

UC Santa Barbara

UC Santa Barbara Electronic Theses and Dissertations

Title

A Disagreement Metric for Loss-Landscape Visualization

Permalink

<https://escholarship.org/uc/item/4rz33506>

ISBN

9798186382300

Author

Zhang, Justin

Publication Date

2026-06-16

Peer reviewed|Thesis/dissertation

University of California
Santa Barbara

A Disagreement Metric for Loss-Landscape Visualization

A Thesis submitted in partial satisfaction
of the requirements for the degree

Master of Science
in
Electrical and Computer Engineering

by

Justin Zhang

Committee in charge:

Professor Nina Miolane, Chair
Professor Yasamin Mostofi
Professor Peng Li

June 2026

The Thesis of Justin Zhang is approved.

Yasamin Mostofi

Peng Li

Nina Miolane, Committee Chair

June 2026

A Disagreement Metric for Loss-Landscape Visualization

Copyright © 2026

by

Justin Zhang

Acknowledgements

This thesis is the product of a great deal of help, and I am grateful to everyone who made it possible.

My deepest thanks go to my advisor, Fatih Dinç, who guided this project from its earliest questions to its final form, and whose insight shaped both the work and the way I think about research.

I am especially grateful to Professor Nina Miolane, the chair of my committee, in whose class this project first took shape.

I thank Jasper Ou, my fellow graduate student and collaborator throughout this project.

I am also grateful to my committee members, Professors Yasamin Mostofi and Peng Li, for their time and feedback.

Abstract

A Disagreement Metric for Loss-Landscape Visualization

by

Justin Zhang

Advancements in deep learning optimization have sparked significant interest in the geometry of high-dimensional loss landscapes. To make these landscapes interpretable, researchers heavily rely on low-dimensional projection techniques, such as random directions, filter normalization, and trajectory-based PCA. However, these visualizations are typically validated through heuristic visual inspection. A key open question remains: can we define a principled, task-agnostic criterion to quantify how well a chosen projection captures the underlying loss landscape?

This thesis introduces the disagreement metric, a theoretically grounded criterion that evaluates projection fidelity by measuring the unexplained fraction of projected gradient energy. Geometrically, this metric quantifies the divergence between the true landscape gradients and the gradients evaluated over the projected subspace. We demonstrate theoretically that in the local quadratic region near a minimum, minimizing the normalized disagreement metric exactly recovers the principal eigenvectors of the Hessian matrix. We further characterize the mathematical behavior of this metric beyond the near-minimum assumption for general smooth loss functions.

Empirically, we employ Recurrent Neural Networks (RNNs) trained on a flip-flop memory task to evaluate the effectiveness of various visualization techniques. By applying PCA directions, filter-normalized random directions, and start-to-end trajectory directions, we score and rank these methods using the disagreement metric, separating faithful planes from ones that capture essentially none of the loss.

Ultimately, this work provides a rigorous, fit-free benchmark for evaluating loss landscape projections, bridging the gap between heuristic visualizations and exact geometric methods in neural network optimization.

Contents

Abstract	v
1 Introduction	1
1.1 The problem: visualizations validated by eye	2
1.2 This thesis: the disagreement metric	3
1.3 Contributions	4
1.4 Outline of the thesis	5
2 Background	7
2.1 The loss landscape and why it matters	7
2.2 Visualizing the landscape by projection	8
2.3 Dimensional reduction of parameter space	10
2.4 Validation by eye, and the gap	11
3 Method	12
3.1 The disagreement metric	12
3.2 The captured fraction of gradient energy	14
3.3 A significance score for any projection	15
4 Theory of the Loss Landscape Near a Minimum	17
4.1 Near a minimum, the loss is quadratic	17
4.2 The two-dimensional Gaussian model	18
5 Theory of the Loss Landscape in General	22
5.1 The criterion is global, not local	23
5.2 Which structure wins: the dominant basin	24
5.3 What the criterion selects in general	25
6 Testing Visualization Methods on Task-Trained RNNs	27
6.1 Setup	27
6.2 Scores	28
6.3 What the scores reveal	30

7	Discussion and Future Work	31
7.1	Summary	31
7.2	Future directions: low-dimensional training	32
7.3	Open questions	33
	References	34

Chapter 1

Introduction

This thesis began with a narrower question about recurrent neural networks: how a network, viewed as a dynamical system, reorganizes itself as its parameters change. A trained RNN realizes a particular arrangement of fixed points and the flow between them, but that arrangement is not static: as the weights move, the dynamics can change qualitatively through bifurcations. We approached this through a sequence of increasingly complex models: first a scalar RNN, whose dynamics reduce to the normal form of the pitchfork bifurcation, $\dot{x} = -x^3 + rx + mu$; then rank-one RNNs; and finally full-rank RNNs. To see how the loss behaved across the weights of these networks, we visualized it the way the field does: by taking a two-dimensional slice through parameter space and plotting the loss over it.

That practice raised the question that became this thesis. A slice is a choice, and nothing about the resulting picture tells you whether it was a good one: whether the plane you happened to pick captures the landscape or hides it. The problem is not special to recurrent networks. Training any modern neural network is the minimization of a scalar loss over a parameter space whose dimension d routinely reaches millions or billions, and the geometry of that loss surface (the shape of its minima, the barriers between

them, the breadth of its flat regions) is widely believed to govern how easily a network trains, how well it generalizes, and how robust it is to perturbation [1]. Yet a function over $\mathbb{R}^d$ cannot be seen. To reason about loss geometry, researchers restrict the loss to a two-dimensional plane through parameter space and plot the resulting surface; such loss-landscape plots have become a standard illustration in the deep-learning literature. Which plane to choose, and how to know whether it was the right one, is the question this thesis answers.

Throughout, we write $\alpha : \mathbb{R}^d \rightarrow \mathbb{R}$ for the scalar function whose landscape we wish to visualize (typically the training loss) and we visualize it on the k -dimensional subspace ($k = 2$ for a plane) spanned by the orthonormal columns of a matrix $V \in \mathbb{R}^{d \times k}$.

1.1 The problem: visualizations validated by eye

A two-dimensional plane is a slice of a high-dimensional space, and the picture it produces depends entirely on which plane is chosen. The same network, viewed through two different planes, can appear as a sharp ravine or a gentle bowl. The choice of V is therefore not a cosmetic detail; it determines what the visualization claims about the landscape.

We can extend this idea beyond just two dimensions. The same construction projects the loss onto a subspace of any dimension k , and the choice of subspace matters for more than visualization. If a low-dimensional subspace captures the directions along which the loss actually changes, it becomes a candidate for low-dimensional training. One could restrict optimization to it (training the network in $k \ll d$ coordinates) and still hope to reach a good, even optimal, solution. The optimization trajectory is itself known to be effectively low-dimensional [2], which hints that such a subspace exists. Whether the goal is an honest picture or low-dimensional training, then, everything turns on choosing

the subspace well.

Existing work supplies heuristics for choosing V . The simplest is to take random directions, optionally normalized to undo the scale invariances of the weights [1]. Others choose data-driven directions: the leading principal components of the optimization trajectory, or the single direction from the start to the end of training [1, 2]. Each of these is a recipe for producing a plane, and several come with partial motivations. None, however, answers the question that matters: how faithfully does the chosen plane capture the landscape?

In practice that question is settled by inspection. A plot that looks smooth, that shows a clean basin around the solution, is taken to be a good picture of the landscape, a process that amounts to judging a projection by its appearance rather than by what it preserves. This is unsatisfying for two reasons. First, a flattering picture can be an artifact of the projection: a plane can hide the sharp directions of a landscape and show only its flat ones. Second, without a quantitative score there is no way to compare two visualization methods, to report how much better one plane is than another, or to say whether any projection captures the landscape at all. What the field lacks is a principled, task-agnostic criterion, a goodness-of-fit measure for a chosen projection.

1.2 This thesis: the disagreement metric

This thesis proposes such a criterion. The guiding idea is that a faithful projection is one on which the projected gradient tells the truth: moving within the plane should change the loss in the way the full landscape’s gradient says it should. Where the gradient of the loss restricted to the plane and the projection of the true gradient disagree, the picture is distorting the landscape.

Chapter 3 makes this precise. A *disagreement* $D(V)$ measures the average squared gap

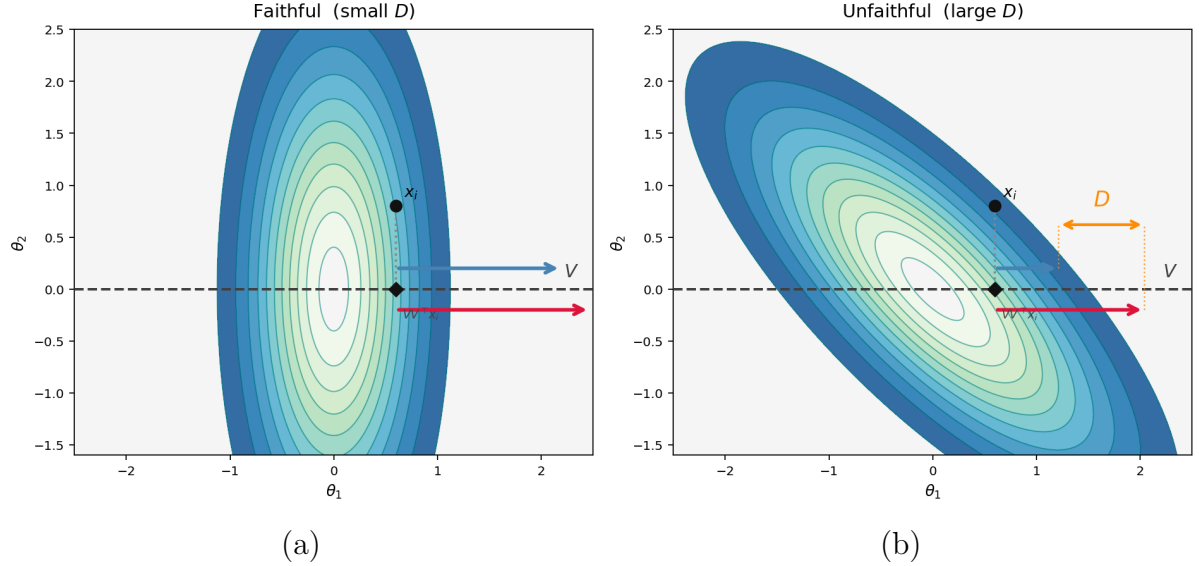


Figure 1.1: (a) A faithful plane, where the subspace V (dashed line) is aligned with the long axis of the loss basin; the projected true gradient (blue) and the gradient of the loss restricted to V (red) nearly agree. (b) An unfaithful plane, where the basin is tilted 45° relative to V : the two arrows diverge, and their gap is marked by D .

between these two gradients over a sample of points in parameter space (Figure 1.1). The criterion is to choose the projection V that makes the disagreement small: $D(V) = 0$ exactly when the plane captures the landscape’s gradient everywhere on the sampled region, and larger values mean greater distortion.

Although we introduce the disagreement metric to evaluate visualizations, the projection that minimizes it does more: it points to the directions that matter most for learning (Chapter 4) and to a subspace small enough to train in, which we take up as future work (Chapter 7).

1.3 Contributions

The contributions of this thesis are as follows.

- A principled, task-agnostic disagreement metric for measuring how faithfully a low-

dimensional projection captures a loss landscape, together with a significance test that turns it into a comparable score for any visualization method (Chapter 3).

- A demonstration that near a minimum, where the loss is quadratic, minimizing the metric recovers the eigenvectors of the Hessian, the principal curvature directions, derived in closed form on a two-dimensional Gaussian model (Chapter 4).
- A characterization of the metric away from a minimum: it is a global criterion that follows the gradient field over the sampled region rather than the local Hessian, and among competing basins selects the one that dominates the gradient (Chapter 5).
- An empirical evaluation that uses the metric to score the standard visualization methods on task-trained RNNs, ranking them and exposing a plane that captures essentially none of the loss’s gradient (Chapter 6).

1.4 Outline of the thesis

Chapter 2 reviews existing loss-landscape visualization techniques and the dimensional reductions they rely on, and assesses how they are validated, identifying the absence of a quantitative goodness-of-fit criterion as the gap this thesis addresses. Chapter 3 defines the disagreement metric and the normalized criterion, and introduces a significance test against random projections. Chapter 4 works out the near-minimum case, where every smooth loss is quadratic, and shows on a two-dimensional Gaussian model that minimizing the metric recovers the Hessian eigenvectors. Chapter 5 drops the near-minimum assumption and finds that the criterion is global: it follows the gradient field over the region sampled rather than the local Hessian, and among competing basins selects the one that dominates the gradient. Chapter 6 applies the metric to recurrent networks trained on the flip-flop memory task, ranking the standard visualization meth-

ods and flagging the ones that capture little of the loss. Chapter 7 summarizes the metric as a benchmark for loss-landscape visualization and lays out future directions, chiefly low-dimensional training within the subspace the metric selects, along with the open problems of minimizing D/total efficiently in high dimensions and characterizing it in closed form away from a minimum.

Chapter 2

Background

The training of a neural network is the minimization of a scalar loss $\alpha : \mathbb{R}^d \rightarrow \mathbb{R}$ over a parameter space of very high dimension d . The geometry of this loss surface is a natural object of study, and visualizing it has become a standard way to reason about optimization and generalization. This chapter reviews how the field visualizes loss landscapes, the broader sense in which parameter space reduces to a few important directions, and how these visualizations are validated. The last of these exposes the gap the thesis addresses.

2.1 The loss landscape and why it matters

The shape of the loss surface around a solution is widely believed to bear on how a network trains and how well it generalizes [1]. A recurring theme is the contrast between *sharp* and *flat* minima: flat minima, in which the loss changes slowly in every direction, have been associated with better generalization, while the sharp minima found by large-batch training have been linked to a generalization gap [3]. This picture comes with a caveat. Dinh et al. [4] show that sharpness is sensitive to reparametrization: because the loss is invariant to rescalings of the weights, a minimum can be made to look

arbitrarily sharp or flat without changing the function the network computes. In recurrent networks the same curvature underlies familiar training difficulties such as exploding and vanishing gradients [5]. The geometry of the landscape carries real information, but any measurement of that geometry must respect the symmetries and scales of the parameters, or it risks reporting an artifact rather than a property of the loss.

Because d is enormous, none of this geometry can be observed directly. The practical response has been to reduce the landscape to a picture.

2.2 Visualizing the landscape by projection

Loss-landscape visualization works by restricting the loss to a low-dimensional affine subspace through a chosen point θ_0 and plotting the result. The earliest such plots are one-dimensional: Goodfellow et al. [6] interpolate linearly between two parameter settings and plot the loss along the connecting line. The two-dimensional version, which produces the now-familiar surface and contour plots, requires choosing two directions, and the methods in the literature differ chiefly in how they make that choice.

A natural way to compare these methods is by how much of the optimization trajectory’s variance their plane retains, a quantity that can be computed in closed form once the two directions are fixed. The three methods below run from least to most trajectory-aware.

2.2.1 Filter normalization

The simplest choice is a pair of random Gaussian directions. Because the loss is invariant to rescalings of the weights, random directions of a fixed norm can over- or under-state the curvature of the plot; Li et al. [1] correct for this with *filter normalization*, rescaling each random direction filter by filter so that the perturbation applied to a filter

matches that filter’s own norm. This makes plots comparable across networks and is the de facto standard for generic loss-landscape pictures. Its weakness is that random directions in a space of dimension d almost never align with the few directions along which training actually moved, so the plane retains very little of the trajectory’s variance and the projected path collapses to a small, uninformative blob (Figure 2.1, left).

2.2.2 Start-to-end

A trajectory-aware alternative fixes one axis to the normalized vector from the initial parameters θ_0 to the final parameters θ_* , the net displacement of training, and takes the second axis to be the orthogonal direction of greatest trajectory variance. The resulting plane contains both the overall progress of optimization and its largest lateral excursion, so the projected path runs from its starting corner to the solution rather than collapsing to a point (Figure 2.1, middle).

2.2.3 Trajectory PCA

The most trajectory-aware choice uses the whole path rather than its endpoints. Collecting the parameter iterates along training into a matrix, centering them at the final parameters, and taking the top two principal components yields, by construction, the plane of maximum trajectory variance [2] (Figure 2.1, right). Ding et al. [2] show that grounding the plane in the trajectory this way avoids a common failure of single-point methods, in which the center of the plot is rendered as a minimum whether or not training has converged.

The three planes therefore capture progressively more of the trajectory, from almost none for filter normalization to the maximum for trajectory PCA. Capturing the trajectory, however, is not the same as faithfully capturing the landscape, a distinction we

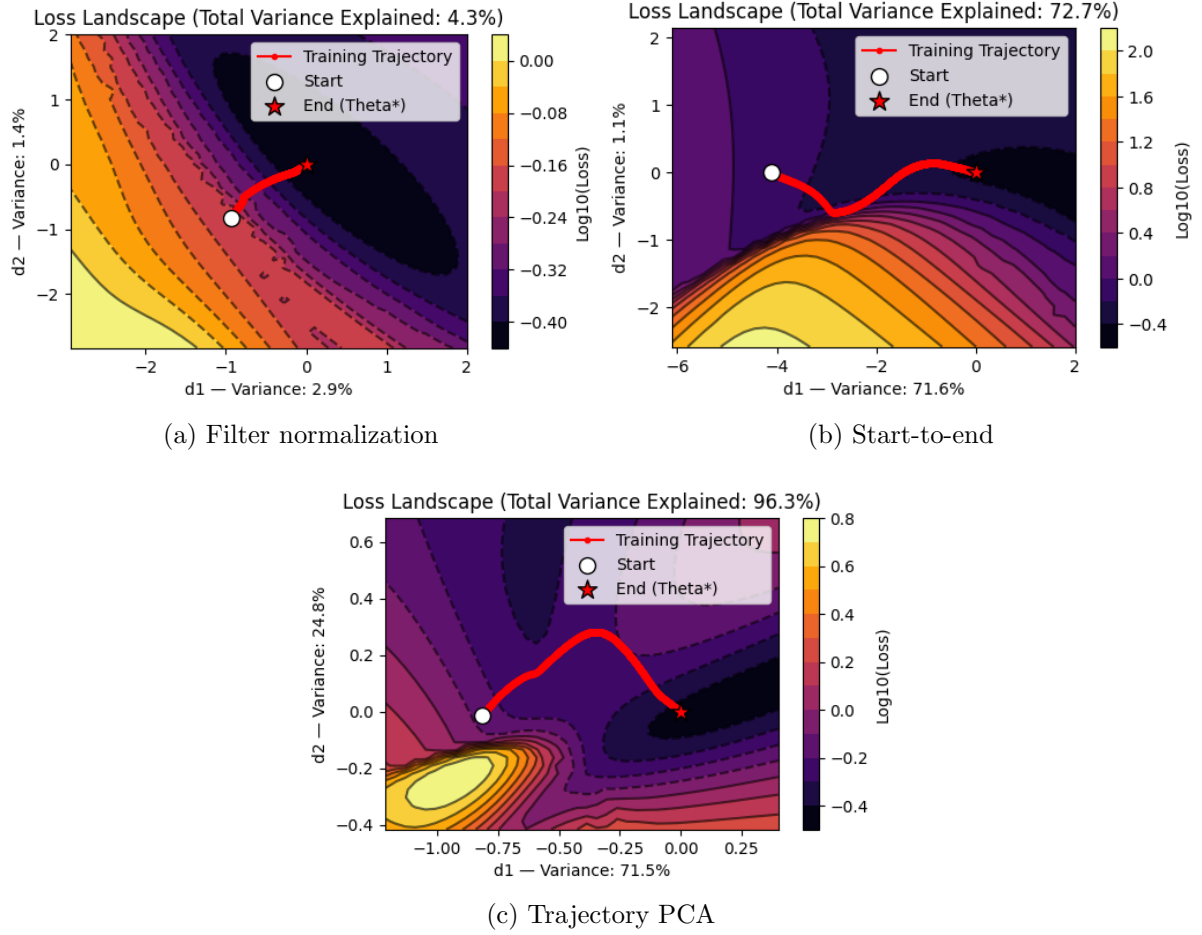


Figure 2.1: The loss landscape of a single rank-one RNN under the three projection methods of this section, with the training trajectory projected onto each plane. Panels (a), (b), and (c) use filter normalization, the start-to-end plane, and trajectory PCA, respectively.

return to below.

2.3 Dimensional reduction of parameter space

Choosing a plane for a picture is a special case of a broader idea: that the behavior of a high-dimensional model is governed by a small number of directions in parameter space. Several lines of work make this precise. Li et al. [7] measure the *intrinsic dimension* of

an objective by training within a random affine subspace and recording the smallest subspace dimension at which a good solution is still reachable, which is often far smaller than d . Gur-Ari et al. [8] observe that, after a short initial phase, gradient descent moves almost entirely within the small subspace spanned by the top eigenvectors of the Hessian.

These observations connect loss-landscape visualization to a body of theory on gradient-based dimension reduction. The closest formal relative is the *active subspace* of Constantine [9], the leading eigenvectors of the average outer product of the gradient with itself, $C = \mathbb{E}[\nabla\alpha \nabla\alpha^\top]$. These are exactly the directions along which the function changes most on average. Near a minimum the gradient covariance is $C \propto H^2$ for the Hessian H , so the active subspace coincides with the Hessian eigenbasis; this is precisely the subspace the disagreement metric recovers there, as Chapter 4 shows.

2.4 Validation by eye, and the gap

What unites the visualization methods above is the way they are justified. Most defend a plane either by construction, arguing that it captures some quantity such as the variance of the trajectory, or by appearance, presenting a smooth basin as evidence that the plane is a good one. Neither is a measure of how faithfully the plane represents the landscape. Variance captured, for instance, describes how the trajectory is spread within the plane, not whether the loss off the plane behaves as the plane suggests; a plane can account for most of the trajectory and still hide the sharpest directions of the loss.

The result is that loss-landscape visualizations are, in practice, validated by eye. There is no standard, task-agnostic criterion that takes a proposed projection and returns how much of the landscape it captures, no way to compare two methods on the same footing, and no way to certify that any plane is faithful at all. Supplying such a criterion is the purpose of this thesis, and the next chapter defines it.

Chapter 3

Method

This chapter defines the disagreement metric previewed in Chapter 1, develops the intuition for why captured gradient energy is the right quantity to measure, turns it into a significance score that applies to any visualization method. Throughout, $\alpha : \mathbb{R}^d \rightarrow \mathbb{R}$ is the scalar function whose landscape we wish to visualize, where d is the dimension of the parameter space, that is, the number of model parameters; we evaluate α at points $x \in \mathbb{R}^d$. We view the landscape on a k -dimensional subspace, with $k \ll d$ and $k = 2$ for a plane, spanned by the orthonormal columns of a matrix $V \in \mathbb{R}^{d \times k}$.

3.1 The disagreement metric

A faithful projection is one on which the projected gradient tells the truth: moving within the plane should change the loss in the way the full landscape’s gradient says it should. There are two natural notions of “the gradient seen in the plane,” and a good projection is one on which they agree.

The first comes from restricting the loss to the subspace,

$$\tilde{\alpha}(y) = \alpha(Vy), \quad y \in \mathbb{R}^k; \quad (3.1)$$

its gradient $\nabla_y \tilde{\alpha}(V^\top x)$ is the first notion of the gradient seen in the plane. The second is simply the projection of the true gradient, $V^\top \nabla \alpha(x)$. The two coincide exactly when α varies only along the subspace; in general they differ, and their squared discrepancy, averaged over a sample $x_1, \dots, x_N$ of points in parameter space, defines the *disagreement*

$$D(V) = \frac{1}{N} \sum_{i=1}^N \left\| \nabla_y \tilde{\alpha}(V^\top x_i) - V^\top \nabla \alpha(x_i) \right\|^2. \quad (3.2)$$

The first notion can be made explicit. By the chain rule,

$$\nabla_y \tilde{\alpha}(V^\top x) = V^\top \nabla \alpha(VV^\top x), \quad (3.3)$$

the in-plane gradient evaluated at the projection $VV^\top x$ of the point x . Substituting this into the definition, the disagreement is

$$D(V) = \frac{1}{N} \sum_{i=1}^N \left\| V^\top \nabla \alpha(VV^\top x_i) - V^\top \nabla \alpha(x_i) \right\|^2. \quad (3.4)$$

Geometrically, it measures how much the in-plane gradient changes when each sample point is replaced by its projection onto the plane; $D(V) = 0$ precisely when the landscape, on the sampled region, depends on x only through $V^\top x$.

3.2 The captured fraction of gradient energy

Raw disagreement is not yet a fair score, because it is biased toward flat directions: where gradients are small, D is small regardless of how well the plane represents the landscape. We therefore normalize by the projected gradient energy,

$$\text{total}(V) = \frac{1}{N} \sum_{i=1}^N \|V^\top \nabla \alpha(x_i)\|^2 = \text{Tr}(V^\top C V), \quad C = \frac{1}{N} \sum_{i=1}^N \nabla \alpha(x_i) \nabla \alpha(x_i)^\top, \quad (3.5)$$

where C is the empirical covariance of the gradient. The ratio $D(V)/\text{total}(V)$ is then a dimensionless, scale-free measure of how much of the projected gradient the plane leaves unexplained, and the criterion is to make it small. Equivalently, we maximize the *captured fraction of projected gradient energy*

$$F(V) = 1 - \frac{D(V)}{\text{total}(V)}, \quad V^\star = \arg \max_V F(V) = \arg \min_V \frac{D(V)}{\text{total}(V)}. \quad (3.6)$$

A value of $F(V) = 1$ means the plane captures the landscape perfectly on the sampled region; $F(V) = 0$ means it captures none of it; and $F(V) < 0$, which arises when D exceeds total, means the in-plane gradient departs from the true projected gradient by more than the gradient's own size, so the plane actively misrepresents the landscape.

The normalization is not cosmetic; without it the criterion is misled by flat directions. Figure 3.1 scores directions on a two-dimensional Gaussian bump $\alpha(x) = 1 - \exp(-\frac{1}{2}x^\top \Sigma x)$, whose Hessian Σ has a steep eigenvector u_1 and a flat one u_2 , using its sampled gradients $\nabla \alpha(x) = e^{-\frac{1}{2}x^\top \Sigma x} \Sigma x$. The raw disagreement D has the units of a squared gradient, so along the flat, low-gradient axis it is small simply because the gradients there are small, regardless of how well that axis represents the landscape: minimizing D alone selects the flat direction u_2 . Dividing by the projected gradient energy total cancels this scale and leaves the dimensionless fraction of projected-gradient vari-

ation that the slice explains, which is largest along the steep, informative axis u_1 . The normalized criterion recovers the right direction where the raw one does not.

3.3 A significance score for any projection

The captured fraction $F(V)$ scores a projection on an absolute scale, but to judge whether a given visualization method has found a good plane we also want a reference: how does its score compare with what an arbitrary plane would achieve? Random directions, the starting point of filter normalization [1], supply a natural null model. Drawing the projection V uniformly at random, as an orthonormal $d \times k$ matrix, induces a null distribution of scores $F(V)$ against which any proposed projection can be compared.

Given a projection V_0 produced by any method, we draw M random projections $V_1, \dots, V_M$ and report the empirical p -value

$$p(V_0) = \frac{1}{M} \sum_{m=1}^M \mathbf{1}\{F(V_m) \geq F(V_0)\}, \quad (3.7)$$

the fraction of random planes at least as faithful as V_0 . A small p -value certifies that the method has found a plane far better than chance, and the test applies to any method, however its plane was produced, without fitting or tuning.

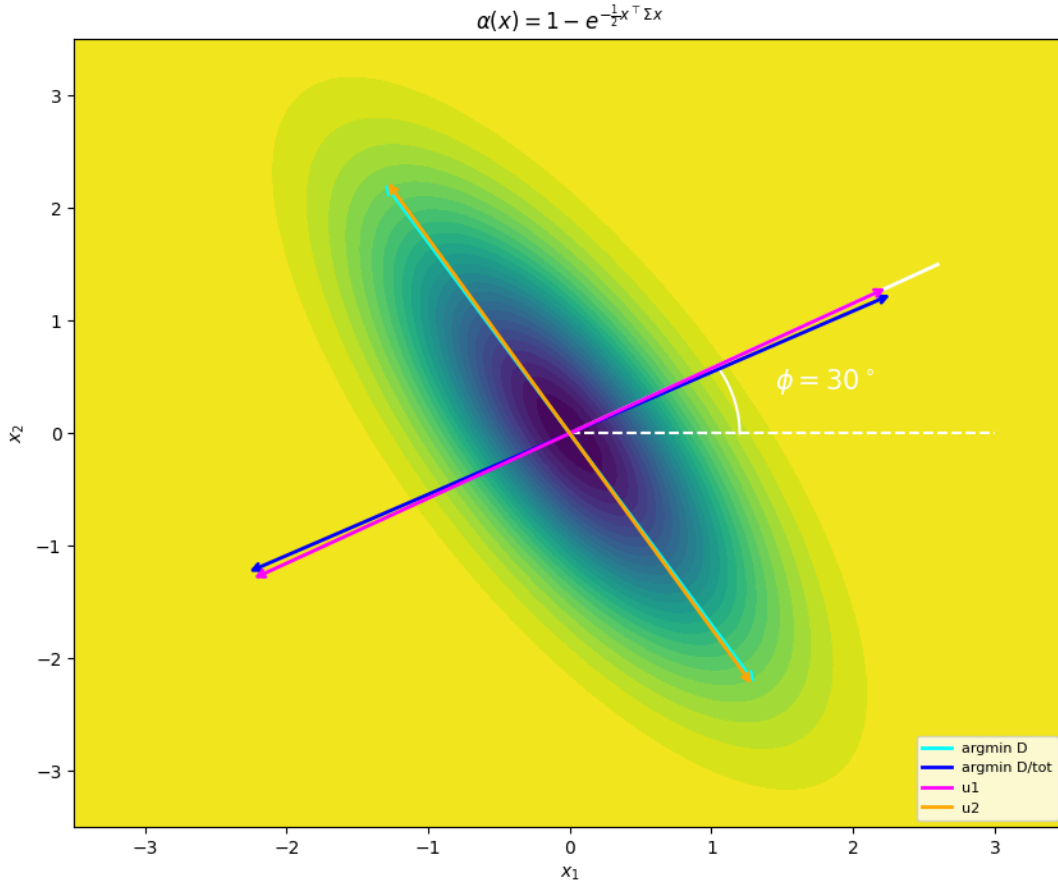


Figure 3.1: Why the disagreement is normalized. A two-dimensional Gaussian bump $\alpha(x) = 1 - \exp(-\frac{1}{2}x^\top \Sigma x)$ with eigenvalues (4, 0.5) and eigenvectors rotated 30° (white arc): steep along u_1 (30°), flat along u_2 (120°); the legend labels them e_1, e_2 . Computed from the gradient field alone, the raw disagreement D is minimized by the flat axis u_2 (cyan), whereas the normalized disagreement D/total is minimized by the steep axis u_1 (blue). Only the normalized criterion recovers the informative principal axis.

Chapter 4

Theory of the Loss Landscape Near a Minimum

The disagreement metric is defined for any smooth loss, but it is sharpest near a minimum. There the loss is a quadratic bowl, and we show that the metric is governed entirely by the Hessian: the projection it selects is the span of the leading Hessian eigenvectors, the principal axes of the bowl. Because every smooth loss is locally quadratic at a minimum, the analysis applies to any trained model.

4.1 Near a minimum, the loss is quadratic

Two things make the quadratic the right object to study. First, it is where visualization actually happens: a trained network sits at, or very close to, a minimum of its loss, and loss-landscape plots are drawn centered on that solution, so the near-minimum behavior is exactly what those plots depict. Second, near a minimum every smooth loss is the same kind of object. Let x^* be a local minimum of α . Since $\nabla\alpha(x^*) = 0$, the

second-order Taylor expansion is

$$\alpha(x) \approx \alpha(x^*) + \frac{1}{2}(x - x^*)^\top H(x - x^*), \quad H = \nabla^2 \alpha(x^*) \succeq 0, \quad (4.1)$$

a quadratic bowl whose shape is set entirely by the Hessian H . A statement about the metric on quadratics is therefore a statement about the metric near every minimum of every model, not about one example.

Because $D(V)$ and $\text{total}(V)$ depend on α only through its gradient, and the constant term vanishes under differentiation, the metric near x^* is fixed by the quadratic part alone. Centering coordinates at the minimum we study

$$\alpha(x) = \frac{1}{2} x^\top H x, \quad \nabla \alpha(x) = H x. \quad (4.2)$$

The quadratic is also the one regime in which the metric is analytically tractable, so it is where we can prove, rather than merely observe, what the criterion selects. The rest of the chapter shows that it recovers the eigenvectors of H .

4.2 The two-dimensional Gaussian model

To make the recovery concrete and visualizable we take a two-dimensional example, the Gaussian bump

$$\alpha(x) = 1 - \exp\left(-\frac{1}{2} x^\top \Sigma x\right), \quad \Sigma \succ 0, \quad (4.3)$$

whose minimum is at the origin and whose second-order expansion there is $\frac{1}{2} x^\top \Sigma x$, so that Σ is its Hessian. We work in the near-minimum regime, $\nabla \alpha(x) = \Sigma x$, and probe the landscape with an isotropic cloud $x_1, \dots, x_N \sim \mathcal{N}(0, I)$. In two dimensions Σ has

eigenvalues $\lambda_1 \geq \lambda_2 > 0$ with orthonormal eigenvectors u_1, u_2 , the principal axes of the elliptical level sets, with u_1 the high-curvature axis.

For a single unit direction v (the case $k = 1$) the two in-plane gradients are $(v^\top \Sigma v)(v^\top x)$ and $v^\top \Sigma x$; squaring their difference and averaging over $x \sim \mathcal{N}(0, I)$, so that $\mathbb{E}[xx^\top] = I$, gives the closed forms

$$\text{total}(v) = v^\top \Sigma^2 v, \quad D(v) = v^\top \Sigma^2 v - (v^\top \Sigma v)^2, \quad F(v) = \frac{(v^\top \Sigma v)^2}{v^\top \Sigma^2 v}. \quad (4.4)$$

Figure 4.1 makes this concrete. We fix the eigenvalues to $\lambda = (4, 0.5)$ and describe the Gaussian by an orientation angle, rotating its eigenvectors 30° from the coordinate frame so that the high-curvature axis u_1 lies at 30° and u_2 at 120° . As the slice direction v sweeps from 0° to 180° , the total energy $\text{total}(v)$ peaks at 30° , where v aligns with u_1 and the gradient is largest, and is smallest at 120° . The disagreement $D(v)$ carries the structure of interest: it vanishes exactly at 30° and 120° and rises to a maximum between them, where Σ tilts v the most. Their ratio $F(v)$ spikes to one at the two eigen-directions and dips to about 0.4 in between. The minimizers of D , equivalently the maximizers of F , are therefore exactly the eigenvectors of Σ , the principal axes of the bowl and the eigenvectors of its Hessian, recovered here even though they are not aligned with the coordinate axes.

Figure 4.2 checks the closed form against direct simulation: at each angle F is estimated from 10^6 sampled gradients, and the Monte Carlo estimate matches the analytic curve to within 0.0018, spiking to one at the eigen-directions 30° and 120° .

Finally, we ask whether the recovered direction could have arisen by chance. Applied to the full Gaussian bump $\alpha(x) = 1 - \exp(-\frac{1}{2}x^\top \Sigma x)$ rather than its quadratic approximation, the normalized criterion is minimized at 28.5° , within 1.5° of the high-curvature eigenvector u_1 at 30° , where $D/\text{total} = 0.38$. Against a null of 50,000 directions drawn

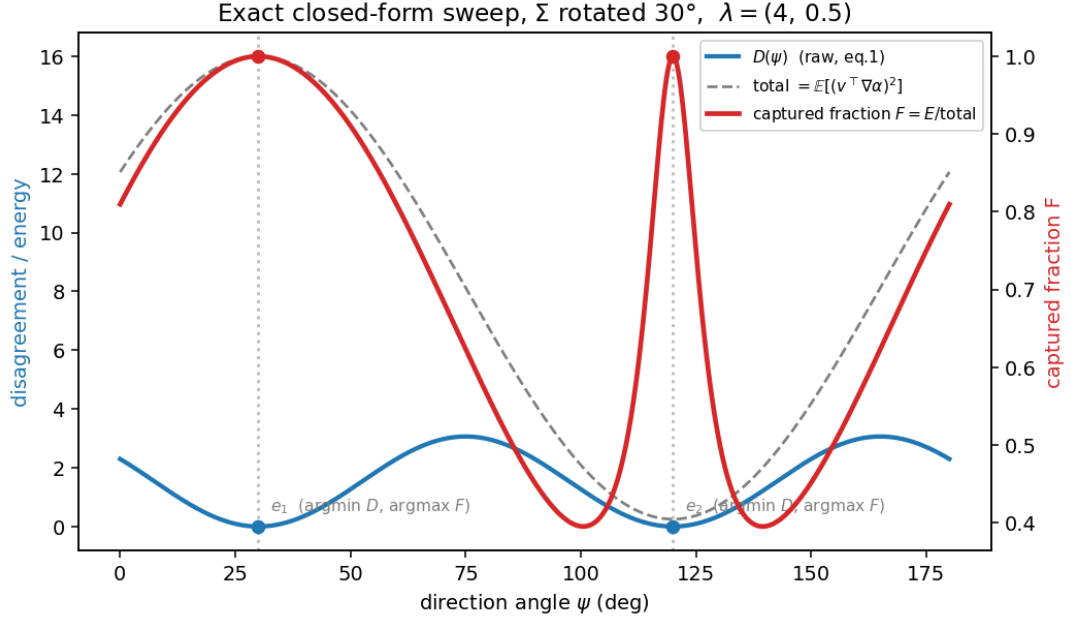


Figure 4.1: The captured fraction and its parts for the two-dimensional Gaussian model with eigenvalues $\lambda = (4, 0.5)$ and eigenvectors rotated 30° from the coordinate frame, as the slice direction sweeps from 0° to 180° . The total energy $\text{total}(v) = v^\top \Sigma^2 v$ peaks at the high-curvature axis (30°); the disagreement $D(v)$ vanishes at both eigen-directions (30° and 120°) and peaks between them; and the captured fraction $F(v)$ spikes to one at the eigen-directions and dips to about 0.4 in between. The minimizers of D and maximizers of F coincide with the Hessian eigenvectors, recovered even though they are not axis-aligned.

uniformly on $[0, \pi)$, only 69 score at least this low, a p -value of 1.4×10^{-3} under the significance test of Chapter 3. The criterion recovers the leading Hessian eigenvector, and significantly so, even on the non-quadratic loss.

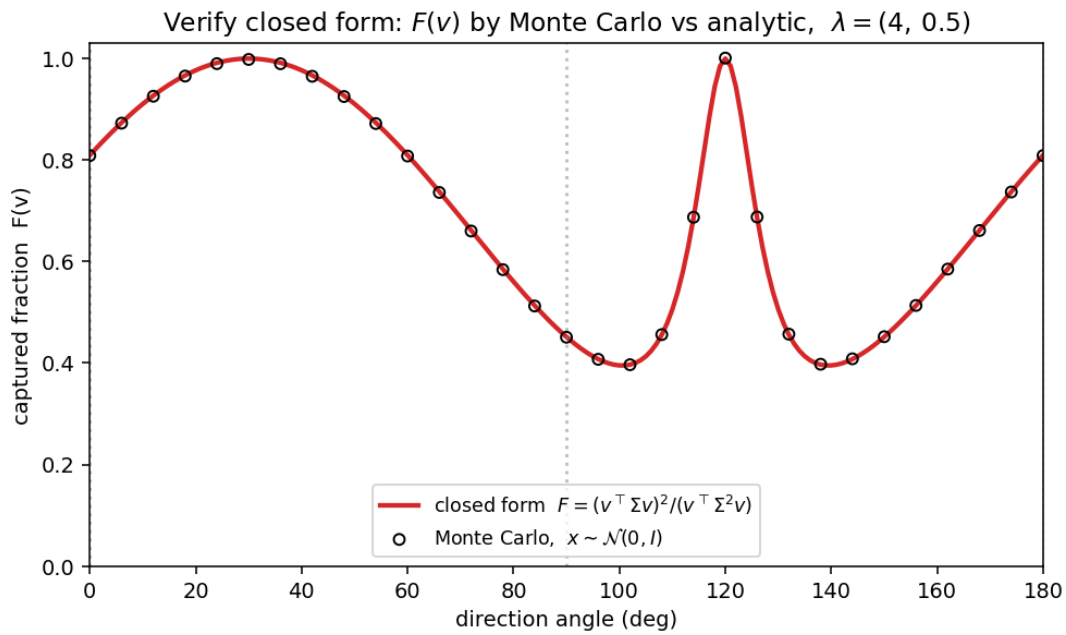


Figure 4.2: Verification of the closed form. Sweeping the direction angle, the captured fraction estimated from 10^6 Monte Carlo samples ($x \sim \mathcal{N}(0, I)$, gradients $g = \Sigma x$) matches the closed form $F(v) = (v^\top \Sigma v)^2 / (v^\top \Sigma^2 v)$ to within 0.0018 at every angle. F spikes to one at the eigen-directions (30° and 120°) and dips to about 0.4 between them.

Chapter 5

Theory of the Loss Landscape in General

In Chapter 4 we analyzed the near-minimum case. We found that when the loss is quadratic, the criterion recovers the Hessian eigenvectors. Away from a minimum the loss is no longer quadratic, there is no single Hessian to recover, and the local curvature can be actively misleading. What does D/total select then?

Unlike the quadratic case of Chapter 4, we do not solve for the minimizer in closed form here; we proceed empirically, sweeping D/total numerically on a controlled analytic model and reporting what its minimizer does. The conclusions of this chapter are therefore *observations* from those sweeps, with the single exception of the local Hessian, which we compute exactly. The model is the landscape of Chapter 4, $\alpha(x) = 1 - \exp(-\frac{1}{2}x^\top \Sigma x)$ with Σ rotated 30° and $\lambda = (4, 0.5)$, swept over 361 directions in $[0^\circ, 180^\circ]$ from $N = 2 \times 10^5$ samples.

The finding is that D/total is a global criterion: it integrates the gradient field over the whole sampling distribution and selects the direction along which the projected gradient is most predictable from the projected coordinate, which is in general not the local

Hessian eigenvector.

5.1 The criterion is global, not local

We first move the sampling center off the minimum: $x \sim \mathcal{N}(\mu, I)$ with $\mu = r e_1(\Sigma)$ for increasing radius r , where $e_1(\Sigma)$ is the leading eigenvector of the Hessian at the minimum (the high-curvature axis of Chapter 4). The question is whether the criterion tracks this global structure or the *local* structure, the leading eigenvector $e_1(H(\mu))$ of the Hessian at the sampling center.

The local Hessian we can write down exactly. With $q(x) = \frac{1}{2}x^\top \Sigma x$, so that $\alpha = 1 - e^{-q}$ and $\nabla q = \Sigma x$, the gradient is $\nabla \alpha = e^{-q} \Sigma x$, and differentiating again, with the product rule and $\nabla e^{-q} = -e^{-q} \Sigma x$,

$$\begin{aligned} \nabla^2 \alpha(x) &= e^{-q} \Sigma + (\Sigma x) (\nabla e^{-q})^\top \\ &= e^{-q} [\Sigma - (\Sigma x)(\Sigma x)^\top]. \end{aligned} \tag{5.1}$$

Evaluated at the sampling center this gives the local Hessian

$$H(\mu) = e^{-\frac{1}{2}\mu^\top \Sigma \mu} [\Sigma - (\Sigma \mu)(\Sigma \mu)^\top] \tag{5.2}$$

Whether or not the criterion follows it is an empirical question, which Figure 5.1 answers. Across every swept radius from $r = 0$ to $r = 3$, $\arg \min D/\text{total}$ stays locked to the global eigenvector $e_1(\Sigma)$ with zero error, even where the local leading eigenvector has flipped away from it. We observe, in other words, that the criterion does not track the local Hessian: it integrates over the whole cloud and reports the direction along which the gradient field is globally organized.

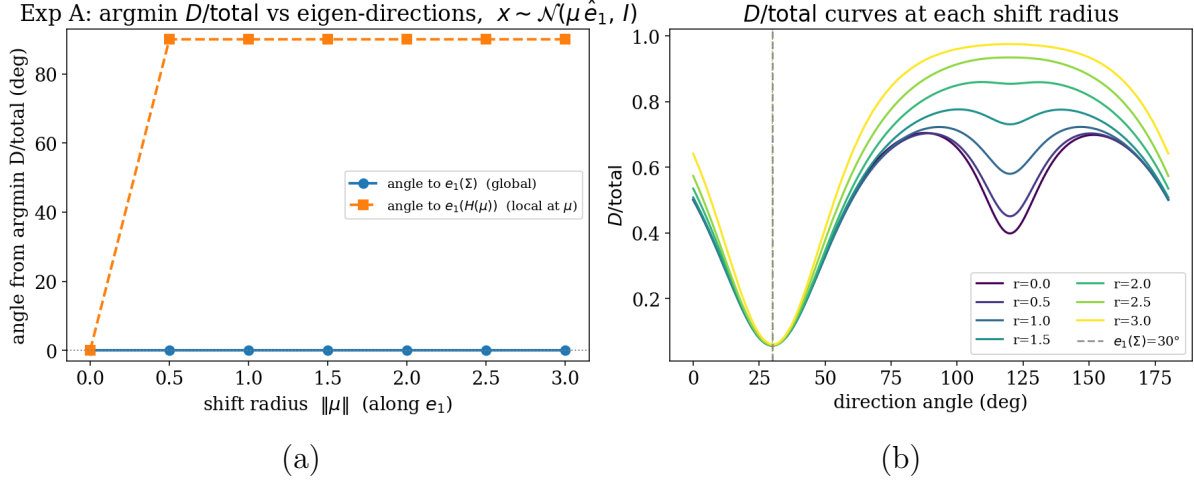


Figure 5.1: The criterion is global. The sampling center is shifted along $e_1(\Sigma)$ by radius r . (a) The angle from $\text{argmin } D/\text{total}$ to the global eigenvector $e_1(\Sigma)$ stays at 0° for all r (blue), while its angle to the local Hessian eigenvector $e_1(H(\mu))$ jumps to 90° as the center moves off the minimum (orange). (b) The D/total curves at each radius all share the same minimizer at $e_1(\Sigma) = 30^\circ$ (dashed line), even as their overall level rises. The minimizer is insensitive to the local Hessian.

The significance is that the metric is anchored to the loss, not to the vantage point from which it is viewed. This is what makes the criterion usable away from a minimum. A visualization centered at an early checkpoint could still reflect the global organization of the loss rather than where the landscape is centered.

5.2 Which structure wins: the dominant basin

A harder test puts two competing structures in one landscape. Let

$$\alpha(x) = 1 - \frac{1}{2}e^{-\frac{1}{2}(x-a)^\top \Sigma_1(x-a)} - \frac{1}{2}e^{-\frac{1}{2}(x+a)^\top \Sigma_2(x+a)}, \quad (5.3)$$

two Gaussian wells with Σ_1 axis-aligned ($0^\circ/90^\circ$) and $\Sigma_2 = \Sigma_1$ rotated 60° , offset by $a = [1.5, 0]$ and sampled from $x \sim \mathcal{N}(0, I)$. The wells pull in different directions: well 1's principal axis is 0° , well 2's is 60° . We want to test if the criterion picks one or averages

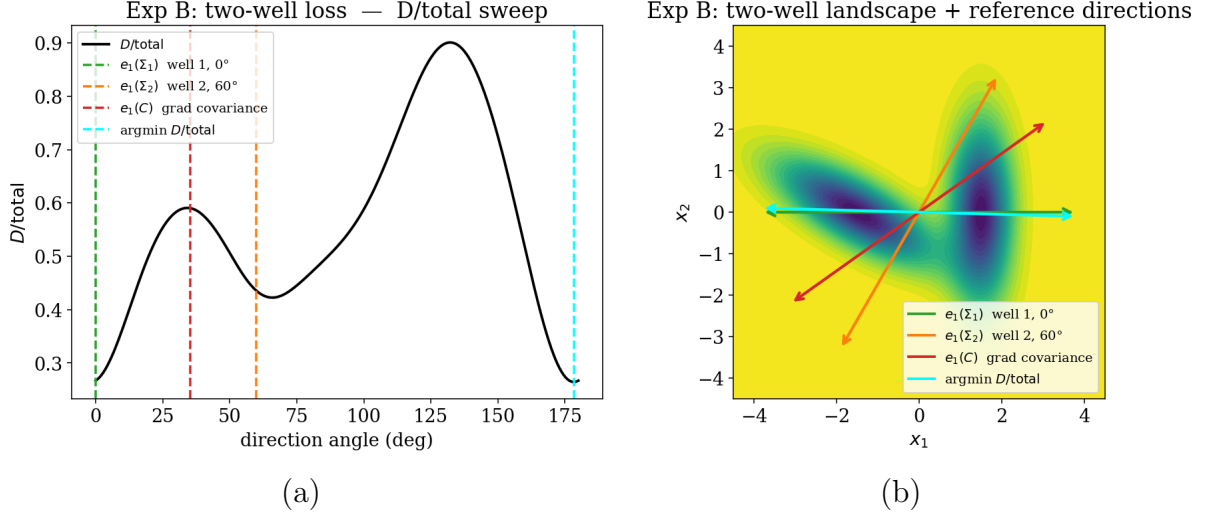


Figure 5.2: With competing basins the criterion is winner-take-all. (a) The D/total sweep over a two-well loss has a deep minimum at well 1's axis (0°) and a shallow one at well 2's (60°); $\text{argmin } D/\text{total}$ (cyan, $178.5^\circ \equiv 0^\circ$) lands on $e_1(\Sigma_1)$, not on $e_1(\Sigma_2)$ (60°) and not on the gradient-covariance direction $e_1(C)$ (35°). (b) The two-well landscape with the reference directions overlaid.

them.

Figure 5.2 shows that the criterion picks one. The D/total sweep has two local minima, a deep one at 0° (well 1, $D/\text{total} \approx 0.26$) and a shallow one near 60° (well 2, ≈ 0.44), so both axes are visible as structures; but the global minimum sits at $e_1(\Sigma_1)$. It is not the gradient-covariance direction $e_1(C)$ at 35° , which splits the difference, and it is not a blend of the two axes. The criterion is winner-take-all, returning the axis of the basin that best organizes the gradient field rather than a consensus.

5.3 What the criterion selects in general

Together the experiments support the following empirical characterization of D/total away from a minimum:

- It is *global*: it integrates the gradient over the full sampling distribution and is insensitive to the local Hessian at the sampling center (Figure 5.1).

- It selects the direction v along which the projected gradient $v^\top \nabla \alpha$ is most *predictable* from the projected coordinate $v^\top x$. For a quadratic this is the Hessian eigenvector (Chapter 4); in general it is the direction of strongest *structured* gradient variation.
- It is *not* the gradient-covariance direction $e_1(C)$, the active subspace of Constantine [9].
- With several competing structures it is *winner-take-all*: the basin whose principal axis best organizes the gradient field is selected, rather than a mixture (Figure 5.2).

Chapter 6

Testing Visualization Methods on Task-Trained RNNs

The previous chapters developed the metric and characterized it on analytic landscapes. We now turn it on trained networks and use it for its intended purpose: scoring the loss-landscape visualizations that the field actually produces. We train recurrent networks on the flip-flop memory task [10], visualize the loss around the trained solution with the standard methods of Chapter 2, and attach a number to each plane, its normalized disagreement D/total , the share of the projected gradient the plane leaves unexplained, with smaller meaning more faithful. The planes themselves were shown in Figure 2.1, where they looked plainly different from one another but carried no quantity to say which, if any, was faithful; that is the gap the metric fills.

6.1 Setup

A recurrent network carries a hidden state that it updates at each time step through a recurrent weight matrix and a nonlinearity, reading in an input stream and emitting

an output. We study two such networks that differ only in that recurrent matrix. The *rank-one RNN* constrains it to a single outer product, $J = m n^\top$ for two vectors m and n ; this low-rank parametrization keeps the recurrent dynamics low-dimensional and lets connectivity, dynamics, and computation be related directly, making it a minimal but expressive recurrent model [11]. The *full-rank RNN* leaves the recurrent matrix unconstrained, as in a standard network. Comparing the two lets us watch the metric as the loss landscape grows in complexity.

Both are trained on the flip-flop memory task [10], a standard probe of how a recurrent network stores information. The network receives brief input pulses on one or more independent channels and must report, on each channel, the sign of the most recent pulse, holding that value until the next pulse on the same channel flips it. Solving the task requires persistent memory, which the trained network builds by forming a set of stable fixed points in its state space, one for each memory state it must hold. The task is a clean benchmark: it demands genuine memory yet stays simple enough that the trained solution can be examined directly.

We train each model to convergence and, around the trained parameters, build the three visualization planes of Chapter 2: a pair of filter-normalized random directions, the start-to-end plane (one axis the vector from the initial to the final parameters), and the trajectory-PCA plane (the top principal components of the optimization path). For each plane we sample gradients of the loss about the solution and compute the disagreement D , the projected gradient energy total, and their ratio D/total .

6.2 Scores

Table 6.1 reports the result. On the rank-one model the three methods are cleanly separated. The start-to-end plane is the most faithful by a wide margin, $D/\text{total} = 0.10$:

Model	Plane	D	total	D/total
Rank-one RNN	start-to-end	5.9×10^{-4}	6.1×10^{-3}	0.10
Rank-one RNN	trajectory PCA	2.8×10^{-3}	4.5×10^{-3}	0.62
Rank-one RNN	filter normalization	2.1×10^{-4}	2.6×10^{-4}	0.82
Full-rank RNN	trajectory PCA	2.9×10^{-2}	6.6×10^{-2}	0.44
Full-rank RNN	start-to-end	1.9×10^{-2}	3.9×10^{-2}	0.49
Full-rank RNN	filter normalization	3.8×10^{-1}	3.7×10^{-1}	1.04

Table 6.1: Normalized disagreement D/total of loss-landscape visualization planes for RNNs trained on the flip-flop task, scored on the planes of Figure 2.1. Smaller is more faithful: $D/\text{total} = 0$ is a perfectly faithful plane, $D/\text{total} = 1$ a plane that captures none of the projected gradient, and $D/\text{total} > 1$ a plane whose in-plane gradient differs from the true projected gradient by more than the gradient’s own size.

the direction along which training travelled is also the direction along which the loss genuinely changes, so the in-plane gradient matches the true gradient almost everywhere on the plane. Trajectory PCA is middling, $D/\text{total} = 0.62$, and the filter-normalized random plane is the worst, $D/\text{total} = 0.82$; its projected gradient energy is also the smallest of the three (total = 2.6×10^{-4}), so the random plane both sees little of the gradient and leaves most of it unexplained.

On the full-rank model the ordering shifts, but the random plane is again the worst, now decisively so. Filter normalization scores $D/\text{total} = 1.04$, above one: its in-plane gradient is no better a description of the loss than no gradient at all, so the plane captures essentially none of the landscape’s structure. The two data-driven planes fare far better, each capturing roughly half the projected gradient: trajectory PCA reaches 0.44 and start-to-end 0.49. A filter-normalized plot of this network would look as clean and convincing as any, yet the metric marks it, with a single number above one, as showing nothing.

6.3 What the scores reveal

Two readings follow. First, the random plane is consistently the least faithful: filter normalization is the worst method on both models, and on the full-rank one it crosses the threshold where the plane captures nothing at all. This is exactly the failure visual inspection cannot catch, since a random plane produces a picture as smooth and plausible as any other. The data-driven planes do better, sometimes dramatically, as with the almost perfectly faithful rank-one start-to-end plane. Second, the ranking of methods depends on the landscape, not on the method alone: start-to-end is the clear winner on the rank-one model, while trajectory PCA edges it out on the full-rank one. No method is uniformly best, which is precisely why a per-instance, quantitative score is needed rather than a fixed rule of thumb.

Chapter 7

Discussion and Future Work

7.1 Summary

Loss-landscape plots are everywhere in deep learning, yet the choice of plane that produces them has been validated by eye: a picture that shows a smooth basin is taken to be a good one. This thesis replaces that process with a quantity. The *disagreement metric* D asks whether the gradient seen inside a projection plane agrees with the projection of the true gradient, and reports how much of the projected gradient the plane leaves unexplained.

The metric is grounded in the development of Chapters 3–5. Chapter 3 defined it, together with the significance test that turns it into a comparable score. Near a minimum, where every smooth loss is quadratic, Chapter 4 showed that minimizing D/total recovers the eigenvectors of the Hessian. Away from a minimum, Chapter 5 showed that the criterion is global: it integrates the gradient field over the sampled region and selects the direction along which that field is most structured, ignoring the local Hessian, which can rotate completely as the sampling center moves. Finally, Chapter 6 turned the metric on task-trained recurrent networks, where it ranked the standard visualization methods and

exposed a filter-normalized plane that visual inspection would accept yet that captures essentially none of the loss’s gradient. Together these give loss-landscape visualization the quantitative goodness-of-fit measure it has lacked.

7.2 Future directions: low-dimensional training

The metric does more than grade a picture; it selects a subspace, the minimizer of D/total . That subspace is the set of directions along which the loss’s gradient is most predictable from the in-plane coordinates, which is to say the directions that carry the structure of the loss. A natural question is whether one can *train* there. Restricting optimization to the selected subspace, writing $\theta = \theta_0 + Vy$ and updating only the k coordinates y , gives *low-dimensional training*; if the subspace captures the gradient structure, optimization confined to it should recover much of what full-parameter training achieves.

The prediction is concrete and testable: subspaces with smaller D/total should support more effective low-dimensional training, tying the diagnostic of this thesis directly to a downstream task and giving an independent check on what the metric measures. The idea connects to the intrinsic-dimension program of Li et al. [7], to the observation of Gur-Ari et al. [8] that gradient descent already concentrates in a small subspace, and to trajectory-plane training [2], but it replaces a random or trajectory-defined subspace with one chosen by a principled, landscape-aware criterion.

Chapter 6 is both a caution and a guide here. A single plane captured the rank-one landscape almost perfectly (start-to-end, $D/\text{total} = 0.10$), but the most faithful plane found for the full-rank model still left close to half of the projected gradient unexplained (trajectory PCA, 0.44). Richer models may therefore need subspaces of dimension $k > 2$, and the appropriate k is itself something D/total could help set, namely the dimension at which the unexplained fraction becomes small.

7.3 Open questions

Two further questions remain. The first is computational. Scoring a given plane is cheap, but minimizing D/total to find the optimal subspace requires fresh gradient evaluations for every candidate and grows expensive as the parameter dimension increases. The second is theoretical. Chapter 5 characterized the global direction empirically rather than in closed form; a general account of what D/total selects on a non-quadratic loss, of the kind available for the quadratic case, remains to be given.

References

- [1] H. Li, Z. Xu, G. Taylor, C. Studer, and T. Goldstein, *Visualizing the loss landscape of neural nets*, in *Advances in Neural Information Processing Systems* (S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, eds.), vol. 31, Curran Associates, Inc., 2018.
- [2] R. Ding, T. Li, and X. Huang, *Better loss landscape visualization for deep neural networks with trajectory information*, in *Proceedings of the 15th Asian Conference on Machine Learning* (B. Yanıkoğlu and W. Buntine, eds.), vol. 222 of *Proceedings of Machine Learning Research*, pp. 311–326, PMLR, 11–14 Nov, 2024.
- [3] N. S. Keskar, D. Mudigere, J. Nocedal, M. Smelyanskiy, and P. T. P. Tang, *On large-batch training for deep learning: Generalization gap and sharp minima*, in *International Conference on Learning Representations (ICLR)*, 2017.
- [4] L. Dinh, R. Pascanu, S. Bengio, and Y. Bengio, *Sharp minima can generalize for deep nets*, in *International Conference on Machine Learning (ICML)*, 2017.
- [5] R. Pascanu, T. Mikolov, and Y. Bengio, *On the difficulty of training recurrent neural networks*, in *Proceedings of the 30th International Conference on Machine Learning* (S. Dasgupta and D. McAllester, eds.), vol. 28 of *Proceedings of Machine Learning Research*, (Atlanta, Georgia, USA), pp. 1310–1318, PMLR, 17–19 Jun, 2013.
- [6] I. J. Goodfellow, O. Vinyals, and A. M. Saxe, *Qualitatively characterizing neural network optimization problems*, in *International Conference on Learning Representations (ICLR)*, 2015.
- [7] C. Li, H. Farkhoor, R. Liu, and J. Yosinski, *Measuring the intrinsic dimension of objective landscapes*, in *International Conference on Learning Representations (ICLR)*, 2018.
- [8] G. Gur-Ari, D. A. Roberts, and E. Dyer, *Gradient descent happens in a tiny subspace*, *arXiv preprint arXiv:1812.04754* (2018).

- [9] P. G. Constantine, *Active Subspaces: Emerging Ideas for Dimension Reduction in Parameter Studies*, vol. 2 of *SIAM Spotlights*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2015.
- [10] D. Sussillo and O. Barak, *Opening the black box: Low-dimensional dynamics in high-dimensional recurrent neural networks*, *Neural Computation* **25** (2013), no. 3 626–649.
- [11] F. Mastrogiuseppe and S. Ostoic, *Linking connectivity, dynamics, and computations in low-rank recurrent neural networks*, *Neuron* **99** (2018), no. 3 609–623.