Abstract. We study the family of functions that are represented by a linear convolutional neural network (LCN). These functions form a semi-algebraic subset of the set of linear maps from input space to output space. In contrast, the families of functions represented by fully-connected linear networks form algebraic sets. We observe that the functions represented by LCNs can be identified with polynomials that admit certain factorizations, and we use this perspective to describe the impact of the network’s architecture on the geometry of the resulting function space. We further study the optimization of an objective function over an LCN, analyzing critical points in function space and in parameter space, and describing dynamical invariants for gradient descent. Overall, our theory predicts that the optimized parameters of an LCN will often correspond to repeated filters across layers, or filters that can be decomposed as repeated filters. We also conduct numerical and symbolic experiments that illustrate our results and present an in-depth analysis of the landscape for small architectures.

Key words. Function space description of neural networks, linear network, Toeplitz matrix, circulant matrix, algebraic statistics, Euclidean distance degree, semi-algebraic set, gradient flow, discriminant, critical point, tensor.

AMS subject classifications. 68T07, 14P10, 14J70, 90C23, 62R01.

1 Introduction

A neural network is a parameterized family of functions. The class of representable functions, also known as the neuromanifold or function space, is determined by the architecture of the network. Optimizing an objective function over such a parametrized set usually depends on both the set and the parametrization. The objective function is typically convex in function space but non-convex in parameter space. This observation has guided numerous recent works on parameter optimization in neural networks, see, e.g., [23, 16, 1, 17]. However, the geometry of function space is not well understood in general. Fully-connected networks, also known as dense networks, have been studied in significantly more detail than convolutional networks, which are the topic of this article. Whereas weight matrices in fully-connected networks have independent and unconstrained entries, weight matrices in convolutional networks are sparse and have repeated entries.

Neural networks with linear activation functions, called linear networks, represent linear functions as compositions of linear functions. An important motivation for studying such networks is to better...
understand the effects of the parameterization when optimizing an objective function. We will see that it also can help us better understand the effects that the geometry of function space has on the optimization problem. Although neural networks are usually presented in a parameterized form, it is in principle possible to characterize the representable functions implicitly as the solutions to certain constraints. Such descriptions have been particularly fruitful in algebraic statistics, e.g., [34, 32, 2, 12, 13]. In the case of fully-connected linear networks, the function space is characterized by rank constraints depending on the layer widths. A study of critical points and local optimizers in parameter and function space for fully-connected linear networks has been carried out in [35]. Here we follow a similar perspective, analyzing the function space and the optimization of parameters for linear convolutional networks (LCNs).

**Main results** In Section 2 we introduce LCNs and present our general setup. Each layer of an LCN is represented by a convolutional matrix or, more generally, a convolutional tensor. We observe that composing convolutional layers corresponds to polynomial multiplication.

In Section 3 we recall some general properties of discriminants and multiplicity root patterns for polynomials, which we use in later sections.

In Section 4 we show that the function space of an LCN (i.e., the set of linear maps that can be represented by the network) is a semi-algebraic set, whose natural ambient space is the vector space of convolutional tensors of a certain format. In contrast to fully-connected linear networks, whose function space is cut out by polynomial equations only, the function space of LCNs is determined by polynomial equations and inequalities. For one-dimensional convolutions with stride one, the function space is a full-dimensional subset of its natural ambient space and its boundary is contained in the discriminant hypersurface of univariate polynomials of a certain degree. The function space is equal to its natural ambient space if and only if at most one of the filters has even size. The latter case is interesting since optimizing a convex loss in function space reduces to a convex optimization problem on a vector space. For higher-dimensional convolutions or larger strides, the function space is typically a lower-dimensional subset of its natural ambient space.

In Section 5 we study the optimization problem for a general loss and one-dimensional convolutions with stride one. Different LCNs can have the same function space, but different parameterizations can influence the optimization behavior. Hence, we distinguish pure critical points given by the optimization on function space from spurious critical points induced only by the parameterization. We show that the critical points in parameter space are either global minima or correspond to polynomials with repeated roots. The latter actually correspond to critical points of the loss in function space restricted to subsets of polynomials with particular real root multiplicity patterns that we characterize combinatorially depending on the LCN architecture. We further describe the gradient flow dynamics in parameter space and provide invariants that allow us to formulate it as a local Riemannian gradient in function space.

In Section 6 we take a closer look at the square loss. We provide upper bounds on the number of critical points in function space in terms of Euclidean distance degrees, and observe that for certain normalizations of the training data, the square loss becomes the Bombieri norm, which is expected to have fewer critical points.

In Section 7 we conduct numerical experiments illustrating our theoretical results. These demonstrate how the type of training data, the geometry of function space, and the parametrization lead to different types of critical points and solutions found using gradient descent.

**Overview of previous works on linear networks** Loss surface. The properties of the optimization problem in linear networks have been studied in [8, 7, 9]. For the square loss in fully-connected linear networks there is a unique local and global minimum, up to equivalence, and all other critical points are saddles. A proof of this statement for the deep case was given in [24]. For deep residual linear networks, [21] showed that the square loss has no critical points other than global minima. Further, [29]
and [37] argue that depth alone does not create bad local minima, although it induces a non-convex loss surface. The analytic form of the critical points for the square loss in fully-connected linear networks was discussed in [38]. For arbitrary differentiable convex losses and fully-connected linear networks having layers at least as wide as the input or output layers, [26] showed that all local minima are global. A geometric analysis was given in [35], showing that the absence of non-global local minima in linear networks is expected for arbitrary smooth convex losses only if the architecture can express all linear maps (as in [26]) or for the quadratic loss and arbitrary architectures (as in [24]). Taking an algebraic standpoint, [30] studied fully-connected linear networks and upper bounds on the number of critical points of the square loss with weight norm regularization. The above works focus mostly on fully-connected networks. A paper dating back to 1989 [7], asked whether non-global local minima for the square loss exist in the case non-fully-connected (i.e., locally-connected) multi-layer linear networks, and this seems to have remained an open question. We show that LCNs can indeed have non-trivial local minima in function space and in parameter space.

Gradient dynamics. An overview of non-convex optimization in low-rank matrix factorization (two-layer linear networks) was given in [14]. Exact solutions to the nonlinear gradient dynamics for the square loss in fully-connected deep linear networks with orthogonal input data have been obtained in [31]. Also for the square loss in fully-connected linear networks, [4] showed linear convergence to a global minimum when the hidden layers are at least as wide as the input or output layers, weight matrices at initialization are approximately balanced, and the initial loss is less than for any rank deficient solution. The work [5] shows that depth amounts to a pre-conditioner which may accelerate convergence. A detailed analysis of gradient flow optimization and convergence for the square loss in fully-connected linear networks was presented in [6]. Using a balancedness condition they obtain a Riemannian gradient in function space and show that the flow always converges to a critical point, which for almost all initializations is the global optimum in function space restricted to the set of rank-$k$ matrices for some $k$. We characterize gradient flow invariants in LCNs and obtain corresponding descriptions in function space. The critical points can also be characterized in function space, in terms of real root multiplicity patterns rather than rank constraints.

Linear convolutional networks and implicit bias of gradient descent. LCNs have been studied in [20] for the case of 1D convolutions with circulant weight matrices (with stride 1) having full-width filters and a single output. Such networks can express any linear map. That work focuses on binary classification with linearly separable data and asks which of the infinitely many linear classifiers with zero loss are obtained by gradient descent. For the exponential loss, it is shown that for fully-connected networks gradient descent converges to the hard margin support vector machine linear classifier independently of the depth of the network (as was previously observed for the direct parameterization in [33]), whereas for convolutional networks it is biased towards linear classifiers that are sparse in frequency domain depending on the depth. In contrast, we consider LCNs with arbitrary filter sizes (and also provide first geometric insights for LCNs with larger strides or higher-dimensional convolutions). As we will see, the filter sizes play an important role. Our results suggest that gradient descent in LCNs will be biased towards solutions with repeated filters (or repeated factors of filters), even when these are not global optimizers of the training objective.

**Notation** Throughout the paper, we use zero-based indices for our vectors and matrices. For $n \in \mathbb{Z}_{\geq 0}$, we write $[n] = \{0, \ldots, n-1\}$. We write $\mathbb{R}[x,y]_d$ (resp. $\mathbb{C}[x,y]_d$) for the space of bivariate homogeneous polynomials of degree $d$ with real (resp. complex) coefficients.
2 Linear convolutional networks

A feedforward neural network is a family of functions $f : \mathbb{R}^{d_0} \to \mathbb{R}^{d_L}$ that are compositions of linear and nonlinear maps:

$$f(x) = (\alpha_L \circ \rho \circ \alpha_{L-1} \circ \rho \circ \ldots \circ \rho \circ \alpha_1)(x).$$  \hfill (1)

Here $\alpha_l : \mathbb{R}^{d_{l-1}} \to \mathbb{R}^{d_l}$ is an affine map and each $\rho : \mathbb{R}^{d_{l}} \to \mathbb{R}^{d_{l}}$ is a (typically nonlinear) activation map that acts on each element of a vector. The composition of consecutive affine and activation maps is called a layer of the network. The coefficients of the affine functions $\alpha_i$ are called the weights and biases of the network and serve as trainable parameters.

A linear network is a feedforward network where the activation function $\rho$ is the identity. In this setting, the end-to-end map $f$ in eq. (1) is affine, however the parameterization of the representable functions provided by the network’s weights is nonlinear. Thus, linear networks preserve some of the nonlinear aspects of general feedforward networks while being theoretically more tractable.

A linear convolutional network (LCN) is a linear network with the additional constraint that the affine functions in eq. (1) have a convolutional structure. The convolutional structure can be seen as constraining general affine maps to satisfy certain linear equations, corresponding to conditions of restricted connectivity (some weight coefficients must be zero) and weight sharing (some weight coefficients must be repeated). Although an LCN function $f$ can also be seen as a particular fully-connected linear network, the family of all representable functions using an LCN architecture is a strict subset of the family of representable functions using a fully-connected architecture. In particular, the study of LCNs cannot be reduced to the study of general linear networks. We will point out several important differences between the geometry and optimization properties of LCNs and those of fully-connected networks.

In this paper, we consider convolutional networks with a single filter per layer and no biases. For most of our discussion, we focus on convolutions applied to one-dimensional signals, which means that the input $x$ of the network function is a vector (we discuss the case where $x$ is an image or higher-dimensional in Section 4.3, but we defer general definitions to that section). In this setting, a convolution is a map $\alpha_l : \mathbb{R}^{d_{l-1}} \to \mathbb{R}^{d_l}$ of the form

$$\left(\alpha_l(x)\right)_i = \sum_{j \in [k_l]} w_{l,j} \cdot x_{i s_l + j},$$  \hfill (2)

where $w_l = (w_{l,0}, \ldots, w_{l,k_l-1}) \in \mathbb{R}^{k_l}$ is a filter of width $k_l$ and $s_l \in \mathbb{Z}_{>0}$ is the stride associated with the convolution.\footnote{The expression in eq. (2) is actually the cross-correlation between $x$ and $w_l$ or equivalently the convolution of $x$ with the adjoint filter $w_l^* = (w_{l,k_l-1}, \ldots, w_{l,0})$. We refer to it simply as a “convolution” following common machine learning terminology. Exchanging all filters with their adjoints would have no effect on our discussion.} We are assuming “no padding” in the way the boundaries of the domain are handled. This tacitly assumes that input dimension and stride fit with the output dimension. We compare this approach with using “circular padding” in Section 2.3 below.

The linear map $\alpha_l$ in eq. (2) can be represented as a generalized Toeplitz matrix (generalized in the sense that standard Toeplitz matrices would correspond to stride $s_l = 1$). For example, for stride $s_l = 2$, filter size $k_l = 3$, and input size $d_{l-1} = 7$, the matrix takes the form

$$W_l = \begin{bmatrix} w_{l,0} & w_{l,1} & w_{l,2} \\ w_{l,0} & w_{l,1} & w_{l,2} \\ w_{l,0} & w_{l,1} & w_{l,2} \end{bmatrix}.$$  \hfill (3)

We will call generalized Toeplitz matrices convolutional matrices. For convolutional matrices as in eq. (3), the input and output sizes $d_{l-1}$, $d_l$, filter size $k_l$, and stride $s_l$ are related by

$$d_l = \frac{d_{l-1} - k_l}{s_l} + 1.$$  \hfill (4)
An \(L\)-layer LCN function is given by \(f(x) = \overline{W}x\), where \(\overline{W} \in \mathbb{R}^{d_L \times d_0}\) can be written as \(\overline{W} = W_LW_{L-1} \cdots W_2W_1\) and each \(W_i \in \mathbb{R}^{d_i \times d_{i-1}}\) is a convolutional matrix. In this work, we study the function space, i.e., the set of linear maps that can be represented by an LCN as the parameters vary:

\[
\mathcal{M}_{d,k,s} = \left\{ W \in \mathbb{R}^{d_L \times d_0} : W = \prod_{l=1}^{L} W_l, W_l \in \mathbb{R}^{d_l \times d_{l-1}} \text{ convolutional} \right\}.
\]

Here \(d = (d_0, \ldots, d_L)\) denotes the dimensions, \(k = (k_1, \ldots, k_L)\) the filter sizes, and \(s = (s_1, \ldots, s_L)\) the strides of the \(L\) layers. We refer to the triple \((d, k, s)\) as the architecture of the LCN. We tacitly assume in the following that the dimensions \(d\) are compatible by satisfying eq. (4). We also note that by eq. (4) the function space \(\mathcal{M}_{d,k,s}\) is fully determined by fixing \(d_L, k, \) and \(s\). As such, we may omit any of the indices \(d, k\) and \(s\) when clear from the context. The function space \(\mathcal{M}_{d,k,s}\) is the image of the parameterization map

\[
\mu_{d,k,s} : \mathbb{R}^{k_1} \times \cdots \times \mathbb{R}^{k_L} \rightarrow \mathcal{M}_{d,k,s} : (w_1, \ldots, w_L) \mapsto \overline{W},
\]

that takes \(L\) filters to the product of their associated convolutional matrices.

**Remark 2.1.** Although we use \(\mathcal{M}_{d,k,s}\) to denote the function space of LCNs, the sequence of dimensions \(d\) does not play an essential role. More precisely, since the product of convolutional matrices is again a convolutional matrix (see Theorem 2.2), we can also define the parameterization map eq. (5) at the level of filters \(\mu_{k,s} : \mathbb{R}^{k_1} \times \cdots \times \mathbb{R}^{k_L} \rightarrow \mathbb{R}^{k}; (w_1, \ldots, w_L) \mapsto \overline{w}\), where \(\overline{w}\) is the filter of the product convolutional matrix.

### 2.1 Compositions of convolutional matrices

It is well known that the composition of convolution operations is again a convolution. More precisely, for two convolutional maps \(\alpha_1 : \mathbb{R}^{d_0} \rightarrow \mathbb{R}^{d_1}\) and \(\alpha_2 : \mathbb{R}^{d_1} \rightarrow \mathbb{R}^{d_2}\) with filters \(w_1\) and \(w_2\) respectively and satisfying (4), the composition is given by

\[
(\alpha_2 \circ \alpha_1)(x)_i = \sum_{j \in [k_1]} w_{2,j} \cdot \alpha_1(x)_{i+k_2+j} = \sum_{j \in [k_1]} w_{2,j} \cdot \sum_{\ell \in [k_2]} w_{1,\ell} \cdot x_{(i+k_2+j)s_1+\ell} = \sum_{m \in [(k_2-1)s_1+k_1]} u_m \cdot x_{is_2s_1+m},
\]

(6)

Here the resulting filter \(u\) has size \((k_2-1)s_1 + k_1\) and has entries

\[
u_m = \sum_{j \in [k_2], \ell \in [k_1]} w_{2,j}w_{1,\ell}.
\]

(7)

**Proposition 2.2.** The composition of \(L\) convolutions with filter sizes \(k = (k_1, \ldots, k_L)\) and strides \(s = (s_1, \ldots, s_L)\) is a convolution with filter size \(k = k_1 + \sum_{l=2}^{L}(k_l - 1) \prod_{m=1}^{l-1} s_m\) and stride \(s = \prod_{m=1}^{L} s_m\). If all strides \(s_m\) are equal to \(s\), then \(k = d_0 - (d_L - 1)s\) and \(s = s^L\).

**Proof.** The expression in eq. (6) shows that the composition of two convolutions with filter sizes \(k_1, k_2\) and strides \(s_1, s_2\) is again a convolution, with filter size \(k = k_1 + (k_2 - 1)s_1\) and stride \(s_2s_1\). If the strides \(s_1, s_2\) are equal to \(s\), then by eq. (4), we have \(k_1 = d_0 - (d_1 - 1)s\) and \(k_2 = d_1 - (d_2 - 1)s\), which yields \(k = k_1 + (k_2 - 1)s = d_0 - (d_2 - 1)s^2\). The statement then follows by induction, whereby at the \(l\)-th iteration we multiply the convolution represented up to layer \(l - 1\) with the \(l\)-th layer convolution. \(\square\)
Consider the LCN architecture with equations. For LCNs, the matrix sizes are decreasing by eq. (4), and such rank constraints do not occur. The space of representable matrices are rank constraints (vanishing of certain minors, which are polynomial equalities and inequalities). Below we give examples of filling and non-filling LCN architectures. For fully-connected linear networks, the only constraints on the corresponding function space is simply convex optimization in a vector space. We emphasize that the vector space $M$ in Theorem 2.2.

**Example 2.5.** Consider the LCN with $d = (3, 2, 1)$, $k = (2, 2)$, $s = (1, 1)$. The corresponding function space $\mathcal{M} \subset \mathbb{R}^{1 \times 3}$ consists of matrices

$$W = W_2W_1 = \begin{bmatrix} c & d \\ 0 & a \\ b & 0 \end{bmatrix} = \begin{bmatrix} ac & ad + bc & bd \\ 0 & a & b \\ d & c \end{bmatrix} =: \begin{bmatrix} A & B & C \end{bmatrix}.$$

We can see that this architecture is not filling, i.e., that the containment $\mathcal{M} \subset \mathbb{R}^{1 \times 3}$ is strict. Indeed, the entries in $W$ are given by the coefficients $(A, B, C)$ of quadratic polynomials that factor into two real linear forms:

$$(ax + b)(cx + d) = acx^2 + (ad + bc)x + bd = Ax^2 + Bx + C.$$

These correspond to quadratic equations with one or more real roots (or linear equations with $A = 0$). Hence the set of representable coefficients $(A, B, C)$ is characterized by the discriminant condition $B^2 - 4AC \geq 0$. See Figure 1 (left).

**Example 2.5.** Consider the LCN with $d = (5, 3, 2)$, $k = (3, 2)$, $s = (1, 1)$. Then

$$W = W_2W_1 = \begin{bmatrix} d & e & 0 \\ 0 & d & e \\ a & b & c \end{bmatrix} = \begin{bmatrix} A & B & C & D & 0 \\ 0 & A & B & C & 0 \end{bmatrix},$$

where $(A, B, C, D) = (ad, bd + ae, cd + be, ce)$. The set $\mathcal{M}$ is the set of all possible $W$ where its entries are related by the above equation as $(a, b, c, d, e)$ vary over $\mathbb{R}^5$. Rephrasing this question as which real cubic polynomials factor into a real quadratic and linear factor, i.e.,

$$(ax^2 + bx + c)(dx + e) = adx^3 + (bd + ae)x^2 + (cd + be)x + ce = Ax^3 + Bx^2 + Cx + D,$$

one can immediately draw on the fundamental theorem of algebra to conclude that every choice of $(A, B, C, D)$ has a corresponding choice of $(a, b, c, d, e)$. From this, we note that $\mathcal{M}$ is filling, so $\mathcal{M}(5,3,2),(3,2),(1,1) = \mathcal{M}(5,2),4,1$.

### 2.2 Polynomial multiplication

The previous examples illustrate a useful identification between convolutional matrices and polynomials. More precisely, we associate a filter $w \in \mathbb{R}^k$ with the polynomial

$$\pi(w) = w_0x^{k-1} + w_1x^{k-2} + \cdots + w_{k-2}x^2y + w_{k-1}y^{k-1} \in \mathbb{R}[x, y]_{k-1}. \quad (8)$$
This is a formal polynomial, where the monomials $x^iy^j$ are used to index the weights of the filter. To better handle situations where leading coefficients vanish, it is convenient to use homogeneous bivariate polynomials instead of univariate affine polynomials as in the two previous examples. Recall that a root of a homogeneous polynomial $p(x, y)$ is an equivalence class of pairs $(r_x, r_y) \in \mathbb{C}^2 \setminus \{(0, 0)\}$ such that $p(r_x, r_y) = 0$, up to complex scalar factors (i.e., a point in $\mathbb{CP}^1$). With some abuse of notation, we will also use $\pi$ to map a convolutional matrix $W$ with filter $w$ to the polynomial in eq. (8).

**Proposition 2.6.** Consider convolutional matrices $W_1, \ldots, W_L$ with compatible sizes and stride one. Then the convolutional matrix $W = W_L \cdots W_1$ satisfies $\pi(W) = \pi(W_L) \cdots \pi(W_1)$.

**Proof.** For the product of two convolutional matrices, the statement follows from eq. (7). The general case follows by induction. $\square$

**Corollary 2.7.** Let $(d, k, s)$ be an LCN architecture of depth $L$ with $s = (1, \ldots, 1)$. Then $\pi$ identifies $\mathcal{M}_{d,k,s}$ with the set of polynomials that factor into polynomials of degrees $k_1 - 1, \ldots, k_L - 1$, that is, $\pi(\mathcal{M}_{d,k,s}) = \{p = q_L \cdots q_1 : q_i \in \mathbb{R}[x, y]_{k_i-1} \subseteq \mathbb{R}[x, y]_{k-1} \text{ where } k = \sum_{i=1}^L (k_i - 1) + 1 \text{ is the size of the end-to-end filter.}

Since the ordering of the factors does not affect the product polynomial, the function space of an LCN with stride-one 1D convolutional layers is the same for any permutation of the layers. This also holds for higher-dimensional convolutions, as we will see in Section 4.3.

**Remark 2.8.** Theorem 2.6 can be generalized to deal with arbitrary strides $s$. For example, for two convolutional matrices $W_2, W_1$ with strides $s_2, s_1$, the product $W_2W_1$ can be identified with the polynomial $p = \pi(W_2W_1) \in \mathbb{R}[x, y]$ that factors as $p = q_2q_1$, where $q_1 = \pi(W_1)$ as in eq. (8), while $q_2$ is defined as follows: if $w$ denotes the filter of $W_2$, we see from eq. (7) that

$$q_2 = w_0x^{s_1(k_2-1)} + w_1x^{s_1(k_2-2)}y^{s_1} + \cdots + w_{k_2-2}x^{s_1}y^{s_1(k_2-2)} + w_{k_2-1}y^{s_1(k_2-1)}.$$

### 2.3 Toeplitz vs. circulant matrices

Discrete convolutions over finite-dimensional data can be defined in slightly different ways depending on how they deal with the boundary of the input. Until now, we have been discussing a “no padding” setting where the convolution is not applied across the boundary, and the dimension of the output is smaller than the dimension of the input (see eq. (3)). An alternative approach would be to consider cyclic convolutions, where the input space $\mathbb{R}^d$ is viewed as representing “cyclic signals” $\mathbb{Z}/d \to \mathbb{R}$. In this section we point out that the two approaches are effectively equivalent in terms of the function space.

Let $W_1, \ldots, W_L$ be the weight matrices of an LCN. These matrices are generalized Toeplitz matrices and must have compatible sizes eq. (4). If $W_l$ has size $d_l \times d_{l-1}$ and corresponding filter $w_l$ of size $k_l$, then we construct a generalized circulant matrix of size $d_0 \times d_0$ as follows. If the stride is $s_l = 1$, the circulant matrix is

$$
\begin{bmatrix}
  w_{l,0} & w_{l,1} & \cdots & w_{l,d_0-1} \\
  w_{l,d_0-1} & w_{l,0} & \cdots & w_{l,d_0-2} \\
  \vdots & \vdots & \ddots & \vdots \\
  w_{l,1} & w_{l,2} & \cdots & w_{l,0}
\end{bmatrix},
$$

where $w_{l,i} = 0$ for $i \geq k_l$. For Toeplitz matrices with larger strides $s_l$, we can similarly define a $d_0 \times d_0$ generalized circulant matrix of stride $s_l$, where each row is obtained by shifting the previous row by $s_l$ steps.

The generalized Toeplitz matrix $W_l$ is the top-left $d_l \times d_{l-1}$ block of its associated generalized circulant matrix. Under this identification, the multiplication of generalized Toeplitz matrices with compatible
sized is the multiplication of the corresponding generalized circulant matrices. In particular, the resulting filter \( \mathbf{W} \) of the Toeplitz matrix product also results from the product of the circulant matrices. Therefore both formulations are equivalent.

Example 2.9. For the Toeplitz matrices in Theorem 2.4, the corresponding product of circulant matrices is
\[
\begin{bmatrix} c & d & 0 \\ 0 & c & d \\ d & 0 & c \end{bmatrix} \begin{bmatrix} a & b & 0 \\ 0 & a & b \\ b & 0 & a \end{bmatrix} = \begin{bmatrix} ac & ad + bc & bd \\ bd & ac & ad + bc \\ ad + bc & bd & ac \end{bmatrix}.
\]

3 Discriminants and multiple roots

In this section, we review some properties of the structure of real polynomials with repeated roots, recalling the definitions of discriminants, multiple root loci, and real root multiplicity patterns (rrmp for short). We will use these notions in the next sections to study the function space of LCNs and to investigate the critical points of the associated loss functions. A reader who is more interested in the applications of these concepts to LCNs might skip this section on a first read and return to it later to better understand the structure of repeated roots in polynomials.

The discriminant of a polynomial \( p = a_nx^n + a_{n-1}x^{n-1}y + \cdots + a_1xy^{n-1} + a_0y^n \in \mathbb{R}[x, y] \) is a polynomial in the coefficients \( a_n, \ldots, a_0 \) that vanishes if and only if \( p \) has a (complex) double root \([10, Chapter 4]\). The iterated singular loci of the discriminant hypersurface are the coincident/multiple root loci.

Definition 3.1. For a partition \( \lambda = (\lambda_1, \ldots, \lambda_r) \) of \( k-1 \), the multiple root locus \( \Delta_\lambda \subset \mathbb{R}[x, y]_{k-1} \) is the set of homogeneous polynomials that have \( r \) (not necessarily distinct) complex roots with multiplicities \( \lambda_1, \ldots, \lambda_r \). That is, \( \Delta_\lambda \) is the set of polynomials \( P \in \mathbb{R}[x, y]_{k-1} \) that can be written as \( P = Q_1^{\lambda_1} \cdots Q_r^{\lambda_r} \), where each \( Q_i \in \mathbb{C}[x, y]_1 \) is a complex linear form.

For instance, the discriminant hypersurface is \( \Delta_{(2,1,\ldots,1)} \). The singular locus of a multiple root locus is a union of certain higher-order multiple root loci; see \([25]\) for a combinatorial description. We are also interested in distinguishing between real and complex roots. This more fine-grained information is captured by the real root multiplicity pattern.

Definition 3.2. A homogeneous polynomial \( P \in \mathbb{R}[x, y]_{k-1} \) has real root multiplicity pattern, short rrmp, \( (\rho \mid \gamma) = (\rho_1, \ldots, \rho_r \mid \gamma_1, \ldots, \gamma_c) \) if it can be written as
\[
P = p_1^{\rho_1} \cdots p_r^{\rho_r} q_1^{\gamma_1} \cdots q_c^{\gamma_c}, \tag{9}
\]
where \( p_i \in \mathbb{R}[x, y]_1 \) and \( q_j \in \mathbb{R}[x, y]_2 \) are irreducible and pairwise linearly independent.

Recall that each factor \( q_j \) has two complex conjugate roots. Polynomials with rrmp \( (\rho \mid \gamma) \) are smooth points on the multiple root locus \( \Delta_{\lambda_{\rho\mid\gamma}} \), where \( \lambda_{\rho\mid\gamma} = (\rho_1, \ldots, \rho_r, \gamma_1, \ldots, \gamma_c) \).

Example 3.3. We describe the possible rrmp of a quadratic, cubic, and quartic polynomial. The discriminant polynomial \( \Delta \) of \( ax^2 + bxy + cy^2 \) distinguishes three rrmp’s:

\[
\begin{array}{ll}
11|0 & 2 \text{ distinct real roots; } \Delta = b^2 - 4ac > 0. \\
2|0 & 1 \text{ double real root; } \Delta = 0. \\
0|1 & \text{ no real roots; } \Delta < 0.
\end{array}
\]

For cubic polynomials \( ax^3 + bx^2y + cxy^2 + dy^3 \) with real coefficients, there are four rrmp’s shown below, discussed in \([3, 11, 18]\).
Following [3], a polynomial $f_n = x^n + a_1 x^{n-1} y + \cdots + a_n y^n$ can be translated to a polynomial $g_n = f_n(x - a_1 / n y, y) = x^n + b_2 x^{n-2} y^2 + \cdots + b_n y^n$. The two polynomials $f_n$ and $g_n$ have the same rrmp. For a quartic polynomial $g_4 = x^4 + px^2 y^2 + qxy^3 + ry^4$, letting
\[
\delta(p, q, r) = 256r^3 - 128r^2 p^2 + 144pq^2 r + 16p^4 r - 27q^4 - 4p^3 q^2
\]
\[
\delta'(p, q, r) = 8pr - 9q^2 - 2p^3,
\]
where $\delta$ corresponds to the discriminant, the different rrmp’s are as follows:

| $\delta(p, q, r)$ | $\delta'(p, q, r)$ |
|-------------------|-------------------|
| $0$               | $0$               |
| $1$               | $1$               |
| $2$               | $2$               |
| $3$               | $3$               |
| $4$               | $4$               |
| $5$               | $5$               |
| $6$               | $6$               |
| $7$               | $7$               |
| $8$               | $8$               |
| $9$               | $9$               |
| $10$              | $10$              |
| $11$              | $11$              |
| $12$              | $12$              |
| $13$              | $13$              |
| $14$              | $14$              |
| $15$              | $15$              |
| $16$              | $16$              |
| $17$              | $17$              |
| $18$              | $18$              |
| $19$              | $19$              |
| $20$              | $20$              |
| $21$              | $21$              |
| $22$              | $22$              |
| $23$              | $23$              |
| $24$              | $24$              |
| $25$              | $25$              |
| $26$              | $26$              |
| $27$              | $27$              |
| $28$              | $28$              |
| $29$              | $29$              |
| $30$              | $30$              |
| $31$              | $31$              |
| $32$              | $32$              |
| $33$              | $33$              |
| $34$              | $34$              |
| $35$              | $35$              |
| $36$              | $36$              |
| $37$              | $37$              |
| $38$              | $38$              |
| $39$              | $39$              |
| $40$              | $40$              |
| $41$              | $41$              |
| $42$              | $42$              |
| $43$              | $43$              |
| $44$              | $44$              |
| $45$              | $45$              |
| $46$              | $46$              |
| $47$              | $47$              |
| $48$              | $48$              |
| $49$              | $49$              |
| $50$              | $50$              |
| $51$              | $51$              |
| $52$              | $52$              |
| $53$              | $53$              |
| $54$              | $54$              |
| $55$              | $55$              |
| $56$              | $56$              |
| $57$              | $57$              |
| $58$              | $58$              |
| $59$              | $59$              |
| $60$              | $60$              |

4 Function space of linear convolutional networks

We show that the function space of an LCN is a semi-algebraic set, discuss its dimension, and whether it is filling (Theorem 2.3).

4.1 Convolutional matrices of stride one

We first focus on one-dimensional convolutions and stride one. Recall that a semi-algebraic set is a solution set to a finite set of polynomial equations and polynomial inequalities, or a finite union of such sets.

Theorem 4.1. Let $(d, k, s)$ be an LCN architecture with $s = (1, \ldots, 1)$ and depth $L$. Then $\mathcal{M}_{d,k,s}$ is a full-dimensional semi-algebraic subset of $\mathcal{M}_{(d_0,d_L),k,s}$, where $k = \sum_{i=1}^{L} k_i - L + 1$ and $s = 1$. In particular, $\mathcal{M}_{d,k,s}$ is cut out from $\mathcal{M}_{(d_0,d_L),k,s}$ by inequalities only. Moreover, the architecture is filling if and only if at most one of the filter sizes $k_1, \ldots, k_L$ is even.

This result explains the difference between Theorems 2.4 and 2.5. To prove it, we use Theorem 2.7: We identify $\mathcal{M}_{(d_0,d_L),k,s}$ with the space $\mathbb{R}[x, y]_{k-1}$ of polynomials of degree $k - 1$, and the subset $\mathcal{M}_{d,k,s}$ with the polynomials that factor as a product of polynomials of degrees $k_1 - 1, \ldots, k_L - 1$. Under this identification, we have the following.

Lemma 4.2. Let $s = (1, \ldots, 1)$. A polynomial $p \in \mathbb{R}[x, y]_{k-1}$ corresponds to a convolutional matrix in $\mathcal{M}_{d,k,s}$ if and only if $p$ has at least $e := \{k_i : k_i \text{ is even}\}$ real roots (counted with multiplicity), i.e., if the rrmp $(\rho | \gamma)$ of $p$ satisfies $\sum_{i=1}^{r} \rho_i \geq e$. 
Proof. By Theorem 2.7, it suffices to show that a polynomial \( p \) of degree \( k - 1 \) has at least \( e \) real roots if and only if \( p \) is a product of polynomials of degrees \( k_1 - 1, \ldots, k_L - 1 \). One direction is immediate. A factorization of \( p \) of the desired type has \( e \) odd-degree factors, and any polynomial \( p \) with \( e \) odd-degree factors has at least \( e \) real roots.

For the other direction, let \( e' \) denote the number of real roots of \( p \). The irreducible factors of \( p \) are \( e' \) linear terms and \( (k - 1 - e')/2 \) quadratic terms. If \( e' \geq e \), these factors can be multiplied to build a set of factors \( q_1, \ldots, q_L \) of degrees \( k_1 - 1, \ldots, k_L - 1 \) as follows. We first assign one of the \( e' \) linear terms to each of the \( e \) factors \( q_i \) of odd degree and then we use the remaining \( (e' - e) + (k - 1 - e')/2 \) irreducible factors of \( p \) to fill up the factors \( q_1, \ldots, q_L \) until their degrees are indeed \( k_1 - 1, \ldots, k_L - 1 \). Note that this procedure works as the degrees of the irreducible factors of \( p \) and the degrees of \( q_1, \ldots, q_L \) both sum up to the same value \( k - 1 \).

**Example 4.3.** When \( s = (1, \ldots, 1) \) and \( k = (k_1, \ldots, k_L) \) are chosen such that the resulting filter size is \( k = 5 \), by Theorem 4.2 the matrices in \( \mathcal{M}_{d,k,s} \) correspond to quartic polynomials with a particular real root structure, specified by the individual filter sizes \( k_i \):

1. If \( L = 2 \) and \( (k_1, k_2) = (3, 3) \), then every polynomial of degree four is in \( \pi(\mathcal{M}_{d,k,s}) \).
2. If \( L = 2 \) and \( (k_1, k_2) = (4, 2) \), then a quartic polynomial \( p \) corresponds to a matrix in the function space if and only if it has two or four real roots. Hence, the complement of the function space corresponds to all quartics with no real roots.
3. If \( L = 3 \) and \( (k_1, k_2, k_3) = (3, 2, 2) \), this architecture has the same function space as the previous one. However, the parameterization of the function space is different: Here \( \pi(\mathcal{M}_{d,k,s}) \) arises by taking the product of two linear and one quadratic factor, whereas in the previous architecture one considers the product of a linear and a cubic factor.
4. If \( L = 4 \) and \( (k_1, k_2, k_3, k_4) = (2, 2, 2, 2) \), a polynomial of degree four is in \( \pi(\mathcal{M}_{d,k,s}) \) if and only if all its roots are real.

**Proof of Theorem 4.1.** We first consider the case where all filter sizes are odd. Using Theorem 4.2, we see that \( e = 0 \), so every polynomial of degree \( k - 1 \) corresponds to a matrix in the function space \( \mathcal{M}_{d,k,s} \). If exactly one filter size is even, then \( k - 1 \) is odd. Hence, every polynomial \( p \) of degree \( k - 1 \) has at least one real root. In Theorem 4.2 we have \( e = 1 \), which shows that \( p \in \pi(\mathcal{M}_{d,k,s}) \). We conclude that the architecture is filling if at most one filter size is even. Finally, assume that two or more filter sizes are even. In this case, we consider a polynomial \( p \) of degree \( k - 1 \) that has zero (if \( k - 1 \) is even) or one (if \( k - 1 \) is odd) real root. Then \( p \) does not belong to \( \pi(\mathcal{M}_{d,k,s}) \) by Theorem 4.2. This shows that the architecture is not filling. That the function space is semi-algebraic follows from the fact that it has a polynomial parametrization (by Tarski-Seidenberg). Finally, we argue that the set of polynomials representable by a given LCN is full dimensional in \( \mathbb{R}[x, y]_{k-1} \). For this, it is enough to notice that small perturbations of the coefficients of a generic polynomial will not affect its root structure. We will discuss this in more details in the following statements.

When the architecture of an LCN is not filling, it is interesting to study the boundary of the function space.

**Proposition 4.4.** Let \( (d, k, s) \) be an LCN architecture with \( e := \left| \{ k_i : k_i \text{ is even} \} \right| \geq 2 \) and \( s = (1, \ldots, 1) \). In the Euclidean topology on \( \mathcal{M}_{(d_0, d_1), k,s} \), the function space \( \mathcal{M}_{d,k,s} \) is closed and its boundary consists of all convolutional matrices that correspond to polynomials with \( \text{rrmp} \ (\rho \mid \gamma) \) satisfying \( \sum_{i=1}^{e} \rho_i \geq e \) and \( \left| \{ \rho_i : \rho_i \text{ is odd} \} \right| \leq e - 2 \).
Proof. A sequence of polynomials, each having at least $e$ real roots counted with multiplicity, will also have at least $e$ real roots in the limit. Hence, by Theorem 4.2, the function space is closed. Using the identification $\pi$ with polynomials, the boundary of $\mathcal{M}_{d,k,s}$ consists of all polynomials in $\mathcal{M}_{d,k,s}$ that are limits of sequences of polynomials in its complement.

Given a sequence of polynomials $P^{(j)}$ of degree $(k - 1) = \sum_{i=1}^{L} (k_i - 1)$ in the complement of $\mathcal{M}_{d,k,s}$ that converges to a polynomial in $\mathcal{M}_{d,k,s}$, we may assume (by restricting to a sub-sequence) that all $P^{(j)}$ have the same rrmp $(\rho \mid \gamma)$. Since $e$ and $(k - 1)$ have the same parity and $P^{(j)} \notin \mathcal{M}_{d,k,s}$, we have $\sum_{i=1}^{L} \rho_i \leq e - 2$. In particular, the number of odd $\rho_i$ is at most $e - 2$. In the limit of the $P^{(j)}$, pairs of complex conjugate roots might become real double roots (which does not affect the number of odd $\rho_i$), or distinct real roots might become equal (which cannot increase the number of odd $\rho_i$). Hence, the limit of the $P^{(j)}$ has rrmp $(\rho' \mid \gamma')$ with at most $e - 2$ odd $\rho'_i$ and $\sum \rho'_i \geq e$ (since we assumed the limit to be in $\mathcal{M}_{d,k,s}$).

Conversely, given a polynomial $P$ with rrmp $(\rho \mid \gamma)$ satisfying $|\{\rho_i : \rho_i \text{ is odd}\}| \leq e - 2$ and $\sum_{i=1}^{L} \rho_i \geq e$, we write $\rho_i = 2\alpha_i + \beta_i$, where $\beta_i \in \{0, 1\}$. We consider its factorization as in eq. (9) and perturb each factor $p_i^{\alpha_i}$ to a polynomial with $\alpha_i$ pairs of complex conjugate roots, plus one real root if $\beta_i = 1$. That way, $P$ gets perturbed to a polynomial $\tilde{P}$ with exactly $\sum \beta_i$ real roots. We have that $\sum \beta_i = |\{\rho_i : \rho_i \text{ is odd}\}| \leq e - 2$, so $\tilde{P} \notin \mathcal{M}_{d,k,s}$. \hfill \qed

Corollary 4.5. Let $(d, k, s)$ be an LCN architecture with $s = (1, \ldots, 1)$ and two or more even filter sizes. The Zariski closure of the Euclidean boundary of the function space $\mathcal{M}_{d,k,s}$ is the discriminant hypersurface $\pi^{-1}(\Delta_{(2,1,\ldots,1)})$.

Proof. Let $e := |\{k_i : k_i \text{ is even}\}|$. Since the rrmp $(\rho \mid \gamma)$ of any polynomial on the Euclidean boundary of $\mathcal{M}_{d,k,s}$ satisfies by Theorem 4.4 that at least one of the $\rho_i$ is larger 1, the boundary is contained in the discriminant hypersurface. Moreover, any polynomial with one real double root, $e - 2$ pairwise distinct real single roots, and all other roots non-real with multiplicity one, is both on the boundary of $\mathcal{M}_{d,k,s}$ and a smooth point of the discriminant. \hfill \qed

Example 4.6. The following table lists all non-filling architectures with stride one and where the end-to-end filter has size 3, 4, 5 (up to permutations of the filters):

| $k$       | function space $\mathcal{M}_k$ | complement | Euclidean boundary |
|-----------|---------------------------------|------------|--------------------|
| (2, 2)    | 11|0, 2|0                  | 0|1          | 2|0          |
| (2, 2, 2) | 11|1|0, 12|0, 3|0          | 1|1          | 12|0, 3|0          |
| (3,2,2)   | 111|1|0, 12|0, 22|0, 13|0, 4|0, 11|1|2|1          | 0|2|0, 11     | 2|1, 22|0, 4|0          |
| (2, 2, 2) | 111|1|0, 12|0, 22|0, 13|0, 4|0          | 11|1|2|1, 0|2|0|11           | 112|0, 13|0, 22|0, 4|0          |

For each architecture $k$, the function space $\mathcal{M}_k$ consists of all polynomials with one of the rrmp’s listed above. We described the possible rrmp’s for quadratic, cubic, and quartic polynomials in Theorem 3.3. Similarly, the complement and boundary of $\mathcal{M}_k$ are as shown above.

When exactly two filter sizes are even, the non-filling geometry of $\mathcal{M}_{d,k,s}$ has a particularly simple description.

Proposition 4.7. Let $(d, k, s)$ be an LCN architecture with $s = (1, \ldots, 1)$ and exactly two even filter sizes. The complement of $\pi(\mathcal{M}_{d,k,s})$ in $\mathbb{R}[x,y]_{k-1}$ is a union of two open convex cones, each consisting of all positive or respectively negative polynomials.

Proof. If exactly two filters have even size, then $k - 1$ is even, and a polynomial of degree $k - 1$ is in $\pi(\mathcal{M}_{d,k,s})$ if and only if it has at least two real roots (by Theorem 4.2). Hence, the complement of $\pi(\mathcal{M}_{d,k,s})$ consists of all polynomials of even degree $k - 1$ without real roots. \hfill \qed
This behaviour is exhibited in Theorem 2.4, illustrated on the left of Figure 1, and in Theorem 4.3 cases 2 and 3 (see also Theorem 4.6 first and third row). In general, if strictly more than two filter sizes are even, the complement of $\pi(\mathcal{M}_{d,k,s})$ is not a union of two convex cones, as the following example demonstrates.

**Example 4.8.** Consider the case $\{|k_i: k_i \text{ is even}\}| = 3$ with stride one. A polynomial of odd degree $k-1$ is in $\pi(\mathcal{M}_{d,k,s})$ if and only if $p$ has at least three real roots (by Theorem 4.2). Hence, the complement $\mathbb{R}[x,y]_{k-1} \setminus \pi(\mathcal{M}_{d,k,s})$ is the set of all polynomials of degree $k-1$ with exactly one real root. Even when restricting ourselves to degree-$(k-1)$ polynomials with one real root such that the remaining degree-$(k-2)$ factor is positive, this set does not form a convex cone. For instance, if $k-1 = 3$, consider the polynomials $(x-y)x(x+y)+2y^3$ and $(x-y)x(x+y)−2y^3$: Both have exactly one real root such that the remaining quadratic factor is positive. However, their sum clearly has three real roots.

### 4.2 Generalization to larger strides

In this section, we consider one-dimensional convolutions that may have stride larger than one. In this setting, the identification with polynomials holds as described in Remark 2.8. Specifically, we write

$$\pi_s: \mathbb{R}^k \longrightarrow \mathbb{R}[x,y]_{(k-1)s}, \quad w \mapsto w_0x^{(k-1)s} + w_1x^{(k-2)s}y^s + \cdots + w_{k-2}x^2y^{(k-2)s} + w_{k-1}y^{(k-1)s}. $$

This generalizes the map $\pi = \pi_1$ defined in eq. (8). If $W_2$ and $W_1$ are convolutional matrices with strides $s_2$ and $s_1$, then $\pi(W_2W_1) = \pi_{s_2}(W_2)\pi(W_1)$. Note that $\pi_{s_1}$ is applied to $W_2$, while $s_2$ does not have any effect on the filter of $W_2W_1$ (only on its stride which is given by $s_2s_1$). Based on this fact, we deduce the following two results.

**Lemma 4.9.**

a) If $s = (s_1, \ldots, s_L)$ and $s' = (s_1, \ldots, s_{L-1}, 1)$, then $\pi(\mathcal{M}_{d,k,s}) = \pi(\mathcal{M}_{d,k,s'})$.

b) If $k = (k_1, \ldots, k_{L-1}, 1)$ and $k' = (k_1, \ldots, k_{L-1})$, $d' = (d_0, \ldots, d_{L-1}, s') = (s_1, \ldots, s_{L-1})$, then $\pi(\mathcal{M}_{d,k,s}) = \pi(\mathcal{M}_{d',k',s'})$.

**Proof.** This follows from the fact that $\pi(W_L \cdots W_1) = \pi_s(W_L)\pi_{s_1}(W_{L-1} \cdots W_1)$, where $s = s_L \cdots s_1$ is the stride of $W_{L-1} \cdots W_1$. Part a) holds since this fact is independent of $s_L$. Part b) holds since $k_L = 1$ implies that $\pi_{s_1}(W_L)$ is a degree-0 polynomial, which does not play a role in the factorization of the end-to-end filter.

**Corollary 4.10.** Let $s_1 = \cdots = s_{L-1} = 1$ and $s_L$ arbitrary. Then the LCN architecture $(d, k, s)$ satisfies the properties in Theorem 4.1 (with $s = s_L$) and Theorems 4.4 and 4.7.

The corollary is immediately implied by Theorem 4.9 a). This covers the case where only the last layer has an arbitrary stride. Similarly, Theorem 4.9 b) says that if the last filter size is one, we can understand the function space from the smaller architecture that only consists of the first $L-1$ layers. For the remaining cases, we obtain the following result.

**Proposition 4.11.** If $k_L > 1$ and $s_i > 1$ for some $i \leq L-1$, the LCN architecture is non-filling. In fact, the function space is a lower-dimensional semi-algebraic subset of $\mathcal{M}_{(d_0, d_L), k,s}$.

**Proof.** Consider the last product $W_L \cdot W'$ where $W' = W_{L-1} \cdots W_1$. We write $k' = k_1 + \sum_{i=2}^{L-1} (k_i - 1)\prod_{m=1}^{i-1} s_m$ for the filter size of $W'$, and $s' = s_1 \cdots s_{L-1} > 1$ for its stride. Even if $W'$ is an arbitrary convolutional matrix with filter size and stride $(k', s')$ (i.e., even if the LCN architecture from the first $L-1$ layers is filling), the product $W_L \cdot W'$ is not an arbitrary convolutional matrix with filter size $k = k' + (k_L - 1)s'$ and stride $s = s's_L$. Otherwise, every polynomial $p \in \mathbb{R}[x,y]_{k-1}$ could be factored into a real polynomial $p'$ of degree $k' - 1$ in $x, y$ and a real polynomial $p_L$ of degree $k_L - 1$ in
We see that the coefficients in eq. (10) are given by

\[ \dim(\text{coefficients}) = \dim(M) \]

and thus cut out by polynomial equations and inequalities. This is in contrast to the non-filling architectures appearing for stride one in Theorem 4.1: in that case, \( M \) has a strictly smaller dimension than \( M_{(d_0,d_0),k,s} \) and is thus cut out by polynomial inequalities.

**Example 4.12.** Consider the LCN with \( d = (5, 2, 1), k = (3, 2), s = (2, 1) \). Note \( s_2 = 1 \) is inconsequential by Theorem 4.9 a). This setting is similar to Theorem 2.5 where we considered \( s = (1, 1) \). In that case, the architecture was filling, but this is no longer the case for stride size 2 as we will see. In the current example, the function space \( M \subseteq \mathbb{R}^{1 \times 5} \) consists of matrices

\[ \Pi = W_2 W_1 = \begin{bmatrix} d & e \\ a & b & c & 0 & 0 \\ 0 & 0 & a & b & c \end{bmatrix} = \begin{bmatrix} ad & bd & ae + cd & be & ce \end{bmatrix}. \]  

We see that the coefficients in eq. (10) are given by

\[ \pi(\Pi) = (dx^2 + cy^2)(ax^2 + bxy + cy^2). \]  

The architecture is filling if and only if every real quartic polynomial \( Ax^4 + Bx^3y + Cx^2y^2 + Dxy^3 + Ey^4 \) can be factored into two real quadratic polynomials as above. However, it is easy to verify that the coefficients of the polynomial in eq. (11) satisfy the equation \( AD^2 + B^2E = BCD \). In fact, we can fully characterize the function space as follows:

\[ M = \{(A, B, C, D, E) \in \mathbb{R}^5: AD^2 + B^2E = BCD \text{ and } C^2 \geq 4AE \}. \]  

Indeed, the coefficients in eq. (10) satisfy \( (ae + cd)^2 - 4adce = (ae - cd)^2 \geq 0 \). For the other containment, we consider a tuple \( (A, \ldots, E) \) in the right-hand side of eq. (12) and explicitly find a factorization as in eq. (11). For the case \( E = D = 0 \), set \( a = A, b = B, c = C, d = 1, e = 0 \). For the case \( E = 0, D \neq 0 \), set \( a = C/D, b = 1, c = 0, d = B, e = D \), which works since \( AD = BC \). It remains to show the case \( E \neq 0 \). We may assume without loss of generality that \( E = 1 \) and we set \( c = 1, e = 1, b = D \). To determine \( a \) and \( d \), we distinguish four cases:

- \( D \neq 0, B \neq 0 \): Set \( d = B/D \) and \( a = AD/B \).
- \( D \neq 0, B = 0 \): Set \( d = 0 \) and \( a = C \).
- \( D = 0, A \neq 0 \): At least one of the two values \( \frac{C}{2} \pm \sqrt{\frac{C^2}{4} - A} \) is non-zero since \( A \neq 0 \). Set \( d \) to be such a non-zero value and \( a = A/d \).
- \( D = 0, A = 0 \): Set \( a = C \) and \( d = 0 \).

Using the equation and inequality in the right-hand side of eq. (12), we verify that these four cases provide factorizations as in eq. (11).
4.3 Higher dimensional convolutions

We briefly discuss convolutions over inputs having a $D$-dimensional structure, $D > 1$ (e.g., raster images). The output of each layer is now a tensor of order $D$. Thus, we identify $\mathbb{R}^{d_l} \cong \mathbb{R}^{\Lambda_l}$ with $\Lambda_l = [d_l^1] \times \cdots \times [d_l^D]$. Each filter is also a tensor $w_l \in \mathbb{R}^{\Lambda_l}$ with $\lambda_l = [k_l^1] \times \cdots \times [k_l^D]$. The convolution of $x \in \mathbb{R}^{\Lambda_l-1}$ with $w_l \in \mathbb{R}^{\Lambda_l}$ is

$$\alpha_l(x) = \sum_{j \in \Lambda_l} w_{l,j} \cdot x_{i_{1}j}, \quad i \in \Lambda_l,$$

(13)

where $s_l \in \mathbb{Z}_{>0}$ is the stride of the convolution (cf. eq. (2) for 1D convolutions). The map (13) is linear in $x$ and can be represented as a convolutional tensor $T_l \in \mathbb{R}^{\Lambda_l \times \Lambda_l-1}$ of order $2D$. The dimensions of the tensor $T_l$ satisfy (4) along each dimension: $d_l^h = \frac{d_l^{h-1} - k_l^h}{s_l} + 1$.

Example 4.13. Let $D = 2$ and let us consider a filter $w$ of size $2 \times 2$ with stride one. The convolution of an input $x = \begin{bmatrix} x_1 & \cdots & x_N \end{bmatrix}$ of size $3 \times 2$ with the filter $w$ yields an output of size $2 \times 1$: Its first resp. last entry is the inner product of $w$ with the first resp. last four entries of $x$ (marked by $*$ resp. $\circ$). The associated convolutional tensor $T$ has size $2 \times 1 \times 3 \times 2$ and its entries (shown in two slices) are as follows:

$$T_{00} := \begin{bmatrix} w_{00} & w_{01} \\ w_{10} & w_{11} \end{bmatrix} \quad \text{and} \quad T_{10} := \begin{bmatrix} 0 & 0 \\ w_{10} & w_{11} \end{bmatrix}.$$

In the following, we assume that all strides are one. Similar to the one-dimensional case, we can represent convolutions using polynomials. If $w \in \mathbb{R}^{\Lambda_l}$, we define

$$\pi(w) = \sum_{i \in \Lambda_l} w_{i_1,\ldots,i_D} x_1^{j_1-1-i_1} \cdots x_D^{j_D-1-i_D} \in \mathbb{R}\{x_1, y_1, \ldots, x_D, y_D\},$$

(14)

where the weights of the filter are the coefficients of a multivariate polynomial that is multi-homogeneous, that is, it has degree $k^D_l - 1$ in each pair $x_j, y_j$. For instance, in Theorem 4.13 we have $\pi(w) = w_{00}y_1y_2 + w_{10}x_1y_2 + w_{10}y_1x_2 + w_{11}x_1x_2$. If $T$ is a convolutional tensor with filter $w$, we also write $\pi(T)$ for the polynomial in (14). The following is a direct generalization of Theorem 2.6.

Proposition 4.14. Consider a collection of convolutional tensors $T_1, \ldots, T_L$ with compatible sizes and stride one. The tensor $\overline{T} = T_L \circ \cdots \circ T_1$ corresponding to the composition of the associated linear maps (13) is a convolutional tensor and satisfies $\pi(\overline{T}) = \pi(T_L) \cdots \pi(T_1)$.

Proof. The proof follows from the composition of two maps as in (13) with stride one:

$$\alpha_2 \circ \alpha_1(x) = \sum_{j \in \Lambda_2} w_{2,j} \cdot \alpha_1(x)_{i+j} = \sum_{j \in \Lambda_2} w_{2,j} \cdot \sum_{\ell \in \Lambda_1} w_{1,\ell} x_{(i+j)+\ell} = \sum_{m \in \mu} u_m \cdot x_{i+m},$$

where $\mu = [k_1^1 + k_2^1 - 1] \times \cdots \times [k_1^D + k_2^D - 1]$ and $u \in \mathbb{R}^\mu$ is the filter with entries $u_m = \sum_{j \in \Lambda_1, \ell \in \Lambda_2, j+\ell=m} w_{2,j} w_{1,\ell}$. From this expression it follows that $\pi(u) = \pi(w_2)\pi(w_1). \Box$

Corollary 4.15. Given an LCN architecture with $L$ layers, inputs of format $[d_0^h] \times \cdots \times [d_0^D]$, and stride one, we write $([k_1^h, \ldots, k_1^D])$ for the filter size in the $i$-th layer. The function space can be identified with the family of polynomials $P \in \mathbb{R}\{x_1, y_1, \ldots, x_D, y_D\}$ that can be factored as $P = Q_L \cdots Q_1$, where $Q_i$ is homogeneous of degree $k_i^1 - 1$ in the variables $x_j, y_j$. 

14
The natural ambient space of the LCN function space is the set of polynomials $P \in \mathbb{R}[x_1, y_1, \ldots, x_D, y_D]$ that are homogeneous of degree $k^j - 1$ in $x_j, y_j$ where $k^j := (k^j_1 - 1) + \cdots + (k^j_D - 1) + 1$. We denote this vector space by $\mathcal{M}(d_0, d_L), k, s$, where $d_0 = (d_0^1, \ldots, d_0^D)$, $d_L = (d_L^1, \ldots, d_L^D)$, $k = (k^1, \ldots, k^D)$, $s = (1, \ldots, 1)$. Using Theorem 4.15, we show that for higher-dimensional convolutions and at least two layers (and excluding trivial cases where filters have size one) the LCN function space is always a lower-dimensional semi-algebraic subset of $\mathcal{M}(d_0, d_L), k, s$ and in particular is not filling.

**Corollary 4.16.** Consider an LCN architecture with stride one, $D > 1$, and $L > 1$, where no layer has trivial filter size $(1, \ldots, 1)$. If there are distinct $j_1, j_2 \in \{1, \ldots, D\}$ such that $k^{j_1} > 1$ and $k^{j_2} > 1$, the architecture is not filling. In fact, the function space is a lower-dimensional semi-algebraic subset of $\mathcal{M}(d_0, d_L), k, s$.

**Proof.** The statement is a consequence of the fact that multivariate polynomials are generically irreducible. More precisely, we may assume without loss of generality that $k^j > 1$ for all $1 \leq j \leq n$ and that $k^j = 1$ for all $n < j \leq D$. The zero loci of the polynomials in the ambient space $\mathcal{M}(d_0, d_L), k, s$ are hypersurfaces in $(\mathbb{P}^1)^n$ of multidegree $(k^1 - 1, \ldots, k^n - 1)$. These are hyperplane sections of $(\mathbb{P}^1)^n$ under the appropriate Segre-Veronese embedding $(\mathbb{P}^1)^n \to \mathbb{P}^N$. Since $n \geq 2$, Bertini’s Theorem [22, Ch. II Thm. 8.18 + Ch. III Rem. 7.9.1] yields that a generic polynomial in $\mathcal{M}(d_0, d_L), k, s$ is irreducible over $\mathbb{C}$. Since at least two layers have non-trivial filter sizes, the function space is contained in the set of polynomials that are reducible over $\mathbb{C}$, which is Zariski closed and thus lower-dimensional in the ambient space $\mathcal{M}(d_0, d_L), k, s$. As before, the fact that the function space is semi-algebraic follows from Tarski-Seidenberg.

**Example 4.17.** Consider an LCN architecture with $3 \times 3$ matrices as inputs (i.e., $D = 2$ and $d_0 = (3, 3)$), $L = 2$ layers, filter sizes $(k^1_1, k^2_1) = (2, 2) = (k^1_2, k^2_2)$, and strides one. The two convolutions are maps $\mathbb{R}^{3 \times 3} \to \mathbb{R}^{2 \times 2} \to \mathbb{R}^{1 \times 1}$. Their composition is a convolution with filter size $3 \times 3$. Hence, the natural ambient space of the LCN function space is the 9-dimensional vector space of polynomials in two pairs of variables such that the degree in each pair is two, and the function space is the subset of polynomials that factor into two polynomials that are linear in each pair. The function space has codimension two in the ambient space. A Macaulay2 [19] computation reveals that its Zariski closure has degree 10 and that its vanishing ideal is minimally generated by 39 homogeneous polynomials of degree 6.

## 5 Optimization

We study the optimization of an objective function, or *loss*, using LCNs with 1D convolutions and stride one. A loss $\mathcal{L}(\theta)$ is a function of the filters $\theta = (w_1, \ldots, w_L)$ that can be written as a composition $\mathcal{L} = \ell \circ \mu$, where $\mu = \mu_{d_L, k, s}$ is the polynomial map described in eq. (5) and $\ell$ is a smooth function in matrix space (so $\mathcal{L}$ only depends on the end-to-end convolutional matrix $\overrightarrow{W} = \mu(\theta)$). This includes the square loss discussed further in Section 6, but also classification losses (such as cross-entropy) or any other smooth function that depends only on the output of the LCN. A tuple of filters $\theta = (w_1, \ldots, w_L)$ is said to be *critical* for the loss $\mathcal{L}$ if all partial derivatives of $\mathcal{L}$ vanish at $\theta$.

In Section 5.1 we analyze critical points of the loss function, explaining their relation to polynomials with repeated roots. In Section 5.2, we discuss the training dynamics and invariants in parameter space for LCNs, which allow us to formulate local Riemannian gradient dynamics in function space.

### 5.1 Critical points in parameter and function space

In this section, we discuss how critical points of an LCN correspond to polynomials that have repeated roots. We provide the proofs of the following proposition and theorems at the end of this section.
Proposition 5.1. Let $\mathcal{L} = \ell \circ \mu$ be a loss function for an LCN with stride one, where $\ell$ is a smooth and convex function on the space of convolutional matrices. If $\theta = (w_1, \ldots, w_L)$ is critical for $\mathcal{L}$, then one of the following holds:

1. $\theta$ is a global minimum for $\mathcal{L}$, or
2. for some $i \neq j$, the homogeneous polynomials $\pi(w_i)$ and $\pi(w_j)$ have a common factor.

Polynomials with multiple roots form in fact the boundaries of non-filling architectures (as shown in Theorem 4.4), however we will see that they arise as critical points more generally: Every critical point $\theta$ for the loss of an LCN corresponds to a critical point $\mu(\theta)$ of $\ell$ restricted to one of the multiple root loci $\Delta_\lambda$, i.e., $\mu(\theta) \in \text{Crit}(\ell|_{\pi^{-1}(\Delta_\lambda)})$, even if $\mu(\theta)$ does not necessarily lie on the boundary of the function space. In other words, $\mu(\theta)$ is a smooth point of $\pi^{-1}(\Delta_\lambda)$ and the differential $d\ell$ at $\mu(\theta)$ vanishes on the tangent space $T_{\mu(\theta)}\pi^{-1}(\Delta_\lambda)$.

Theorem 5.2. Let $\mathcal{L} = \ell \circ \mu$ be a loss function for an LCN with stride one. Let $\theta = (w_1, \ldots, w_L) \in \text{Crit}(\mathcal{L})$ and let $\overline{w} = \mu(\theta) \in \mathbb{R}^k$ be the end-to-end filter. Let $\lambda = (\lambda_1, \ldots, \lambda_r)$ be the partition of $k - 1$ corresponding to the root structure of $\pi(\overline{w})$. Then $\overline{w} \in \text{Crit}(\ell|_{\pi^{-1}(\Delta_\lambda)})$.

To reverse Theorem 5.2, we characterize which critical points on the multiple root loci come from critical points of a given LCN.

Theorem 5.3. Let $\mathcal{L} = \ell \circ \mu$ be a loss function for an LCN with stride one. Let $\lambda$ be a partition of $k - 1$ and let $\overline{w} \in \text{Crit}(\ell|_{\pi^{-1}(\Delta_\lambda)})$. If there exists $\theta = (w_1, \ldots, w_L)$ such that $\mu(\theta) = \overline{w}$ and each polynomial $\pi(w_1), \ldots, \pi(w_L)$ has no repeated roots, then $\theta \in \text{Crit}(\mathcal{L})$.

Moreover, if the loss $\ell$ in function space is sufficiently generic (e.g., if the training data is sufficiently generic), then we do not expect there to be any further critical points of the LCN other than the ones described in Theorem 5.3. We provide a formal argument for this assertion when $\ell = \ell_{X,Y}$ is the square loss (defined in eq. (21)) with generic training data.

Theorem 5.4. Let $X \in \mathbb{R}^{d_0 \times N}$ and $Y \in \mathbb{R}^{d_L \times N}$ be generic data matrices with $N \geq d_0$ and let $\mathcal{L} = \ell_{X,Y} \circ \mu$ be the square loss for an LCN with stride one. Then every $\theta = (w_1, \ldots, w_L) \in \text{Crit}(\mathcal{L})$ satisfies that each polynomial $\pi(w_1), \ldots, \pi(w_L)$ has no repeated roots.

To summarize, a critical point $\overline{w}$ of a generic loss $\ell$ on a multiple root locus $\pi^{-1}(\Delta_\lambda)$ comes from a critical point of the LCN if and only if $\pi(\overline{w})$ can be factored according to the LCN architecture such that no factor has a double root. Whether a polynomial with rrmp $(\rho \mid \gamma)$ can be factored according to the LCN architecture without double roots in any of the factors depends on a purely combinatorial property of $(\rho \mid \gamma)$ and the partition $(k_1 - 1, \ldots, k_L - 1)$ of $k - 1$ that is given by the architecture. Indeed, the factorization of the polynomial is equivalent to the placing of $\rho_i$ balls of size 1 and color $i$ (for $1 \leq i \leq r$) and $\gamma_j$ balls of size 2 and color $r + j$ (for $1 \leq j \leq c$) into $L$ bins of sizes $k_1 - 1, \ldots, k_L - 1$ such that no bin contains two balls of the same color. If this placement of balls into bins is possible, we say that $(\rho \mid \gamma)$ is compatible with the LCN architecture.

All in all, the previous Theorems 5.2 to 5.4 show that the critical points of an LCN with a generic loss $\ell$ correspond exactly to the critical points on multiple root loci with an rrmp that is compatible with the architecture.

Corollary 5.5. Let $X \in \mathbb{R}^{d_0 \times N}$ and $Y \in \mathbb{R}^{d_L \times N}$ be generic data matrices with $N \geq d_0$ and let $\mathcal{L} = \ell_{X,Y} \circ \mu$ be the square loss for an LCN with stride one. Let $(\rho \mid \gamma)$ be such that $\lambda = \lambda_{\rho,\gamma}$ is a partition of $k - 1$. If $(\rho \mid \gamma)$ is compatible with the LCN architecture, then for every $\overline{w} \in \text{Crit}(\ell|_{\pi^{-1}(\Delta_\lambda)})$ with rrmp $(\rho \mid \gamma)$ there exists some $\theta \in \mu^{-1}(\overline{w}) \cap \text{Crit}(\mathcal{L})$. Otherwise, every $\overline{w} \in \text{Crit}(\ell|_{\pi^{-1}(\Delta_\lambda)})$ with rrmp $(\rho \mid \gamma)$ satisfies $\mu^{-1}(\overline{w}) \cap \text{Crit}(\mathcal{L}) = \emptyset$. 

16
Example 5.6. We compare the LCN architectures with \( k = (3, 2, 2) \) and \( k' = (4, 2) \). We discussed in Theorem 4.3 that the function space associated with these architectures is the same. The polynomial \( \pi(w) = p_1 p_2^2 \), where \( p_1 \in \mathbb{R}[x, y]_1 \), has rmp \((13 | 0)\). We can find a factorization compatible with the architecture \( k = (3, 2, 2) \) where each polynomial associated with a filter does not have any repeated roots: \( \pi(w) = \left(p_1 p_2\right) \cdot p_2 \). In contrast, for \( k' = (4, 2) \), any factorization of \( \pi(w) \) according to the architecture will be such that the cubic polynomial associated with the filter of size 4 will have at least one pair of repeated roots.

For both architectures, we now list all compatible \((\rho \mid \gamma)\) by providing a feasible factorization; the remaining ones are marked with a dash:

| \( \rho \mid \gamma \) | 11110 | 11210 | 22101 | 13011 | 40111 | 11121 | 21011 | 02011 | 01111 |
|-------------------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| (3, 2, 2)         | \( p_1 p_2 p_3 p_4 \) | \( p_1 p_2 p_3 p_4 \) | \( p_1 p_2 p_3 p_4 \) | \( p_1 p_2 p_3 p_4 \) | \( p_1 p_2 p_3 p_4 \) | \( p_1 p_2 p_3 p_4 \) | \( p_1 p_2 p_3 p_4 \) | \( p_1 p_2 p_3 p_4 \) | \( p_1 p_2 p_3 p_4 \) |

(4, 2) \( p_1 p_2 p_3 p_4 \) \( p_1 p_2 p_3 p_4 \) \( p_1 p_2 p_3 p_4 \) \( p_1 p_2 p_3 p_4 \) \( p_1 p_2 p_3 p_4 \) \( p_1 p_2 p_3 p_4 \) \( p_1 p_2 p_3 p_4 \) \( p_1 p_2 p_3 p_4 \) \( p_1 p_2 p_3 p_4 \)

To prove the theorems above, we make use of the composition rule of differentials:

\[
\mathcal{d} \mathcal{L} = \mathcal{d} \ell \circ \mathcal{d} \mu_{d,k,s}.
\]

Even if the parameterization map \( \mu \) is surjective (i.e., the function space \( \mathcal{M} = \mathcal{M}_{d,k,s} \) is filling), the differential \( \mathcal{d} \mu \) (equivalently, the Jacobian matrix of \( \mu \)) might not have maximal rank everywhere. This implies that the parameterization may give rise to critical points for \( \mathcal{L} \) in addition to the global minimum even if the function space \( \mathcal{M} \) is a vector space and \( \ell \) is convex. Following [35], we call this kind of critical points spurious. In contrast, a critical point \( \theta \) for \( \mathcal{L} \) is pure if \( \mu(\theta) \) is a critical point of the loss in function space, i.e., of \( \ell|_{\mathcal{M}} \).

Given an end-to-end filter \( \overline{w} \in \mathbb{R}^k \) of an LCN, we describe the set of parameters \( \theta \) such that \( \mu(\theta) = \overline{w} \), i.e., the fiber of the parameterization. We say that two parameters \( \theta = (w_1, \ldots, w_L) \) and \( \theta' = (w'_1, \ldots, w'_L) \) are equivalent up to scalar factors if there exists \((\kappa_1, \ldots, \kappa_L) \in (\mathbb{R}^*)^L \) with \( \prod_{i=1}^L \kappa_i = 1 \) such that \( w_i = \kappa_i w'_i \).

Lemma 5.7. Let \( \overline{w} \in \mathbb{R}^k \setminus \{0\} \) be an end-to-end filter of an LCN with stride one. Then the fiber \( \mu^{-1}(\overline{w}) \) consists of a finite set of scaling equivalence classes. Moreover, either all points in the same equivalence class are critical or none are.

Proof. The fact that \( \mu^{-1}(\overline{w}) \) consists of a finite set of scaling equivalence classes follows from the uniqueness of the decomposition of \( \pi(\overline{w}) \in \mathbb{R}[x, y] \) into irreducible factors: Each equivalence class corresponds to a way of aggregating factors of \( \pi(\overline{w}) \) into polynomials of degrees \( k_1 - 1, \ldots, k_L - 1 \), where \( (k_1, \ldots, k_L) \) are the filter widths of the LCN. For the second claim, we use the fact that if \( \theta \) and \( \theta' \) are equivalent under rescaling, then the image of the differential \( \mathcal{d} \mu \) at these points is the same. This will follow from our characterization of the image of \( \mathcal{d} \mu \) in (15). Hence, \( \mathcal{d} \mathcal{L}(\theta) = \mathcal{d} \ell(\overline{w}) \circ \mathcal{d} \mu(\theta) = 0 \) if and only if \( \mathcal{d} \mathcal{L}(\theta') = \mathcal{d} \ell(\overline{w}) \circ \mathcal{d} \mu(\theta') = 0 \).

Example 5.8. We discuss the parameters corresponding to particular points in the function space of the LCN with \( k = (2, 2) \) from Theorem 2.4. Using the identification \( \pi \) with polynomials, a point in the interior of the function space is a quadratic polynomial with two distinct real roots: \( P = p_1 p_2 \), where \( p_1 = ax + by \) and \( p_2 = cx + dy \) are not scalar multiples of each other. The fiber \( \mu^{-1}(P) \) of the parameterization has four connected one-dimensional components: \( \{(k \kappa p_1, \frac{1}{k} p_2) : \kappa > 0\} \), \( \{(k \kappa p_2, \frac{1}{k} p_1) : \kappa > 0\} \), \( \{(k \kappa p_1, \frac{1}{k} p_2) : \kappa < 0\} \), and \( \{(k \kappa p_2, \frac{1}{k} p_1) : \kappa < 0\} \).

The points on the boundary of the function space are exactly those \( P = p_1 p_2 \) where \( p_1 \) and \( p_2 \) are linearly dependent. In other words, the parameters mapping to boundary points are those which satisfy \( bc - ad = 0 \). We will see in Theorem 5.11 that these are exactly the parameters where the Jacobian of the parameterization \( \mu \) drops rank.

Hence, a non-zero point on the boundary of the function space is of the form \( P = \pm p^2 \) for some linear
term $p \neq 0$. Thus, the fiber $\mu^{-1}(P)$ has two connected one-dimensional components: $\{ (\pm \kappa p, \frac{1}{\kappa} p) : \kappa > 0 \}$ and $\{ (\pm \kappa p, \frac{1}{\kappa} p) : \kappa < 0 \}$. Finally, the fiber $\mu^{-1}(0)$ over the origin is the union of the two planes $\{ (p, 0) : p \in \mathbb{R}[x, y]_1 \}$ and $\{ (0, p) : p \in \mathbb{R}[x, y]_1 \}$.

**Remark 5.9.** In the case of fully-connected linear networks, spurious critical points are always saddles. More precisely, for a fully-connected network, a loss function $\mathcal{L} = \ell \circ \mu$ where $\ell$ is a smooth convex function has non-global minima if and only if the restriction of $\ell$ on the function space has non-global minima [35, Proposition 10]. In Example 5.10, we will see that this is not true for LCNs, and it is possible for the loss function $\mathcal{L}$ to have non-global minima even if the function space is filling.

The intuitive geometric reason for this different behavior between fully-connected and convolutional linear networks lies in the very different structure of the fibers. For fully-connected networks, as shown in [35, Proposition 9], it is possible to perturb a spurious critical point within the fiber to obtain another point that is not a critical point for the loss. This in turn implies that the original spurious critical point could not have been a minimum for the loss. In the case of LCNs, however, this argument does not apply, by Theorem 5.7. Thus, for LCNs, it is possible for spurious critical points to be (non-global) local minima.

**Example 5.10.** Let us consider an LCN with filter sizes $k = (2, 3)$. This is a filling architecture. In terms of polynomials, its function space is the set of all cubic polynomials. The two layers of the LCN factor such a polynomial into a linear and a quadratic term. We fix the target polynomial in $\{ \}$. Let us consider an LCN with filter sizes $k = (2, 3)$. This is a filling architecture. In terms of polynomials, its function space is the set of all cubic polynomials. The two layers of the LCN factor such a polynomial into a linear and a quadratic term. We fix the target polynomial $u = (x + y)(x^2 + 1/10 y^2)$. In the space of polynomials $\mathbb{R}[x, y]_3$, we use the loss $\ell(p)$ that is the squared sum of the coefficients of $p - u$. We will see that $\mathcal{L} = \ell \circ \mu$ has a non-global minimum, although $\ell$ clearly only has one local and global minimum.

Almost every cubic polynomial $p$ can be factored as $p = q_1 q_2$ where $q_1 = x + ay$ and $q_2 = bx^2 + cxy + dy^2$. We assume that $q_1$ is monic (i.e., the coefficient for $x$ is 1) since there is otherwise a scaling ambiguity among $q_1$ and $q_2$ which does not affect the product $q_1 q_2$ nor the property of critical points of $\mathcal{L}$. Our objective function $\mathcal{L}$ thus becomes

$$
\mathcal{L} : \mathbb{R}^4 \rightarrow \mathbb{R} : (a, b, c, d) \mapsto (a - 1)^2 + (ab + c - 1)^2 + (ac + d - 1/10)^2 + (ad - 1/10)^2.
$$

A direct computation reveals that $\mathcal{L}$ has 10 complex critical points: Three of them (1 real, 2 complex) correspond to factorizations of the target polynomial $u$. The other seven (3 real, 4 complex) are spurious critical points and yield factorizations of seven different cubic polynomials. One of the latter three real spurious critical points is

$$
a \approx 0.0578445483987, \ b \approx 1.0000187825172, \ c \approx 0.941829719725, \ d \approx 0.0511336556138.
$$

The Hessian of $\mathcal{L}$ at this point is positive definite, showing that it is indeed a local minimum.

We now investigate the existence of spurious critical points for LCNs by studying the rank of the differential of the parameterization map $\mu$.

**Proposition 5.11.** Let $\mu$ be the parameterization map of an LCN with stride one. Given filter weights $\theta = (w_1, \ldots, w_L)$, we write $\overline{w} = \mu(\theta) \in \mathbb{R}^k$ for the end-to-end filter. Using the identification $\pi$, the image of the differential $d\mu$ at the point $\theta$ is the set of polynomials

$$
H(\theta) := \{ h g : h \in \mathbb{R}[x, y]_{k-1 - \deg(g)} \} \subset \mathbb{R}[x, y]_{k-1},
$$

where $g = \gcd \left( \frac{\pi(w_1)}{\pi(w_1)}, \ldots, \frac{\pi(w_L)}{\pi(w_L)} \right) \in \mathbb{R}[x, y]$. In particular, $d\mu(\theta)$ is surjective if and only if $\gcd(\pi(w_i), \pi(w_j)) = 1$ for every pair $i, j \in \{1, \ldots, L\}$ with $i \neq j$.

To prove this result, we make use of the following homogeneous version of Bezout’s identity.
Lemma 5.12. Let \( p_1, \ldots, p_n \) be homogeneous polynomials in \( \mathbb{R}[x, y] \) of arbitrary degree and let \( g = \gcd(p_1, \ldots, p_n) \) and \( l = \text{lcm}(p_1, \ldots, p_n) \). For any \( d \geq \deg(l) - 1 \) the following sets of homogeneous polynomials in \( \mathbb{R}[x, y]_d \) are equal:

\[
I_d := \{ \alpha_1 p_1 + \cdots + \alpha_n p_n : \alpha_i \in \mathbb{R}[x, y]_{d-\deg(p_i)} \}, \quad J_d := \{ h \in \mathbb{R}[x, y]_{d-\deg(g)} \}.
\]

Proof. If \( g = l \) (i.e. \( p_1 = \cdots = p_n \)), the assertion is clear. Hence, we assume that \( \deg(g) < \deg(l) \). We always have that \( I_d \subseteq J_d \), since any algebraic combination of \( p_1, \ldots, p_n \) is a multiple of \( g \). For the converse, we observe that it is enough to prove that \( J_{\deg(l)-1} \subseteq I_{\deg(l)-1} \) (if \( hg \) belongs to the ideal generated by \( p_1, \ldots, p_n \), then so do \( xhg \) and \( yhg \)). Moreover, it is sufficient to assume that the linear form \( f \) is not a factor of \( l \) (otherwise, we apply a general invertible linear transformation to the input variables, e.g. \( x' = x, y' = y + tx \) for appropriate \( t \)). We can thus safely dehomogenize and obtain univariate polynomials \( \tilde{p}_i = p_i(x, 1), \tilde{g} = g(x, 1), \tilde{l} = l(x, 1) \) in \( \mathbb{R}[x] \). Note that these polynomials have the same degrees as their homogeneous counterparts and also that \( \tilde{g} = \gcd(\tilde{p}_1, \ldots, \tilde{p}_n) \) and \( \tilde{l} = \text{lcm}(\tilde{p}_1, \ldots, \tilde{p}_n) \). By Bezout’s identity, we can write \( \beta_1 \tilde{p}_1 + \cdots + \beta_n \tilde{p}_n = \tilde{g} \) for some \( \beta_i \in \mathbb{R}[x] \). Thus, for any \( h \in \mathbb{R}[x] \), we get a decomposition \( \tilde{h} \tilde{p}_1 \cdots + \tilde{h}_n \tilde{p}_n = \tilde{h} \tilde{g} \). We now claim that we can assume \( \deg(\tilde{h}_i) < \deg(l) - \deg(\tilde{p}_i) \) whenever \( \deg(g) \leq \deg(l) - 1 \). Indeed, if this is not the case, then we write \( \tilde{h}_i = \tilde{q}_i \tilde{p}_i^{*} + \tilde{r}_i \) with \( \tilde{p}_i^{*} = \tilde{l}/\tilde{p}_i \) and \( \deg(\tilde{r}_i) < \deg(\tilde{p}_i^{*}) = \deg(l) - \deg(\tilde{p}_i) \), and obtain \( (\sum_{i=1}^n \tilde{q}_i) \tilde{l} + \tilde{r}_1 \tilde{p}_1 + \cdots + \tilde{r}_n \tilde{p}_n = \tilde{h} \tilde{g} \). Since \( \deg(\tilde{h}) \leq \deg(l) \), we deduce that \( \sum_{i=1}^n \tilde{q}_i = 0 \), which yields a decomposition with the desired degree bounds. We can now homogenize this relation so that the total degree is \( \deg(l) - 1 = \deg(l) - 1 \) to obtain \( \alpha_1 p_1 + \cdots + \alpha_n p_n = hg \), where \( h(x, y) = y^{\deg(l)-1-\deg(p_i)} \tilde{h}_i \left( \frac{x}{y} \right) \), \( g(x, y) = y^{\deg(g)} \tilde{g} \left( \frac{x}{y} \right) \).

Note that \( p_i \) and \( g \) are the original homogeneous polynomials from the statement. Since \( \tilde{h} \) can be chosen arbitrarily in \( \mathbb{R}[x]_{\leq \deg(l)-1-\deg(g)} \), we have that \( h(x, y) \) can be an arbitrary polynomial in \( \mathbb{R}[x, y]_{\deg(l)-1-\deg(g)} \).

We thus conclude that \( J_{\deg(l)-1} \subseteq I_{\deg(l)-1} \).

Proof of Theorem 5.11. For notational simplicity, we omit writing the map \( \pi \) and identify \( \mathbb{R}^k \) with \( \mathbb{R}[x, y]_{k-1} \). The differential of the parameterization can be written as

\[
d\mu(\theta) : \mathbb{R}^{k_1 + \cdots + k_L} \to \mathbb{R}^k ; \quad (\dot{w}_1, \ldots, \dot{w}_L) \mapsto \dot{w}_1 \frac{\bar{w}}{w_1} + \cdots + \dot{w}_L \frac{\bar{w}}{w_L},
\]

where the product means polynomial multiplication and each \( \dot{w}_i \) is simply a symbol for the coordinates of the tangent vector corresponding to the variable \( w_i \). Since \( \bar{w} \) is a common multiple of \( \frac{\bar{w}}{w_1}, \ldots, \frac{\bar{w}}{w_L} \), using Theorem 5.12 we deduce that the image of (16) is equal to \( H(\theta) \) in (15). This shows that the differential is surjective if and only if \( \gcd(\frac{\bar{w}}{w_1}, \ldots, \frac{\bar{w}}{w_L}) = 1 \). The latter is equivalent to \( \gcd(w_i, w_j) = 1 \) for every pair \( i \neq j \). Indeed, any common factor of \( w_i, w_j \) is a factor of each \( \frac{\bar{w}}{w_i} \); conversely, if \( q \) is an irreducible common factor of \( \frac{\bar{w}}{w_1}, \ldots, \frac{\bar{w}}{w_L} \), then \( q|w_i \) for some \( i \in \{1, \ldots, L\} \), and from \( q|\bar{w}_i \), we deduce \( q|w_j \) for some \( j \neq i \).

We are finally ready to prove the main theorems of this section. As in the proof of Theorem 5.11, we omit writing the map \( \pi \) and identify filters with polynomials.

Proof of Theorem 5.1. Let \( \overline{W} = \mu w_1, \ldots, w_L \). If \( (d\ell \circ d\mu)(\theta) = 0 \) then either \( d\ell(\overline{W}) = 0 \) or \( d\mu(\theta) \) must not have full rank. The assumption that \( \ell \) is convex means that if \( d\ell(\overline{W}) = 0 \) then \( \overline{W} \) is a global minimum for \( \ell \) and thus \( \theta \) is a global minimum for \( L \). Alternatively, if \( d\mu(\theta) \) does not have full rank, then by Theorem 5.11 the second condition holds.
Proof of Theorem 5.2. If \( \lambda = (1, \ldots, 1) \), then \( \mathcal{W} \) has \( k - 1 \) distinct roots. By Theorem 5.11, the differential \( d\mu(\theta) \) has full rank, i.e., \( \theta \) is a pure critical point. This in turn implies that \( \mathcal{W} \) is a critical point for \( \ell \) in \( \Delta_\lambda = \mathbb{R}[x, y]_{k-1} \).

We next assume that \( \lambda = (\lambda_1, \ldots, \lambda_r) \) is such that \( \lambda_1 > 1 \). Let \( \mathcal{W} = q_1^{\lambda_1} \cdots q_r^{\lambda_r} \) be the decomposition of \( \mathcal{W} \) into pairwise distinct complex linear forms \( q_i \in \mathbb{C}[x, y] \). Note that if \( \mathcal{W} = \prod_{i=1}^r q_i^{\lambda_i} \) is real, then so is \( \prod_{i=1}^r q_i^{\lambda_i - 1} \). We claim that the (real) tangent space of \( \Delta_\lambda \) at the smooth point \( \mathcal{W} \) is given by

\[
T_{\mathcal{W}} \Delta_\lambda = \left\{ h \prod_{i=1}^r q_i^{\lambda_i - 1} : h \in \mathbb{R}[x, y]_r \right\} \subset \mathbb{R}[x, y]_{k-1}.
\]  

Indeed, let us assume without loss of generality that \( q_{i_1}, \ldots, q_{i_r} \) are real linear forms, while \( q_{j_1}, q_{j_1+1}, \ldots, q_{j_s}, q_{j_s+1} \) are such that each \( (q_{j_1}, q_{j_1+1}) \) is a conjugate pair of complex linear forms, with \( r'+2r'' = r \) (in particular we must also have \( \lambda_{j_i} = \lambda_{j_i+1} \) for all \( l = 1, \ldots, r'' \)). We write \( t_{j_l} = q_{l_1}q_{l_1+1} \) for the irreducible real quadratic form that is the product of the pair \( q_{l_1}, q_{l_1+1} \). In a neighborhood of \( \mathcal{W} \), the multiple root locus \( \Delta_\lambda \) is the image of

\[
(q_{i_1}, \ldots, q_{i_r}, t_{j_1}, \ldots, t_{j_{r''}}) \mapsto \prod_{i=1}^r q_i^{\lambda_i} \prod_{l=1}^{r''} t_{j_l}^{\lambda_{j_l}},
\]

as \( q_{i_m} \in \mathbb{R}[x, y]_1 \) and \( t_{j_l} \in \mathbb{R}[x, y]_2 \) vary. Arguing as in the proof of Theorem 5.11 (based on Theorem 5.12), we have the image of the differential of (18) is the set of homogeneous polynomials of degree \( k - 1 \) that are multiples of \( g' = \gcd(h, q_{i_1}, \ldots, q_{i_r}, t_{j_1}, \ldots, t_{j_{r''}}) \). We finally observe that \( g' = \prod_{i=1}^r q_i^{\lambda_i - 1} \).

Having shown (17), we recall that the image of the differential of the parameterization \( \mu \) is \( \text{Im}(d\mu(\theta)) = \{ h g : h \in \mathbb{R}[x, y]_{k-1 - \deg(g)} \} \subset \mathbb{R}[x, y]_{k-1} \), where \( g \) is as in Theorem 5.11. To conclude the proof, we note that \( T_{\mathcal{W}} \Delta_\lambda \subset \text{Im}(d\mu(\theta)) \) implies the statement of the theorem since \( 0 = d\mathcal{L}(\theta) = d\mathcal{L}(\mathcal{W}) \circ d\mu(\theta) \) implies \( d\mathcal{L}(\mathcal{W}) = 0 \), i.e., \( \mathcal{W} \) is a critical point of \( \ell \mid_{ \Delta_\lambda } \).

Proof of Theorem 5.3. Below we show the following claim for \( \mathcal{W} \) being a smooth point of \( \Delta_\lambda \): For every \( \theta \in \mu^{-1}(\mathcal{W}) \), it holds \( \text{Im}(d\mu(\theta)) = T_{\mathcal{W}}(\Delta_\lambda) \) if and only if each \( w_i \) has no repeated roots. Hence, under the assumptions in the theorem, \( d\mathcal{L}(\theta) = d\mathcal{L}(\mathcal{W}) \circ d\mu(\theta) = d\mathcal{L}(\mathcal{W})|_{T_{\mathcal{W}} \Delta_\lambda} = 0 \).

To prove our claim, we use the fact that \( T_{\mathcal{W}}(\Delta_\lambda) = \{ h \prod_{i=1}^r q_i^{\lambda_i - 1} : h \in \mathbb{R}[x, y]_r \} \) as in eq. (17) if \( \mathcal{W} = \prod_{i=1}^r q_i^{\lambda_i} \) and that \( \text{Im}(d\mu(\theta)) = \{ h g : h \in \mathbb{R}[x, y]_{k-1 - \deg(g)} \} \), where \( g \) is the gcd defined in Theorem 5.11. We observe that \( w_i \) has no repeated roots if and only if \( \prod_{i=1}^r q_i^{\lambda_i - 1} \mid g \). This holds for every \( i = 1, \ldots, L \) if and only if \( \prod_{i=1}^r q_i^{\lambda_i - 1} \mid g \). Since \( g \mid \prod_{i=1}^r q_i^{\lambda_i - 1} \) always holds, as seen in the proof of Theorem 5.2, the latter is equivalent to \( \text{Im}(d\mu(\theta)) = T_{\mathcal{W}}(\Delta_\lambda) \).

Proof of Theorem 5.4. As we will see in eqs. (21) to (23), if \( XX^T \) has full rank, any square loss \( \ell_{XY}(\mathcal{W}) \) is equivalent up to a constant additive term to a function of the form \( \ell_{\Sigma, u}(\mathcal{W}) = \| u - \mathcal{W} \|^2_\Sigma \), where \( u \) is a target filter and \( \langle \cdot, \cdot \rangle_\Sigma \) is a non-degenerate inner product on \( \mathbb{R}^k \). Here \( k \) is the size of the filters \( u \) and \( \mathcal{W} \). We now show that for arbitrary inner products \( \langle \cdot, \cdot \rangle_\Sigma \) and generic target filters \( u \), any critical point \( \theta = (w_1, \ldots, w_L) \) of \( \mathcal{L}_{\Sigma, u} = \ell_{\Sigma, u} \circ \mu \) is such that each polynomial \( w_1, \ldots, w_L \) has no repeated
roots. Fixing the inner product arbitrarily, we consider for all \( i = 1, \ldots, L \) the set
\[
Z_i = \{ (\theta, \overline{w}, u) : \theta \in \text{Crit}(\mathcal{L}_{\Sigma, u}), \mu(\theta) = \overline{w}, \theta = (w_1, \ldots, w_L), w_i \text{ has a repeated root} \}.
\]
Note that this is an algebraic set. We claim that \( \dim(\text{pr}_3(Z_i)) \leq k - 1 \) holds for all \( i = 1, \ldots, L \), where \( \text{pr}_3 \) denotes the projection on the third factor. This implies the statement of the theorem since it shows that the set of target filters \( u \) such that any of the corresponding critical points has a filter with repeated roots has positive codimension in \( \mathbb{R}^k \). To prove our claim on the dimension of \( Z_i \), we further consider the sets
\[
Z_{i, \lambda} = \{(\theta, \overline{w}, u) : (\theta, \overline{w}, u) \in Z_i, \overline{w} \in \text{Reg}(\Delta_\lambda) \},
\]
where \( \lambda \) is a partition of \( k - 1 \) and \( \text{Reg}(\Delta_\lambda) \) denotes the locus of smooth points on \( \Delta_\lambda \). Clearly, \( Z_i = \bigcup \lambda Z_{i, \lambda} \). As shown in the proof of Theorem 5.3, if \( \overline{w} \) is a smooth point of \( \Delta_\lambda \) and \( w_i \) has a repeated root, then \( T_{\overline{w}}\Delta_\lambda \subseteq \text{Im}(d\mu(\theta)) \). The fiber \( \mu^{-1}(\overline{w}) \) consists of finitely many scaling equivalence classes (see Theorem 5.7) and all \( \theta \) in the same equivalence class yield the same \( \text{Im}(d\mu(\theta)) \). We note that \( \theta \in \text{Crit}(\mathcal{L}_{\Sigma, u}) \) means equivalently that \( u - \overline{w} \perp \Sigma \text{ Im}(d\mu(\theta)) \), or in other words \( u \in \overline{w} + \text{Im}(d\mu(\theta))^{⊥} \). We write \( N_{\Sigma, \overline{w}} \) for the finite union of the affine spaces \( \overline{w} + \text{Im}(d\mu(\theta))^{⊥} \) where \( \theta = (w_1, \ldots, w_L) \in \mu^{-1}(\overline{w}) \) and some \( w_i \) has a repeated root. Hence, \( u \in \text{pr}_3(Z_{i, \lambda}) \) if and only if there is some smooth point \( \overline{w} \) on \( \Delta_\lambda \) such that \( u \in N_{\Sigma, \overline{w}} \). Since \( \dim(N_{\Sigma, \overline{w}}) < \dim(T_{\overline{w}}\Delta_\lambda)^{⊥} = k - \dim(\Delta_\lambda) \), we see that \( \dim(\text{pr}_3(Z_{i, \lambda})) \leq \dim(\Delta_\lambda) + \max_{\overline{w} \in \text{Reg}(\Delta_\lambda)} \dim(N_{\Sigma, \overline{w}}) < k \). This concludes the proof.

5.2 Training dynamics

In this section, we discuss some matters related to the training dynamics of linear convolutional networks. We begin by proving the existence of dynamical invariants in parameter space. Similar results have been shown for linear networks [6] and ReLU networks [36]. We restrict ourselves to networks with stride one.

Proposition 5.13. Let \( \mathcal{L} = \ell \circ \mu \) be a loss function for an LCN with stride one, where \( \ell \) is a smooth function on the space of convolutional matrices. Let \( \theta(t) \) be an integral curve in parameter space for the negative gradient field of \( \mathcal{L} \), i.e., \( \dot{\theta}(t) = -\nabla(\mathcal{L}(\theta(t))) \). If we write \( \theta(t) = (w_1(t), \ldots, w_L(t)) \), where \( w_i \) are filter vectors, then the quantities \( \delta_{ij}(t) = \|w_i(t)\|^2 - \|w_j(t)\|^2 \) for \( i, j \in \{1, \ldots, L\} \) remain constant for all \( t \in \mathbb{R} \).

Proof. For any \( \kappa \in \mathbb{R} \setminus \{0\} \), let \( \nu_{ij}(\theta, \kappa) = (\kappa w_1, \ldots, \kappa w_i, 1/\kappa w_j, \ldots, w_L) \). Because the map \( \mu \) is multilinear in the filters, we have that \( \mathcal{L}(\nu_{ij}(\theta, \kappa)) \) does not depend on \( \kappa \). This implies that \( \frac{d}{d\kappa} \mathcal{L}(\nu_{ij}(\theta, \kappa)) = 0 \) and in particular
\[
0 = \left[ \frac{d}{d\kappa} \mathcal{L}(\nu_{ij}(\theta, \kappa)) \right]_{\kappa=1} = \left[ \frac{d}{d\kappa} \nu_{ij}(\theta, \kappa) \right]_{\kappa=1} = \frac{d}{d\theta} \mathcal{L}(\theta) \cdot \left[ \begin{array}{c} w_1(t) \\ \vdots \\ w_i(t) \\ -w_j(t) \\ \vdots \\ -w_L(t) \end{array} \right]^T.
\]
We now observe that
\[
\frac{d}{dt} \delta_{ij}(t) = \frac{d}{d\theta} \delta_{ij} \cdot \frac{d}{dt} \theta(t) = -2 [0, \ldots, w_i, \ldots, -w_j, \ldots, 0]^T \cdot \frac{d}{d\theta} \mathcal{L}(\theta) = 0.
\]
This shows that \( \delta_{ij} \) remains constant for all \( t \).
As discussed in Theorem 5.7, each connected component of the fiber \( \mu^{-1}(p) \) of a polynomial \( p \) under the parameterization map \( \mu \) consists of the different rescalings of a fixed factorization of \( p \). Theorem 5.13 can be used to eliminate the scaling ambiguity, assuming that the parameters (or the invariants) are known at initialization.

**Corollary 5.14.** Let \( \mu \) be the parameterization map of an LCN with stride one. Given filters \( \theta = (w_1, \ldots, w_L) \), the set of parameter values \( \theta' \) such that \( \mu(\theta) = \mu(\theta') \) and \( \delta_{i,i+1} = \delta'_{i,i+1} \) for all \( i = 1, \ldots, L - 1 \) is finite, where \( \delta_i \) and \( \delta'_i \) are the invariants associated with \( \theta \) and \( \theta' \), respectively, as defined in Theorem 5.13.

**Proof.** Let \( q_i = \pi(w_i) \) be the polynomials associated with the filters \( w_i \), and let \( p = q_L \cdots q_1 = \pi(\mu(\theta)) \). As in Lemma 5.7, the factors \( q_i \) are determined by \( p \) up to a finite number of root permutations and also up to rescalings: \( q'_i = \kappa_i q_i \), where \( \prod_{i=1}^L \kappa_i = 1 \). It remains to show that there are only finitely many choices of the scale factors \( \kappa_i \) such that \( \delta_{i+1,i} = \|\kappa_{i+1} q_{i+1}\|^2 - \|\kappa_i q_i\|^2 \) for \( i = 1, \ldots, L - 1 \). We can recover the possible \( \kappa_i \) by solving a simple polynomial system. Setting \( \beta_i = \|\kappa_i q_i\|^2 = \kappa_i^2 \|q_i\|^2 \) and \( \delta_i = \delta_{i+1,i} \), we have

\[
\beta_{i+1} - \beta_i = \delta_i, \quad (i = 1, \ldots, L - 1), \quad \prod_{i=1}^L \beta_i = H,
\]

where \( H = \prod_{i=1}^L \|q_i\|^2 \). In particular, all \( \beta_i \) are determined by \( \beta_1 \); \( \beta_{i+1} = \beta_1 + \delta_1 + \cdots + \delta_i \) for all \( i = 1, \ldots, L - 1 \). Hence, we obtain one polynomial equation of degree \( L \) in \( \beta_1 \), namely

\[
\beta_1 \cdot (\beta_1 + \delta_1) \cdots (\beta_1 + \delta_1 + \cdots + \delta_{L-1}) - H = 0. \tag{19}
\]

There are finitely many real solutions for \( \beta_1 \) in eq. (19) such that \( \beta_1, \ldots, \beta_1 + \delta_1 + \cdots + \delta_{L-1} \) are all positive. Each such solution gives finitely many choices for the \( \kappa_i \).

The previous statement can be used to study the impact of the initialization (within the fiber) on the training dynamics. More precisely, we can compute the Riemannian metric that locally characterizes the gradient dynamics in function space. In general if \( L = \ell \circ \mu \) and \( \theta(t) \) is an integral curve for the negative gradient field of \( L \), then setting \( \bar{W}(t) = \mu(\theta(t)) \) we have

\[
\frac{d}{dt} \bar{W}(t) = -K(\theta(t)) \cdot \nabla \ell(\bar{W}(t)), \tag{20}
\]

where \( K(\theta) = \text{Jac}_\mu(\theta) \text{Jac}_\mu(\theta)^\top \) and \( \text{Jac}_\mu(\theta) \) is the Jacobian matrix of \( \mu \) at \( \theta \). The matrix \( K(\theta) \) corresponds to the neural tangent kernel \([23]\) popular in the analysis of neural networks and can be viewed as a “pre-conditioning” matrix that modifies the gradient of \( \ell \) in function space. If \( K(\theta) \) could be written as a function of \( \bar{W} = \mu(\theta) \), then we could view eq. (20) as a Riemannian gradient flow for the metric tensor \( K^{-1} \) (this assumes \( K \) is not singular, which is generically the case; note that \( K \) is always positive semi-definite). In general, however, the matrix \( K(\theta) \) depends on the parameter \( \theta \), and different parameters within the fiber of \( \mu \) will yield different matrices. In particular, the trajectories in function space are susceptible to the relative scalings of the filters at initialization. Using the dynamical invariants, however, we can bypass this problem by restricting ourselves to the parameters that are dynamically reachable, defining a local metric in function space associated with the initialization. We show this idea with a simple example.

**Example 5.15.** Let us consider the factorization of a cubic polynomial into a linear and a quadratic factor. We write \( C = A \cdot B \), where

\[
C = c_3 x^3 + c_2 x^2 y + c_1 xy^2 + c_0 y^3, \quad A = a_2 x^2 + a_1 xy + a_0 y^2, \quad B = b_1 x + b_0 y.
\]
The Jacobian matrix of the parameterization map \( \mu : (a_2, a_1, a_0, b_1, b_0) \mapsto (c_3, c_2, c_1, c_0) = (a_2 b_1, a_2 b_0 + a_1 b_1, a_1 b_0 + a_0 b_1, a_0 b_0) \) is

\[
\text{Jac}_\mu(a_2, a_1, a_0, b_1, b_0) = \begin{bmatrix} b_1 & 0 & 0 & a_2 & 0 \\ b_0 & b_1 & 0 & a_1 & a_2 \\ 0 & b_0 & b_1 & a_0 & a_1 \\ 0 & 0 & b_0 & 0 & a_0 \end{bmatrix} \in \mathbb{R}^{4 \times 5}.
\]

The matrix \( K(\theta) = \text{Jac}_\mu \text{Jac}^\top_\mu \in \mathbb{R}^{4 \times 4} \) can be written as \( K(\theta) = K_B + K_A \), where

\[
K_B = \begin{bmatrix} b_1^2 & b_0 b_1 & 0 & 0 \\ b_0 b_1 & b_0^2 + b_1^2 & b_0 b_1 & 0 \\ 0 & b_0 b_1 & b_0^2 + b_1^2 & b_0 b_1 \\ 0 & 0 & b_0 b_1 & b_0^2 \end{bmatrix}, \quad K_A = \begin{bmatrix} a_2^2 & a_1 a_2 & a_0 a_2 & 0 \\ a_1 a_2 & a_1^2 + a_2^2 & a_0 a_1 + a_1 a_2 & a_0 a_2 \\ a_0 a_2 & a_0 a_1 + a_1 a_2 & a_0^2 + a_1^2 & a_0 a_1 \\ 0 & a_0 a_2 & a_0 a_1 & a_0^2 \end{bmatrix}.
\]

This matrix clearly depends on the coefficients of \( A \) and \( B \), however we can recover it only if \( C \) and the value of \( \delta = \|A\|^2 - \|B\|^2 \), which is fixed from the initialization. To resolve the scale ambiguity, we replace \( A, B \) with \( \gamma A, 1/\gamma B \) with \( \gamma \in \mathbb{R} \setminus \{0\} \) (here \( A \) and \( B \) can be fixed arbitrarily so that \( A \cdot B = C \)). We see that \( K(\theta) \) becomes a one-dimensional (quadratic) family of matrices \( K(\gamma) = (1/\gamma)^2 K_B + \gamma^2 K_A \). Using \( \|\gamma A\|^2 - \|1/\gamma B\|^2 = \delta \) and writing \( \beta = \|1/\gamma B\|^2 \) and \( H = \|A\|^2 \) we deduce that

\[
\beta^2 + \beta \delta - H = 0, \quad \beta > 0, \beta + \delta > 0, H > 0,
\]

which is the same as eq. (19). This yields \( \beta = \frac{-\delta + \sqrt{\delta^2 + 4H}}{2} \) and \( \gamma^2 = \frac{2\|B\|^2}{-\delta + \sqrt{\delta^2 + 4H}} \).

## 6 Optimization of the square loss

In this section, we investigate the minimization of quadratic losses and the Euclidean distance degrees for LCNs.

### 6.1 The square loss

Given data \( X \in \mathbb{R}^{d_0 \times N} \) and \( Y \in \mathbb{R}^{d_2 \times N} \) (where \( N \) is the number of data samples), and assuming that \( XX^\top \) has full rank, we write the quadratic loss as:

\[
\ell(W) = \ell_{X,Y}(W) = \|WX - Y\|^2 = \langle Y, Y \rangle - 2\langle WX, Y \rangle + \langle WX, WX \rangle
\]

\[
= \text{const.} + 2\langle WX(XX^\top)^{-1}X^\top, Y \rangle + \langle WX, WX \rangle
\]

\[
= \text{const.} + 2\langle U, Y \rangle_{XX^\top} + \langle WX, WX \rangle_{XX^\top}
\]

\[
= \text{const.} + \|WX - U\|^2_{XX^\top},
\]

where \( U \) is the unconstrained optimizer of this quadratic problem, over the set of all matrices:

\[
U = \arg\min_{U \in \mathbb{R}^{d_2 \times d_0}} \|WX - Y\|^2 = YX^\top(XX^\top)^{-1}.
\]

As discussed in the previous sections, an LCN corresponds to a semi-algebraic set \( M \) of convolutional matrices \( \mathbf{W} \). By eq. (21), the optimization of the square error \( \ell(W) \) over the function space \( M \) corresponds to a constrained linear regression problem, which is equivalent to solving

\[
\min_{W \in M} \|U - W\|^2_{XX^\top}.
\]

23
If $U \in \mathcal{M}$, then the solution is $U$. If $U \notin \mathcal{M}$, we can optimize the objective over the boundary of the function space. We also may assume that the matrix $U$ is convolutional of the same format as the matrices in $\mathcal{M}$, for if not, we can consider the orthogonal projection (with respect to $\| \cdot \|_{X^TX}$) to the appropriate linear space of convolutional matrices.

**Remark 6.1.** The same expressions as above hold for population losses, by replacing $\frac{1}{N}XX^T = \frac{1}{N} \sum_{i=1}^{N} x_i x_i^T$ with $E_{x \sim D}[xx^T]$ where $D$ is the distribution over the input data.

If we assume that $U$ is a convolutional matrix with filter $u$, then we can also write the square loss in terms of the filter $\overline{w}$ of $\overline{W}$:

$$
\ell(\overline{w}) = \| \overline{W} - U \|_{XX^T}^2 = \text{tr}[(\overline{W} - U)XX^T (\overline{W} - U)^\top] = \sum_{m=0}^{d_L-1} [(\overline{W} - U)_{m,:}XX^T (\overline{W} - U)^\top_{m,:}],
$$

(22)

where we write $M_{m,:}$ for the $m$-th row of a given matrix $M$. We denote by $k$ and $s$ the filter size and stride of the convolutional matrices in the function space $\mathcal{M}$. Since the $m$-th row of $\overline{W} - U$ is obtained by shifting $\overline{w} - u$ by $sm$ positions to the right, we see that

$$
\ell(\overline{w}) = (\overline{w} - u)^\top \tau(XX^T)(\overline{w} - u),
$$

(23)

where $\tau : \mathbb{R}^{d_0 \times d_0} \to \mathbb{R}^{k \times k}$ is an operator that sums all shifts of rows and columns of a matrix:

$$
\tau(M)_{ij} = \sum_{m=0}^{d_L-1} M_{i+sm,j+sm}.
$$

(24)

**Remark 6.2.** If we consider circulant matrices instead of convolutional matrices, then all matrices appearing in eq. (22) would be square matrices of size $d_0 \times d_0$ and the trace would sum from $m = 0$ to $d_0 - 1$. The sum in eq. (24) would change accordingly and would be cyclic.

**Lemma 6.3.** Let $XX^T$ be a multiple of the identity matrix. For any stride $s$, the square loss $\ell$ on the filters is equivalent to the Euclidean distance between $\overline{w}$ and $u$, i.e., $\ell(\overline{w}) \propto \| \overline{w} - u \|^2$.

**Proof.** If $XX^T$ is a multiple of the identity, then $\tau(XX^T) = d_L XX^T$ for convolutional matrices resp. $\tau(XX^T) = d_0 XX^T$ for circulant matrices. Now the claim follows from eq. (23). \qed

**Remark 6.4.** The lemma above holds both for convolutional and circulant matrices. If we restrict ourselves to the circulant setting, then the statement holds more generally if $XX^T$ is an arbitrary diagonal matrix as long as $s$ and $d_0$ are coprime. This follows from Theorem 6.2. Indeed, if $s$ and $d_0$ are coprime, $\tau(XX^T)_{ij}$ sums over all entries of the diagonal matrix $XX^T$ and so $\tau(XX^T)$ is a multiple of the identity matrix.

We can apply Theorem 6.3 also for population losses (see Theorem 6.1) where the data distribution $D$ is such that the covariance matrix $E_{x \sim D}[xx^T]$ is a multiple of the identity (e.g., Gaussian data), and conclude that using the square loss on the LCN is equivalent to using a standard Euclidean distance at the level of filters. If we make use of circulant convolutions and restrict ourselves to a stride that is coprime to $d_0$, then this is also the case if the covariance matrix is simply diagonal.

Next we give a detailed example illustrating the critical points of the quadratic loss for an LCN and how these depend on the particular training data. In the next subsection we will discuss the number of critical points in more generality.
Example 6.5. We return to Theorem 2.4 to consider minimizing the square loss over the set of functions \( \mathcal{M} = \{ W = (A, B, C) : B^2 \geq 4AC \} \) represented by the two layer LCN with architecture specified by \( d = (3, 2, 1), k = (2, 2), s = (1, 1) \). Given data matrices \( X, Y \), we are minimizing an objective \( \ell(W) = \|WX - Y\|^2 \) which is convex over the space of all matrices \( W \in \mathbb{R}^{1 \times 3} \). Hence, if the unconstrained minimizer is outside \( \mathcal{M} \) and \( XX^\top \) is full rank, then the set of minimizers over \( \mathcal{M} \) will be at the boundary of \( \mathcal{M} \). The situation is illustrated in Figure 1. The boundary \( \partial \mathcal{M} = \{ W = (A, B, C) : B^2 = 4AC \} \) can be parametrized by \( (A, C) \). The case where both non-negative or both non-positive. In the following we consider the case \( A, C \geq 0 \) (the case \( A, C \leq 0 \) is similar). The gradient of the objective function with respect to \( A, C \) is

\[
\nabla_{A,C} \ell(A, C) = (-2YX^\top + 2WXX^\top)\nabla_{A,C} W^\top.
\]

The first order criticality condition \( \nabla_{A,C} \ell(A, C) = 0 \) means that

\[
-YX^\top + WXX^\top = \lambda WJ,
\]

where we introduce the matrix \( J = \begin{bmatrix} 0 & 0 & 1 \\ 0 & -1/2 & 0 \\ 1 & 0 & 0 \end{bmatrix} \), so that \( WJ = [C, -\sqrt{AC}, A] \) is a basis of the left null space of \( \nabla_{A,C} W^\top \). We can solve eq. (25) by writing

\[
W = YX^\top (XX^\top - \lambda J)^{-1},
\]

and plugging this into the defining equation of the boundary, \( B^2 = 4AC \), which gives us an equation in \( \lambda \). As discussed in [28, Section 4.4], computing a square distance from a point to a quadric surface results in a degree 6 polynomial in \( \lambda \). Since our boundary equation is homogeneous, we can clear the determinant in Cramer’s rule for the matrix inverse and use simply the adjugate matrix, which results in a degree 4 polynomial in \( \lambda \):

\[
(YX^\top \text{adj}(XX^\top - \lambda J)_{:,2})^2 - 4(YX^\top \text{adj}(XX^\top - \lambda J)_{:,1})(YX^\top \text{adj}(XX^\top - \lambda J)_{:,3}).
\]

Solving this, we typically obtain 4 values of \( \lambda \), which when inserted in eq. (26) gives us the critical points in the boundary of the function space.

Now we take a closer look at the special case with \( XX^\top = I \). In this case, the unconstrained optimal solution is \( U = (u_1, u_2, u_3) = YX^\top \) and eq. (27) simplifies to

\[
(u_2^2 - u_1 u_3)\lambda^4 + (-u_1^2 - 4u_1 u_3 - u_3^2)\lambda^3 + (-4u_1^2 - 2u_2^2 - 5u_1 u_3 - 4u_3^2)\lambda^2 \\
+ (-4u_1^2 - 4u_1 u_3 - 4u_3^2)\lambda + (u_2^2 - 4u_1 u_3).
\]

We classify the points \((u_1, u_2, u_3)\) for which the problem has different numbers of solutions. When \( u_2^2 - u_1 u_3 \neq 0 \), eq. (28) is a degree 4 polynomial in \( \lambda \) which generally has 4 or 2 real solutions (the global minimum corresponds to a real root). The cases are separated by the discriminant of eq. (28) with factors \( u_2^2, (u_1 - u_3)^2 \), and

\[
32u_1^6 + 435u_1^4 u_2^2 + 384u_1^2 u_2^4 + 256u_2^6 - 240u_1^5 u_3 - 960u_1^3 u_2 u_3 - 960u_1^4 u_3^2 + 696u_1^4 u_3^2 \\
+ 1098u_1^2 u_2^2 u_3^2 - 980u_1^3 u_3^2 - 960u_1^2 u_2^2 u_3^2 + 696u_1^2 u_3^4 + 435u_2^4 u_3^4 - 240u_1 u_3^5 + 32u_3^6.
\]

We visualize the situation in Figure 1. The figure shows the slice \( u_3 = -u_1 + 1 \). Since the discriminant is homogeneous, each point in the slice is representative of a line through the origin in \( \mathbb{R}^3 \). The zeros of the factor \((u_1 + u_3)^2\) are not visible over this slice. We see that the discriminant separates 12 generic cases outside of the function space (i.e., inside the blue ellipse, where \( u_2^2 - u_1 u_3 < 0 \)). By evaluating individual instances in each of these cases, we find that the points inside the green curvy diamond (caustic) have two local minima and two saddles, the points between the green curvy diamond and the red circle have one minimum and one saddle, and the points between the red circle and the blue ellipse have two local minima (whereby we observed that only one of them satisfies \( A, C \geq 0 \)).
Figure 1: Left: Boundary of the function space in $\mathbb{R}^{1 \times 3}$ from Theorem 2.4. The complement of the function space is inside the elliptic double cone with axis $(1, 0, 1)$. The boundary is colored by values of the square loss for random data $X, Y$ with $U = YX^\top (XX^\top)^{-1}$ shown as a red dot. For this data there are two local minima and two saddles shown as green dots. Right: Discrimination of datasets with $XX^\top = I$ in terms of $U = (u_1, u_2, u_3) = YX^\top$ by the types of critical points of the square loss as discussed in Theorem 6.5. Shown is the slice $u_3 = -u_1 + 1$. For points with $u_2^2 - 4u_1u_3 < 0$ outside the function space, we have cases with two local minima and two saddles, one minimum and one saddle, and two local minima.

6.2 Euclidean distance degrees

We now discuss the number of critical points of the optimization problem in function space using the square loss eq. (21). This problem is given by

$$\min_{W \in \mathcal{M}} \|U - W\|^2_{XX^\top},$$

where $\mathcal{M}$ is the function space of an LCN and $U$ and $X$ are fixed data matrices. We focus on one-dimensional convolutions with stride one. We first assume that $XX^\top$ is the identity matrix (i.e., we are interested in minimizing the standard Euclidean distance of the data $U$ to the function space $\mathcal{M}$) and comment on other choices of $X$ later.

By Theorem 5.2, each critical point in the parameter space of the LCN corresponds in function space to a critical point of the Euclidean distance from the data $U$ to a multiple root locus $\Delta_\lambda$. For almost every choice of $U$ the number of complex critical points of minimizing the Euclidean distance from $U$ to $\Delta_\lambda$ is the same. This number is known as the Euclidean distance degree (short ED degree) of $\Delta_\lambda$ [15]. Hence, if we assume that the data $U$ is generic, Theorem 5.5 shows that the number of $\mu(\theta)$, where $\theta$ is a critical point in the parameter space of the given LCN, is upper bounded by the sum of the ED degrees of all multiple root loci $\Delta_\lambda \subset \mathbb{R}[x, y]_{k-1}$, where $k$ is the filter size in the function space $\mathcal{M}$ and $\lambda = \lambda_{\rho|\gamma}$ comes from an rmp $(\rho \mid \gamma)$ that is compatible with the LCN architecture.

Example 6.6. For generic data $U$, Theorem 5.6 shows the following: The number of $\mu(\theta)$, where $\theta$ is a critical point of an LCN with architecture $k = (3, 2, 2)$ resp. $k' = (4, 2)$ is at most the sum of the ED degrees of $\Delta_{(1,1,1,1)}, \Delta_{(1,1,2)}, \Delta_{(2,2)}, \Delta_{(1,3)}$ resp. $\Delta_{(1,1,1,1)}, \Delta_{(1,1,2)}$.

The state-of-the-art on ED degrees of multiple root loci is presented in [27]. For instance, the ED degree of the discriminant hypersurface of degree-$(k - 1)$ polynomials is $3(k - 1) - 2$ [15, Example
5.12]. More generally, the ED degree of the multiple root locus $\Delta_{(\alpha,1,\ldots,1)}$ (with $\alpha > 1$ and $k - 1 - \alpha$ ones) is $(2\alpha - 1)(k - 1) - 2(\alpha - 1)^2$ [27, Theorem 5.1]. Many more ED degrees are computed in [27, Table 1], but a closed formula for general partitions is not known.

**Example 6.7.** $(k = 5)$ The discriminant of quartic polynomials has ED degree 10. Its iterated singular loci are the following multiple root loci: $\Delta_{(2,2)}$ with ED degree 13, $\Delta_{(1,3)}$ with ED degree 12, and $\Delta_{(4)}$ with ED degree 10. Hence, the upper bounds on the number of $\mu(\theta)$ for critical points $\theta$ given in Theorem 6.6 are $1 + 10 + 13 + 12 = 36$ resp. $1 + 10 = 11$.

**Remark 6.8.** The ED degrees discussed in this section are often referred to as *generic ED degrees* in the algebraic literature. The *special ED degree* refers to the following slightly different norm to be optimized: for two polynomials $P = \sum_{i=0}^{k-1} w_i x^i$ and $Q = \sum_{i=0}^{k-1} u_i x^i$,

$$\langle P, Q \rangle_B := \sum_{i=0}^{k-1} \frac{i!(k-i-1)!}{(k-1)!} w_i u_i.$$  (30)

The letter “$B$” stands for *Bombieri norm*. For example, if $P$ and $Q$ are quadratic polynomials, then $\langle P, Q \rangle_B = u_2 w_2 + \frac{1}{2} u_1 w_1 + u_0 w_0$. The Euclidean inner product that arises for LCNs if $XX^\top$ is the identity is instead simply $\langle P, Q \rangle_2 = u_2 w_2 + u_1 w_1 + u_0 w_0$. This slight difference has an impact on the ED degrees. Indeed, the ED degree of the discriminant hypersurface $\{(w_0, w_1, w_2) \colon w_1^2 = u_0 w_0\}$ is 2 for the Bombieri norm, and 4 for the standard Euclidean norm (see Example 6.5 for a computation of the four critical points). More generally, if we consider the multiple root loci for polynomials of degree $k - 1$, then the special ED degree of the discriminant hypersurface is $k - 1$ [15, Corollary 8.7] (whereas the generic ED degree is $3(k - 1) - 2$, as noted above). The special ED degree of the multiple root locus $\Delta_{(\alpha,1,\ldots,1)}$ (with $\alpha > 1$ and $k - 1 - \alpha$ ones) is $k - 1$, independently of $\alpha$ [27, Theorem 5.1].

**Example 6.9.** $(k = 5)$ The generic ED degrees of the multiple root loci of quartic polynomials are listed in Example 6.7. The special ED degrees are smaller: for $\Delta_{(1,1,2)}$ it is 4, for $\Delta_{(2,2)}$ it is 7, for $\Delta_{(1,3)}$ it is 4, and for $\Delta_{(4)}$ it is 4.

**Remark 6.10.** The above discussion shows that the Bombieri norm can be thought of as a “special” norm that leads to lower ED degrees and thus makes the optimization landscape simpler. Perhaps counterintuitively, the standard Euclidean distance is instead *generic*, in the sense that almost all pairings $\langle \cdot, \cdot \rangle$ will lead to the same ED degrees for multiple-root loci. In other words, for *almost all* data matrices the ED degrees of the multiple root loci would be the same as assuming $XX^\top$ is the identity.

A natural question in this setting is if there exists a “special” data distribution that induces the Bombieri norm eq. (30) in polynomial space. For circulant convolutions this is not possible. Indeed, for any matrix $M$, the matrix $\tau(M)$ when $d_0 = d_L$, will always have constant diagonal entries by the same reasoning as in Remark 6.4. In the convolutional (Toeplitz) case, however, such special data distributions can exist, as the following simple example demonstrates.

**Example 6.11.** We consider a single convolutional layer with $k = 3$, $d_0 = 4$ and $d_1 = 2$. If the data covariance matrix is such that

$$XX^\top = \begin{bmatrix} 3/4 & 1/4 & 1/4 \\ 1/4 & 1/4 & 3/4 \end{bmatrix},$$

then the induced metric in the space of quadratic polynomials is indeed the Bombieri norm $\langle P, Q \rangle_B = u_2 w_2 + \frac{1}{2} u_1 w_1 + u_0 w_0$. This example suggests that non-standard normalizations of the data might lead to more favorable optimization landscapes for convolutional networks.
Example 6.12. We return to Example 6.5, an LCN with $M = \{W = (A, B, C): B^2 \geq 4AC\}$, and investigate the structure of critical points of the square loss over the boundary $\partial M = \{W = (A, B, C): B^2 = 4AC\}$ when $XX^T$ is compatible with the Bombieri norm,

$$XX^T = \begin{bmatrix} 1 & 1/2 \\ 1/2 & 1 \end{bmatrix}. \tag{31}$$

A geometric interpretation of this setting is that the balls $\{W: \|W - U\|_{XX^T} \leq c\}$ are ellipsoids with the same $1 \times 2 \times 1$ aspect ratio as the boundary of the function space. As before, we write $U = (u_1, u_2, u_3) = Y X^T (X X^T)^{-1}$ for the unconstrained least squares solution. With $XX^T$ taking the form eq. (31), equation eq. (27) for the case $A, C \geq 0$ becomes

$$(-1)(\lambda + 1)(\lambda + 1)((u_1 u_3 - 4 u_2^2)\lambda^2 + (u_1^2 + 8 u_2^2 + u_3^2)\lambda + (u_1 u_3 - 4 u_2^2)). \tag{32}$$

This quartic polynomial in $\lambda$ has a repeated root $\lambda = -1$ for any values of $U$ and hence its discriminant vanishes identically. The distinction in terms of $U$ is determined by the last factor alone. We discuss the zeros of the individual factors in turn.

- Consider the factor $(\lambda + 1)$ of eq. (32). The root $\lambda = -1$ plugged into eq. (25) gives equation $W (XX^T + J) = Y X^T$ for $W$. Since the matrix $XX^T + J = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 1 \end{bmatrix}$ is not of full rank, for generic $Y X^T$ there is no solution and this case can be ignored. Solutions only exist in the non-generic case that $Y X^T = c \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$, namely $\{W = (A, B, C): A + C = c\}$, which is a hyperplane. The intersection of this hyperplane with the boundary of the function space gives us the set of critical points $\{(A, B, C): A + C = c, B^2 = 4AC\}$. For the assumed case $A, C \geq 0$, this is empty over the reals for $c < 0$, a single point for $c = 0$, and an ellipse for $c > 0$. All critical points have the same Bombieri distance $c$ from $U = \begin{bmatrix} c & 0 & -c \end{bmatrix}$.

- Consider now the last factor of eq. (32), which is quadratic in $\lambda$. Its discriminant has factors $(u_1 + u_3)^2$ and $u_1^2 - 2 u_1 u_3 + 16 u_2^2 + u_3^2$. The former has zero set $\{u_1 = -u_3\}$, which also appeared in the case $XX^T = I$ and is inside the function space. The latter has real zero set $\{u_2 = 0, u_1 = u_3\}$, which is the center line of our boundary cone $\partial M$ and can be interpreted as a degeneration of the caustic that appeared in the case $XX^T = I$. It also corresponds to the non-generic case discussed above.

Hence, when $XX^T$ takes the form eq. (31) compatible with the Bombieri norm, there is a single generic case for the datasets. We find that generic datasets $U = (u_1, u_2, u_3)$ outside the function space have two critical points over the boundary of the function space: 1 global minimum and 1 saddle. This is in contrast with the more complicated case distinction in Example 6.5 and Figure 1 when $XX^T = I$, where some generic datasets led to 4 critical points. On the other hand, for the Bombieri norm there are special datasets, with $\{u_2 = 0, u_1 = u_3 > 0\}$, with infinitely many critical points, namely entire ellipses of global minima.

7 Examples and numerical evaluation

We present computational experiments on LCNs with 1D convolutions with stride one. In theorem 5.1 we saw that critical parameters are global minimizers of the objective or are contained in the discriminant hypersurface. In fact, in Theorem 5.2 we observed that any critical parameter corresponds to a critical point over the locus with the same root structure. We discuss the relevant root loci and evaluate their appearance in gradient optimization using different LCN architectures. As observed in the previous sections, the order of the filters does not affect the function space nor the gradient dynamics. For this reason, we consider architectures with filter widths $k_1 \geq \cdots \geq k_L$. 

28
We perform our experiments with the five non-filling architectures from Theorem 4.6. The left hand side of Figure 1 visualizes the function space \( M_{(2,2)} \). The function spaces representing cubic and quartic polynomials are depicted in Figure 2: On the left, \( M_{(2,2)} \) is the “inside” of the “cuspidal wedge.” On the right, the complement of \( M_{(2,2)} = M_{(4,2)} \) are the points above the blue “bowl.” These points correspond to one of the two convex cones described in theorem 4.7. The other cone is not visible since the figure fixes the highest degree coefficient to 1. Moreover, \( M_{(2,2,2,2)} \) is the set inside the triangular region.

Figure 2: Left: The surface \( \Delta = 0 \) of coefficients \((a, b, c, d)\), \( a \neq 0 \), for which a cubic polynomial \( ax^3 + bx^2y + cxy^2 + dy^3 \) has a double root. The coefficients with triple roots are shown in red. The coefficients with \( \Delta > 0 \) (the interior of \( M_{(2,2)} \)) are on the side with smaller values of \( c \). Right: The surfaces \( \delta = 0 \) and \( \delta' = 0 \) (as in Theorem 3.3) of coefficients \((p, q, r)\), that separate the real root multiplicities of a depressed quartic polynomial \( x^4 + px^2y^2 + qxy^3 + ry^4 \). The positive values of \( \delta \) and \( \delta' \) are on the sides that do not contain the labels. The complement of \( M_{(3,2,2)} \) is above the blue bowl. The interior of \( M_{(2,2,2,2)} \) is within the triangular blue region.

7.1 Critical points reached by gradient descent

For each of the architectures listed above, we generated 10,000 random data sets \( X, Y \) each consisting of 10 iid input-output pairs from a standard normal distribution, with inputs in \( \mathbb{R}^{d_0} \) and outputs in \( \mathbb{R} \). Since \( d_0 = 1 \), we have from Theorem 2.2 that \( d_0 = k \). For each dataset and architecture, we initialized the parameters at random from a standard normal distribution and ran gradient descent optimization of the square loss \( L \) with fixed step size \( 0.01 \) and stopping criterion \( \|\nabla L\|^2 \leq 10^{-14} \). Instances which did not reach the stopping criterion within a maximum of 15,000 iterations were discarded. Since the final coefficients are only approximately critical points, they rarely correspond to polynomials with exactly repeated roots. To assess the rrmp’s of solutions, we compute the roots of all factor filters and consider a root \( r \) as real if \( |r - \text{real}(r)| \leq \text{tol} \cdot |r| \), and consider two roots \( r, r' \) as same if \( |r - r'| \leq \text{tol} \cdot \max\{|r|, |r'|\} \) with \( \text{tol} = 10^{-4} \). We use a similar procedure for the initializations and for the target. This classification scheme is relatively robust, but is not exact. The results are shown in Table 1.

- For the architecture \( k = (2, 2) \), about 66% of the datasets correspond to targets with rrmp \( 11|0 \) inside the function space, and gradient descent finds a solution with nearly zero training error. About 34% of the datasets correspond to targets with rrmp \( 0|1 \) outside the function space and gradient descent finds solutions with rrmp \( 2|0 \) at its boundary.
Figure 3: The square loss \((w_1, \ldots, w_L) \mapsto \|U - \mu(w_1, \ldots, w_L)\|^2_{XX^T}, U = YX^T(XX^T)^{-1}\), for random data \(X, Y\), plotted in log-scale over random 2D affine subsets of parameter values. The bottom shows the discriminant at the corresponding filters, \((w_1, \ldots, w_L) \mapsto |\Delta(\pi(\mu(w_1, \ldots, w_L)))|\). As can be seen, deeper networks tend to have a more intricate loss surface and critical parameters often correspond to functions where the discriminant vanishes.

- For \(k = (2, 2, 2)\), about 25% of the datasets have \(\text{rrmp } 1\{1\}\) inside the function space, and for most of these, gradient descent finds a solution of the same \(\text{rrmp}\) with nearly zero loss. About 75% of the datasets have \(\text{rrmp } 1\{0\}\) outside the function space. Of these, about 70% led to a solution with \(\text{rrmp } 1\{2\}\), i.e., a smooth point of the discriminant hypersurface, and about 30% led to a solution with \(\text{rrmp } 3\{0\}\), i.e., a point on the red curve in Figure 2.

- For the architectures \(k = (4, 2)\) or \((3, 2, 2)\), which have the same function space, the different parametrizations lead to different optimization behaviors. First, the initialization procedure leads to different distributions in function space, with \(k = (4, 2)\) being more frequently initialized with \(\text{rrmp } 1\{1\}\) than \(k = (3, 2, 2)\). For targets with \(\text{rrmp } 0\{1\}\), outside the function space, the solutions often correspond to zeros of the discriminant and are critical points in parameter space but they do not lie on the Euclidean boundary of \(M_k\). Hence, they are spurious critical points. The effect is more pronounced in the deeper architecture, which has a more intricate parametrization. The optimization landscapes are illustrated in Figure 3. For targets with \(\text{rrmp } 1\{1\}\), inside the function space, a good fraction of the optimization instances with \(k = (4, 2)\) converged to sub-optimal solutions with \(\text{rrmp } 1\{2\}\) or \(2\{1\}\), and for \(k = (3, 2, 2)\) an even larger fraction converged to sub-optimal solutions with \(\text{rrmp } 1\{2\}, 2\{2\}, 0, 13\{0\}, 0, 2\{1\}\). This is in line with our theoretical discussion, by which \(k = (3, 2, 2)\), but not \(k = (4, 2)\), can have critical points in parameter space with \(\text{rrmp}'s\) \(2\{2\}\) and \(13\{0\}\) for generic data (see Theorem 5.5 and Theorem 5.6).

- For \(k = (2, 2, 2, 2)\), in contrast to the previous architectures, a fraction of the targets outside the function space (i.e., with \(\text{rrmp } 1\{1\}\) or \(0\{11\}\)) led to a solution with \(\text{rrmp } 4\{0\}\), i.e., the codimension 3 point at the origin in Figure 2, which was predicted by Theorem 5.5.

### 7.2 Euclidean vs. Bombieri norm

In this experiment we generated 500 random targets \(U\) for each architecture. For each of them, we ran gradient descent with 50 random initializations for the loss with Euclidean and Bombieri norms. For each target we recorded the list of converged solutions, using the settings of the previous experiment. To determine the number of distinct solutions we considered two filters \(w_i\) and \(w_{i'}\) as equal if on each entry
Table 1: Rrmp’s of solutions returned by gradient descent optimization of the square loss in parameter space for random datasets and initializations. The table omits a small fraction of targets and initializations numerically classified in rrmp’s of positive codimension.

| $\mathcal{M}_{(2,2)}$ | target | % | initial 11|0 | solution | % | 100% | mean loss |
|------------------------|--------|---|-----------|---|----------|---|------|-----------|
|                         | 11|0   | 65.8      | 11|0 | 100      | 1.15e-15 |
|                         | 0|1   | 34.2      | 2|0 | 100      | 0.185    |

| $\mathcal{M}_{(2,2,2)}$ | target | % | initial 11|1|0 | solution | % | 100% | mean loss |
|------------------------|--------|---|-----------|---|----------|---|------|-----------|
|                         | 11|1   | 24.9      | 11|0 | 99.7     | 2.32e-15 |
|                         | 12|0   | 0.051     | 94.5|
|                         | 3|0   | 0.205     | 1.19 |

| $\mathcal{M}_{(4,2)}$ | target | % | initial 11|1|1|0 | solution | % | 24.6% | mean loss |
|------------------------|--------|---|-----------|---|----------|---|------|-----------|
|                         | 11|1   | 72.6      | 11|0 | 15.5     | 0.228    |
|                         | 11|1   | 83.2      | 1.94e-15|
|                         | 2|1   | 1.36      | 0.54    |

| $\mathcal{M}_{(3,2,2)}$ | target | % | initial 11|1|1|0 | solution | % | 65.1% | mean loss |
|------------------------|--------|---|-----------|---|----------|---|------|-----------|
|                         | 11|1   | 72.9      | 11|0 | 27.1     | 0.221    |
|                         | 22|0   | 1.28      | 0.992    |
|                         | 13|0   | 25.8      | 0.798    |
|                         | 11|1   | 45.5      | 1.78e-15 |
|                         | 2|1   | 0.381     | 0.446    |

| $\mathcal{M}_{(2,2,2,2)}$ | target | % | initial 11|1|1|1|0 | solution | % | 99.8% | mean loss |
|--------------------------|--------|---|-----------|---|----------|---|------|-----------|
|                         | 11|1   | 72.7      | 11|0 | 33.3     | 0.259    |
|                         | 22|0   | 2.24      | 0.908    |
|                         | 13|0   | 52.1      | 0.774    |
|                         | 4|0   | 12.4      | 1.53     |

| $\mathcal{M}_{(2,2,2,2,2)}$ | target | % | initial 11|1|1|1|1|0 | solution | % | 99.8% | mean loss |
|-----------------------------|--------|---|-----------|---|----------|---|------|-----------|
|                             | 11|1   | 72.5      | 11|0 | 33.3     | 0.259    |
|                             | 22|0   | 2.24      | 0.908    |
|                             | 13|0   | 52.1      | 0.774    |
|                             | 4|0   | 12.4      | 1.53     |
Table 2: Percentages of targets for which gradient descent with different initializations converged to a given number of distinct solutions. For the Bombieri norm $B$ we usually observe a smaller number of distinct solutions compared with the generic Euclidean norm $I$.

| $k$    | metric | number of distinct solutions |
|--------|--------|-----------------------------|
|        |        | 1  | 2  | 3  | 4  |
| (2, 2) | $I$    | 95.8 | 4.2 |      |    |
|        | $B$    | 100 |      |      |    |
| (2, 2) | $I$    | 84.7 | 14.5 | 0.84 |    |
|        | $B$    | 95.6 | 1.1  | 3.29 |    |
| (4, 2) | $I$    | 66.3 | 33.7 |      |    |
|        | $B$    | 73.8 | 26.2 |      |    |
| (3, 2) | $I$    | 22.5 | 58.8 | 17.7 | 1.01|
|        | $B$    | 32  | 48.3 | 19.5 | 0.209|
| (2, 2, 2) | $I$    | 70.7 | 26.5 | 2.61 | 0.217|
|        | $B$    | 84.8 | 8.21 | 6.97 |    |

$|w_{jk} - w_{ivk}| ≤ tol \cdot \max\{\max\{|w_{jk}| : j\}, 1\}$ with $tol = 10^{-4}$. The results are shown in Table 2. In line with our theoretical discussion in Section 6.2 we observed fewer distinct solutions for the Bombieri norm than for the generic norm.

7.3 Symbolic computations

We illustrate several of our results with a final example. We consider linearly generated data $D_N = \{(x_i, y_i) \in \mathbb{R}^5 \times \mathbb{R} : i = 1, \ldots, N, y_i = u^\top x_i\}$ where $u = [2, 0, 5, 0, 2]^\top \in \mathbb{R}^5$ is a “target filter” and $x_i$ is sampled from a standard Gaussian distribution. We study the square loss regression problem for different 1D LCN architectures with compatible dimensions, as described in Section 6. In light of Theorem 6.3, if the data is normalized so that the empirical covariance $XX^\top$ is the identity, or if $N \to \infty$, then this corresponds to an objective function $\mathcal{L}(\theta) = \ell(\mu(\theta)) = \|\mu(\theta) - u\|^2$, where $\bar{w} = \mu(\theta)$ is the end-to-end filter and $\| \cdot \|$ is the standard Euclidean norm.

The possible LCN architectures with stride one, $d_0 = 5, d_L = 1$ and non-trivial filter sizes $1 < k_i < 5$ are $k = (4, 2), (3, 3), (3, 2, 2), (2, 2, 2)$ (not considering permutations). By Theorem 4.1, only the architecture $k = (3, 3)$ is filling. The target filter corresponds to the polynomial $\pi(u) = 2x^2 + 5x^2y^2 + 2y^4 = (x^2 + 2y^2)(2x^2 + y^2)$, which does not have any real roots and is thus not contained in $\mathcal{M}_{(4, 2)}, \mathcal{M}_{(3, 2, 2)}$, nor $\mathcal{M}_{(2, 2, 2)}$.

Following our results in Section 5.1, for all architectures listed above, any critical point $\theta \in \text{Crit}(\mathcal{L})$ with $\mu(\theta) \neq u$ corresponds to polynomials with repeated roots. More precisely, if $\theta$ is a critical point for $\mathcal{L}$, then $\mu(\theta)$ is a critical point for $\ell|_{\Delta_\lambda}$, where $\ell(\bar{w}) = \|\bar{w} - u\|^2$ and $\Delta_\lambda$ is the multiple root locus where $\lambda$ describes the (complex) root pattern of $\mu(\theta)$. Based on this, we can compute the critical points for $\ell|_{\Delta_\lambda}$ for any $\lambda$ by solving algebraic systems. Recall that the number of complex critical points was predicted by the ED degrees in Theorem 6.7.

- $\lambda = (2, 1, 1)$: out of the 10 complex critical points on $\Delta_\lambda$, four are real and in fact rational, namely $[\frac{1}{3}, \pm \frac{2}{3}, \pm \frac{5}{3}, \pm \frac{1}{3}] \equiv \frac{1}{3}(x \pm y)^2(x^2 \pm 7xy + y^2), [0, 0, 5, 0, 2]^\top \equiv y^2(5x^2 + 2y^2)$, and $[2, 0, 5, 0, 0]^\top \equiv x^2(5y^2 + 2x^2)$.
- $\lambda = (3, 1)$: out of the 12 complex critical points on $\Delta_\lambda$, four are real and none are rational.
- $\lambda = (2, 2)$: out of the 13 complex critical points on $\Delta_\lambda$, five are real and three are rational: $[-1, 0, 2, 0, -1]^\top \equiv -(x - y)^2(x + y)^2, [0, 0, 5, 0, 0]^\top \equiv 5x^2y^2, \left[\frac{7}{3}, 0, \frac{15}{3}, 0, \frac{2}{3}\right]^\top \equiv \frac{7}{3}(x^2 + y^2)^2$. 

32
• \( \lambda = (4) \): out of the 10 complex critical points on \( \Delta_3 \), four are real and also rational: \([0, 0, 0, 0, 2]^T = 2y^4, [2, 0, 0, 0, 0]^T = 2x^5, [\frac{17}{35}, \pm \frac{68}{35}, \frac{102}{35}, \pm \frac{68}{35}, \frac{17}{35}]^T = \frac{17}{35} (x \pm y)^4 \).

For any choice of quartic architecture, all critical points are mapped to either \( \mu \) or one of the 17 real critical points listed above. We have verified numerically that gradient descent indeed always converges to one of these critical points. In fact, it always seems to converge to one of the rational critical points. For example, for \( k = (4, 2) \), gradient descent converges to either \([2, 0, 5, 0, 0]^T \) or \([0, 0, 5, 0, 2]^T \) which arise with \( \lambda = (2, 1, 1) \). Note that in this case we do not expect critical points with \( \lambda = (3, 1), (2, 2), (4) \) due to Theorems 5.5 and 5.6.

Finally, we remark that the previous list only describes the critical points in function space but not the actual parameters. However, as discussed in Section 5.2, it is possible recover these parameters up to a finite ambiguity given the initialization of gradient descent. For example, assume that \( k = (4, 2) \) and we initialize gradient descent at \( \theta(0) = (w_1(0), w_2(0)) = ([1, 6, 11, 6], [4, 1]) \), i.e., a factorization of the polynomial \((x + y)(x + 2y)(x + 3y) \cdot (4x + y)\). Then, following the approach outlined in Theorem 5.14, we can derive that if the gradient descent path \( \theta(t) \) converges to \( \mu(\theta) = [2, 0, 5, 0, 0]^T \) in function space, then in parameter space \( \theta(t) \) converges to one of \((w_1, w_2) = (1/\kappa[2, 0, 5, 0]^T, \kappa[1, 0]^T)\) with \( \kappa = \pm \sqrt{\frac{1}{2} (\sqrt{31445} - 177)} \approx \pm 0.4045867 \). Numerically, we verify that gradient descent converges to these filters with \( \kappa > 0 \).

**Acknowledgment**

Kathlén Kohn was partially supported by the Knut and Alice Wallenberg Foundation within their WASP (Wallenberg AI, Autonomous Systems and Software Program) AI/Math initiative. Guido Montúfar and Thomas Merkh have been supported by the European Research Council (ERC) under the European Union’s Horizon 2020 research and innovation programme (grant no 757983).

**References**

[1] Z. Allen-Zhu, Y. Li, and Z. Song, *A convergence theory for deep learning via over-parameterization*, in Proceedings of the 36th International Conference on Machine Learning, K. Chaudhuri and R. Salakhutdinov, eds., vol. 97 of Proceedings of Machine Learning Research, PMLR, 09–15 Jun 2019, pp. 242–252, [http://proceedings.mlr.press/v97/allen-zhu19a.html](http://proceedings.mlr.press/v97/allen-zhu19a.html).

[2] E. Allman, H. Banos, R. Evans, S. Hosten, K. Kubjas, D. Lemke, J. Rhodes, and P. Zwiernik, *Maximum likelihood estimation of the latent class model through model boundary decomposition*, Journal of Algebraic Statistics, 10 (2019), pp. 51–84, [https://doi.org/10.18409/jas.v10i1.75](https://doi.org/10.18409/jas.v10i1.75).

[3] D. S. Arnon, *Geometric reasoning with logic and algebra*, Artificial Intelligence, 37 (1988), pp. 37–60, [https://doi.org/10.1016/0004-3702(88)90049-5](https://doi.org/10.1016/0004-3702(88)90049-5), [https://www.sciencedirect.com/science/article/pii/0004370288900495](https://www.sciencedirect.com/science/article/pii/0004370288900495).

[4] S. Arora, N. Cohen, N. Golowich, and W. Hu, *A convergence analysis of gradient descent for deep linear neural networks*, in International Conference on Learning Representations, 2019, [https://openreview.net/forum?id=SkMqg3C5K7](https://openreview.net/forum?id=SkMqg3C5K7).
[5] S. Arora, N. Cohen, and E. Hazan, *On the optimization of deep networks: Implicit acceleration by overparameterization*, vol. 80 of Proceedings of Machine Learning Research, Stockholmsmässan, Stockholm Sweden, 10–15 Jul 2018, PMLR, pp. 244–253, http://proceedings.mlr.press/v80/arora18a.html.

[6] B. Bah, H. Rauhut, U. Terstiege, and M. Westdickenberg, *Learning deep linear neural networks: Riemannian gradient flows and convergence to global minimizers*, CoRR, abs/1910.05505 (2019), http://arxiv.org/abs/1910.05505, https://arxiv.org/abs/1910.05505.

[7] P. Baldi, *Linear learning: Landscapes and algorithms*, in Advances in Neural Information Processing Systems 1, D. S. Touretzky, ed., Morgan-Kaufmann, 1989, pp. 65–72, http://papers.nips.cc/paper/123-linear-learning-landscapes-and-algorithms.pdf.

[8] P. Baldi and K. Hornik, *Neural networks and principal component analysis: Learning from examples without local minima*, Neural Networks, 2 (1989), pp. 53–58, https://doi.org/https://doi.org/10.1016/0893-6080(89)90014-2, https://www.sciencedirect.com/science/article/pii/0893608089900142.

[9] P. Baldi and K. Hornik, *Learning in linear neural networks: A survey*, Trans. Neur. Netw., 6 (1995), p. 837–858, https://doi.org/10.1109/72.392248, https://doi.org/10.1109/72.392248.

[10] S. Basu, R. Pollack, and M.-F. Roy, *Algorithms in Real Algebraic Geometry (Algorithms and Computation in Mathematics)*, Springer-Verlag, Berlin, Heidelberg, 2006.

[11] J. F. Blinn, *How to solve a cubic equation, part I: The shape of the discriminant*, IEEE Computer Graphics and Applications, 26 (2006), pp. 84–93, https://doi.org/10.1109/MCG.2006.60.

[12] T. Ö. Çelik, A. Jamneshan, G. Montúfar, B. Sturmfels, and L. Venturello, *Optimal transport to a variety*, in Mathematical Aspects of Computer and Information Sciences, D. Salamanig, E. Tsigaridas, and Z. Zafeirakopoulos, eds., Cham, 2020, Springer International Publishing, pp. 364–381.

[13] T. Ö. Çelik, A. Jamneshan, G. Montúfar, B. Sturmfels, and L. Venturello, *Wasserstein distance to independence models*, Journal of Symbolic Computation, 104 (2021), pp. 855–873, https://doi.org/https://doi.org/10.1016/j.jsc.2020.10.005, https://www.sciencedirect.com/science/article/pii/S0747717120301152.

[14] Y. Chi, Y. M. Lu, and Y. Chen, *Nonconvex optimization meets low-rank matrix factorization: An overview*, IEEE Transactions on Signal Processing, 67 (2019), pp. 5239–5269, https://doi.org/10.1109/TSP.2019.2937282.

[15] J. Draisma, E. Horobeț, G. Ottaviani, B. Sturmfels, and R. R. Thomas, *The Euclidean distance degree of an algebraic variety*, Foundations of computational mathematics, 16 (2016), pp. 99–149.

[16] S. S. Du, X. Zhai, B. Poczos, and A. Singh, *Gradient descent provably optimizes over-parameterized neural networks*, in International Conference on Learning Representations, 2019, https://openreview.net/forum?id=S1eK3i09YQ.
[17] Y. DUKLER, Q. GU, AND G. MONTÚFAR, Optimization theory for ReLU neural networks trained with normalization layers, in Proceedings of the 37th International Conference on Machine Learning, H. D. III and A. Singh, eds., vol. 119 of Proceedings of Machine Learning Research, PMLR, 13–18 Jul 2020, pp. 2751–2760, http://proceedings.mlr.press/v119/dukler20a.html.

[18] E. GONZALEZ AND D. WEINBERG, Root configurations of real univariate cubics and quartics, arXiv: Commutative Algebra, (2015).

[19] D. GRAYSON AND M. STILLMAN, Macaulay 2, a system for computation in algebraic geometry and commutative algebra. faculty.math.illinois.edu/Macaulay2/.

[20] S. GUNASEKAR, J. D. LEE, D. SOUDRY, AND N. SREBRO, Implicit bias of gradient descent on linear convolutional networks, in Advances in Neural Information Processing Systems 31, S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, eds., Curran Associates, Inc., 2018, pp. 9461–9471, http://papers.nips.cc/paper/8156-implicit-bias-of-gradient-descent-on-linear-convolutional-networks.pdf.

[21] M. HARDT AND T. MA, 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, https://openreview.net/forum?id=ryxB0Rtxx.

[22] R. HARTSHORNE, Algebraic geometry, vol. 52, Springer Science & Business Media, 2013.

[23] A. JACOT, F. GABRIEL, AND C. HONGLER, Neural tangent kernel: Convergence and generalization in neural networks, in Advances in Neural Information Processing Systems 31, S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, eds., Curran Associates, Inc., 2018, https://proceedings.neurips.cc/paper/2018/file/5a4be1fa34e62bb8a6ec6b91d2462f5a-Paper.pdf.

[24] K. KAWAGUCHI, Deep learning without poor local minima, in Advances in Neural Information Processing Systems 29, D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, eds., Curran Associates, Inc., 2016, pp. 586–594, http://papers.nips.cc/paper/6112-deep-learning-without-poor-local-minima.pdf.

[25] S. KURMANN, Some remarks on equations defining coincident root loci, Journal of Algebra, 352 (2012), pp. 223–231.

[26] T. LAURENT AND J. VON BRECHT, Deep linear networks with arbitrary loss: All local minima are global, in Proceedings of the 35th International Conference on Machine Learning, J. Dy and A. Krause, eds., vol. 80 of Proceedings of Machine Learning Research, Stockholmsmässan, Stockholm Sweden, 10–15 Jul 2018, PMLR, pp. 2902–2907, http://proceedings.mlr.press/v80/laurent18a.html.

[27] H. LEE AND B. STURMFELS, Duality of multiple root loci, Journal of Algebra, 446 (2016), pp. 499–526.

[28] C. LENNERZ AND E. SCHOMER, Efficient distance computation for quadratic curves and surfaces, in Geometric Modeling and Processing. Theory and Applications. GMP 2002. Proceedings, 2002, pp. 60–69.

[29] H. LU AND K. KAWAGUCHI, Depth creates no bad local minima, CoRR, abs/1702.08580 (2017), http://arxiv.org/abs/1702.08580, https://arxiv.org/abs/1702.08580.
[30] D. Mehta, T. Chen, T. Tang, and J. Hauenstein, *The loss surface of deep linear networks viewed through the algebraic geometry lens*, IEEE Transactions on Pattern Analysis and Machine Intelligence, (2021), pp. 1–1, https://doi.org/10.1109/TPAMI.2021.3071289.

[31] A. M. Saxe, J. L. McClelland, and S. Ganguli, *Exact solutions to the nonlinear dynamics of learning in deep linear neural networks*, in 2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14–16, 2014, Conference Track Proceedings, Y. Bengio and Y. LeCun, eds., 2014, http://arxiv.org/abs/1312.6120.

[32] A. Seigal and G. Montúfar, *Mixtures and products in two graphical models*, Journal of algebraic statistics, 9 (2018), pp. 1–20, https://doi.org/10.18409/jas.v9i1.90.

[33] D. Soudry, E. Hoffer, M. S. Nacson, S. Gunasekar, and N. Srebro, *The implicit bias of gradient descent on separable data*, Journal of Machine Learning Research, 19 (2018), pp. 1–57, http://jmlr.org/papers/v19/18-188.html.

[34] S. Sullivan, *Algebraic Statistics*, Graduate Studies in Mathematics, American Mathematical Society, 2018, https://books.google.de/books?id=EXj1uQEACAAJ.

[35] M. Trager, K. Kohn, and J. Bruna, *Pure and spurious critical points: a geometric study of linear networks*, in International Conference on Learning Representations, 2020, https://openreview.net/forum?id=rkgOlCVYvB.

[36] F. Williams, M. Trager, D. Panozzo, C. Silva, D. Zorin, and J. Bruna, *Gradient dynamics of shallow univariate relu networks*, in Advances in Neural Information Processing Systems, H. Wallach, H. Larochelle, A. Beygelzimer, F. d’Alché-Buc, E. Fox, and R. Garnett, eds., vol. 32, Curran Associates, Inc., 2019, https://proceedings.neurips.cc/paper/2019/file/1f6419b1cbe79c71410cb320fc094775-Paper.pdf.

[37] L. Zhang, *Depth creates no more spurious local minima*, 2019, https://arxiv.org/pdf/1901.09827.pdf.

[38] Y. Zhou and Y. Liang, *Critical points of linear neural networks: Analytical forms and landscape properties*, in International Conference on Learning Representations, 2018, https://openreview.net/forum?id=SysEexbRb.
A Gradients

For convenience we give a description of the gradient for LCNs.

**Proposition A.1.** Consider a loss \( \ell : \mathbb{R}^{d_{x} \times d_{y}} \to \mathbb{R} \) on the space of matrices. For an LCN parametrization \( \mu : (w_{1}, \ldots, w_{L}) \mapsto \bar{W} \) as in eq. (5), denote the loss on the space of filters by \( \mathcal{L} = \ell \circ \mu \). The gradient is

\[
(\nabla_{w_{l}} \mathcal{L}(w_{1}, \ldots, w_{L}))_{\alpha} = \sum_{\beta, \gamma}(\nabla_{w_{l}} W_{l})_{\alpha, \beta, \gamma}(W_{l+1}^{T} \cdots W_{L}^{T})^{\beta} \nabla_{\bar{W}} \ell(\bar{W}) W_{1}^{T} \cdots W_{l-1}^{T})_{\beta, \gamma},
\]

where \((\nabla_{w_{l}} W_{l})_{\alpha, \beta, \gamma} = \partial_{w_{l, \alpha}}(W_{l})_{\beta, \gamma}\). For the square loss \( \ell(\bar{W}) = \|\bar{W}X - Y\|^{2} \) with data \( X, Y \),

\[
\nabla_{\bar{W}} \ell(\bar{W}) = \bar{W}XX^{T} - YY^{T}.
\]

The previous result describes the gradient of the loss \( \mathcal{L} \) in terms of convolutional matrices \( W_{l} \), however the expression does not depend on the actual matrices but only on the filters involved. We can make this fact explicit in the case of stride 1 as follows.

**Proposition A.2.** Let \( \mathcal{L} = \ell \circ \mu \) be a loss function for an LCN with stride 1 and let \( \theta = (w_{1}, \ldots, w_{L}) \) denote the filters. For any vector \( w \in \mathbb{R}^{k} \), let \( \pi_{-1}(w) := w_{0}x^{-k} + w_{1}x^{-k+2} + \cdots + w_{k-2}x^{-1} + w_{k-1} \) be the Laurent polynomial (polynomial with negative exponents) whose coefficients are the elements of \( w \). For any \( i = 1, \ldots, L \), the gradient \( \nabla_{w_{l}} \mathcal{L}(\theta) \in \mathbb{R}^{k_{i}} \) is given by the coefficients of degree \((0, \ldots, k_{i} - 1)\)

\[
\hat{\pi}(\nabla \ell) \cdot \pi_{-1}(w_{1}) \cdots \pi_{-1}(w_{i-1}) \cdot \pi_{-1}(w_{i+1}) \cdots \pi_{-1}(w_{L}) \in \mathbb{R}[x, x^{-1}],
\]

where \( \hat{\pi}(\nabla \ell) \) is the univariate polynomial representation of the gradient \( \nabla \ell(\mu(\theta)) \) obtained from (8) by dehomogenizing \( y = 1 \).

**Proof.** The Laurent polynomial \( \pi_{-1}(w) \) is a representation of the adjoint filter \( w^{*} := (w_{k-1}, \ldots, w_{0}) \). Convolution with the adjoint filter \( w^{*} \) is the adjoint operator of the convolution with \( w \), for the standard Euclidean inner product. In terms of polynomials, this means that

\[
\langle \hat{\pi}(v), \hat{\pi}(w_{2}) \rangle = \langle \hat{\pi}(v) \cdot \pi_{-1}(w_{2}), \hat{\pi}(w_{1}) \rangle,
\]

for any vectors \( v, w_{1}, w_{2} \), where \( \langle \cdot, \cdot \rangle \) is coefficient-wise inner-product and \( \cdot \) is product operation for polynomials (missing coefficients are treated as zero so we need not constrain the sizes of \( v, w_{1}, w_{2} \)).

The claim now follows from the fact that \( \nabla_{w_{l}} \mathcal{L} \) is such that

\[
\langle \hat{\pi}(\nabla_{w_{l}} \mathcal{L}), \hat{\pi}(\hat{w}_{l}) \rangle = \langle \hat{\pi}(\nabla \ell), \hat{\pi}(\hat{w}_{1}) \rangle \cdots \langle \hat{\pi}(\nabla \ell), \hat{\pi}(\hat{w}_{L}) \rangle = \langle \hat{\pi}(\nabla \ell) \cdot \pi_{-1}(w_{1}) \cdots \pi_{-1}(w_{i-1}) \cdot \pi_{-1}(w_{i+1}) \cdots \pi_{-1}(w_{L}), \hat{\pi}(\hat{w}_{l}) \rangle.
\]

\[ \square \]