Spurious Valleys and Clustering Behavior of Neural Networks

Samuele Pollaci [1 2]

Abstract

Neural networks constitute a class of functions that are typically non-surjective, with high-dimensional fibers and complicated image. We prove two main results concerning the geometry of the loss landscape of a neural network. First, we provide an explicit effective bound on the sizes of the hidden layers so that the loss landscape has no spurious valleys, which guarantees the success of gradient descent methods. Second, we present a novel method for analyzing whether a given neural network architecture with monomial activation function can represent a target function of interest. The core of our analysis method is the study of a specific set of error values, and its behavior depending on different training datasets.

1. Introduction

In various neural network applications, the need of analyzing high-dimensional datasets has prompted the use of increasingly deep and complex neural network architectures. Although the practical results are remarkable and promising, a complete theoretical understanding of the reason why training methods, such as the Stochastic Gradient Descent (SGD), work so well is yet to be reached. In this paper, we study the geometrical properties of the loss landscapes of neural networks to shed some light on this topic. In particular, we present two main results: the first one relates the presence of spurious valleys in the loss landscape with the architecture of the neural network (Section 3), and the second one uses the geometry of the loss landscape to define a novel criterion to determine whether the target function is representable by the given neural network (Section 4).

We view a neural network $\mathcal{N}$ as a family of functions indexed by the set of its *parameters* (*weights* and *biases*). In particular, we formalize a neural network as a function sending a tuple A of parameters to a map $\mathcal{N}(A)$, defined as an alternating composition of affine transformations with a given *activation function* (Section 2.1). Intuitively, each affine transformation expresses the behavior of a layer of neurons, and the image of a tuple A of parameters via $\mathcal{N}$ is the corresponding neural network with fixed parameters[1] A. As usual, we are going to use neural networks for approximating with a good representative η from the image of $\mathcal{N}$ an unknown real *target function* $\phi\colon \mathbb{R}^p \to \mathbb{R}^q$. The choice of η can be determined by gradient descent methods based on a given finite sample $\mathcal{T}$ of points of the graph of ϕ, called a *training dataset*, and a fixed, chosen *error function*. During this process, a path on the graph of the error function, i.e. the *loss landscape*, is defined, moving closer to points of lower error and eventually ending in η. It is patent that the geometry of the loss landscape can influence how good of an approximation η is. In particular, the gradient descent can get stuck in particular connected components of the loss landscape, called *spurious valleys* (Definition 3.2), that do not contain any global minimum of the error function, forcing the path to end in a sub-optimal point. The absence of spurious valleys implies that all the local minima of the error function are global, and thus it guarantees, under certain assumptions, the success of gradient descent methods. Our first result provides sufficient conditions for the absence of spurious valleys:

Theorem 1.1. *Let $\mathcal{N}$ be a neural network. If the error function has a global minimum, then there are explicit effective bounds on the sizes of the hidden layers so that the loss landscape of $\mathcal{N}$ has no spurious valleys.*

The bounds mentioned in Theorem 1.1 depend only on the sizes of the input and output layers, the number of hidden layers, and the activation function. Hence, Theorem 1.1 can be used to choose the sizes of the hidden layers of $\mathcal{N}$ to prevent the loss landscape from having spurious valleys. We have precise formulations of Theorem 1.1 in the spurious valleys Theorems (Theorems 3.7 and 3.10) for neural networks with and without bias. Our results generalize a theorem from (Venturi et al., 2019), denoted as Theorem 8, to wider classes of neural networks in the context of empirical risk minimization.

[1]Department of Mathematics, University of Bonn, Bonn, Germany [2]Department of Computer Science, Vrije Universiteit Brussel, Brussels, Belgium. Correspondence to: Samuele Pollaci <Samuele.Pollaci@vub.be>.

Proceedings of the 40^{th} International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023.

[1]We call $\mathcal{N}(A)$ a *neural network function*, to distinguish it from our definition of neural network which does not fix the parameters.

The second main contribution is presented in Section 4 and it concerns a novel method for analyzing whether a given neural network $\mathcal{N}$ can represent a target function ϕ of interest, i.e. if ϕ is in the image of $\mathcal{N}$. For this result, we focus just on *polynomial* neural networks, which are neural networks with an r-th power exponentiation as activation function. This environment allows us to use the powerful machinery of algebraic geometry (Hartshorne, 1977), and, other than being a satisfactory testing ground, it could constitute a good stepping stone for further results in more general settings. More in detail, our criterion is based on the study of a certain set $S_{\mathcal{T}}$ of error values depending on the training dataset $\mathcal{T}$. We provide now a brief and technical description of the construction of $S_{\mathcal{T}}$, as an anticipation of what is presented in Section 4. We denote by $\mathcal{H}_{\mathcal{T}}$ the graph of the error function restricted to a generic 2-plane. If we consider the projection $\pi_{\mathcal{T}}\colon \mathcal{H}_{\mathcal{T}} \to \mathbb{A}^1$ onto the 1-dimensional affine space representing the error, we obtain a family of plane curves, which are the fibers of the restricted error function. We denote by $S_{\mathcal{T}} \subseteq \mathbb{A}^1$ the set of error values $\mathfrak{s}$ where $\pi_{\mathcal{T}}^{-1}(\mathfrak{s})$ is a singular curve. We observe that the behavior of the points in $S_{\mathcal{T}}$ as we change the training dataset $\mathcal{T}$ contains information on the ability of the neural network to approximate the target function. This approach involving cross-sections is a mathematical elaboration of an idea from the paper (Li et al., 2018) on the visualization of loss landscapes.

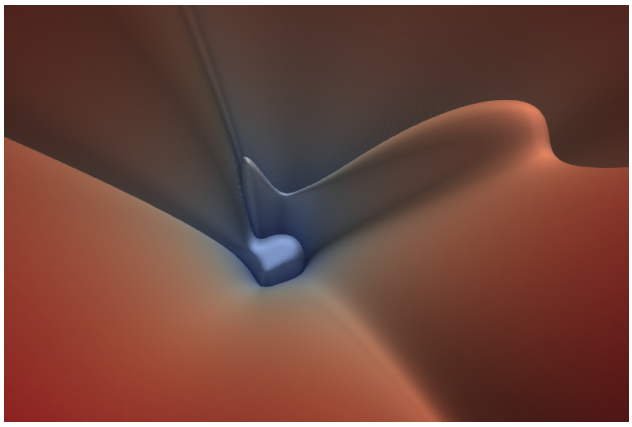

Figure 1. Visualisation of a cross-section of the loss landscape of a 1-3-3-1 NN, with activation function $\rho(x) = x^2$ and target function $\phi(x) := 5x^4 + 3x^2 - 13$, made with Paraview (Ayachit, 2015). To train the NN and output the image files we adapted the code used in (Li et al., 2018).

We show that for a one-hidden layer polynomial neural network $\mathcal{N}$, the points of $S_{\mathcal{T}}$ organize along $\mathbb{A}^1$ into 3 distinct clusters. Moreover, a particular behavior can be observed when we modify the training dataset $\mathcal{T}$. More specifically, let $\mathcal{T}_\delta := \{(i+\delta, \psi(i+\delta)) \mid (i, \psi(i)) \in \mathcal{T}\}$ be a training dataset depending on $\delta \in \mathbb{A}^1$, for a target polynomial ψ of degree d. We proved that, as δ goes to ∞, the clusters of points of $S_{\mathcal{T}_\delta}$ move towards different limit values in $\mathbb{A}^1$, depending on whether the target function f is in the image of the neural network or not.

Theorem 1.2. *Let $\mathfrak{s}_\delta \in S_{\mathcal{T}_\delta}$. Then, as δ goes to $+\infty$, one of the followings holds:*

1. *$\mathfrak{s}_\delta$ is asymptotic to $a\delta^{2\max\{r,d\}}$, for some constant $a \in \mathbb{C}^*$;*
2. *$\mathfrak{s}_\delta$ is asymptotic to $b\delta^{2(m-1)}$, for some constant $b \in \mathbb{C}^*$;*
3. *if ψ is in the image of $\mathcal{N}$ then $\mathfrak{s}_\delta = 0$, otherwise $\mathfrak{s}_\delta$ is asymptotic to $c\delta^{2(m-2)}$, for some constant $c \in \mathbb{C}^*$,*

where m depends on the coefficients of ψ.

We have a more precise statement in the clustering Theorem (Theorem 4.4). We would like to highlight that the middle cluster (2) of points of $S_{\mathcal{T}_\delta}$ tends to zero if and only if the target function ψ is in the image of $\mathcal{N}$.

1.1. Overview

The rest of the paper is structured as follows. In Section 2, we introduce neural networks and the mathematical formalism needed to deal with them. In Section 3, we generalize the result presented as Theorem 8 in (Venturi et al., 2019) to wider classes of neural networks, by proving Theorems 3.7 and 3.10. They provide a sufficient condition on the sizes of the hidden layers of a neural network so that its loss landscape has no spurious valleys. In Section 4, we study the points of $S_{\mathcal{T}} \subseteq \mathbb{A}^1$, coming from 2D cross-sections of the loss landscape. The core of the chapter is formed by Theorem 4.4, which describes the limit behavior of the points of $S_{\mathcal{T}}$ for different target functions. In Section 5 we discuss related work, and then we conclude.

Due to constraints on the number of pages, the full proofs of the main results (Theorems 3.7, 3.10, and 4.4) are contained in the Appendices A, B, and C, after the bibliography. Nevertheless, a sketch or idea of each proof is provided in the main body of the paper.

2. Neural Networks

We provide a short and precise overview of artificial neural networks (NNs), with the mathematical formalism needed for a more in-depth analysis. For further insight and information about algebraic-geometric concepts and definitions, you can refer to (Hartshorne, 1977) or (Goertz & Wedhorn, 2020).

2.1. Artificial Neural Networks

In this subsection, we introduce the nomenclature concerning NNs. As usual, we denote by $\mathbb{A}^n_{\mathbb{F}}$ the n-dimensional affine space over the field $\mathbb{F}$. Moreover, we are always going to consider $\mathbb{F}$ to be either $\mathbb{R}$ or $\mathbb{C}$.

We use NNs to approximate an unknown real *target function* $\phi\colon \mathbb{A}^p_{\mathbb{R}} \to \mathbb{A}^q_{\mathbb{R}}$, given a finite *training dataset* $\mathcal{T}$ for ϕ. In this setting, $\mathcal{T}$ is a finite subset of the graph $\mathcal{G}(\phi) \subseteq \mathbb{A}^p_{\mathbb{R}} \times \mathbb{A}^q_{\mathbb{R}}$ of the target function, and the *inputs* and *outputs* of $\mathcal{T}$ are the points in the projections of $\mathcal{T}$ onto $\mathbb{A}^p_{\mathbb{R}}$ and $\mathbb{A}^q_{\mathbb{R}}$, respectively.

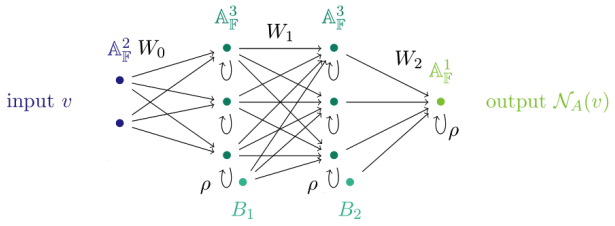

Figure 2. A NN map $\mathcal{N}_A\colon \mathbb{A}^2_{\mathbb{F}} \to \mathbb{A}^1_{\mathbb{F}}$ with $l = 2$.

A *neural network architecture* $\mathfrak{N}$ is a tuple (l, H, ρ), where $l \in \mathbb{N}$ is the *number of hidden layers*, $H = (h_0, \dots, h_{l+1}) \in \mathbb{Z}_+^{l+2}$ is the tuple of *sizes* of the layers[2], and $\rho\colon \mathbb{A}^1_{\mathbb{F}} \to \mathbb{A}^1_{\mathbb{F}}$ is the *activation function*. We denote by $\mathfrak{w} := h_0h_1 + \sum_{j=1}^{l}(h_j+1)h_{j+1}$ the *number of parameters* of the NN. For $i \in [0, l+1]$, the affine space $\mathbb{A}^{h_i}_{\mathbb{F}}$ is called the *i-layer*. As usual, the i-layers with $i \in [1, l]$ are called *hidden layers*, and the 0-layer and $l+1$-layer are called the *input* and *output layer*, respectively. By abuse of notation, we denote by ρ also the function $\mathbb{A}^{h_i}_{\mathbb{F}} \to \mathbb{A}^{h_i}_{\mathbb{F}}$ defined on the whole i-layer by $(a_1, \dots, a_{h_i}) \mapsto (\rho(a_1), \dots, \rho(a_{h_i}))$.

For a fixed architecture $\mathfrak{N} = (l, H, \rho)$, let $A := (A_0, \dots, A_l)$ be a tuple of affine transformations $A_i\colon \mathbb{A}^{h_i}_{\mathbb{F}} \to \mathbb{A}^{h_{i+1}}_{\mathbb{F}}$, where A_0 is linear. A *NN function* $\mathcal{N}_A\colon \mathbb{A}^{h_0}_{\mathbb{F}} \to \mathbb{A}^{h_{l+1}}_{\mathbb{F}}$ for the architecture $\mathfrak{N}$ and parameters A is a map defined by $x \mapsto A_l \circ \rho \circ \dots \circ \rho \circ A_0(x)$. Intuitively, this is just a neural network with fixed parameters. In fact, we can write each affine transformation A_i as the composition of a translation and a linear map using matrices $B_i \in \mathcal{M}_{h_{i+1}\times 1}(\mathbb{F})$ and $W_i \in \mathcal{M}_{h_{i+1}\times h_i}(\mathbb{F})$, i.e. $A_i(x) = W_ix + B_i$. The entries of W_i and B_i are the *weights* and the *biases* of $\mathcal{N}_A$, respectively.

The map $\mathcal{N}\colon \mathbb{A}^{\mathfrak{w}}_{\mathbb{F}} \to \mathrm{Hom}(\mathbb{A}^{h_0}_{\mathbb{F}}, \mathbb{A}^{h_{l+1}}_{\mathbb{F}})$ sending parameters $A \in \mathbb{A}^{\mathfrak{w}}_{\mathbb{F}}$ to $\mathcal{N}_A$ is the *artificial neural network (induced by the architecture $\mathfrak{N}$)*. The *artificial neural network without bias (induced by the architecture $\mathfrak{N}$)* is the map $\mathcal{N}$ restricted to the space where all biases are zero.

A function ψ is *representable by $\mathcal{N}$* if it is in the image of $\mathcal{N}$. Moreover, ψ is *representable* by an architecture $\mathfrak{N}$ if it is representable by the NN induced by $\mathfrak{N}$. A set $\mathcal{T}$ is *representable* by $\mathcal{N}$ if there exists a map ϕ representable by $\mathcal{N}$ such that $\mathcal{T}$ is a training dataset for ϕ.

2.2. Training

Let $\mathcal{N}$ be a NN, $\mathcal{T}$ be a training dataset, and $L\colon \mathbb{A}^{h_{l+1}}_{\mathbb{F}} \times \mathbb{A}^{h_{l+1}}_{\mathbb{F}} \to \mathbb{R}$ be a semimetric on the output layer. During the training process, we look for the minimizers of the *error function* $err_{\mathcal{T}}\colon \mathbb{A}^{\mathfrak{w}}_{\mathbb{F}} \to \mathbb{R}$ defined by

$$err_{\mathcal{T}}(A) := \frac{1}{|\mathcal{T}|} \sum_{(i,o)\in\mathcal{T}} L\left(o, \mathcal{N}_A(i)\right).$$

The map L and the graph of $err_{\mathcal{T}}$ are called the *loss function* and the *loss landscape of $\mathcal{N}$*, respectively.

Notice that the error function might have no global minimum. In any case, we can end the training process when we find parameters $A \in \mathbb{A}^{\mathfrak{w}}_{\mathbb{F}}$ such that the error is sufficiently small, i.e. $err_{\mathcal{T}}(A) < \epsilon$, for some chosen constant $\epsilon \in \mathbb{R}^+$.

2.3. Algebraic Setup

In Section 4, we need a fully-algebraic setup. Here we introduce the choices that allow us to have such algebraic environment. First, we choose the activation function to be a monomial.

Definition 2.1. An architecture $\mathfrak{N} := (l, H, \rho)$ is *polynomial* if $\rho = (\cdot)^r$ for some $r \in \mathbb{Z}_+$. In such case, the NN (without bias) induced by $\mathfrak{N}$ is called the h_0-...-h_{l+1} *r-polynomial NN (without bias)*.

Given a polynomial NN $\mathcal{N}$, a NN function is a tuple of polynomials with monomials of a specific degree, as it expresses the following proposition.

Proposition 2.2. *Let $\mathcal{N}\colon \mathbb{A}^{\mathfrak{w}}_{\mathbb{F}} \to Hom(\mathbb{A}^p_{\mathbb{F}}, \mathbb{A}^q_{\mathbb{F}})$ be an r-NN, with $l \geq 1$ hidden layers. Then, the q components of a NN function are polynomials in p variables with monomials of degree multiple of r, up to degree r^l.*

Proof. Clear, as a NN is a composition of affine transformations and degree r exponentiations. □

By Proposition 2.2, for a polynomial NN $\mathcal{N}$, it is possible to find the set $\mathcal{D}_{\mathcal{N}}$ of algebraic functions which includes the image of $\mathcal{N}$. In particular, we have

$$\mathrm{Im}(\mathcal{N}) \subseteq \mathcal{D}_{\mathcal{N}} := \left\{ \left(\sum_{i=0}^{r^{l-1}} P_{1,i}, \dots, \sum_{i=0}^{r^{l-1}} P_{q,i} \right) \middle| \, P_{j,i} \in \mathbb{F}[X_p]_{ir} \right\},$$

[2] Whenever we use the term "layers" with no further specification, we mean input layer, hidden layers, and output layer.

where $\mathbb{F}[X_p]_{ir}$ is the group of polynomials in p variables, with coefficients in $\mathbb{F}$, and of degree ir. If a NN $\mathcal{N}\colon \mathbb{A}_{\mathbb{F}}^{\mathfrak{w}} \to \mathcal{D}_{\mathcal{N}}$ is surjective, we say $\mathcal{N}$ is *filling* (Kileel et al., 2019).

Finally, since our goal is to approximate a *real* target function ϕ, we choose the loss function to be the real squared Euclidean distance $L(y, \hat{y}) := \sum_{i=1}^{h_{l+1}} (y_i - \hat{y}_i)^2$. In this way, $err_{\mathcal{T}}\colon \mathbb{A}_{\mathbb{R}}^{\mathfrak{w}} \to \mathbb{R}$ is an algebraic function, and the loss landscape $\mathcal{L}$ is an algebraic hypersurface in the affine space $\mathbb{A}_{\mathbb{R}}^{\mathfrak{w}+1}$.

Remark 2.3. When studying the geometry of $\mathcal{L}$ in Section 4, we consider the base change $\mathcal{L} \times_{\mathbb{R}} \mathbb{C}$, which corresponds to linearly extending the error function on $\mathbb{C}$. Even though the loss function is not a semimetric on $\mathbb{C}$, the linear extension allows us to get valuable geometric insights.

3. Spurious Valleys

The presence of spurious valleys (Definition 3.2) in the loss landscape may cause gradient-descent methods to fail. In this section, we provide a sufficient condition (Theorems 3.7 and 3.10) on the number of parameters $\mathfrak{w}$ for having no spurious valleys in the loss landscape. The Theorems 3.7 and 3.10 generalize a result from (Venturi et al., 2019), denoted as Theorem 8, to wider classes of neural networks in the context of empirical risk minimization. We use the definition of spurious valley given in (Venturi et al., 2019).

Definition 3.1. For all $c \in \mathbb{R}$, a *sub-level set* of a function $f\colon X \to \mathbb{R}$ is $\Omega_f(c) := \{x \in X \mid f(x) \leq c\}$.

Definition 3.2. We define a *spurious valley* of the loss landscape as a path-connected component of a sub-level set $\Omega_{err_{\mathcal{T}}}(c)$ which does not contain any element of argmin $\text{err}_{\mathcal{T}}$.

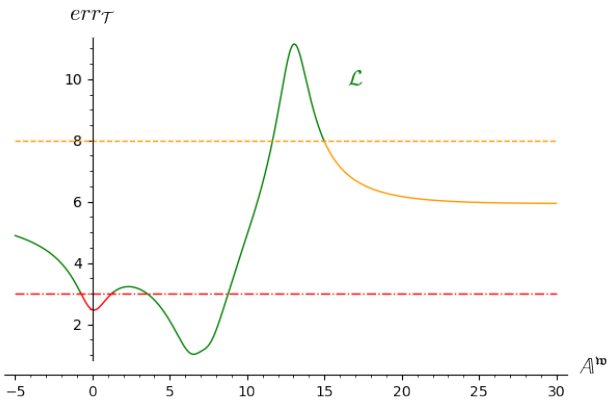

Figure 3. Simplified 2D visualization of a loss landscape $\mathcal{L}$. Two spurious valleys are clearly visible: the red one for the sub-level set $\Omega_{err_{\mathcal{T}}}(3)$, and the orange one for $\Omega_{err_{\mathcal{T}}}(8)$ (assuming the right branch is asymptotic to $err_{\mathcal{T}} = 6$).

We will use the following setup for a NN.

Setup 3.3. Let $\mathfrak{N} := (l, H, \rho)$ be an architecture, with continuous ρ, and $\mathcal{T}$ be a training dataset. We consider the NN $\mathcal{N}\colon \Theta := \mathbb{A}_{\mathbb{R}}^{\mathfrak{w}} \to \mathcal{C}$ (with or without bias) induced by $\mathfrak{N}$. We take as loss function a convex semimetric L on real vectors. We denote by $\mathcal{L} \subseteq \mathbb{A}_{\mathbb{R}}^{\mathfrak{w}+1}$ the loss landscape of $\mathcal{N}$ for $\mathcal{T}$ with base space $\mathbb{R}$.

Throughout this section, we always assume the error function has a global minimum, i.e. $\arg\min err_{\mathcal{T}} \neq \emptyset$. For the sake of simplicity, we start by considering the no-bias case. Obtaining the analogous results for NNs with bias will require some careful adjustments.

3.1. No-bias Case

We start by redefining the set of functions representable by $\mathfrak{N}$ in a more convenient way. For any $m \in \mathbb{Z}_+$ and any vector $w \in \mathbb{R}^m$, we define the functions $\psi_w := \psi_{m,\rho,w}\colon \mathbb{R}^m \to \mathbb{R}$ by sending x to $\rho\langle w, x\rangle$, i.e. the activation function applied to the scalar product between w and x. Moreover, for any $k \in \mathbb{Z}_+$ and $K \in \mathbb{Z}_+^k$, let $\mathcal{C}_{\rho,K}^k$ be the set of functions representable by a NN without bias induced by the architecture $\mathfrak{N}_K := (k, (p, K, 1), \rho)$. We define $\mathcal{C}_\rho^k := \bigcup_{K \in \mathbb{Z}_+^k} \mathcal{C}_{\rho,K}^k$ as the set of all the functions representable by a NN without bias induced by an architecture $\mathfrak{N}_K$ with arbitrary tuple K. By the structure of a NN and our definition of ψ_w, the $\mathcal{C}_\rho^k$'s are vector spaces of the following form:

$$\begin{aligned} \mathcal{C}_\rho^1 &= \text{span}_{\mathbb{R}}\{\psi_w \mid w \in \mathbb{R}^p\} \\ \mathcal{C}_\rho^k &= \text{span}_{\mathbb{R}}\{\rho f \mid f \in \mathcal{C}_\rho^{k-1}\}, k > 1. \end{aligned} \tag{1}$$

Definition 3.4. Let ρ be a continuous activation function, and k a positive integer. The *k-intrinsic dimension* dim_ρ^k is the dimension $\dim_{\mathbb{R}}(\mathcal{C}_\rho^k)$ of the vector space $\mathcal{C}_\rho^k$.

Example 3.5 (Polynomial NN). Let $\mathcal{N}$ be a 2-h_1-h_2-3 2-polynomial NN without bias. Notice that in this case $\rho = (\cdot)^2$ is the squaring operation. Then, for any $w := (w_1, w_2) \in \mathbb{R}^2$ we have $\psi_w(x, y) = w_1^2 x^2 + w_2^2 y^2 + 2w_1 w_2 xy$. Hence,

$$\begin{aligned} \mathcal{C}_\rho^1 &= \text{span}_{\mathbb{R}}\{w_1^2 x^2 + w_2^2 y^2 + 2w_1 w_2 xy \mid (w_1, w_2) \in \mathbb{R}^2\} \\ &= \{ax^2 + by^2 + cxy \mid a, b, c \in \mathbb{R}\}, \\ \mathcal{C}_\rho^2 &= \text{span}_{\mathbb{R}}\{\rho f \mid f \in \mathcal{C}_\rho^1\} \\ &= \{ax^4 + bx^3 y + cx^2 y^2 + dxy^3 + ey^4 \mid a, b, c, d, e \in \mathbb{R}\}. \end{aligned}$$

Moreover, we get $dim_\rho^1 = 3$ and $dim_\rho^2 = 5$.

By the definition of the vector space $\mathcal{C}_\rho^k$ in (1), and Definition 3.4, the intrinsic dimensions dim_ρ^k depend solely on the size of the input layer and the activation function ρ.

Example 3.6. Let $\mathcal{N}$ be a NN with h_0-dimensional input layer, and activation function $\rho(x) := \sum_{n=0}^{N} a_n x^n$ a real

polynomial of degree $N \in \mathbb{N}$. The 1-intrinsic dimension is $dim_\rho^1 = \sum_{n=0}^{N} \binom{h_0+n-1}{n}$.

We have now all the elements to state the spurious valleys theorem for NNs without bias.

Theorem 3.7 (Spurious valleys theorem: no-bias case)**.** *Consider Setup 3.3 for a NN without bias, and suppose $dim_\rho^k < \infty$ for all $k \in [1, l]$. If $h_k \geq dim_\rho^k$ for all $k \in [1, l]$, then the loss landscape $\mathcal{L}$ has no spurious valleys.*

It is interesting to notice that Theorem 3.7 does not depend on the size of the training dataset. On the contrary, it relies solely on the expressiveness gained from the overparameterization of the hidden layers: the proof (Appendix A) uses the fact that the sizes of the hidden layers are greater than the dimensions of the corresponding functional spaces to carefully change the matrices of weights and build a non-increasing path from any point of the loss landscape to the global minimum. These considerations clearly hold also for the bias version (Theorem 3.10), presented in Section 3.3,

Example 3.8 (Polynomial NN (continues)). By applying Theorem 3.7 to the polynomial NN $\mathcal{N}$ from Example 3.5, we obtain that, if $h_1 \geq 3$ and $h_2 \geq 5$, then the loss landscape of $\mathcal{N}$ has no spurious valleys.

In the next subsection, we provide a sketch of the proof of Theorem 3.7. The complete proof can be found in Appendix A.

3.2. Proof of Theorem 3.7 (sketch)

The technique of the proof is analogous to the one used in (Venturi et al., 2019). As pointed out in (Freeman & Bruna, 2017) and (Venturi et al., 2019), the following property implies that the loss landscape $\mathcal{L}$ does not have any spurious valley:

Property 3.9. *Given any initial $\mathfrak{w}$-tuple of parameters $\tilde{A} \in \Theta$, there exists a continuous path $\theta\colon t \in [0,1] \mapsto \theta_t \in \Theta$ such that $\theta_0 = \tilde{A}$, $\theta_1 \in argmin_{\theta\in\Theta} err_{\mathcal{T}}(\theta)$, and the function $t \mapsto err_{\mathcal{T}}(\theta_t)$ is non-increasing.*

Hence, we reduce to proving that for any $A \in \Theta$ there exists a continuous path satisfying the conditions in Property (3.9). Let $\tilde{A} := (\tilde{A}_0, \dots, \tilde{A}_l) \in \Theta$ be our starting set of parameters, where l is the number of hidden layers of the NN as usual. In order to build the desired path θ, we construct $l+1$ intermediate continuous paths $P_1, \dots, P_{l+1}\colon [0,1] \to \Theta$, and then we concatenate them.

Each path P_i with $i \in [1, l]$ modifies just the matrices $\tilde{A}_{i-1}$ and $\tilde{A}_i$ of the tuple of parameters $\tilde{A}$, in such a way that the image $P_i([0,1])$ of the path is entirely contained in the fiber $\mathcal{N}^{-1}(\mathcal{N}_{\tilde{A}}) \subseteq \Theta$. In particular, since we are moving in the fiber above $\mathcal{N}_{\tilde{A}}$, the NN function $\mathcal{N}_{\tilde{A}}$ remains the same along the path and the error remains constant. Hence, the requirement for the path of being non-increasing is satisfied.

The final goal of the first l paths is to be able to move from the starting tuple of parameters $\tilde{A}$ to another point A of the space Θ with some special properties. Such properties together with the convexity of the loss function guarantee the existence of a continuous path P_{l+1} with endpoint a global minimum of the error function, and such that $t \mapsto err_{\mathcal{T}}(P_{l+1}(t))$ is non-increasing, as desired. For more details, we refer to Appendix A.

3.3. Bias Case

To adapt Theorem 3.7 to NNs with bias, we use Setup 3.3 and the notation adopted in Subsection 3.1 with some slight changes:

- $\mathcal{N}$ has bias. As a consequence, the parameter space Θ has a higher dimension.
- The matrices $A_0, \dots, A_l$ of a set of parameters $A \in \Theta$ contain also the bias terms of the respective layers, i.e. they have now one additional column on the right.
- The sets of representable functions are now

$$\mathcal{C}_\rho^1 = \text{span}_{\mathbb{R}}\{\{\psi_w \mid w \in \mathbb{R}^p\} \cup \{1\}\}$$
$$\mathcal{C}_\rho^k = \text{span}_{\mathbb{R}}\{\{\rho f \mid f \in \mathcal{C}_\rho^{k-1}\} \cup \{1\}\}, k > 1.$$

Theorem 3.10 (Spurious valleys theorem: bias case)**.** *Consider Setup 3.3 for a NN with bias, and suppose dim_ρ^k is finite for any $k \in [1, l]$. If $h_k \geq dim_\rho^k - 1$ for all $k \in [1, l]$, then the loss landscape $\mathcal{L}$ has no spurious valleys.*

The proof of Theorem 3.10 is analogous to the proof of Theorem 3.7 and it is written in detail in Appendix B.

Remark 3.11. Notice that, by Proposition 2.2 it is clear that, for a polynomial NN with or without bias, $\mathcal{C}_\rho^k$ is a finitely generated $\mathbb{R}$-vector space, and $dim_\rho^k < \infty$. Moreover, since the squared Euclidean distance is a convex loss function and the monomial exponentiation $(\cdot)^r$ is continuous, we can apply the (no-bias) spurious valleys theorem to any polynomial NN (without bias) with sufficiently wide layers.

4. Clustering

In this section, we present Theorem 4.4 which describes a behavior of polynomial NNs that seems to be a good indicator of the representability of a function. Since we consider just polynomial NNs in this section, we omit the term "polynomial" when referring to NNs and architectures.

4.1. The Idea

As the number of parameters $\mathfrak{w}$ is typically huge, it is often useful to consider cross-sections when studying loss landscapes. In particular, we compose the error function

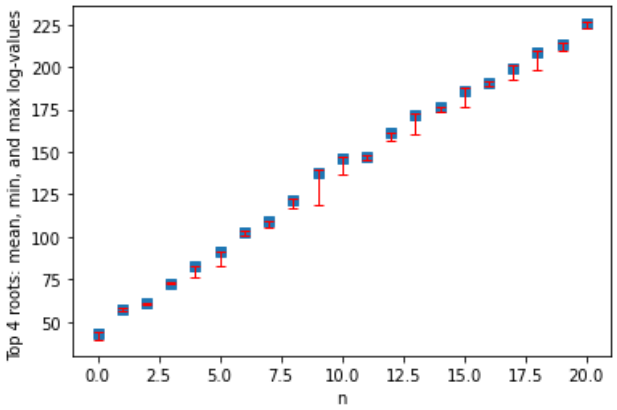

Figure 4. Average, maximum and minimum error values (in logarithmic scale) of the points in the cluster c_4 depending on the input interval $[10^{n+2} - 100, 10^{n+2}]$. The data were collected by conducting 21 trials on SageMath (Stein et al., 2020), using the NN architecture described in Example 4.1, with $N = 10$, $\Delta = 100$, and $\delta(n) := 10^{n+2}$, for $n \in [0, 20] \subseteq \mathbb{Z}$.

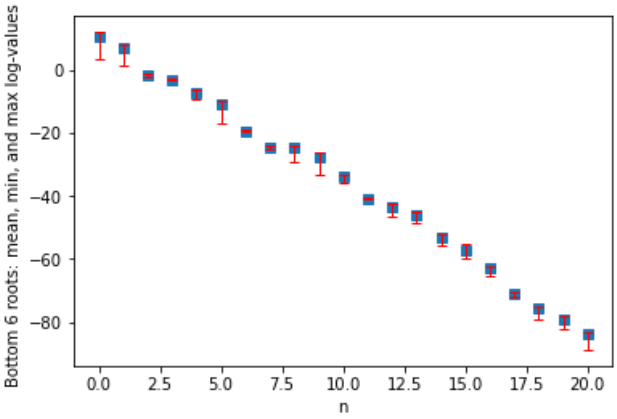

Figure 5. Average, maximum and minimum error values (in logarithmic scale) of the points in the cluster c_6 depending on the input interval $[10^{n+2} - 100, 10^{n+2}]$. The data were collected by conducting 21 trials on SageMath (Stein et al., 2020), using the NN architecture described in Example 4.1, with $N = 10$, $\Delta = 100$, and $\delta(n) := 10^{n+2}$, for $n \in [0, 20] \subseteq \mathbb{Z}$.

$err_{\mathcal{T}}$ with a 2-plane $p\colon \mathbb{A}^2_{\mathbb{F}} \to \mathbb{A}^{w}_{\mathbb{F}}$, so that the graph of the restricted error function $f := err_{\mathcal{T}} \circ p$ is an algebraic hypersurface $\mathcal{H}_{\mathcal{T}} := \mathcal{Z}(f - z) \subseteq \mathbb{A}^3_{\mathbb{F}}$. In other words, we reduce to study a 2-dimensional cross-section $\mathcal{H}_{\mathcal{T}}$ of the loss landscape in a 3-dimensional space. We call $\mathcal{H}_{\mathcal{T}}$ the *cut loss landscape*, and we consider it over $\mathbb{C}$. If we look at the projection $\pi_{\mathcal{T}}\colon \mathcal{H}_{\mathcal{T}} \to \mathbb{A}^1_{\mathbb{C}}$ onto the error coordinate, we obtain a family of plane curves, which are the fibers of f. We denote by $S_{\mathcal{T}} \subseteq \mathbb{A}^1_{\mathbb{C}}$ the set of points $\mathfrak{s}$ where $\pi_{\mathcal{T}}^{-1}(\mathfrak{s})$ is a singular curve. Notice that here we consider the error to be a complex number, not just a non-negative real number.

If we modify the input points of $\mathcal{T}$ by translating them towards infinity, and we change the output points accordingly, we observe that the points of $S_{\mathcal{T}}$ organize into clusters. Moreover, the mean values of these clusters move towards different limit values, depending on whether the target function is representable or not.

4.2. Examples

We study in detail the clustering behavior of a 1-h-1 neural network $\mathcal{N}$ when considering representable and non-representable training datasets.

It is easy to see that $\mathcal{N}$ is filling and a NN function is of the form $ax^r + b$, for $a, b \in \mathbb{F}$. Hence, it is really easy to verify whether a training dataset (or even a target function) is representable. Nevertheless, the results in this section, other than proving an interesting fact about representability, provide valuable insights into the limit behavior of a 1-hidden layer NN that might be used to study analogous behaviors for NNs with more hidden layers. We start our analysis with two examples.

Example 4.1 (Representable case). Let $\mathcal{N}$ be a 1-3-1 2-NN, and $\psi(x) = 3x^2 - 13$ be the target function. The map ψ is representable because $\mathcal{N}$ is filling and $\psi \in \mathcal{D}_{\mathcal{N}}$ (see Proposition 2.2). Let $\{\mathcal{T}_\delta\}$ be a family of training datasets for ψ, such that, for any $\delta \in \mathbb{R}$, the input points of $\mathcal{T}_\delta$ are taken uniformly at random from the real interval $[\delta - \Delta, \delta]$, for some fixed $\Delta \in \mathbb{R}_+$. It is easy to see that, for a generic cutting plane p, the minimum value of the function $f := err_{\mathcal{T}_\delta} \circ p$ is 0, for all $\delta \in \mathbb{R}$.

By using SageMath (Stein et al., 2020), we compute the polynomial defining the cut loss landscape $\mathcal{H}_{\mathcal{T}_\delta}$. Then we can use Groebner basis and elimination on Magma (Bosma et al., 1997) to find the polynomial e which defines the projection of $\mathcal{Z}\left(\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}, f - z\right)$ onto the error axis. The roots of e are the points of $S_{\mathcal{T}_\delta}$. When plotting the points of $S_{\mathcal{T}_\delta}$ (as in Figures 4 and 5), we apply an absolute value to get real positive numbers.

After multiple trials with different values of δ and Δ, some interesting empirical observations can be formulated:

- We obtain 11 different error values for each trial, divided into 3 clusters. The first cluster has just the point 0, with multiplicity 3. This is of course expected since ψ is representable. The second and third clusters, denoted by c_6 and c_4, contain 6 and 4 points respectively.
- The difference between the highest and lowest error value in each cluster tends to get smaller by reducing Δ.

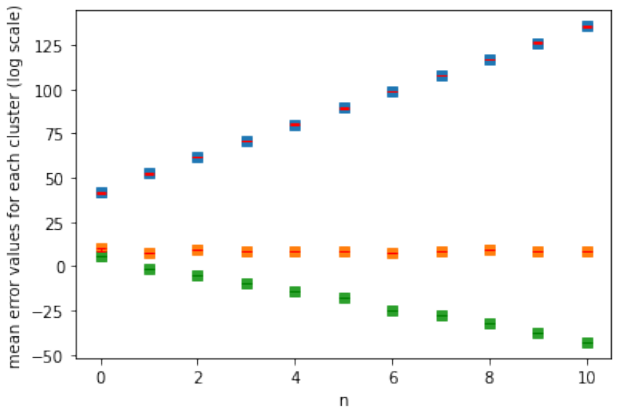

Figure 6. Average error values (in logarithmic scale) of the points in the 3 clusters of $S_{\mathcal{T}_\delta}$ depending on the input interval $[10^{n+2} - 100, 10^{n+2}]$. We used $\psi(x) = -x^2 + x + 5$ as target function, $\rho(x) = x^2$ as activation function.

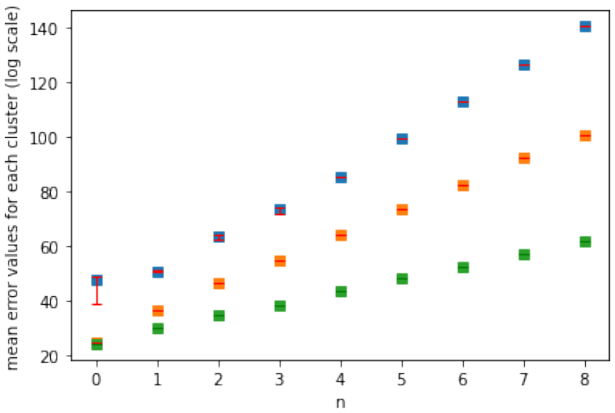

Figure 7. Average error values (in logarithmic scale) of the points in the 3 clusters of $S_{\mathcal{T}_\delta}$ depending on the input interval $[10^{n+2} - 100, 10^{n+2}]$. We used $\psi(x) = x^3 + 2x^2 - 3$ as target function, $\rho(x) = x^2$ as activation function.

- Modifying δ deeply influences the three aforementioned clusters. In particular, the values in c_6 and c_4 tend to 0 and ∞ respectively as δ goes to $\pm\infty$, whereas the 0 value stays the same. In Figures 4 and 5 we collected some data which clearly show the limit behavior of the clusters c_4 and c_6, respectively.

Example 4.2 (Non-representable case). We consider the same setting of Example 4.1, but ψ is a polynomial target function not representable by $\mathcal{N}$. We still observe 3 clusters of points in $S_{\mathcal{T}_\delta}$, but their limit behavior is different. We again use the absolute value to plot the points of $S_{\mathcal{T}_\delta}$. As can be noticed from the graphs in Figures 6, 7, and 8, there is always one cluster whose points tend towards $+\infty$, as in the representable case. Moreover, the bottom cluster is not constant at 0, and the middle cluster does not tend to 0, which was the behavior observed in Example 4.1. The data in Figures 6, 7, and 8 were collected and elaborated using SageMath (Stein et al., 2020) and Magma (Bosma et al., 1997).

4.3. The Clustering Theorem

Now that we have a better understanding of the clustering behavior, we can introduce Theorem 4.4, which describes the limit behaviour of the error values of the points in the set $S_{\mathcal{T}}$ introduced in Section 4.1.

Definition 4.3. Let $\mathcal{T} \subseteq \mathbb{R}^p \times \mathbb{R}^q$ be a training dataset for a target function $\psi\colon \mathbb{R}^p \to \mathbb{R}^q$, and $\delta \in \mathbb{R}^p$. The *δ-translated training dataset* $\mathcal{T}_\delta$ for ψ is the training dataset $\{(i+\delta, \psi(i+\delta)) \mid (i, \psi(i)) \in \mathcal{T}\}$. The *family of translated training datasets* $\{\mathcal{T}_\delta\}$ for ψ is the set of training datasets $\{\mathcal{T}_\delta\colon \delta \in \mathbb{R}^p\}$.

We will use the following notation to compare growing rates: if ϕ and ψ are two functions such that $\lim_{x\to+\infty} \frac{\phi(x)}{\psi(x)} = 1$, then we say ϕ and ψ *have the same growth rate as* $x \to +\infty$, or *are asymptotic at* $+\infty$, and we write $\phi \sim_{+\infty} \psi$.

Theorem 4.4 (Clustering theorem)**.** *Let $r, h \in \mathbb{Z}$, $r > 1$, $h > 2$. Let $\mathcal{N}$ be a 1-h-1 r-NN, and $\psi(x) := \sum_{j=0}^{d} \alpha_j x^j$ be a real polynomial target function. Let $\{\mathcal{T}_\delta\}$ be a family of translated training datasets for ψ with at least 3 distinct points each. For a generic 2-plane, let $\mathfrak{s}_\delta \in S_{\mathcal{T}_\delta}$. Then one of the following holds:*

1. *$\mathfrak{s}_\delta \sim_{+\infty} a\delta^{2\max\{r,d\}}$, for some non-zero constant $a \in \mathbb{C}^*$;*
2. *$\mathfrak{s}_\delta \sim_{+\infty} b\delta^{2(m-1)}$, for some non-zero constant $b \in \mathbb{C}^*$;*
3. *if ψ is in the image of $\mathcal{N}$ then $\mathfrak{s}_\delta = 0$, otherwise $\mathfrak{s}_\delta \sim_{+\infty} c\delta^{2(m-2)}$, for some non-zero constant $c \in \mathbb{C}^*$,*

where $m := max(\{j \mid \alpha_j \neq 0 \text{ and } j \neq r\} \cup \{0\})$.

Remark 4.5. Notice that if $r \neq d$ then $m = d$, else $m < d$. Hence $m \leq \max\{r, d\}$.

Proposition 4.6. *Let $r, h \in \mathbb{Z}$, $r > 1$, $h > 2$. Let $\mathcal{N}$ be a 1-h-1 r-NN, and $\psi(x) := \sum_{j=0}^{d} \alpha_j x^j$ be a real polynomial target function. Then ψ is representable by $\mathcal{N}$ if and only if $m = 0$.*

Proof. It is easy to see that $\psi\colon \mathbb{R} \to \mathbb{R}$ is representable by $\mathcal{N}$ if and only if ψ is of the form $\psi(x) = \alpha_r x^r + \alpha_0$, with $\alpha_r, \alpha_0 \in \mathbb{R}$. We conclude by the definition of m (see Theorem 4.4). □

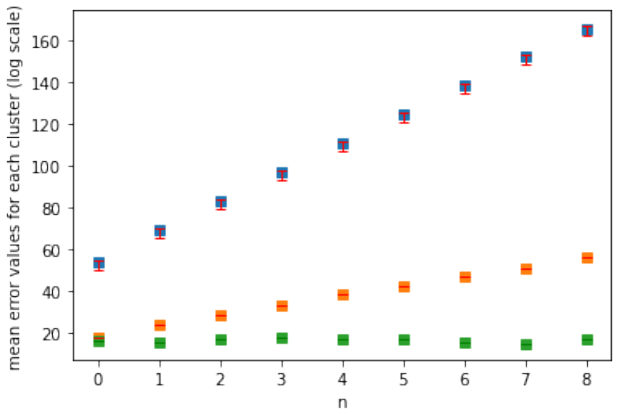

Figure 8. Average error values (in logarithmic scale) of the points in the 3 clusters of $S_{\mathcal{T}_\delta}$ depending on the input interval $[10^{n+2} - 100, 10^{n+2}]$. We used $\psi(x) = x^3 + 2x^2 - 3$ as target function, $\rho(x) = x^3$ as activation function.

The points of $S_{\mathcal{T}_\delta}$ satisfying 2. in Theorem 4.4 are the points composing the *middle cluster*. This middle cluster can be used to tell whether the target function is representable. In fact, by Theorem 4.4 and Proposition 4.6, the target function is representable if and only if the points of the middle cluster tend to 0. The points satisfying 3. might be misleading when using numerical methods to compute them, as they may result close to 0 for both representable and non-representable training datasets.

We conclude this section with the idea of the proof of Theorem 4.4. The full proof is contained in Appendix C.

4.4. Proof of Theorem 4.4 (sketch)

The proof of Theorem 4.4 uses some notions from algebraic geometry. We refer to (Hartshorne, 1977) or (Goertz & Wedhorn, 2020) for the basic definitions and results.

Let f be the error function restricted to a generic 2-plane, and $\mathcal{H}_{\mathcal{T}_\delta} = \mathcal{Z}(f(x,y) - z) \subseteq \mathbb{A}^3_{\mathbb{C}}$ be the cut loss landscape. Then, we can write $S_{\mathcal{T}_\delta} = \{f(x,y) \mid (x,y) \in Z_\delta\} \subseteq \mathbb{A}^1_{\mathbb{C}}$, where $Z_\delta \subseteq \mathbb{A}^2_{\mathbb{C}}$ is the zero locus defined by the partial derivatives of f. The idea of the proof is to study the limit behavior of the points in $S_{\mathcal{T}_\delta}$ through the analysis of the points of Z_δ.

Since the partial derivatives of f have a complicated expression, we study instead a larger set of points, namely the points of $Z'_\delta \supseteq Z_\delta$, where $Z'_\delta := \mathcal{Z}(a(x,y)b(x,y), a(x,y)c(x,y)) \subseteq \mathbb{A}^2_{\mathbb{C}}$, and the polynomials a, b, and c depend on the partial derivatives of f. To study the limit behavior, we would like to express the coordinate x and y as functions of δ.

Let $\overline{Z} \subseteq \mathbb{P}^2_{\mathbb{C}} \times \mathbb{P}^1_{\mathbb{C}}$ be the projective closure of Z_δ w.r.t. the variables (x,y) and δ separately. Morevoer, let $\pi_2 \colon \overline{Z} \subseteq \mathbb{P}^2_{\mathbb{C}} \times \mathbb{P}^1_{\mathbb{C}} \to \mathbb{P}^1_{\mathbb{C}}$ be the second projection. Since we are only interested in positive real values of δ, we can consider an analytic local section $\sigma' \colon D \to \overline{Z}$ of π_2, where $D := (M, +\infty) \subseteq \mathbb{R}_{>0}$. Let $\sigma := \pi_1 \circ \sigma' \colon D \to \mathbb{P}^2_{\mathbb{C}}$. It is possible to show that we may reduce to $\sigma \colon D \subseteq \mathbb{A}^1_{\mathbb{C}} \to \mathbb{A}^2_{\mathbb{C}}$, which expresses the coordinates of a point in Z_δ as a function of δ, as desired. Hence, we can study the limit behaviour of $f(\sigma)$ as δ goes to $+\infty$.

To do so, we prove several lemmata. In the majority of them, we use the genericity assumption (Griffiths & Harris, 1994) for the 2-plane in Theorem 4.4.

5. Related Work

The geometry and topology of the loss landscapes have been studied extensively for shallow neural networks (Venturi et al., 2019; Chizat & Bach, 2018; Mei et al., 2018), and deep linear networks (Montúfar et al., 2014; Arora et al., 2018; 2019; Hardt & Ma, 2017). In the last years, more attention has been drawn to the geometry of deep non-linear neural networks (Freeman & Bruna, 2017; Larsen et al., 2022). in particular, (Pittorino et al., 2022) and (Simsek et al., 2021) focus on the simmetries encountered in the loss landscapes. Our Theorems 3.7 and 3.10 on spurious valleys extend to deep non-linear NNs a result from (Venturi et al., 2019), in the context of empirical risk minimization. Other works with a focus on level sets include (Draxler et al., 2018) and (Freeman & Bruna, 2017).

As far as polynomial NNs are concerned, the literature is quite substantial. This is especially true for linear activation functions, as cited above, and quadratic activation functions, like in (Soltanolkotabi et al., 2019; Kawaguchi, 2016; Du & Lee, 2018). Moreover, the study of NNs through the lenses of algebraic geometry has sparked some interest, as can be seen in recent papers (Yang, 2021; Mehta et al., 2022; Trager et al., 2020). In particular, Kileel (Kileel et al., 2019) studied the dimension of the space of functions representable by a polynomial NN without bias.

6. Conclusions and Future Work

In this paper, we have presented and proved two novel main results. The first one (Theorems 3.7, and 3.10) provides a sufficient condition on the sizes of the hidden layers of a neural network to have no spurious valleys in the loss landscape. This guarantees, under certain assumptions, the success of gradient descent methods.

The main limitation both in Theorem 3.7 and in Theorem 3.10 is the assumption on the finiteness of the intrinsic dimensions. This requirement is fundamental for the tech-

nique used in the proofs, as allows to carefully modify the matrices of weights and build the non-increasing function required by Property 3.9. Unfortunately, commonly used activation functions like the ReLU or the sigmoid have infinite intrinsic dimensions. Nevertheless, some insight may be gained into the presence of spurious valleys for these cases too. First, we believe that if the size of the k-th hidden layer is less than the k-th intrinsic dimension dim_ρ^k, then there exists a training dataset $\mathcal{T}$ such that the resulting loss landscape has spurious valleys (this, of course, holds in particular for the case with infinite intrinsic dimensions). Second, in the case of infinite intrinsic dimensions, one could consider finite-dimensional vector subspaces $\mathcal{S}_\rho^k$ of the functional spaces $\mathcal{C}_\rho^k$. Each functional subspace corresponds to a restriction R^k on the weights used. The same technique used in the proofs of Theorems 3.6 and 3.9 may allow to show that, if the size of the k-th hidden layer is greater or equal to the dimension of the k-th vector subspace $\mathcal{S}_\rho^k$, then there are no spurious valleys in the corresponding subset of the loss landscape (i.e. the landscape obtained by considering the graph of the error function restricted on the subset of weights given by the R^k's). Notice that, increasing the sizes of the hidden layers would allow to use the modified result with less restrictive limitations on the weights, i.e. the considered subset of the loss landscape has a higher dimension. This hints at the fact that increasing the sizes of the hidden layers reduces the dimension of the subspace of the loss landscape in which spurious valleys may occur, i.e. it reduces the chances of encountering spurious valleys during gradient descent. Both the conjectures presented above require some more work and care, and they are left for future work. It would also be interesting to investigate whether Theorems 3.7 and 3.10 hold in the population risk setting. We believe they do, since the core of the proofs seems to work for any input-output distribution, and the technique used would remain the same. However, adapting the setting for population risk would lead to the introduction of some additional concepts and notation (in order to properly deal with the expectation substituting the sum in the error computation), which would make the paper even more complex and less accessible. Such extension is certainly desirable but it is left for future work.

Our second result (Theorem 4.4) focuses on the expressive power of neural networks with a polynomial activation function. In more detail, Theorem 4.4 expresses the behavior of the errors of certain singular points of the loss landscape as we modify the training dataset. Moreover, such behavior provides information on the representability of the target function. Even though Theorem 4.4 holds for 1-k-1 polynomial NNs, we observed a similar clustering behavior in deeper polynomial NNs. Hence, the next step would be to extend Theorem 4.4 to a wider class of NNs. This could yield valuable novel insight into the set of functions representable by a NN architecture.

Acknowledgements

The results contained in this paper are part of the author's Master's thesis project, carried out at Bonn University with advisors Daniel Huybrechts and Emre Setöz. The paper was completed during the first year of the author's doctoral studies at Vrije Universiteit Brussel with supervisor Bart Bogaerts. This work was partially supported by Fonds Wetenschappelijk Onderzoek – Vlaanderen (project G0B2221N) and the Flemish Government (Onderzoeksprogramma Artificiële Intelligentie (AI) Vlaanderen).

References

Arora, S., Cohen, N., and Hazan, E. On the optimization of deep networks: Implicit acceleration by overparameterization. In Dy, J. G. and Krause, A. (eds.), *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, volume 80 of *Proceedings of Machine Learning Research*, pp. 244–253. PMLR, 2018.

Arora, S., Cohen, N., Golowich, N., and Hu, W. A convergence analysis of gradient descent for deep linear neural networks. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019.

Ayachit, U. *The ParaView Guide: A Parallel Visualization Application*. Kitware, Inc., Clifton Park, NY, USA, 2015. ISBN 1930934300.

Bosma, W., Cannon, J., and Playoust, C. The Magma algebra system. I. The user language. *J. Symbolic Comput.*, 24(3-4):235–265, 1997. ISSN 0747-7171. doi: 10.1006/jsco.1996.0125. Computational algebra and number theory (London, 1993).

Chizat, L. and Bach, F. R. On the global convergence of gradient descent for over-parameterized models using optimal transport. In Bengio, S., Wallach, H. M., Larochelle, H., Grauman, K., Cesa-Bianchi, N., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pp. 3040–3050, 2018.

Draxler, F., Veschgini, K., Salmhofer, M., and Hamprecht, F. A. Essentially no barriers in neural network energy landscape. In Dy, J. G. and Krause, A. (eds.), *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, volume 80 of *Proceedings of Machine Learning Research*, pp. 1308–1317. PMLR, 2018.

Du, S. S. and Lee, J. D. On the power of over-parametrization in neural networks with quadratic activation. In Dy, J. G. and Krause, A. (eds.), *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, volume 80 of *Proceedings of Machine Learning Research*, pp. 1328–1337. PMLR, 2018.

Freeman, C. D. and Bruna, J. Topology and geometry of half-rectified network optimization. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017.

Goertz, U. and Wedhorn, T. *Algebraic Geometry I*. Springer, 2020. ISBN 978-3-8348-0676-5. doi: 10.1007/978-3-8348-9722-0.

Griffiths, P. and Harris, J. *Principles of Algebraic Geometry*. Wiley, 1994. ISBN 978-0-471-05059-9.

Hardt, M. and Ma, T. Identity matters in deep learning. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017.

Harris, J. W. Cones, projections, and more about products. In *Algebraic Geometry: a first course*, 1992.

Hartshorne, R. *Algebraic geometry*. Springer-Verlag, New York, 1977. ISBN 0-387-90244-9. Graduate Texts in Mathematics, No. 52.

Kawaguchi, K. Deep learning without poor local minima. In Lee, D. D., Sugiyama, M., von Luxburg, U., Guyon, I., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, pp. 586–594, 2016.

Kileel, J., Trager, M., and Bruna, J. On the expressive power of deep polynomial neural networks. In Wallach, H. M., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E. B., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pp. 10310–10319, 2019.

Larsen, B. W., Fort, S., Becker, N., and Ganguli, S. How many degrees of freedom do we need to train deep networks: a loss landscape perspective. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net, 2022.

Li, H., Xu, Z., Taylor, G., Studer, C., and Goldstein, T. Visualizing the loss landscape of neural nets. In Bengio, S., Wallach, H. M., Larochelle, H., Grauman, K., Cesa-Bianchi, N., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pp. 6391–6401, 2018.

Mehta, D., Chen, T., Tang, T., and Hauenstein, J. D. The loss surface of deep linear networks viewed through the algebraic geometry lens. *IEEE Trans. Pattern Anal. Mach. Intell.*, 44(9):5664–5680, 2022. doi: 10.1109/TPAMI.2021.3071289.

Mei, S., Montanari, A., and Nguyen, P. A mean field view of the landscape of two-layers neural networks. *CoRR*, abs/1804.06561, 2018.

Montúfar, G., Pascanu, R., Cho, K., and Bengio, Y. On the number of linear regions of deep neural networks. In Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N. D., and Weinberger, K. Q. (eds.), *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*, pp. 2924–2932, 2014.

Pittorino, F., Ferraro, A., Perugini, G., Feinauer, C., Baldassi, C., and Zecchina, R. Deep networks on toroids: Removing symmetries reveals the structure of flat regions in the landscape geometry. In Chaudhuri, K., Jegelka, S., Song, L., Szepesvári, C., Niu, G., and Sabato, S. (eds.), *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA*, volume 162 of *Proceedings of Machine Learning Research*, pp. 17759–17781. PMLR, 2022.

Simsek, B., Ged, F., Jacot, A., Spadaro, F., Hongler, C., Gerstner, W., and Brea, J. Geometry of the loss landscape in overparameterized neural networks: Symmetries and invariances. In Meila, M. and Zhang, T. (eds.), *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pp. 9722–9732. PMLR, 2021.

Soltanolkotabi, M., Javanmard, A., and Lee, J. D. Theoretical insights into the optimization landscape of over-parameterized shallow neural networks. *IEEE Trans. Inf. Theory*, 65(2):742–769, 2019. doi: 10.1109/TIT.2018.2854560.

Stein, W., Joyner, D., Kohel, D., Cremona, J., and Eröcal, B. *SageMath, the Sage Mathematics Software System (Version 9.0)*, 2020. `https://www.sagemath.org`.

Trager, M., Kohn, K., and Bruna, J. Pure and spurious critical points: a geometric study of linear networks. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020.

Venturi, L., Bandeira, A. S., and Bruna, J. Spurious valleys in one-hidden-layer neural network optimization landscapes. *J. Mach. Learn. Res.*, 20:133:1–133:34, 2019.

Yang, X. The landscape of multi-layer linear neural network from the perspective of algebraic geometry. *CoRR*, abs/2102.04338, 2021.

A. Proof of Theorem 3.7

In this appendix, we present the proof of Theorems 3.7.

Theorem (3.7)**.** *Consider Setup 3.3 for a NN without bias, and suppose $dim^k_\sigma < \infty$ for all $k \in [1, l]$. If $h_k \geq dim^k_\sigma$ for all $k \in [1, l]$, then the loss landscape $\mathcal{L}$ has no spurious valleys.*

We need some additional definitions and a few preliminary results. As pointed out in (Freeman & Bruna, 2017) and (Venturi et al., 2019), the following property implies that the loss landscape $\mathcal{L}$ has no spurious valley:

Property (3.9)**.** *Given any initial $\mathfrak{w}$-tuple of parameters $\tilde{\theta} \in \Theta$, there exists a continuous path $\theta\colon t \in [0, 1] \mapsto \theta_t \in \Theta$ such that $\theta_0 = \tilde{\theta}$, $\theta_1 \in \arg\min_{\theta\in\Theta} err_{\mathcal{T}}(\theta)$, and the function $t \mapsto err_{\mathcal{T}}(\theta_t)$ is non-increasing.*

We will use Property 3.9 to prove Theorem 3.7 and its bias case analogue (Theorem 3.10).

First, we define a few more maps, linking the parameters of the NN to functions in $\mathcal{C}^k_\sigma$. For any positive integer m we define

$$\begin{aligned} \psi^0\colon \mathbb{R}^m &\to \mathcal{C}^1_\sigma \\ w &\mapsto \psi_w. \end{aligned}$$

Whenever we want to apply ψ^0 to the rows of a matrix $A = (a_{i,j})_{i,j} \in \mathbb{R}^{s\times t}$ we will write $\psi^1(A) := (\psi_{a_{1,\cdot}}, \dots, \psi_{a_{s,\cdot}})^\top$. Finally, for any $k \in \mathbb{Z}_{>1}$, and for any $k+1$-tuple of positive integers $(m_0, m_1, \dots, m_k) \in \mathbb{Z}^{k+1}_{>0}$ we define the map

$$\psi^k\colon \mathbb{R}^{m_1\times m_0} \times \mathbb{R}^{m_2\times m_1} \times \dots \times \mathbb{R}^{m_k\times m_{k-1}} \to \left(\mathcal{C}^k_\sigma\right)^{m_k},$$

by sending $(W_0, \dots, W_{k-1})$ to $\psi^1(W_{k-1})(\psi^{k-1}(W_0, \dots, W_{k-2}))$.

Notation A.1. For any tuple $a = (b_0, \dots, b_k)$ and $i \in [0, k]$ we denote $a'_i := (b_0, \dots, b_i)$.

Remark A.2. If $\mathcal{N}$ is a NN without bias with l hidden layers, then for any tuple of parameters $A = (W_0, \dots, W_k)$ we can write $\mathcal{N}_A(x) = W_l\psi^l(A'_{l-1})(x)$.

Proposition A.3. *Let σ be a continuous map, and $k \in \mathbb{Z}_+$. If $d := dim^k_\sigma < \infty$, then for any basis of $\mathcal{C}^k_\sigma$ there exist an inner product $\langle\cdot,\cdot\rangle$ on $\mathcal{C}^k_\sigma$ and a map $\phi^k\colon \mathbb{R}^p \to \mathcal{C}^k_\sigma$ such that $\langle f, \phi^k(x)\rangle = f(x)$, for any $f \in \mathcal{C}^k_\sigma$ and $x \in \mathbb{R}^p$.*

Proof. The proof is analogous to the one for Lemma 18 in (Venturi et al., 2019). □

Proposition A.4. *Let $k \in \mathbb{Z}_{>0}$. If σ is continuous and $d := dim^k_\sigma < \infty$, then the map ψ^k is continuous.*

Proof. For ψ^0 see the proof of Lemma 18 in (Venturi et al., 2019). ψ^1 is just ψ^0 applied row-wise, hence continuity of ψ^1 follows immediately from continuity of ψ^0. Then we proceed by induction. Consider ψ^k and assume ψ^{k-1} is continuous. We fix a base $\{f_1, \dots, f_d\}$ for $\mathcal{C}^k_\sigma$, and consider the inner product and the map $\phi^k(x) := \sum_{i=1}^d f_i(x)f_i$, defined in Lemma A.3. Let $\mathcal{R} := \mathbb{R}^{h_1\times p} \times \mathbb{R}^{h_2\times h_1} \times \dots \times \mathbb{R}^{h_{k-1}\times h_{k-2}} \times \mathbb{R}^{h_{k-1}}$, and $s\colon \mathcal{R} \to \mathbb{R}^d$ be the map defined by sending $(W_0, \dots, W_{k-2}, w)$ to a d-tuple $(t_1, \dots, t_d)$, where $t_i := \langle\psi^0(w)(\psi^{k-1}(W'_{k-2})), f_i\rangle$. By the definitions of ψ^k and ψ^1, it is sufficient to show that, for $i \in [1, d]$, s is continuous. We choose $x_1, \dots, x_d \in \mathbb{R}^p$ such that $\{\phi^k(x_1), \dots, \phi^k(x_d)\}$ is a basis of $\mathcal{C}^k_\sigma$. We define the function $z\colon \mathcal{R} \to \mathbb{R}^d$ by sending $(W_0, \dots, W_{k-2}, w)$ to a d-tuple $(t'_1, \dots, t'_d)$, where $t'_i := \psi^0(w)(\psi^{k-1}(W'_{k-2}))(x_i)$. Let $M := (f_j(x_i))_{i,j} \in \mathbb{R}^{d\times d}$. Then $s = M^{-1}z$ and we reduce to prove continuity of z, as M^{-1} is a fixed matrix. This is equivalent to show that, for any $i \in [1, d]$, $\psi^0(\cdot)(\psi^{k-1}(\cdot))(x_1)$ is continuous. Since $\psi^0(w)(\psi^{k-1}(A))$ is in $\mathcal{C}^k_\sigma$ for any $(A, w) \in \mathcal{R}$, by Lemma A.3 we have

$$\psi^0(w)(\psi^{k-1}(A))(x_i) = \langle\psi^0(w)(\psi^{k-1}(A)), \phi^k(x_i)\rangle = \langle\sigma\langle w, \psi^{k-1}(A)\rangle, \phi^k(x_i)\rangle.$$

Notice that $\phi^k(x_i)$ is fixed, ψ^{k-1} is continuous in the parameters A by induction hypothesis, and σ is continuous by hypothesis. By continuity of the inner products we conclude. □

By Remark A.2 and Proposition A.3, a NN function can be written as $\mathcal{N}_A(x) = \langle W_l\psi^l(A'_{l-1}), \phi(x)\rangle$, for any $A = (W_1, \dots, W_l) \in \Theta$, and $x \in \mathbb{R}^p$, where the scalar product is applied row-wise on $W_l\psi^l(A'_{l-1})$. Moreover, notice that, for any $k \in [1, l]$, the h_k-vector $\psi^k(A'_{k-1})$ can be written as a $h_k \times dim^k$ matrix by using a basis of $\mathcal{C}^k_\sigma$.

The next two lemmata will be the building blocks used in the proof of Theorem 3.7. Lemma A.5 shows that, under certain conditions, for any set of parameters $A \in \Theta$, it is possible to find a path in the fiber $\mathcal{N}^{-1}(\mathcal{N}_A) \subseteq \Theta$ starting from A and ending in another set of parameters $\tilde{A}$ such that the matrix $\psi^{k+1}(\tilde{A}'_k)$ is full rank. In particular, since we are moving in the fiber above $\mathcal{N}_A$, the NN function remains the same along the path and the error remains constant.

Lemma A.5. *Let $k \in [1, l-1]$, and $A = (A_0, \dots, A_l) \in \Theta$. Suppose that $dim_\sigma^{k+1} < \infty$, and $\operatorname{rank}(\psi^k(A'_{k-1})) = dim_\sigma^k < \infty$. If $h_{k+1} \geq dim_\sigma^{k+1}$, then there exists a path $\theta\colon t \in [0,1] \mapsto \theta_t \in \Theta$ such that*

1. *$\theta_0 = A$;*
2. *$\theta_1 =: \tilde{A} \in \Theta$ such that $A_i = \tilde{A}_i \ \ \forall i \in [0, l] \setminus \{k, k+1\}$, and $\operatorname{rank}(\psi^{k+1}(\tilde{A}'_k)) = dim_\sigma^{k+1}$;*
3. *$\mathcal{N}_{\theta_t} = \mathcal{N}_A \ \forall t \in [0,1]$.*

Proof. If $\operatorname{rank}(\psi^{k+1}(A'_k)) = dim^{k+1}$ we take the constant path. Suppose otherwise $\operatorname{rank}(\psi^{k+1}(A'_k)) = z < dim^{k+1}$. Let $a_1, \dots, a_{h_{k+1}} \in \mathbb{R}^{h_k}$ be the rows of A_k, and let $d_i := \psi^0(a_i)(\psi^k(A_0, \dots, A_{k-1}))$, with $i \in [1, h_{k+1}]$, be the corresponding rows of $\psi^{k+1}(A'_k)$ (remember that we can write the functions $\psi^0(a_i)(\psi^k(A_0, \dots, A_{k-1})) \in \mathcal{C}_\sigma^{k+1}$ as vectors with entries the coefficients of a linear decomposition w.r.t. a basis $\{f_i\}_i$ of $\mathcal{C}_\sigma^{k+1}$). Moreover suppose that $L := \{d_{i_1}, \dots, d_{i_z}\}$ is a linearly independent set, $i_1, \dots, i_z \in [1, h_{k+1}]$. We will construct a new set of parameters $B \in \Theta$ by modifying just A_{k+1}. Let $I := \{i_1, \dots, i_z\}$, $J := [1, h_{k+1}] \setminus I = \{j_1, \dots, j_{h_{k+1}-z}\}$, and $b_1, \dots, b_{h_{k+1}}$ the columns of A_{k+1}. For $j \in J$ we can write

$$d_j = \sum_{s=1}^{z} \alpha_j^s d_{i_s},$$

for some $\alpha_j^i \in \mathbb{R}$. Then define $B_t := A_t$ for $t \in [0, l] \setminus \{k+1\}$, and the columns $\bar{b}_1, \dots \bar{b}_{h_{k+1}}$ of B_{k+1} as

$$\begin{aligned} \bar{b}_i :=& b_i + \sum_{s=1}^{h_1 - z} \alpha_s^i b_{j_s} && \text{for } i \in I, \\ \bar{b}_j :=& 0 && \text{for } j \in J. \end{aligned}$$

In this way we have that $A_{k+1}\psi^{k+1}(A'_k) = B_{k+1}\psi^{k+1}(B'_k)$, i.e. $\psi^{k+2}(A'_{k+1}) = \psi^{k+2}(B'_{k+1})$. Hence the path $t \in [0, \frac{1}{2}] \mapsto \theta_t = (A'_k, 2tB_{k+1} + (1-2t)A_{k+1}, A_{k+2}, \dots, A_l)$ leaves the network unchanged, i.e. $\mathcal{N}_{\theta_t} = \mathcal{N}_A$ for all $t \in [0, \frac{1}{2}]$.

Notice that the linearly independent set L is contained in the spanning set $S := \{\sigma f \mid f \in \mathcal{C}_\sigma^k\}$ defining $\mathcal{C}_\sigma^{k+1}$. Hence there exists a basis T such that $L \subseteq T \subseteq S$. In particular the elements of the basis T are all of the form σf with $f \in \mathcal{C}_\sigma^k$. Since $\operatorname{rank}(\psi^k(B'_{k-1})) = \operatorname{rank}(\psi^k(A'_{k-1})) = dim_\sigma^k$, we can choose $c_{j_1}, \dots, c_{j_{h_{k+1}-z}} \in \mathbb{R}^{h_k}$ such that the matrix $\hat{A}_k$ with rows $\hat{a}_j := c_j$ for $j \in J$ and $\hat{a}_i := a_i$ for $i \in I$, satisfies $\operatorname{rank}(\psi^{k+1}(\hat{A}'_k)) = dim_\sigma^{k+1}$. Then we define $\hat{A} = (B'_{k-1}, \hat{A}_k, B_{k+1}, \dots, B_l) \in \Theta$. We have again that the path $t \in [\frac{1}{2}, 1] \mapsto \theta_t = (B'_{k-1}, (2t-1)\hat{A}_k + (2-2t)B_k, B_{k+1}, \dots, B_l)$ leaves the network unchanged, i.e. $\mathcal{N}_{\theta_t} = \mathcal{N}_B = \mathcal{N}_A$ for all $t \in [\frac{1}{2}, 1]$. Concatenating the two paths built above gives the path with the desired properties. □

It is easy to see from the proof of Lemma A.5 that we can also maximize the rank of the first matrix $\psi^1(A_0)$ in an over-parametrized regime, and all the above considerations also apply.

Lemma A.6. *Let $A = (A_0, \dots, A_l) \in \Theta$. Suppose $dim_\sigma^1 < \infty$. If $h_1 \geq dim_\sigma^1$ then there exists a path $\theta\colon t \in [0,1] \mapsto \theta_t \in \Theta$ such that*

1. *$\theta_0 = A$;*
2. *$\theta_1 =: \tilde{A}$ such that $\operatorname{rank}(\psi^1(\tilde{A}_0)) = dim_\sigma^1$, and $A_i = \tilde{A}_i \ \ \forall i \in [2, l]$;*
3. *$\mathcal{N}_{\theta_t} = \mathcal{N}_A \ \forall t \in [0,1]$.*

Proof. Analogous to the proof of Lemma A.5. It is sufficient to take the spanning set $S := \{\psi_w \mid w \in \mathbb{R}^{h_0}\} = \{\sigma\langle w, \cdot\rangle \mid w \in \mathbb{R}^{h_0}\}$ and considering that the $\mathbb{R}$-vector space of functions $\{\langle w, \cdot\rangle \mid w \in \mathbb{R}^{h_0}\}$ has finite dimension h_0. □

We can now prove the spurious valleys theorem for a NN without bias.

Proof. (Theorem 3.7) For the sake of simplicity, we will write $\mathcal{C}^k$ and dim^k instead of $\mathcal{C}^k_\sigma$ and dim^k_σ respectively. The technique is analogous to the one used in (Venturi et al., 2019): for any $A \in \Theta$ we want to build a path satisfying the conditions in Property 3.9.

Let $\tilde{A} \in \Theta$ be our starting set of parameters. In order to build a path satisfying the conditions in Property 3.9, we will construct $l+1$ intermediate paths $P_1, \dots, P_{l+1} \colon [0,1] \to \Theta$. The final goal of the first l paths is to be able to assume w.l.o.g. $\operatorname{rank}(\psi^l(\tilde{A}'_{l-1})) = dim^l$. The first path P_1 is built by applying Lemma A.6 on $\tilde{A}$. The endpoint of P_1 is a tuple of parameters $\overline{A}$ with $\operatorname{rank}(\psi^1(\overline{A}_0)) = dim^1$. Since the NN function remains the same for all the tuples of parameters on the path P_1, the error function remains constant on the path, i.e. $err_{\mathcal{T}}(\tilde{A}) = err_{\mathcal{T}}(P_1(t)) \, \forall t \in [0,1]$. Hence we can assume w.l.o.g. that the starting tuple of parameters $\tilde{A}$ is such that $\operatorname{rank}(\psi^1(\overline{A}_0)) = dim^1$. Following the same reasoning, we can proceed by induction by applying Lemma A.5 on $\tilde{A}$, $l-1$ more times. More in details, for each $k \in [2, l]$, we build a path P_k such that the endpoint is a tuple of parameters $\overline{A}$ with $\operatorname{rank}(\psi^k(\overline{A}'_{k-1})) = dim^k$, and the error function is constant on P_k. Hence we can assume w.l.o.g. $\operatorname{rank}(\psi^l(\tilde{A}'_{l-1})) = dim^l$.

For the $l+1$-th path consider the following. By initial assumption, for any training sample $\mathcal{T}$, the error function $err_{\mathcal{T}}$ has a global minimum. Since the NN function has the form $\mathcal{N}_A(x) = \langle A_l \psi^l(A'_{l-1}), \phi(x) \rangle$ and $\operatorname{rank}(\psi^l(\tilde{A}'_{l-1})) = dim^l$, there exists $B \in \mathbb{R}^{h_{l+1} \times h_l}$ such that $\hat{A} := (\tilde{A}'_{l-1}, B) \in \arg\min_{A \in \Theta} err_{\mathcal{T}}(A)$. By convexity of L, the error function is convex in the parameters of the last matrix, i.e. $err_{\mathcal{T}}(\tilde{A}'_{l-1}, \cdot)$ is convex. Hence the path $t \in [0,1] \mapsto \theta_t = (\tilde{A}'_{l-1}, tB + (1-t)\tilde{A}_l)$ satisfies all the conditions in Property 3.9, as desired. □

B. Proof of Theorem 3.10

Now we can prove the bias version of the spurious valleys theorem.

Theorem (3.10)**.** *Consider Setup 3.3 for a NN with bias, and suppose dim^k_σ is finite for any $k \in [1, l]$. If $h_k \geq dim^k_\sigma - 1$ for all $k \in [1, l]$, then the loss landscape $\mathcal{L}$ has no spurious valleys.*

We use Setup 3.3 and the notation adopted for Theorem 3.7 with some slight changes:

- $\mathcal{N}$ has bias. As a consequence, the parameter space Θ has a higher dimension.
- The matrices $A_0, \dots, A_l$ of a set of parameters $A \in \Theta$ contain also the bias terms of the respective layers, i.e. they have now one additional column on the right.
- The sets of representable functions are now

$$\mathcal{C}^1_\sigma = \operatorname{span}_{\mathbb{R}} \{\{\psi_w \mid w \in \mathbb{R}^p\} \cup \{1\}\}$$
$$\mathcal{C}^k_\sigma = \operatorname{span}_{\mathbb{R}} \{\{\sigma f \mid f \in \mathcal{C}^{k-1}_\sigma\} \cup \{1\}\}, \quad k > 1.$$

- Finally we modify the definitions of the functions "ψ". If $v = (v_1, \dots, v_m)$ is a m-dimensional vector, we denote by $\overline{v}$ the $m+1$-dimensional vector $(v_1, \dots, v_m, 1)$. For any positive integer m, and any real-valued vector $w \in \mathbb{R}^{m+1}$, we define the new functions

$$\tilde{\psi}_w \colon \mathbb{R}^m \to \mathbb{R}$$
$$x \mapsto \sigma\langle w, \overline{x} \rangle.$$

  Moreover, for any matrix $A \in M_{s \times t}(\mathbb{R})$, we define $\tilde{\psi}(A) := (\tilde{\psi}_{a_1}, \dots, \tilde{\psi}_{a_s})^\top$, where $a_1, \dots, a_s$ are the rows of A. As the first affine transformation of a neural network is always linear, even for the NNs with bias, the functions $\psi^0 \colon w \mapsto \psi_w$ and ψ^1 are defined as for the no-bias case. Finally, for any $k \in \mathbb{Z}_{>1}$, and for any $k+1$-tuple of positive integers $(m_0, m_1, \dots, m_k) \in \mathbb{Z}^{k+1}_{>0}$ we make a slight modification to ψ^k, and we define, by abuse of notation, a map ψ^k from $\mathbb{R}^{m_1 \times m_0} \times \mathbb{R}^{m_2 \times (m_1+1)} \times \dots \times \mathbb{R}^{m_k \times (m_{k-1}+1)}$ to $\left(\mathcal{C}^k_\sigma\right)^{m_k}$, by

$$(W_0, \dots, W_{k-1}) \mapsto \tilde{\psi}(W_{k-1})(\psi^{k-1}(W_0, \dots, W_{k-2})).$$

The proof of Theorem 3.10 is analogous to the proof of Theorem 3.7:

- It is easy to see that Propositions A.3 and A.4 still hold true.
- The NN functions can be written as $\mathcal{N}_A(x) = \langle A_l \overline{\psi^l(A'_{l-1})}, \phi(x) \rangle$, for any $A \in \Theta, x \in \mathbb{R}^{h_0}$, where the scalar product is applied row-wise on the LHS. Moreover, for any $k \in [1, l]$, the $h_k + 1$-vector $\overline{\psi^k(A'_{k-1})} := \big(\psi^k(A'_{k-1}), 1\big)$ can now be written as a $(h_k + 1) \times dim^k$ matrix by using a basis of $\mathcal{C}^k_\sigma$.
- The Lemmata A.5 and A.6 need just a little adjustment which relaxes the bound on the hidden layer size: for Lemma A.5 it is sufficient to have $h_{k+1} \geq dim^{k+1}_\sigma - 1$, and for Lemma A.6 it is sufficient $h_1 \geq dim^1_\sigma - 1$.

C. Proof of Theorem 4.4

In this final appendix, we present the proof of Theorem 4.4. Since we consider just polynomial NNs in this section, we omit the term "polynomial" when referring to NNs and architectures.

Theorem (4.4)**.** *Let $r, h \in \mathbb{Z}$, $r > 1$, $h > 2$. Let $\mathcal{N}$ be a 1-h-1 r-NN, and $\psi(x) := \sum_{j=0}^{d} \alpha_j x^j$ be a real polynomial target function. Let $\{\mathcal{T}_\delta\}$ be a family of translated training datasets for ψ with at least 3 distinct points each. For a generic 2-plane, let $\mathfrak{s}_\delta \in S_{\mathcal{T}_\delta}$. Then one of the followings holds:*

1. *$\mathfrak{s}_\delta \sim_{+\infty} a\delta^{2\max\{r,d\}}$, for some non-zero constant $a \in \mathbb{C}^*$;*
2. *$\mathfrak{s}_\delta \sim_{+\infty} b\delta^{2(m-1)}$, for some non-zero constant $b \in \mathbb{C}^*$;*
3. *if ψ is in the image of $\mathcal{N}$ then $\mathfrak{s}_\delta = 0$, otherwise $\mathfrak{s}_\delta \sim_{+\infty} c\delta^{2(m-2)}$, for some non-zero constant $c \in \mathbb{C}^*$,*

where $m := max(\{j \mid \alpha_j \neq 0 \text{ and } j \neq r\} \cup \{0\})$.

To prove Theorem 4.4, we will need several lemmata. We first fix some notation that we will extensively use in this section.

Notation C.1. We denote by $p_{u,v,k} \colon \mathbb{A}^2_{\mathbb{C}} \to \mathbb{A}^{\mathfrak{w}}_{\mathbb{C}}$, $p_{u,v,k}(x, y) := ux + vy + k$, a generic 2-plane in the space of parameters $\mathbb{A}^{\mathfrak{w}}_{\mathbb{C}}$, with $u, v, k \in \mathbb{A}^{\mathfrak{w}}_{\mathbb{C}}$. We denote with U, V, and K the last coordinate of u, v, and k respectively.

Let $\mathcal{N}$ be a 1-h-1 r-NN, and $\psi(x) := \sum_{j=0}^{d} \alpha_j x^j$ be a real polynomial target function. Let $\mathcal{T}$ be a training dataset for ψ, with $|\mathcal{T}| > 2$, and let $\{\mathcal{T}_\delta\}$ be the family of translated training datasets. When we restrict the error function onto a generic 2-plane $p_{u,v,k}$ we obtain

$$f(x,y) = \frac{1}{N} \sum_{n=1} \left((s - \alpha_r) p_{n,r} + B - \alpha_0 - \sum_{j=1, j\neq r}^{d} \alpha_j p_{n,j} \right)^2, \tag{2}$$

where $B = B(x, y) := Ux + Vy + K$, and $s = s(x, y) := \sum_{j=1}^{h} (u_j x + v_j y + k_j)^r (u_{h+j} x + v_{h+j} y + k_{h+j})$. Recall that we denote by $\mathcal{H}_{\mathcal{T}_\delta} \subseteq \mathbb{A}^3_{\mathbb{C}}$ the graph of f, i.e. the cut loss landscape, by $\pi \colon \mathcal{H}_{\mathcal{T}_\delta} \to \mathbb{A}^1_{\mathbb{C}}$ the projection onto the last coordinate, and by $S_{\mathcal{T}_\delta} \subseteq \mathbb{A}^1_{\mathbb{C}}$ the set of points $\mathfrak{s}_\delta$ where $\pi^{-1}(\mathfrak{s}_\delta)$ is a singular curve. We want to study the limit behaviour of the points of $S_{\mathcal{T}_\delta}$ as $\delta \to +\infty$.

Notice that $S_{\mathcal{T}_\delta} = \{f(x, y) \mid (x, y) \in Z_\delta\} \subseteq \mathbb{A}^1_{\mathbb{C}}$, where $Z_\delta \subseteq \mathbb{A}^2_{\mathbb{C}}$ is the zero locus defined by the partial derivatives of f, i.e.

$$\begin{cases} 0 = \sum_{n=1}^{N} \mathcal{D}_n \left(p_{n,r} \frac{\partial s}{\partial x} + U \right) \\ 0 = \sum_{n=1}^{N} \mathcal{D}_n \left(p_{n,r} \frac{\partial s}{\partial y} + V \right) \end{cases}, \tag{3}$$

where $\mathcal{D}_n := (s - \alpha_r) p_{n,r} + B - \alpha_0 - \sum_{j=1, j\neq r}^{d} \alpha_j p_{n,j}$. Hence, we study the limit behavior of f evaluated at a point of Z_δ.

By manipulating the equations in (3), it is easy to see that $Z_\delta \subseteq Z'_\delta$, where $Z'_\delta \subseteq \mathbb{A}^2_{\mathbb{C}}$ is the zero locus defined by

$$\begin{cases} 0 = \mathbf{D} s (N P_{2r} - P_r^2) - \mathbf{D} \sum\limits_{\substack{j=1 \\ j\neq r}}^{d} \alpha_j (N P_{r+j} - P_j P_r) \\ 0 = \mathbf{D} (B - \alpha_0)(N P_{2r} - P_r^2) - \mathbf{D} \sum\limits_{\substack{i=1 \\ j\neq r}}^{d} \alpha_j (P_{2r} P_j - P_{r+j} P_r) \end{cases} \tag{4}$$

where $\mathbf{D} := \frac{\partial s}{\partial x}V - \frac{\partial s}{\partial y}U$. We will use Z'_δ to prove Theorem 4.4.

Notation C.2. We denote by $Z \subseteq \mathbb{A}^3_{\mathbb{C}}$ the zero locus describing the points in (3), with x, y, and δ as variables. Let $\overline{Z} \subseteq \mathbb{P}^2_{\mathbb{C}} \times \mathbb{P}^1_{\mathbb{C}}$ be the reduced projective closure of Z w.r.t. the variables (x, y) and δ separately. For $[\delta : \eta] \in \mathbb{P}^1_{\mathbb{C}}$, let $\overline{Z}_{[\delta:\eta]} \subseteq \mathbb{P}^2_{\mathbb{C}}$ be the restriction of $\overline{Z}$ to $\mathbb{P}^2_{\mathbb{C}} \times \{[\delta : \eta]\}$.

Let $\pi_2 \colon \overline{Z} \subseteq \mathbb{P}^2_{\mathbb{C}} \times \mathbb{P}^1_{\mathbb{C}} \to \mathbb{P}^1_{\mathbb{C}}$ be the second projection. We proceed by considering local sections of π_2. Since we are interested just in real values of δ close to $+\infty$, we may consider sections from a real interval $D := (M, +\infty) \subseteq \mathbb{R}_{>0}$. Since D is simply connected, there exists an analytic local section $\sigma' \colon D \subseteq \mathbb{A}^1_{\mathbb{C}} \to \mathbb{P}^2_{\mathbb{C}} \times \mathbb{P}^1_{\mathbb{C}}$ of π_2. Let $\sigma := \pi_1 \circ \sigma' \colon D \to \mathbb{P}^2_{\mathbb{C}}$ be the composition with the first projection. By Lemma C.7, we may reduce to $\sigma \colon D \subseteq \mathbb{A}^1_{\mathbb{C}} \to \mathbb{A}^2_{\mathbb{C}}$.

Now we can study the limit behaviour of $f(\sigma)$ as δ goes to $+\infty$. To do so, we prove some lemmata.

Notation C.3. We will denote by $\overline{s}$ and $\overline{B}$ the homogenizations of s and B restricted to $\{[x : y : 0] \mid [x : y] \in \mathbb{P}^1_{\mathbb{C}}\}$. We denote by $\overline{s}_x$ and $\overline{s}_y$ the partial derivatives of $\overline{s}$. Moreover, we write $\tilde{s}_x$ and $\tilde{s}_y$ for the dehomogeneizations of $\overline{s}_x$ and $\overline{s}_y$ respectively. More explicitly we have:

$$\begin{aligned}
\overline{s}(x,y) &:= \sum_{j=1}^{h} (u_j x + v_j y)^r (u_{h+j} x + v_{h+j} y) \\
\overline{B}(x,y) &:= Ux + Vy \\
\overline{s}_x(x,y) &:= \sum_{j=1}^{h} (u_j x + v_j y)^{r-1} \Big((r+1) u_j u_{h+j} x + r u_j v_{h+j} y + v_j u_{h+j} y \Big) \\
\overline{s}_y(x,y) &:= \sum_{j=1}^{h} (u_j x + v_j y)^{r-1} \Big((r+1) v_j v_{h+j} y + r v_j u_{h+j} x + u_j v_{h+j} x \Big)
\end{aligned} \tag{5}$$

$$\begin{aligned}
\tilde{s}_x(x) &:= \sum_{j=1}^{h} (u_j x + v_j)^{r-1} \Big((r+1) u_j u_{h+j} x + r u_j v_{h+j} + v_j u_{h+j} \Big) \\
\tilde{s}_y(x) &:= \sum_{j=1}^{h} (u_j x + v_j)^{r-1} \Big((r+1) v_j v_{h+j} + r v_j u_{h+j} x + u_j v_{h+j} x \Big).
\end{aligned} \tag{6}$$

Lemma C.4. *The zero locus $\mathcal{Z}(\tilde{s}_x, \tilde{s}_y) \subseteq \mathbb{A}^1_{\mathbb{C}}$ is empty for a generic 2-plane.*

Proof. We would like to understand which conditions on the parameters of the 2-plane imply that $\mathcal{Z}(\tilde{s}_x, \tilde{s}_y)$ is empty, i.e. the GCD$(\tilde{s}_x, \tilde{s}_y)$ is a polynomial of degree 0. When we run the Euclidean algorithm for GCD, the degree of the residual polynomial is decreased of at least 1 in each iteration. Since we want the last non-zero residual to be of degree 0, and $\tilde{s}_x$ and $\tilde{s}_y$ are both polynomial in x of degree r, from the GCD computation we get at most r closed conditions of the form $\mathcal{Z}(c_0, \dots, c_{r-i})$ at iteration i, where $c_0, \dots, c_{r-i}$ are the coefficients of the i-th residual polynomial. We denote the union of such vanishing loci with Z.

It remains to show there exists a 2-plane not in Z. Let p' be the 2-plane defined by $u_1 = u_{h+1} = v_2 = v_{h+2} = U = 1$, $V = 2$, and all the other entries of $u, v, k \in \mathbb{A}^{\mathfrak{w}}_{\mathbb{C}}$ equal to 0. We have $\overline{s}(x, y) = x^{r+1} + y^{r+1}$, $\tilde{s}_x(x) = (r+1)x^r$, and $\tilde{s}_y(x) = r + 1$. Hence, the chosen vectors for p' define a point that is not contained in Z. □

Lemma C.5. *The zero loci $\mathcal{Z}(\overline{s}, \overline{B}), \mathcal{Z}(\overline{s}, V\overline{s}_x - U\overline{s}_y) \subseteq \mathbb{P}^1_{\mathbb{C}}$ are empty for a generic 2-plane.*

Proof. We want to show $\mathcal{Z}(\overline{s}, \overline{B})$ and $\mathcal{Z}(\overline{s}, V\overline{s}_x - U\overline{s}_y)$ are both empty for all 2-planes in a Zariski open A. To achieve this we explicitly construct A as complement of the union of some close sets $Z_0, \dots, Z_4$, and we show A is non-empty.

If $U = V = 0$, then $\mathcal{Z}(\overline{s}, \overline{B}) = \mathcal{Z}(\overline{s}, V\overline{s}_x - U\overline{s}_y) = \mathcal{Z}(\overline{s}) \neq \emptyset$. Hence we set $Z_0 := \mathcal{Z}(U = 0, V = 0)$.

Let $[x' : y'] \in \mathcal{Z}(\overline{s}, \overline{B})$. Hence $0 = \overline{B}(x', y') = Ux' + Vy'$, which implies $[x' : y'] = [-V : U]$. Then we get a closed condition on the parameters of the 2-plane, namely $Z_1 := \mathcal{Z}\Big(\overline{s}(-V, U)\Big) \subseteq \mathbb{A}^{3\mathfrak{w}}_{\mathbb{C}}$.

Now let $[x' : y'] \in \mathcal{Z}(\overline{s}, V\overline{s}_x - U\overline{s}_y)$. Then $0 = (r+1)\overline{s}(x', y') = x'\overline{s}_x(x', y') + y'\overline{s}_y(x', y')$. If $y' = 0$, then $\overline{s}_x(1, 0) = 0$. Thus we obtain another closed condition $Z_2 := \mathcal{Z}\Big(\sum_{j=1}^{h} u_j^r u_{h+j}\Big)$. If $y' \neq 0$, by using $V\overline{s}_x(x', y') - U\overline{s}_y(x', y') = 0$, we obtain $\overline{s}_x(x', y')(Ux' + Vy') = 0$ and $\overline{s}_y(x', y')(Ux' + Vy') = 0$. If $Ux' + Vy' = 0$, we get $[x' : y'] = [-V : U]$ and the same condition encoded in Z_1. If $Ux' + Vy' \neq 0$, then $\overline{s}_x(x', y') = \overline{s}_y(x', y') = 0$, i.e. (x', y') is a common root of the two derivatives. Hence we need $\mathcal{Z}(\overline{s}_x, \overline{s}_y) \subseteq \mathbb{P}^2_{\mathbb{C}}$ to be empty. Since $y' \neq 0$ we may as well de-homogenise and consider the affine chart for $y \neq 0$, i.e. we focus on $\mathcal{Z}(\tilde{s}_x, \tilde{s}_y) \subseteq \mathbb{A}^1_{\mathbb{C}}$. By Lemma C.4, $\mathcal{Z}(\tilde{s}_x, \tilde{s}_y) \subseteq \mathbb{A}^1_{\mathbb{C}}$ is empty for a generic 2-plane.

Notice that by how we defined the 2-plane $p_{u,v,k}$ not all triples of vectors (u, v, k) defines a 2-plane: u and v have to be linearly independent. In other words, the $2 \times \mathfrak{w}$ matrix M formed by stacking the vectors u and v must have a 2×2 non-vanishing minor. Thus we can build one last zero locus Z_3, defined by the 2×2 minors of M.

Hence we can define the Zariski open $A := \mathbb{A}^{3\mathfrak{w}}_{\mathbb{C}} \setminus \bigcup_{i=0}^{3} Z_i$. It is easy to verify that the 2-plane p', defined in the proof of Lemma C.4, is in A. In particular, A is not empty. □

Lemma C.6. *Let $a, b, c \in \mathbb{C}$. If $\frac{\partial s}{\partial x}(\sigma) - b$ and $\frac{\partial s}{\partial y}(\sigma) - c$ both tends to 0 as δ goes to $+\infty$, then $\lim_{\delta \to +\infty} s(\sigma) - a \neq 0$ for a generic 2-plane.*

Proof. We show the statement holds for a generic 2-plane, i.e. for 2-planes in a non-empty Zariski open A.

Let s', $\overline{\frac{\partial s}{\partial x}}$, and $\overline{\frac{\partial s}{\partial y}}$ be the homogeneizations of s and its partial derivatives, respectively. Consider $Z := \mathcal{Z}\left(s' - az^{r+1}, \overline{\frac{\partial s}{\partial x}} - bz^r, \overline{\frac{\partial s}{\partial y}} - cz^r\right) \subseteq \mathbb{A}^{3\mathfrak{w}}_{\mathbb{C}} \times \mathbb{P}^2_{\mathbb{C}}$, where the parameters of the generic 2-plane, and x, y, z are the variables. Then, the projection $\pi(Z)$ of Z on $\mathbb{A}^{3\mathfrak{w}}_{\mathbb{C}}$ is again close (for example by Theorem 3.12 in (Harris, 1992)). We take $A := \mathbb{A}^{3\mathfrak{w}}_{\mathbb{C}} \setminus \pi(Z)$, which clearly is open. It remains to show A is non-empty. We have to find the parameters $u, v, k \in \mathbb{A}^{\mathfrak{w}}_{\mathbb{C}}$ of a 2-plane for which $\overline{Z} \cap \{(u, v, k)\} \times \mathbb{P}^2_{\mathbb{C}}$ is empty. Here we use the notation for u, v, k we have adopted in the rest of the section, namely $u := (u_1, \ldots, u_h, u_{h+1}, \ldots, u_{2h}, U)$, and analogously for v and k. We take $u_1 = u_{h+1} = v_2 = v_{h+2} = k_3 = U = 2V = 1$, and $k_{h+3} \in \mathbb{C}$ such that $k_{h+3} \neq c + \eta\frac{a}{r+1} + \mu\frac{b}{r+1}$ for any $\eta, \mu \in \mathbb{C}$ with $\eta^r = \frac{a}{r+1}$, and $\mu^r = \frac{b}{r+1}$. We take all the other entries of u, v, and k equal to 0. Then we get $s'(x, y, z) = x^{r+1} + y^{r+1} + k_{h+3}z^{r+1}$. With the choices made for u, v, k it is easy to verify that $Z \cap \{(u, v, k)\} \times \mathbb{P}^2_{\mathbb{C}}$ is empty. Hence, the chosen (u, v, k) is a point in $\mathbb{A}^{3\mathfrak{w}}_{\mathbb{C}} \setminus \pi(Z)$, concluding the proof. □

Lemma C.7. *Let $\delta \in \mathbb{A}^1_{\mathbb{C}}$. Then $\overline{Z}_{[\delta:1]}$ has no points at infinity for a generic 2-plane.*

Proof. We re-write the polynomials in (3) in a more convenient way:

$$p_x := (s - \alpha_r)\left(\frac{\partial s}{\partial x}P_{2r} + P_r U\right) + (B - \alpha_0)\left(\frac{\partial s}{\partial x}P_r + NU\right) - \sum_{j=1, j\neq r}^{d} \alpha_j \left(\frac{\partial s}{\partial x}P_{r+j} + UP_j\right),$$

and

$$p_y := (s - \alpha_r)\left(\frac{\partial s}{\partial y}P_{2r} + P_r V\right) + (B - \alpha_0)\left(\frac{\partial s}{\partial y}P_r + NV\right) - \sum_{j=1, j\neq r}^{d} \alpha_j \left(\frac{\partial s}{\partial y}P_{r+j} + VP_j\right).$$

Notice that

$$p_3 := \mathbf{D}(B - \alpha_0)\left(P_r^2 - NP_{2r}\right) - \mathbf{D}\sum_{\substack{j=1 \\ j\neq r}}^{d} \alpha_j (P_r P_{r+j} - P_{2r}P_j) = \left(\frac{\partial s}{\partial y}P_{2r} + VP_r\right)p_x - \left(\frac{\partial s}{\partial y}P_{2r} + UP_r\right)p_y$$

is in the ideal generated by p_x and p_y. Hence, $\overline{Z}_{[\delta:1]}$ is contained in $\mathcal{Z}(\overline{p}_x, \overline{p}_y, \overline{p}_3) \subseteq \mathbb{P}^2_{\mathbb{C}}$, where $\overline{p}_x, \overline{p}_y$, and $\overline{p}_3$ are the homogeneizations of p_x, p_y, and p_3 respectively. We are interested in the points of $C := \mathcal{Z}(\overline{p}_x, \overline{p}_y, \overline{p}_3)$ of the form $[x : y : 0]$:

$$C|_{\{[x:y:0] | [x:y] \in \mathbb{P}^1_{\mathbb{C}}\}} \colon \mathcal{Z}\Big(\overline{s}\,\overline{s}_x, \overline{s}\,\overline{s}_y, \overline{B}(V\overline{s}_x - U\overline{s}_y)\Big) = \mathcal{Z}(\overline{s}, \overline{B}) \cup \mathcal{Z}(\overline{s}, V\overline{s}_x - U\overline{s}_y) \tag{7}$$

The equality in 7 comes from the equality $(r+1)\overline{s} = x\overline{s}_x + y\overline{s}_y$. By Lemma C.5, the zero locus in (7) is empty. Hence $\overline{Z}_{[\delta:1]}$ has no points at infinity. □

Lemma C.8. *Let $a, b \in \mathbb{A}^1_{\mathbb{C}}$, and let s, B, and $\sigma(\delta)$ be defined as for Theorem 4.4 for a generic 2-plane. If $\lim_{\delta\to+\infty} \frac{\partial s}{\partial x}(\sigma) = a$ and $\lim_{\delta\to+\infty} \frac{\partial s}{\partial y}(\sigma) = b$, then $\lim_{\delta\to+\infty} \sigma \in \mathbb{A}^2_{\mathbb{C}}$. In particular the limits $\lim_{\delta\to+\infty} s(\sigma)$ and $\lim_{\delta\to+\infty} B(\sigma)$ are both finite.*

Proof. We want to show that $\lim_{\delta\to+\infty} \sigma$ is finite for a generic 2-plane. In order to do this, we homogenize the partial derivatives of s, and we consider the zero locus $Z := \mathcal{Z}\left(\overline{\frac{\partial s}{\partial x}} - az^r, \overline{\frac{\partial s}{\partial y}} - bz^r\right) \subseteq \mathbb{P}^2_{\mathbb{C}}$. The points at infinity of Z are $Z' := \mathcal{Z}(\overline{s}_x, \overline{s}_y) \subseteq \mathbb{P}^1_{\mathbb{C}}$, where $\overline{s}_x$ and $\overline{s}_y$ are like in (5). We will show that Z' is empty for a generic 2-plane. Let $[x^* : y^*] \in Z$. From the expressions of $\overline{s}_x$ and $\overline{s}_y$ is clear that, for a generic 2-plane, $x^* = 0$ if and only if $y^* = 0$. Then we can assume w.l.o.g. that $y^* \neq 0$ and dehomogenize the polynomials $\overline{s}_x$ and $\overline{s}_y$. By Lemma C.4, $\mathcal{Z}(\tilde{s}_x, \tilde{s}_y) \subseteq \mathbb{A}^1_{\mathbb{C}}$ is empty for a generic 2-plane, where $\tilde{s}_x$ and $\tilde{s}_y$ are $\overline{s}_x$ and $\overline{s}_y$ dehomogenized, respectively. Hence Z' is empty and Z has no points at infinity for a generic 2-plane. It follows that if $\frac{\partial s}{\partial x}(\sigma) - a$ and $\frac{\partial s}{\partial y}(\sigma) - b$ are both going to 0, then it is not possible for σ to tend to a point at infinity of $\mathbb{A}^2_{\mathbb{C}}$. □

Lemma C.9. *Suppose $\frac{\partial s}{\partial x}(\sigma)V - \frac{\partial s}{\partial y}(\sigma)U = 0$ on D, and $\frac{\partial s}{\partial x}(\sigma)$ goes to infinity as $\delta \to +\infty$. Then $s(\sigma) - \alpha_r \sim \alpha_m \delta^{m-r}$, as $\delta \to +\infty$.*

Proof. Recall that $B = B(x, y) := Ux+Vy+K$ and $s = s(x, y) := \sum_{j=1}^h (u_j x+v_j y+k_j)^r (u_{h+j}x+v_{h+j}y+k_{h+j})$, for a generic 2-plane $p_{u,v,k}$ defined by the vectors $u := (u_1, \ldots, u_{2h}, U)$, $v := (v_1, \ldots, v_{2h}, V)$, $k := (k_1, \ldots, k_{2h}, K) \in \mathbb{A}^{\mathbf{w}}_{\mathbb{C}}$. From the first equation in (3) we obtain

$$(B(\sigma) - \alpha_0)\left(\frac{\partial s}{\partial x}(\sigma)P_r + NU\right) = -(s(\sigma) - \alpha_r)\left(\frac{\partial s}{\partial x}(\sigma)P_{2r} + UP_r\right) + \sum_{j=1, j\neq r}^{d} \alpha_j \left(\frac{\partial s}{\partial x}(\sigma)P_{r+j} + UP_j\right). \quad (8)$$

We may assume that $s(\sigma) - \alpha_r \sim a\delta^{t-r}$ for some $a \in \mathbb{C}^*$ and $t \in \mathbb{Z}$. Moreover, we can write $\sigma_1 \sim c_1\delta^{q_1}$ and $\sigma_2 \sim c_2\delta^{q_2}$ for some $c_1, c_2 \in \mathbb{C}^*$ and $q_1, q_2 \in \mathbb{Z}$. We proceed by contradiction.

Suppose that either $t > m$ or $t = m$ and $a \neq \alpha_m$. By (8), we must have $B(\sigma) - \alpha_0 \sim b\delta^t$, where $b = -a$ if $t > m$, or $b = \alpha_m - a$ if $t = m$. In particular, $b \neq 0$. Since $t \geq m > 0$, we have $U\sigma_1 + V\sigma_2 \sim b\delta^t$, where $\sigma = (\sigma_1, \sigma_2)$. Then only three cases are possible:

1. $q_1 = q_2 > t$. In this case we necessarily have $c_1 + c_2 = 0$. Hence we have

$$s(\sigma) = c_1^{r+1}\left(\sum_{j=1}^h (u_j - v_j)^r (u_{h+j} - v_{h+j})\right)\delta^{q_1(r+1)} + o\left(\delta^{q_1(r+1)}\right).$$

   Notice that $c_1 \neq 0$ by hypothesis, and $\sum_{j=1}^h (u_j - v_j)^r (u_{h+j} - v_{h+j})$ is non-zero for a generic 2-plane. Hence $s(\sigma)$ is going to infinity of degree $q_1(r + 1) > t(r + 1) \geq m(r + 1) \geq r + 1 > 1$. This means that also $s(\sigma) - \alpha_r$ is going to infinity with order $q_1(r + 1)$. But $q_1(r + 1) > t(r + 1) > t - r$, which is a contradiction.

2. One between q_1 and q_2 is equal to t, and the other is smaller than t. W.l.o.g. $q_1 = t$ and $q_2 < t$. In particular, we must have $Uc_1 = b$. Then we get:

$$s(\sigma) = b^{r+1}\left(\sum_{j=1}^h u_j^r u_{h+j}\right)\delta^{t(r+1)} + o\left(\delta^{t(r+1)}\right).$$

   Again, $b \neq 0$ by hypothesis, and $\sum_{j=1}^h u_j^r u_{h+j}$ is non-zero for a generic 2-plane. Hence $s(\sigma)$ is going to infinity of degree $t(r + 1) \geq m(r + 1) \geq r + 1 > 1$, and so does $s(\sigma) - \alpha_r$. But $t(r + 1) > t - r$, which is a contradiction.

3. $q_1 = q_2 = t$ and $Uc_1 + Vc_2 = b$. Then we get:

$$s(\sigma) = \left(\sum_{j=1}^h (u_j c_1+v_j c_2)^r (u_{h+j}c_1+v_{h+j}c_2)\right)\delta^{t(r+1)} + o\left(\delta^{t(r+1)}\right). \quad (9)$$

Since c_1 and c_2 depend on the generic 2-plane, in this case is more difficult to understand if $s(\sigma)$ is of degree $t(r+1)$ at infinity. Using the notation introduced in this section, we have $(r+1)\overline{s} = x\overline{s}_x + y\overline{s}_y$. Notice that the coefficient of $\delta^{t(r+1)}$ in (9) is equal to $D := \overline{s}(c_1, c_2) = \frac{1}{r+1}\left(c_1\overline{s}_x(c_1, c_2) + c_2\overline{s}_y(c_1, c_2)\right)$. By hypothesis $V\frac{\partial s}{\partial x}(\sigma_1, \sigma_2) = U\frac{\partial s}{\partial y}(\sigma_1, \sigma_2)$. In particular, the limit behaviour of $V\frac{\partial s}{\partial x}(\sigma_1, \sigma_2)$ and $U\frac{\partial s}{\partial y}(\sigma_1, \sigma_2)$ is the same. This means that their leading coefficients must be equal, or, equivalently, $V\overline{s}_x(c_1, c_2) = U\overline{s}_y(c_1, c_2)$. Hence $UD = \frac{1}{r+1}\left(Uc_1 + Vc_2\right)\overline{s}_x(c_1, c_2)$. Since $c_2 \neq 0$ by hypothesis and $U \neq 0$ for a generic 2-plane, we also have $D = \frac{1}{r+1}\left(\frac{c_1}{c_2} + \frac{V}{U}\right)\tilde{s}_x\left(\frac{c_1}{c_2}\right)$ and $\tilde{s}_y\left(\frac{c_1}{c_2}\right) = \frac{V}{U}\tilde{s}_x\left(\frac{c_1}{c_2}\right)$. Notice that $\frac{c_1}{c_2} + \frac{V}{U} \neq 0$ because $Uc_1 + Vc_2 = b \neq 0$ by hypothesis. $\tilde{s}_x\left(\frac{c_1}{c_2}\right) = 0$ would imply that also $\tilde{s}_y\left(\frac{c_1}{c_2}\right) = 0$, which is not possible for a generic 2-plane by Lemma C.4. Hence $D \neq 0$ and $s(\sigma) = D\delta^{t(r+1)}$. Hence, $s(\sigma) - \alpha_r$ is going to infinity of degree $t(r+1)$ too. But $t(r+1) > t - r$, which is a contradiction.

We have taken care of the case $t > m$ and the case $t = m$ with $a \neq \alpha_m$. Hence, it remains to show that it is not possible to have $t < m$ for a generic 2-plane. If we assume $t < m$, then $B(\sigma) - \alpha_0 \sim \alpha_m\delta^m$, by (8). By proceeding analogously to the previous case, we find that $s(\sigma) - \alpha_r$ goes to infinity with order greater than $m(r+1) > 0 > t - m$, which is a contradiction. □

We can now conclude the proof of Theorem 4.4. We consider two cases.

1. *There exists $M \in \mathbb{R}$ such that, for any $\delta > M$, $\frac{\partial s}{\partial x}(\sigma)V - \frac{\partial s}{\partial y}(\sigma)U \neq 0$.* By (4) we get:

$$\begin{cases} (s(\sigma)-\alpha_r)(NP_{2r}-P_r^2) = \sum\limits_{\substack{j=1 \\ j\neq r}}^{d} \alpha_j(NP_{r+j}-P_jP_r) \\ (B(\sigma)-\alpha_0)(NP_{2r}-P_r^2) = \sum\limits_{\substack{j=1 \\ j\neq r}}^{d} \alpha_j(P_{2r}P_j-P_{r+j}P_r) \end{cases}. \tag{10}$$

Moreover, for any integer $0 < j \leq d$, $j \neq r$, we have:

$$\begin{aligned} NP_{r+j} - P_jP_r =& rj\left(N\sum_{n=1}^{N} i_n^2 - \left(\sum_{n=1}^{N} i_n\right)^2\right)\delta^{j+r-2} + \frac{rj(j+r-2)}{2}\left(N\sum_{n=1}^{N} i_n^3 - \sum_{n=1}^{N} i_n\sum_{n=1}^{N} i_n^2\right)\delta^{j+r-3} \\ &+ \frac{rj}{12}\left(\mu_1 N\sum_{n=1}^{N} i_n^4 - 2\mu_2\sum_{n=1}^{N} i_n\sum_{n=1}^{N} i_n^3 - 3\mu_3\Big(\sum_{n=1}^{N} i_n^2\Big)^2\right)\delta^{j+r-4} + o\left(\delta^{j+r-4}\right) \end{aligned},$$

and

$$\begin{aligned} P_{2r}P_j - P_{r+j}P_r =& (r^2 - rj)\left(N\sum_{n=1}^{N} i_n^2 - \left(\sum_{n=1}^{N} i_n\right)^2\right)\delta^{j+2r-2} + \frac{(r^2-rj)(j+2r-2)}{2}\left(N\sum_{n=1}^{N} i_n^3 - \sum_{n=1}^{N} i_n\sum_{n=1}^{N} i_n^2\right)\delta^{j+2r-3} \\ &+ \frac{r}{12}\left(\eta_1 N\sum_{n=1}^{N} i_n^4 + 2\eta_2\sum_{n=1}^{N} i_n\sum_{n=1}^{N} i_n^3 + 3\eta_3\Big(\sum_{n=1}^{N} i_n^2\Big)^2\right)\delta^{j+2r-4} + o\left(\delta^{j+2r-4}\right) \end{aligned},$$

where $\mu_1 := 3rj + 2r^2 + 2j^2 - 9r - 9j + 11$, $\mu_2 := j^2 + r^2 - 3j - 3r + 4$, $\mu_3 := jr - j - r + 1$, $\eta_1 := 7r^3 - 18r^2 + 11r - 3rj^2 - 2r^2j - 2j^3 + 9rj + 9j^2 - 11j$, $\eta_2 := -2r^4 + 4r^3 + 6r^3j - 4r^2 - 5r^2j - r^2j^2 - 3rj^2 + 4rj + rj^3$, and $\eta_3 := -r^4 + 2r^3 - 2r^3j - r^2 - r^2j + 3r^2j^2 + rj - rj^2$. Since $\mathcal{T}$ has at least 2 distinct points, $N\sum_{n=1}^{N} i_n^2 \neq \left(\sum_{n=1}^{N} i_n\right)^2$, and hence $NP_{2r} - P_r^2$ is non-zero for a sufficiently large δ. By dividing the equations in (10) by $NP_{2r} - P_r^2$ we obtain expressions for $s(\sigma) - \alpha_r$ and $B(\sigma) - \alpha_0$ respectively, depending on δ. Thus, we get a new expression for the error:

$$f(\sigma) = \frac{1}{N}\sum_{k=1}^{N}\left(\sum_{j=1, j\neq r}^{d} \frac{\alpha_j\omega_{k,j}}{NP_{2r} - P_r^2}\right)^2, \tag{11}$$

where $\omega_{k,j} := p_{k,r}(NP_{r+j} - P_jP_r) + (P_{2r}P_j - P_{r+j}P_r) - p_{k,j}(NP_{2r} - P_r^2) \in \mathbb{R}[\delta]$. Notice that, if g is representable by $\mathcal{N}$, then $f(\sigma) = 0$ because the sum over j has no terms. By performing some computations, for any $1 \leq k \leq N$ and any $0 < j \leq d$, $j \neq r$, we obtain that $\omega_{k,j} = \Omega\delta^{2r+j-4} + o\left(\delta^{2r+j-4}\right)$, where

$$\Omega = \frac{r^2 j(r-j)}{2}\left(N\sum_{n=1}^{N} i_n^2 - \left(\sum_{n=1}^{N} i_n\right)^2\right) i_k^2 - \frac{r^2 j(r-j)}{2}\left(N\sum_{n=1}^{N} i_n^3 - \sum_{n=1}^{N} i_n \sum_{n=1}^{N} i_n^2\right) i_k + \frac{r}{12}\left(\xi_1 N \sum_{n=1}^{N} i_n^4 + 2\xi_2 \sum_{n=1}^{N} i_n \sum_{n=1}^{N} i_n^3 + 3\xi_3\Big(\sum_{n=1}^{N} i_n^2\Big)^2\right),$$

where $\xi_1, \xi_2, \xi_3 \in \mathbb{Z}[r, j]$. We focus on the coefficient of the leading term of $\omega_{k,j}$. Notice that the term multiplying i_k is non-zero, by the Cauchy-Schwarz inequality. Then, since $\mathcal{T}$ has at least 3 distinct elements, for each $0 < j \leq d$, $j \neq r$, there exists at least one $1 \leq k \leq N$ such that the coefficient of the degree $2r + j - 4$ term of $\omega_{k,j}$ is non-zero. Hence there exists a $k \in [1, N]$ such that $\sum_{j=1, j\neq r}^{d} \frac{\alpha_j \omega_{k,j}}{NP_{2r} - P_r^2} \sim c\delta^{m-2}$ as $\delta \to +\infty$, for some multiplicative non-zero real constant $c \in \mathbb{R}^*$. By (11), f is asymptotic to $\delta^{2(m-2)}$, up to a non-zero real constant. This concludes the first case.

2. *For any $M \in \mathbb{R}^+$, there exists $I > M$ such that $\frac{\partial s}{\partial x}(\sigma(I))V - \frac{\partial s}{\partial y}(\sigma(I))U = 0$.* By the Identity Theorem, we get $\frac{\partial s}{\partial x}(\sigma)V - \frac{\partial s}{\partial y}(\sigma)U = 0$ on D. If $\lim_{\delta\to+\infty} \frac{\partial s}{\partial x}(\sigma) \in \mathbb{A}^1_{\mathbb{C}}$, then also $\lim_{\delta\to+\infty} \frac{\partial s}{\partial y}(\sigma)$ is finite. By Lemma C.8, both $\lim_{\delta\to+\infty} s(\sigma)$ and $\lim_{\delta\to+\infty} B(\sigma)$ are finite. Furthermore, by Lemma C.6, $s(\sigma) - \alpha_r$ does not tend to 0 as $\delta \to +\infty$. Hence, by (2), $f(\sigma)$ goes to $+\infty$ with order $2\max\{d, r\}$ as $\delta \to +\infty$. In particular, if $d > r$ then $f(\sigma) \sim \alpha_d^2 \delta^{2d}$, else if $d \leq r$ then $f(\sigma) \sim c^2\delta^{2r}$, where $c := \lim_{\delta\to+\infty} s(\sigma) - \alpha_r \in \mathbb{C}^*$.

   If $\frac{\partial s}{\partial x}(\sigma)$ goes to infinity, then clearly $\frac{\partial s}{\partial y}(\sigma)$ tends to infinity too. Then from the first equation in (3) we obtain

$$(B(\sigma) - \alpha_0)\left(\frac{\partial s}{\partial x}(\sigma)P_r + NU\right) = -(s(\sigma) - \alpha_r)\left(\frac{\partial s}{\partial x}(\sigma)P_{2r} + UP_r\right) + \sum_{j=1, j\neq r}^{d} \alpha_j \left(\frac{\partial s}{\partial x}(\sigma)P_{r+j} + UP_j\right). \quad (12)$$

   Since $\frac{\partial s}{\partial x}(\sigma)$ tends to infinity, $\frac{\partial s}{\partial x}(\sigma)P_r + NU \neq 0$ for a sufficiently large δ. Hence, from (12) we get an expression for $B(\sigma) - \alpha_0$ that we can substitute in the error f to obtain:

$$f(\sigma) = \frac{1}{N}\sum_{k=1}^{N}\left(\frac{\lambda_k}{\Lambda}\right)^2,$$

where

$$\Lambda := \frac{\partial s}{\partial x}(\sigma)P_r + NU$$
$$\lambda_k := (s(\sigma) - \alpha_r)\left(p_{k,r}\Lambda - \frac{\partial s}{\partial x}(\sigma)P_{2r} + UP_r\right) - \sum_{\substack{j=1 \\ j\neq r}}^{d} \alpha_j \left(p_{k,j}\Lambda - \frac{\partial s}{\partial x}(\sigma)P_{r+j} + UP_j\right).$$

   Then for any $k \in [1, N]$ and for any $j \in [1, d], j \neq r$ we have

$$\begin{aligned} p_{k,j}\Lambda - \frac{\partial s}{\partial x}(\sigma)P_{r+j} + UP_j &= \frac{\partial s}{\partial x}(\sigma)\left(j\left(Ni_k - \sum_{n=1}^{N} i_n\right)\delta^{r+j-1} + o\left(\delta^{r+j-1}\right)\right) \\ &\quad + U\left(j\left(Ni_k - \sum_{n=0}^{N} i_n\right)\delta^{j-1} + o\left(\delta^{j-1}\right)\right) \\ &= j\left(Ni_k - \sum_{n=1}^{N} i_n\right)\frac{\partial s}{\partial x}(\sigma)\delta^{r+j-1} + \frac{\partial s}{\partial x}(\sigma)o(\delta^{r+j-1}), \end{aligned}$$

where the last equality holds because $\frac{\partial s}{\partial x}(\sigma)$ goes to infinity by our assumption. By Lemma C.9, $s(\sigma)-\alpha_r \sim \alpha_m \delta^{m-r}$. Hence, for any $k \in [1,N]$ we have $\lambda_k \sim (r-m)\left(Ni_k - \sum_{n=1}^N i_n\right)\alpha_m \frac{\partial s}{\partial x}(\sigma)\delta^{m+r-1}$. Notice that $m \neq r$ by definition. We conclude that

$$f(\sigma) \sim \alpha_m^2 \frac{(r-m)^2}{N} \sum_{k=1}^N \left(i_k - \frac{1}{N}\sum_{n=1}^N i_n\right)^2 \delta^{2(m-1)},$$

as desired.

This concludes the proof of Theorem 4.4.

C.1. Observation on the proof of Theorem 4.4

Notation C.10. For $n \in [1,N]$, we denote by $p_{n,r}(\delta) := (i_n+\delta)^r$ the degree r polynomial in δ derived from the translation of the dataset $\mathcal{T}$ by δ, as in Definition 4.3. Moreover, for any $j \in \mathbb{N}$, let $P_j := \sum_{n=1}^N p_{n,j}$.

From the proof of Theorem 4.4, it is clear that the points of $S_{\mathcal{T}_\delta}$ satisfying 3. in Theorem 4.4 come from the evaluation of the restricted error function f at the points of the zero locus of $\mathbb{A}^2_{\mathbb{C}}$, defined by

$$\begin{cases} 0 = & (s(x,y)-\alpha_r)\mathbf{P}_{r,r} - \sum\limits_{\substack{j=1 \\ j\neq r}}^{d} \alpha_j \mathbf{P}_{j,r} \\ 0 = & (B(x,y)-\alpha_0)\mathbf{P}_{r,r} - \sum\limits_{\substack{j=1 \\ j\neq r}}^{d} \alpha_j \mathbf{P}_{r,r+j} \end{cases}, \tag{13}$$

where $\mathbf{P}_{i,k} := P_{i+r}P_{i-r} - P_iP_k$. The remaining points of $S_{\mathcal{T}_\delta}$, i.e. the ones verifying 1. or 2. in Theorem 4.4, come from the evaluation of f at the points of the following zero locus of $\mathbb{A}^2_{\mathbb{C}}$, defined by

$$\begin{cases} 0 = V\frac{\partial s}{\partial x}(x,y) - U\frac{\partial s}{\partial y}(x,y) \\ 0 = (B(x,y)-\alpha_0)\,\mathcal{P}_0 + (s(x,y)-\alpha_r)\,\mathcal{P}_r - \sum\limits_{\substack{j=1 \\ j\neq r}}^{d} \alpha_j \mathcal{P}_j \end{cases}, \tag{14}$$

where $\mathcal{P}_i(x,y) := \frac{\partial s}{\partial x}(x,y)P_{i+r} + UP_i$. In particular, to determine which condition, 1. or 2. holds, we have to look at finiteness of the limit of the partial derivatives (more details can be found in the proof of Theorem 4.4). Notice that, in general, (13) has $r+1$ points, and (14) has $r(2r+1)$ points. These numbers agree with the cardinality of the clusters found in Examples 4.1 and 4.2.