ChatGPT

Neural Networks as Manifolds in Function Space

Networks as Submanifolds of $C(X)$ or $L^2(X)$

A neural network with fixed architecture defines a family of functions (the network's hypothesis class) that can be viewed as a **low-dimensional submanifold** of the space of all functions on the input domain [1]. In other words, each setting of the network's parameters θ yields a specific function $f_\theta: X \to \mathbb{R}^m$, and the collection of all such f_θ (for θ in parameter space Θ) forms a subset of $C(X)$ or $L^2(X)$. In favorable cases (e.g. ignoring parameter symmetries), this mapping $\theta \mapsto f_\theta$ is locally an **immersion** – the network's parameter manifold (of dimension equal to the number of free parameters) is embedded in the infinite-dimensional function space. This perspective is implicit in classic results: the **Universal Approximation Theorem** guarantees that by increasing width (or depth), neural networks can approximate any continuous function on a compact domain arbitrarily well [2]. This means that in the limit of large network size, the reachable function-family is dense in $C(X)$, but for a **finite network** the function manifold is limited (having at most as many degrees of freedom as parameters).

Parametrization and Redundancy: In practice, the parameter-to-function map is not one-to-one – e.g. permuting hidden neurons or scaling weights can leave f_θ invariant. Such symmetries mean the image of Θ in function space is an **immersed manifold** with self-overlaps, and the mapping can have singular Jacobian in those redundant directions. Indeed, the **Fisher Information** metric (which measures functional distinguishability of parameter perturbations) is often singular for overparameterized nets [3]. This is a rigorous observation: certain directions in weight space do not change the output, leading to a degenerate (rank-deficient) metric. Researchers handle this by modding out the symmetries or using pseudo-inverses/damping in the metric [3]. It remains an open challenge to fully characterize the global structure of these function-space manifolds—for instance, understanding all the functionally equivalent parameters or the manifold's global curvature is difficult. However, *locally* (away from singular points), one can study neural networks as smooth submanifolds of function space, and many works have formalized aspects of this view.

Riemannian Geometry via Jacobians and NTK

A powerful approach is to equip the network's function-manifold with a **Riemannian metric** induced from the ambient function space. Intuitively, one declares two parameter perturbations $d\theta_1, d\theta_2$ to be orthogonal if they produce orthogonal changes in the output function (under an L^2 inner product on outputs). Formally, one can define an inner product on the tangent space at parameters θ as $\langle d\theta_1, d\theta_2\rangle := \int_X \langle \partial f_\theta(x)/\partial\theta \cdot d\theta_1,\; \partial f_\theta(x)/\partial\theta \cdot d\theta_2\rangle dx$. In matrix terms, the **Jacobian** $J_\theta(x) = \nabla_\theta f_\theta(x)$ maps parameter-space directions to function perturbations, and the pullback of the L^2 metric gives a Gram matrix $G(\theta)=\mathbb{E}*x[*/\theta(x)^\top J_\theta(x)]$. This is precisely the **Fisher Information Matrix** in a probabilistic setting, or an L^2-based metric in a regression setting. The entries $G_{ij}=\mathbb{E}*x[\partial*f]$ can be seen as a kind of }f \;\partial_{\theta_j**Neural Tangent Kernel** (NTK) evaluated on identical inputs [4]. Indeed, Jacot et al. (2018) defined the NTK for a network as $\Theta(\theta; x,x')=\sum_i \partial_{\theta_i}f_\theta(x)\,\partial_{\theta_i}f_\theta(x')$ [4], which for $x=x'$ recovers

the above inner product. When integrated or averaged over the input distribution, this yields the Riemannian metric on the function manifold.

Invariance and Natural Gradient: A key motivation for introducing a Riemannian metric on parameter space is to make analyses and optimization *coordinate-invariant*. *Shun-Ichi Amari's* work on **natural gradient** (1998) was seminal: he showed that using the Fisher metric in gradient descent yields parameter updates that are invariant to smooth reparameterizations of the model. In other words, the Fisher-informed update is "natural" in that it respects the geometry of the underlying function space, rather than the arbitrary coordinates of Θ. Subsequent research formalized this. *Ollivier (2015)*, for example, explicitly studied **Riemannian metrics for feedforward networks** and derived training algorithms using either the Fisher metric or scaled Hessians [5]. These methods are mathematically principled and invariant to many transformations of weights or data, confirming that a proper geometric treatment can improve optimization **and** be independent of parameterization choices [5]. More recently, *Kristiadi et al. (2023)* argue that many apparent inconsistencies in flatness or mode-connectivity analyses arise from ignoring the metric; if one "explicitly represents the metric" and applies the correct tensor transformation rules, many quantities become invariant [6]. They emphasize that a metric is "always present" conceptually, but standard practice often assumes an identity metric by default, losing invariance under reparameterization [6]. By keeping track of the proper Riemannian metric, one can compare quantities like Hessian eigenvalues or flatness across different parameterizations on an equal footing [7] [8].

Several works have *proposed metrics and studied their effects*. *Benjamin et al. (2019)* replaced the usual parameter-space Euclidean distance with an **L^2 function-space distance** in measuring training progress [9]. They showed empirically that parameter distance is a poor proxy for actual function change – for instance, early in training a small weight change can drastically alter the output function, whereas later in training parameters may move a lot with little functional effect. They found the ratio of function-distance to weight-distance shrinks as training proceeds [10], and proposed regularizers to directly *limit how far the network strays in function space* between tasks or updates [11]. These ideas are heuristic (proposed as practical methods), but they highlight the importance of viewing training as a path in function space rather than weight space. On the theoretical side, *Bai, Rosenberg, and Xu (2025)* develop a **Generalized Tangent Kernel (GTK)** framework which unifies natural and standard gradients. They prove that the NTK (generalized appropriately) actually induces a Riemannian metric on the entire function space, one that makes the usual gradient *equivalent* to the natural gradient under that metric [12]. In other words, for a fixed network parameterization, there is an intrinsic metric on the manifold of functions such that using plain gradient descent in parameters is geometrically "natural" [12] [13]. This framework resolves issues when the parameterization is **non-immersive** (degenerate): they offer solutions for cases where the Jacobian is rank-deficient ("non-immersion"), which had caused difficulties for defining natural gradients [13]. The existence of this metric and the orthonormal basis of function space (related to NTK's eigenfunctions) are established rigorously in their work.

Summary of Rigor: Defining a pullback Riemannian metric via the network's Jacobian is a well-founded mathematical construction. Many authors have proven properties of such metrics or used them in proven algorithms (e.g. convergence proofs for natural gradient methods rely on the Fisher metric's positive-definiteness on the immersed manifold). However, one must be careful with degeneracies: it remains partly **open** to develop a complete "regular" Riemannian manifold structure when the network has symmetry-induced singular directions. Researchers typically circumvent this by considering quotient spaces or adding small damping to the metric [3]. In practice and theory, the metric viewpoint has been extremely useful: it underlies *invariant flatness measures*, *trust-region methods*, and our theoretical understanding of how gradient descent traverses the function manifold. Invariance to reparameterization is formally guaranteed when using the proper geometric objects [6], whereas analyses that ignore the metric can lead to contradictory or coordinate-dependent conclusions. Overall,

the **metric geometry of neural networks** is an area where formal differential-geometric theorems (e.g. on invariance, convergence of natural gradient, existence of metrics like GTK) coexist with ongoing research (e.g. handling global manifold structure and singularities).

Infinite-Width Limit: NTK as a Flat Manifold Approximation

One of the most significant theoretical developments is the characterization of neural networks in the **infinite-width limit**, where the network's function manifold becomes "flat" in a certain sense. Pioneering work by *Neal (1996)* and *Williams (1997)* showed that a single-hidden-layer neural network, when the number of hidden units tends to infinity with random initialization, corresponds to a **Gaussian Process** (GP) in function space [14]. In that limit, the random network's output functions have a well-defined distribution and covariance kernel (often called the **Neural Network Gaussian Process kernel**). This was later extended: e.g. *Lee, Matthews et al. (2018)* proved deep fully-connected networks also converge to GPs as width $\to \infty$. These are rigorous results (under mild conditions like i.i.d. initial weights and appropriate variance scaling) that *identify the network's function-family in the infinite limit with a reproducing kernel Hilbert space (RKHS)* given by the GP's covariance kernel. Importantly, in this limit the network's parameter-function map **linearizes**: the infinite network behaves like a linear model in function space, with a fixed basis of "random features."

The **Neural Tangent Kernel (NTK)** result by *Jacot et al. (2018)* makes this concrete for training dynamics. They showed that if one considers gradient descent on an infinitely wide network (with suitable scaling of learning rate and initialization), the NTK converges to a constant kernel (independent of training time) and the network's predictions follow a *linear evolution* in function space [15]. In fact, in the infinite-width limit each network realization has an NTK that is effectively frozen at its initial value, and training the network to minimize mean-square loss will produce the **same solution as kernel ridge regression in that NTK** [16]. This is a formal theorem: for example, under full-batch gradient flow, $ \lim_{width\to\infty} f_\theta(t)$ solves a linear ODE $ \dot f = -\Theta_\infty \,(f - y)$ whose solution is the kernel interpolant of the data [15] [16]. Geometrically, one can view the network's function manifold as becoming so high-dimensional (and the parameter changes so small) that the trajectory never experiences the manifold's curvature – it stays in the *tangent space* around the initial function. The infinite network thus acts like an *affine subspace* of functions: the learned function is just the initial function plus a linear combination of basis functions determined by the NTK. In other words, the rich, nonlinear manifold of reachable functions has been flattened into a single fixed function space (the RKHS of the NTK). Consistent with this, the infinite-width NTK is **data-independent** [16] – all the feature "geometry" is fixed at initialization and does not evolve with the data.

NTK vs Feature Learning: A crucial distinction, emphasized in many works, is that **finite-width networks can move off this linearized manifold**, i.e. they can *learn new features*, whereas the infinite NTK-limit networks cannot. This remains partly an empirical and heuristic observation, but it's strongly supported by both theory and experiments. For example, *Samarin et al. (2022)* compared finite convnets to their "linearized" counterparts (networks frozen to first-order expansion). They found that **finite networks greatly outperform their NTK-linearized versions on complex tasks** – and the gap widens on more difficult datasets [17]. At normal widths, the linearized network (which is equivalent to a random features or kernel method) is markedly worse, indicating that the finite network has changed its representation during training in a way a kernel method cannot [17] [18]. Only when the networks become extremely wide do the linearized and nonlinear models begin to agree, consistent with NTK theory (in one case, a wide linearized network matched the performance of an ultra-wide real network, implying it had entered the lazy regime) [19] [20]. These results are in line with the intuition that *feature learning* – the network adjusting its hidden-layer representations – is a key advantage of neural networks, and this is precisely what the NTK regime "freezes out."

From a theoretical perspective, the NTK analysis is **rigorous** (under certain initialization and width assumptions), and it has even been used to prove global convergence of training for overparameterized nets in that regime. However, the NTK regime is essentially a **linearization** (sometimes dubbed "lazy training" [21]): the weights move only infinitesimally, so the network function changes in the span of the initial tangent vectors. *Chizat et al. (2019)* formalized that this behavior is not inevitable – it results from the standard scaling of initialization and learning rate. They showed that if one scales networks differently (e.g. the so-called "mean-field" or feature-learning scaling), then even as width grows, one can escape the lazy regime [22] [23]. In other words, there are other infinite-width limits where the kernel itself *evolves* (or where features do change over training). This is an area of active research: for two-layer networks, a line of work (e.g. Mei, Montanari 2019; Rotskoff, Vanden-Eijnden 2018) proved that with appropriate scalings, one gets a **mean-field limit** described by a Wasserstein gradient flow in the space of neuron distributions – a scenario where the network *does* learn feature distributions even at infinite width. More recently, *Yang & Hu et al. (2021)* introduced the **Maximal Update Parametrization (μP)** to allow feature learning in deep networks' infinite-limit. They point out that the classical NNGP and NTK limits, while illuminating, "fail to capture what makes NNs powerful – the ability to learn features" [24]. In the NTK limit, the learning rate is taken small enough that weights stay near initialization, *preventing new feature discovery* [25]. Yang and colleagues formally constructed an alternative infinite limit where all layers can evolve ("infinite-width limit that sufficiently captures learning" [26]), and indeed demonstrate theoretical and empirical advantages when using this parametrization [27]. This cutting-edge research is still being refined, but it underscores that the **standard infinite-width (NTK) manifold is essentially a flattened, linearized version** of the true neural network function manifold. The NTK regime is rigorously understood, while the feature-learning regime is only partially understood through special cases and new formalisms.

Metric Geometry in Infinite vs Finite Regimes: Another way to distinguish the infinite-width (kernel) regime is via the **curvature of the function manifold**. At initialization, a sufficiently wide network with random weights induces a metric on input space that is highly symmetric (e.g. in a two-layer ReLU network, the volume element and curvature of the input-space metric are isotropic in the infinite limit) [28]. *Zavatone-Veth et al. (2025)* show that after training on a task, finite networks develop *anisotropic* Riemannian metrics on input space – effectively *warping* the input manifold to expand regions near decision boundaries [28]. In contrast, in the infinite (kernel) limit, the metric would remain fixed and relatively uniform. This is a more geometric way to say that **training molds the network's feature manifold** in finite networks, whereas in the NTK limit the feature manifold is fixed at initialization. The authors connect this to earlier theoretical proposals (Amari & Wu 1999) that one could improve generalization by **expanding the geometry near class boundaries**, which a feature-learning network naturally does [29]. While these insights are partly empirical (measuring curvature, volume elements, etc. for networks), they highlight an important open area: we lack a complete theory for how the **curvature** of the network function manifold (or the induced metric on inputs) relates to generalization. It's understood that NTK essentially assumes zero curvature (a flat tangent space) throughout training, and this is provably suboptimal for complex tasks [17]. But quantifying the *benefit* of traversing the curved manifold (feature learning) is still largely heuristic. Recent work provides perturbative expansions (beyond first-order NTK) to approximate finite-width effects, but a full characterization (especially for deep, multi-layer nets at practical widths) remains **open**.

Proven Results vs. Open Questions

In summary, researchers have made substantial progress in **formalizing neural networks as objects in function space**. It is rigorously established that a neural network's hypothesis class is (under mild conditions) a smooth manifold embedded in L^2 or $C(X)$ – at least locally and modulo symmetries. There are **formal theorems** for: (a) *universal approximation* (density of these manifolds in function space) [2], (b) existence of a well-defined *Riemannian metric* via the network Jacobian (Fisher

Information or NTK) and invariance properties of gradient-based methods using that metric [6] [12], and (c) the *infinite-width limit* where the manifold essentially linearizes (Neural Tangent Kernel behavior) [16]. Particularly, the NTK theory is backed by rigorous proofs and has even been used to explain convergence and generalization in extreme overparameterization. On the other hand, several aspects are **hypothesized or empirical**. The idea that finite networks find a better optimum by moving along the curved manifold (learning new features) is supported by many experiments [17] and heuristic arguments [27], but a complete theory for finite-width feature learning is still emerging. Likewise, claims about how the **geometry of the function manifold relates to generalization** (e.g. "flat minima" in weight space correspond to broad functions in function space) are often made, but they require careful, coordinate-invariant treatment – an active research topic [6] [8]. Some recent works provide **partial theoretical frameworks** (e.g. mean-field limits, higher-order corrections to NTK, or geometry-of-learning empirical studies), but many questions remain open. For example, quantifying the exact gain from feature learning, or describing the global topology of the function submanifold (how complex and curved it can be for a given architecture), are largely unresolved.

In conclusion, viewing neural networks as submanifolds in function space with an induced metric has become a fruitful paradigm. **Fully-connected networks**, especially one- or two-layer ones, have been the testbed for rigorous results – from *exact GP/kernel equivalences* [14] to *convergence theorems in the NTK regime* [15]. Those results are solid. Building on them, we now have a growing geometric understanding (e.g. Riemannian measures of complexity, invariances, curvature calculations), though many of those developments mix **proven theory** (e.g. existence of the Fisher/NTK metric) with **heuristic or empirical insight** (e.g. interpreting changes in that metric during training). As research continues, the gap between the linearized theory and the observed power of nonlinear feature learning is gradually narrowing – with new theories aiming to capture the **true curved manifold** of neural networks in function space rather than just its tangent approximation [24] [30]. Each work outlined above contributes a piece: some provide formal theorems and rigorous frameworks, while others offer conjectures or experimental evidence about the geometry of neural networks, highlighting intriguing directions for future theoretical exploration.

Sources: The concepts and results discussed are drawn from a range of foundational and recent works, including formal theories of natural gradient and network information geometry [5] [6], empirical studies of network function evolution [9] [17], and theoretical analyses of infinite-width limits and kernel equivalences [15] [16], among others [12] [25]. Each cited work deepens our understanding of neural networks as geometric objects in function space, either through rigorous proof or insightful experimentation.

---

[1] [2] Universal approximation theorem - Wikipedia
https://en.wikipedia.org/wiki/Universal_approximation_theorem

[3] [6] [7] [8] proceedings.neurips.cc
https://proceedings.neurips.cc/paper_files/paper/2023/file/395371f778ebd4854b88521100af30ad-Paper-Conference.pdf

[4] [14] [15] [16] [17] [18] [19] [20] proceedings.mlr.press
https://proceedings.mlr.press/v180/samarin22a/samarin22a.pdf

[5] [1303.0818] Riemannian metrics for neural networks I: feedforward networks
https://arxiv.org/abs/1303.0818

[9] [10] [11] [1805.08289] Measuring and regularizing networks in function space
https://arxiv.org/abs/1805.08289

12 13 openreview.net
https://openreview.net/pdf?id=HOnL5hjaIt

21 [PDF] Theories of Neural Networks Training - Challenges and Recent ...
https://lchizat.github.io/files/presentations/chizat2019NNtraining.pdf

22 [PDF] On Lazy Training in Differentiable Programming
http://papers.neurips.cc/paper/8559-on-lazy-training-in-differentiable-programming.pdf

23 On Lazy Training in Differentiable Programming | alphaXiv
https://www.alphaxiv.org/overview/1812.07956v5

24 25 26 27 30 On infinitely wide neural networks that exhibit feature learning - Microsoft Research
https://www.microsoft.com/en-us/research/blog/on-infinitely-wide-neural-networks-that-exhibit-feature-learning/

28 29 How does training shape the Riemannian geometry of neural network representations?
https://arxiv.org/html/2301.11375v4