complexity-control-in-feed-forward-networks/. 8, 16, 33, 35, 36

[6] Christopher M Bishop. *Pattern recognition and machine learning*. Information science and statistics. Springer, New York, NY, 2006. ISBN 978-0387-31073-2. URL http://cds.cern.ch/record/998831. 4

[7] Peter Craven and Grace Wahba. Smoothing noisy data with spline functions. *Numerische Mathematik*, 31(4):377–403, December 1978. ISSN 0945-3245. doi: 10.1007/BF01404567. URL https://doi.org/10.1007/BF01404567. 4, 8

[8] G. Cybenko. Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals and Systems*, 2(4):303–314, December 1989. ISSN 1435-568X. doi: 10.1007/BF02551274. URL https://doi.org/10.1007/BF02551274. 6

[9] Jerome Friedman and Bogdan E. Popescuy. Gradient directed regularization for linear regression and classification. *Tech rep*, January 2004. URL https://www.researchgate.net/publication/244258820_Gradient_Directed_Regularization_for_Linear_Regression_and_Classification. 8, 16, 33, 36

[10] Carl Friedrich Gauß. *Theoria motus corporum coelestium in sectionibus conicis solem ambientium auctore Carolo Friderico Gauss.* sumtibus Frid. Perthes et IH Besser, 1809. URL https://books.google.at/books?id=VKhu8yPcat8C. 2

[11] Carl-Friedrich Gauß. *Theoria combinationis observationum erroribus minimis obnoxiae.-Gottingae, Henricus Dieterich 1823.* Henricus Dieterich, 1823. URL https://books.google.at/books?id=hrZQAAAAcAAJ. 2

[12] Gauthier Gidel, Francis Bach, and Simon Lacoste-Julien. Implicit Regularization of Discrete Gradient Dynamics in Deep Linear Neural Networks. *arXiv e-prints*, art. arXiv:1904.13262, April 2019. URL https://arxiv.org/abs/1904.13262v1. 5, 8, 16, 33, 35

[13] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016. http://www.deeplearningbook.org. 5, 12

[14] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The elements of statistical learning: data mining, inference and prediction.* Springer, 2 edition, 2009. ISBN 978-0-387-84858-7. doi: 10.1007/978-0-387-84858-7. URL http://www-stat.stanford.edu/~tibs/ElemStatLearn/. 8

[15] Kurt Hornik. Approximation capabilities of multilayer feedforward networks. *Neural Networks*, 4(2):251 – 257, 1991. ISSN 0893-6080. doi: 10.1016/0893-6080(91)90009-T. URL https://doi.org/10.1016/0893-6080(91)90009-T. 6

[16] Yoshifusa Ito. Approximation of functions on a compact set by finite sums of a sigmoid function without scaling. *Neural Networks*, 4(6):817 – 826, 1991. ISSN 0893-6080. doi: 10.1016/0893-6080(91)90060-I. URL https://doi.org/10.1016/0893-6080(91)90060-I. 1

[17] George S. Kimeldorf and Grace Wahba. A correspondence between bayesian estimation on stochastic processes and smoothing by splines. *The Annals of Mathematical Statistics*, 41 (2):495–502, 1970. ISSN 00034851. URL http://www.jstor.org/stable/2239347. 4, 5, 8

[18] Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. *arXiv e-prints*, art. arXiv:1412.6980, December 2014. URL https://arxiv.org/abs/1412.6980. 7

[19] Masayoshi Kubo, Ryotaro Banno, Hidetaka Manabe, and Masataka Minoji. Implicit Regularization in Over-parameterized Neural Networks. *arXiv e-prints*, art. arXiv:1903.01997, March 2019. URL https://arxiv.org/abs/1903.01997. 8, 9

[20] Adrien Marie Legendre. *Nouvelles méthodes pour la détermination des orbites des comètes.* Number 1 in Analyse des triangles tracés sur la surface d'un sphéroide. F. Didot, 1805. URL https://books.google.at/books?id=7C9RAAAAYAAJ. 2

[21] Yuanzhi Li and Yingyu Liang. Learning Overparameterized Neural Networks via Stochastic Gradient Descent on Structured Data. *arXiv e-prints*, art. arXiv:1808.01204, August 2018. URL https://arxiv.org/abs/1808.01204v3. 8, 9

[22] Hartmut Maennel, Olivier Bousquet, and Sylvain Gelly. Gradient Descent Quantizes ReLU Network Features. *arXiv e-prints*, art. arXiv:1803.08367, March 2018. URL https://arxiv.org/abs/1803.08367v1. 8, 9, 36

[23] Behnam Neyshabur. Implicit Regularization in Deep Learning. *arXiv e-prints*, art. arXiv:1709.01953, September 2017. URL https://arxiv.org/abs/1709.01953v2. 8

[24] Behnam Neyshabur, Ryota Tomioka, and Nathan Srebro. In Search of the Real Inductive Bias: On the Role of Implicit Regularization in Deep Learning. *arXiv e-prints*, art. arXiv:1412.6614, December 2014. URL https://arxiv.org/abs/1412.6614v4. 8

[25] Tomaso Poggio, Qianli Liao, Brando Miranda, Andrzej Banburski, Xavier Boix, and Jack Hidary. Theory IIIb: Generalization in Deep Networks. *arXiv e-prints*, art. arXiv:1806.11379, June 2018. URL https://arxiv.org/abs/1806.11379v1. 1, 8, 16, 33, 36

[26] Christian H. Reinsch. Smoothing by spline functions. *Numerische Mathematik*, 10(3):177–183, October 1967. ISSN 0945-3245. doi: 10.1007/BF02162161. URL https://doi.org/10.1007/BF02162161. 4, 8

[27] Uri Shaham, Alexander Cloninger, and Ronald R Coifman. Provable approximation properties for deep neural networks. *Applied and Computational Harmonic Analysis*, 44(3): 537–557, 2018. URL https://doi.org/10.1016/j.acha.2016.04.003. 1

[28] Daniel Soudry, Elad Hoffer, Mor Shpigel Nacson, Suriya Gunasekar, and Nathan Srebro. The Implicit Bias of Gradient Descent on Separable Data. *arXiv e-prints*, art. arXiv:1710.10345, October 2017. URL https://arxiv.org/abs/1710.10345v4. 8

ETH

Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Declaration of originality

The signed declaration of originality is a component of every semester paper, Bachelor's thesis, Master's thesis and any other degree paper undertaken during the course of studies, including the respective electronic versions.

Lecturers may also require a declaration of originality for other written papers compiled for their courses.

I hereby confirm that I am the sole author of the written work here enclosed and that I have compiled it in my own words. Parts excepted are corrections of form and content by the supervisor.

Title of work (in block letters):

IMPLICIT REGULARIZATION FOR ARTIFICIAL NEURAL NETWORKS

Authored by (in block letters):

For papers written by groups the names of all authors are required.

Name(s):	First name(s):
HEISS	JAKOB MICHAEL

With my signature I confirm that

- I have committed none of the forms of plagiarism described in the 'Citation etiquette' information sheet.
- I have documented all methods, data and processes truthfully.
- I have not manipulated any data.
- I have mentioned all persons who were significant facilitators of the work.

I am aware that the work may be screened electronically for plagiarism.

Place, date	Signature(s)
ZÜRICH, 12.08.2019	Jakob Heiss

For papers written by groups the names of all authors are required. Their signatures collectively guarantee the entire content of the written paper.