Article

Not peer-reviewed version

Are Flat Minima an Illusion?

Michael Timothy Bennett *

Posted Date: 23 March 2026

doi: 10.20944/preprints202603.1737.v1

Keywords: flat minima; sharpness; generalisation; reparameterisation invariance; weakness; PACBayes; large-batch training; model selection; ReLU networks; loss landscape

 doi:10.20944/preprints202603.1737.v1

Article

Are Flat Minima an Illusion?

Michael Timothy Bennett

School of Computing, The Australian National University, Australia; michael.bennett@anu.edu.au

Abstract

Flat minima are widely believed to generalise better than sharp ones. I present a theoretical and empirical case that this belief is wrong. On the theoretical side, I prove that weakness, the measure-theoretic volume of compatible completions in the learner's embodied language, is reparameterisation-invariant, minimax-optimal, and tracked by the PAC-Bayes bound. Sharpness is none of these things. On the empirical side, I show that the large-batch generalisation advantage *vanishes* as a function of training set size. On MNIST, large-batch networks generalise better than small-batch networks when training data is scarce, and the advantage becomes negligible when data is abundant. A quantity that correlates, anticorrelates, or does not correlate depending on the data regime is not a cause. It is a confound. I confirm reparameterisation non-invariance of the Hessian trace on trained networks (up to 99× change with zero change in function or generalisation). I construct a formal vocabulary for frozen-partition ReLU networks (Appendix H) and measure the pair proxy via linear feasibility on 100 networks (Section 5.7). Weakness outperforms sharpness at model selection ($\rho = +0.374$, $p = 0.00012$ vs $\rho = -0.226$, $p = 0.024$).

Keywords: flat minima; sharpness; generalisation; reparameterisation invariance; weakness; PAC-Bayes; large-batch training; model selection; ReLU networks; loss landscape

---

1. Introduction

Flat minima generalise better than sharp ones. The literature treats this as established. Hochreiter and Schmidhuber (1997) argued it in 1997, Keskar et al. (2017) demonstrated it empirically in 2017. Sharpness-Aware Minimisation (Foret et al. 2021) explicitly seeks flat minima and improves generalisation. The correlation is not in dispute.

The causation is. Dinh et al. (2017) gave strong reason to doubt it. They constructed a reparameterisation that transforms any sharp minimum into a flat one without changing the function the network computes. The Hessian eigenvalues change. The loss landscape geometry changes. The function does not. The predictions do not.

Several authors have proposed parameterisation-invariant sharpness measures. Tsuzuku et al. (2020) normalised by parameter scale. Petzka et al. (2021) used a Fisher information metric. Kwon et al. (2021) introduced adaptive sharpness. They still do not answer the question "what property of the learned *function* makes it generalise?"

This paper offers three contributions.

1. **The vanishing advantage.** The generalisation advantage of large-batch over small-batch training vanishes as a function of training set size. On MNIST with a 3-layer ReLU MLP, large-batch networks generalise better at 500, 2,000, 6,000, and 12,000 training points. At 24,000 the gap is small (+0.08 pp, $p = 0.002$). At 60,000 it is negligible (+0.02 pp). A quantity that correlates at some scales and not others is not a cause. It is a confound.
2. **Reparameterisation invariance.** I prove that weakness (Bennett 2023, 2025) is reparameterisation-invariant by construction. I confirm non-invariance of the Hessian trace on trained networks, with changes of up to 99× while generalisation stays exactly constant.

 doi:10.20944/preprints202603.1737.v1

3. **Weakness beats sharpness.** I construct a formal vocabulary for frozen-partition ReLU networks (Appendix H) and measure weakness via linear feasibility on 100 networks with the same architecture, data, and training. Weakness outperforms sharpness at model selection ($\rho = +0.374$, $p = 0.00012$ vs $\rho = -0.226$, $p = 0.024$). Simplicity does not predict at all ($p = 0.848$).

I am honest about what this paper does and does not do. Weakness outperforms sharpness at model selection ($\rho = 0.374$, $p = 0.00012$ vs $\rho = -0.226$, $p = 0.024$). Buffer size (k_{free}) is a significant but modest predictor ($\rho = +0.117$, $p = 0.043$). The region-class vocabulary is an approximation. Formalising the exact shared-weight vocabulary is an open problem.

The structure is as follows. Section 2 reviews the definitions from Stack Theory. Section 3 extends weakness to continuous domains. Section 4 presents the reparameterisation invariance proof, the PAC-Bayes connection, and the confounding diagnosis. Section 5 presents the experiments. Section 6 locates continuous weakness in the landscape of generalisation theory. Section 7 discusses implications, limitations, and future work.

All proofs are self-contained. The definitions used here are shared with the Technical Appendices (Bennett 2025), which serve as the master reference for Stack Theory. The finite-case predecessors of the continuous results are proved in Appendix B for completeness.

2. Background

I need four definitions. Each one builds on the last.

Definition 1 (Environment and programs)**.** *An* environment *is a nonempty set Φ of mutually exclusive states. A* program *is any subset $p \subseteq \Phi$. Write $\mathcal{P} = 2^{\Phi}$ for the set of all programs. A* vocabulary *is any set of programs $\mathfrak{v} \subseteq \mathcal{P}$.*

Think of Φ as the set of all possible configurations of the world. Each state is one complete configuration. A program is a constraint that holds in some configurations and not others. A vocabulary is the set of constraints a system can express. For a neural network, the vocabulary is the set of input-output behaviours the architecture can realise.

Definition 2 (Embodied language and statements)**.** *A vocabulary $\mathfrak{v}$ induces an* embodied language

$$L_{\mathfrak{v}} = \left\{ l \subseteq \mathfrak{v} \;\middle|\; \bigcap_{p \in l} p \neq \emptyset \right\}.$$

Elements $l \in L_{\mathfrak{v}}$ are called statements*. The* truth set *of l is $T(l) = \bigcap_{p \in l} p$.*

A statement is a consistent conjunction of programs. Raising my arm is a statement. It combines programs for muscle contraction, balance, spatial position, and so on. The embodied language $L_{\mathfrak{v}}$ is *not* the full powerset $2^{\mathfrak{v}}$. Most combinations of programs are mutually exclusive. You cannot raise and lower your arm at the same time. The satisfiability constraint $\bigcap_{p \in l} p \neq \emptyset$ is what makes the structure interesting.

Definition 3 (Extensions and weakness)**.** *A* completion *of $x \in L_{\mathfrak{v}}$ is any $y \in L_{\mathfrak{v}}$ with $x \subseteq y$. The* extension *of x is*

$$\text{Ext}(x) = \{ y \in L_{\mathfrak{v}} \mid x \subseteq y \}.$$

For a set of statements $X \subseteq L_{\mathfrak{v}}$, write $\text{Ext}(X) = \bigcup_{x \in X} \text{Ext}(x)$. The weakness *of x is $w(x) = |\text{Ext}(x)|$.*

Weakness measures how non-committal a statement is. The more commitments you can still make, the weaker the statement.

 doi:10.20944/preprints202603.1737.v1

Definition 4 (Tasks, correctness, and learning). *A* $\mathfrak{v}$-task *is a pair* $\alpha = \langle I_\alpha, O_\alpha \rangle$ *where* $I_\alpha \subseteq L_\mathfrak{v}$ *and* $O_\alpha \subseteq \mathrm{Ext}(I_\alpha)$. *A* policy $\pi \in L_\mathfrak{v}$ *is* correct for α *if* $\mathrm{Ext}(I_\alpha) \cap \mathrm{Ext}(\pi) = O_\alpha$. *The set of correct policies is* Π_α. *Task* α *is a* child *of* ω, *written* $\alpha \sqsubset \omega$, *if* $I_\alpha \subsetneq I_\omega$ *and* $O_\alpha \subseteq O_\omega$.

A task specifies inputs and correct outputs. A policy constrains how the system completes inputs. Correctness means the policy produces exactly the right outputs on the given inputs. A child task is a task with fewer examples. Learning means generalising from a child task to its parent.

The foundational result is that under an exchangeable prior over parent tasks, weakness maximisation is both necessary and sufficient for optimal generalisation (Bennett 2023, 2025). Among all correct policies for the child, the ones most likely to also be correct for the unknown parent are the weakest. Minimising description length is neither necessary nor sufficient (Bennett 2024).

3. Extending Weakness to Continuous Domains

When the vocabulary is finite, weakness is a count. You enumerate completions and take the cardinality. When the vocabulary is infinite, every uncountable extension has the same cardinality, so counting breaks. This section replaces counting with measuring.

Definition 5 (Measurable vocabulary). *A* measurable vocabulary *is a triple* $(\mathfrak{v}, \mathcal{A}_L, \mu)$ *where* $\mathfrak{v} \subseteq \mathcal{P}$ *is a vocabulary (possibly infinite),* $\mathcal{A}_L$ *is a* σ*-algebra on* $L_\mathfrak{v}$, *and* μ *is a* σ*-finite measure on* $(L_\mathfrak{v}, \mathcal{A}_L)$. *I require that* $\mathrm{Ext}(l)$ *is* $\mathcal{A}_L$*-measurable for every* $l \in L_\mathfrak{v}$. *Since* $\mathrm{Ext}(X) = \bigcup_{x \in X} \mathrm{Ext}(x)$ *for any* $X \subseteq L_\mathfrak{v}$, *and a countable union of measurable sets is measurable,* $\mathrm{Ext}(I_\alpha)$ *is measurable whenever* I_α *is countable. All tasks in this paper have finite input sets.*

The measure μ does for infinite vocabularies what counting measure does for finite ones. It assigns a size to sets of statements so that extensions can be compared. When $\mathfrak{v}$ is finite, setting μ to counting measure recovers every existing result.

Definition 6 (Continuous weakness). *The* μ-weakness *of* $l \in L_\mathfrak{v}$ *is* $w_\mu(l) = \mu(\mathrm{Ext}(l))$.

Definition 7 (Continuous extension model). *Fix a measurable vocabulary* $(\mathfrak{v}, \mathcal{A}_L, \mu)$ *and a task* α *with output region* O_α. *Define the* unseen region $U = L_\mathfrak{v} \setminus \mathrm{Ext}(I_\alpha)$, *where* $0 < \mu(U) < \infty$ *and* $\mu(O_\alpha) < \infty$. *Assume* $(U, \mathcal{A}_L|_U, \mu|_U)$ *is an atomless standard measure space (Appendix A). A* continuous extension model *is a probability space* $(\Omega, \mathcal{F}, P)$ *with a random set* $S \colon \Omega \to 2^U$ *such that* $\{S \subseteq B\}$ *and* $\{S \subseteq B, S \cap A \neq \emptyset\}$ *are* $\mathcal{F}$*-measurable for every* $A, B \in \mathcal{A}_L|_U$. *The* buffer *of a correct policy* π *is* $B_\pi = \mathrm{Ext}(\pi) \cap U$. *The* P-weakness *is* $w_P(\pi) = P(S \subseteq B_\pi)$.

The parent task will additionally demand some set S of outputs. The policy survives if and only if every demand falls inside its buffer.

Definition 8 (μ-exchangeability and nondegeneracy). *A continuous extension model is* μ-exchangeable *if for every measure-preserving bijection* $\sigma \colon (U, \mu) \to (U, \mu)$ *and every measurable* $B \subseteq U$, $P(S \subseteq B) = P(S \subseteq \sigma(B))$. *It is* nondegenerate *if for every measurable* $A \subseteq U$ *with* $\mu(A) > 0$, *we have* $P(\emptyset \neq S \subseteq A) > 0$. *(This event is measurable because* $\{\emptyset \neq S \subseteq A\} = \{S \subseteq A\} \setminus \{S \subseteq \emptyset\}$.*)*

Write $G(\pi, P) = w_P(\pi) = P(S \subseteq B_\pi)$ for the *generalisation probability* of policy π under extension model P.

Lemma 1 (Buffer measure determines μ-weakness for correct policies). *For any correct* $\pi \in \Pi_\alpha$, $w_\mu(\pi) = \mu(O_\alpha) + \mu(B_\pi)$, *where* $O_\alpha = \mathrm{Ext}(I_\alpha) \cap \mathrm{Ext}(\pi)$ *is the output region (fixed for all* $\pi \in \Pi_\alpha$ *by the correctness condition) and* $B_\pi = \mathrm{Ext}(\pi) \cap U$. *$\mu$-weakness ranking and buffer-measure ranking agree among correct policies.*

 doi:10.20944/preprints202603.1737.v1

Proof. By the correctness condition, $\mathrm{Ext}(\pi) = O_\alpha \sqcup B_\pi$ where $O_\alpha \subseteq \mathrm{Ext}(I_\alpha)$ and $B_\pi \subseteq U = L_\mathfrak{v} \setminus \mathrm{Ext}(I_\alpha)$. These sets are disjoint and both $\mathcal{A}_L$-measurable (by the measurability requirement on extensions in Definition 5). So $w_\mu(\pi) = \mu(\mathrm{Ext}(\pi)) = \mu(O_\alpha) + \mu(B_\pi)$. Since $\mu(O_\alpha)$ is the same for all $\pi \in \Pi_\alpha$, ranking by $w_\mu(\pi)$ is equivalent to ranking by $\mu(B_\pi)$. □

The continuous theory produces four main theorems. Full proofs are in Appendices C–F.

Theorem 1 (Sufficiency)**.** *If P is μ-exchangeable, then $w_P(\pi) = f(\mu(B_\pi))$ for some nondecreasing f. Among correct policies, μ-weakness determines generalisation probability.*

Theorem 2 (Strict necessity)**.** *If P is μ-exchangeable and nondegenerate, and $\mu(B_{\pi_1}) > \mu(B_{\pi_2})$, then $w_P(\pi_1) > w_P(\pi_2)$.*

Theorem 3 (Unified optimality)**.** *Let $\mathcal{E}$ be the class of all μ-exchangeable models.*

1. Sufficiency. $\mu(B_{\pi_1}) \geq \mu(B_{\pi_2}) \implies G(\pi_1, P) \geq G(\pi_2, P)$ *for all* $P \in \mathcal{E}$.
2. Strict necessity. *If P is nondegenerate,* $\mu(B_{\pi_1}) > \mu(B_{\pi_2}) \implies G(\pi_1, P) > G(\pi_2, P)$.
3. Uniqueness. *All optimal policies have the same buffer measure $\mu(B_\pi)$, hence the same μ-weakness.*

Theorem 4 (Minimax optimality)**.** *If a μ-weakest correct policy π^* exists,*

$$\max_{\pi \in \Pi_\alpha} \inf_{P \in \mathcal{E}} G(\pi, P) = \inf_{P \in \mathcal{E}} \max_{\pi \in \Pi_\alpha} G(\pi, P),$$

both achieved by π^*.

This result is a corollary of sufficiency. Because π^* maximises $G(\pi, P)$ for every $P \in \mathcal{E}$ simultaneously, it trivially achieves the max-min and the minimax equality holds. The learner has a dominant strategy. No adversary can make weakness maximisation suboptimal.

4. The Flat Minima Resolution

The continuous theory resolves the flat minima puzzle. The resolution has three parts.

4.1. Reparameterisation Invariance of Weakness

Definition 9 (Policy map and function-preserving reparameterisation)**.** *Fix a measurable vocabulary defined over input-output behaviour. Let $\Theta \subseteq \mathbb{R}^d$ be a parameter space and let $F(\theta)$ denote the function computed by the network with parameters θ. A program $p \in \mathfrak{v}$ is* satisfied *by $F(\theta)$ if the input-output behaviour of $F(\theta)$ lies in p (viewed as a subset of Φ). The* policy map *is*

$$\Pi_F(\theta) = \{p \in \mathfrak{v} \mid F(\theta) \text{ satisfies } p\}.$$

A function-preserving reparameterisation *is a diffeomorphism $\phi\colon \Theta \to \Theta$ with $F(\phi(\theta)) = F(\theta)$ for all θ.*

Theorem 5 (Reparameterisation invariance of μ-weakness)**.** *If ϕ is function-preserving, then $\Pi_F(\phi(\theta)) = \Pi_F(\theta)$, hence $w_\mu(\Pi_F(\phi(\theta))) = w_\mu(\Pi_F(\theta))$.*

Proof. $F(\phi(\theta)) = F(\theta)$ by assumption. The policy map depends only on which programs $F(\theta)$ satisfies. Same function, same programs satisfied, same policy, same extension, same weakness. □

That is the proof. Three sentences. The definitions do the work. The vocabulary lives in function space. The reparameterisation does not move you in function space. The weakness does not change.

 doi:10.20944/preprints202603.1737.v1

4.2. PAC-Bayes Bounds Track Weakness

Theorem 6 (PAC-Bayes ordering agrees with μ-weakness)**.** *Fix a measurable vocabulary with $\mu(L_{\mathfrak{v}}) < \infty$. Set $P_0 = \mu/\mu(L_{\mathfrak{v}})$ (the normalised prior) and $Q_\pi = \mu|_{B_\pi}/\mu(B_\pi)$ (uniform posterior on the buffer). Then for correct π with $\mu(B_\pi) > 0$,*

$$\mathrm{KL}(Q_\pi \| P_0) = \log \frac{\mu(L_{\mathfrak{v}})}{\mu(B_\pi)},$$

strictly decreasing in $\mu(B_\pi)$.

Proof. The Radon-Nikodym derivative is $dQ_\pi/dP_0 = \mu(L_{\mathfrak{v}})/\mu(B_\pi) \cdot \mathbf{1}_{B_\pi}$. Computing the KL-divergence,

$$\mathrm{KL}(Q_\pi \| P_0) = \int \log \frac{dQ_\pi}{dP_0}\, dQ_\pi = \log \frac{\mu(L_{\mathfrak{v}})}{\mu(B_\pi)}.$$

Since $\mu(O_\alpha)$ is fixed for all $\pi \in \Pi_\alpha$ and $\mu(\mathrm{Ext}(\pi)) = \mu(O_\alpha) + \mu(B_\pi)$, the KL is strictly decreasing in $\mu(B_\pi)$, hence strictly decreasing in μ-weakness $w_\mu(\pi) = \mu(\mathrm{Ext}(\pi))$. □

Since the standard PAC-Bayes bound is increasing in $\mathrm{KL}(Q_\pi \| P_0)$ for fixed empirical risk and sample size, and correct policies have zero empirical risk by definition, the μ-weakest correct policy minimises the PAC-Bayes bound. PAC-Bayes bounds work because they track weakness. The KL-divergence is a proxy for a proxy. The fundamental quantity is the buffer measure.

4.3. Diagnosis of Confounding

The preceding results assemble the diagnosis.

1. Weakness is reparameterisation-invariant (Theorem 5).
2. Weakness determines generalisation probability under exchangeable demands (Theorems 1–4).
3. Sharpness is not reparameterisation-invariant (Dinh et al. 2017).
4. Therefore sharpness is not the driver.

5. Experiments

5.1. Setup

The cross-regime experiments (Sections 5.2–5.4) use a 3-layer ReLU MLP (784→256→128→10) trained on subsets of MNIST with no regularisation. The model-selection experiments (Sections 5.6 and 5.7) use smaller architectures as specified in each section. I train two regimes per data scale. The *small-batch* regime uses batch size 64 with learning rate 0.01–0.05. The *large-batch* regime uses batch size equal to the training set size at 500 and 2,000, and batch size 4,096 at 6,000 and above, with learning rate 0.2–0.5. All networks are trained to >99.5% training accuracy. At most scales, all networks reach exactly 100%. Exceptions are one network at 2,000 (99.95%), the 24,000 large-batch regime (99.98–100%), and 4 networks at 60,000 (99.998%). 25 networks per regime, 50 per data scale.

For the reparameterisation test, I use $T_\beta(\theta) = (\beta W_1, \beta b_1, W_2/\beta, b_2, W_3, b_3)$ between layers 1 and 2, and $T_\gamma(\theta) = (W_1, b_1, \gamma W_2, \gamma b_2, W_3/\gamma, b_3)$ between layers 2 and 3. By the positive homogeneity of ReLU, both preserve the function computed by the network. The 3-layer architecture provides two independent reparameterisation axes.

For each network I measure four quantities.

1. **Hessian trace.** Estimated via Hutchinson's method with 50 Rademacher vectors and autodifferentiation Hessian-vector products.
2. **Layer-1 activation patterns.** The number of distinct binary activation patterns $\mathbf{1}[W_1 x + b_1 > 0]$ across all unseen points.
3. **Layer-2 activation patterns.** The number of distinct binary activation patterns $\mathbf{1}[W_2 h_1 + b_2 > 0]$ across all unseen points.
4. **Ensemble agreement.** For each network, among the unseen points it misclassifies, the fraction correctly classified by other networks in the same experiment.

 doi:10.20944/preprints202603.1737.v1

5.2. The Vanishing Advantage

Table 1 presents test accuracy by regime across six training set sizes. At 500, 2,000, 6,000, and 12,000 training points, the large-batch regime generalises better. At 24,000 the gap is small but statistically significant (+0.08 pp, Welch $p = 0.002$). At 60,000 it is negligible (+0.02 pp). The large-batch advantage peaks at 2,000 training points and shrinks thereafter.

Table 1. Test accuracy by training regime across data scales. All rows use the 3-layer architecture. †Large-batch networks that did not reach exactly 100% training accuracy (one at 2,000 at 99.95%, 23 at 24,000 (>99.98%), 4 at 60,000 at 99.998%). All other networks reached exactly 100%.

n	Small-batch	Large-batch	Δ	Hessian (small / large)
500	85.3%	86.5%	+1.2	52.5 / 20.6
2,000	90.4%	92.0%	+1.6†	59.5 / 23.1
6,000	94.0%	94.9%	+1.0	61.9 / 25.9
12,000	96.0%	96.5%	+0.5	44.8 / 23.9
24,000	97.0%	97.1%	+0.08†	39.1 / 38.2
60,000	98.0%	98.0%	+0.02†	30.1 / 26.0

At every data scale, the large-batch regime has *lower* Hessian trace (Table 1). The standard association between large batch size and sharp minima (Keskar et al. 2017) is reversed at every scale tested. The Hessian tracks the generalisation gap but does not explain it.

5.3. Reparameterisation Invariance on Trained Networks

I apply T_β and T_γ to trained networks at three data scales (500, 2,000, 6,000) and measure all four quantities. Table 2 shows representative results for one flat-regime network at 6,000 training points.

Table 2. Reparameterisation invariance test. Net 0 (small-batch regime, 6,000 training points). Hessian trace changes by up to 99×. All other quantities are exactly invariant.

Reparam	Value	Test acc	Hessian	L1	L2	EA
T_β	1	0.9392	66.1	63,998	51,348	0.1909
T_β	2	0.9392	81.3	63,998	51,348	0.1909
T_β	5	0.9392	388.8	63,998	51,348	0.1909
T_β	10	0.9392	1,607.1	63,998	51,348	0.1909
T_β	20	0.9392	6,525.9	63,998	51,348	0.1909
T_γ	1	0.9392	62.9	63,998	51,348	0.1909
T_γ	5	0.9392	255.2	63,998	51,348	0.1909
T_γ	20	0.9392	3,378.9	63,998	51,348	0.1909

The Hessian trace changes by 99× under T_β and 54× under T_γ. Test accuracy does not change. Layer-1 activation patterns do not change. Layer-2 activation patterns do not change. Ensemble agreement does not change. Not approximately. Exactly, to the precision of floating-point arithmetic. This result holds for all 12 networks tested across all three data scales. Appendix G reports the Hessian ratios for representative networks at each scale.

The two independent reparameterisation axes (T_β between layers 1–2, T_γ between layers 2–3) demonstrate that the argument generalises to arbitrary depth. Any pair of adjacent ReLU layers admits a function-preserving scaling. The Hessian changes. The function does not. All function-level quantities stay constant. Figure 1 visualises the invariance for two representative networks.

 doi:10.20944/preprints202603.1737.v1

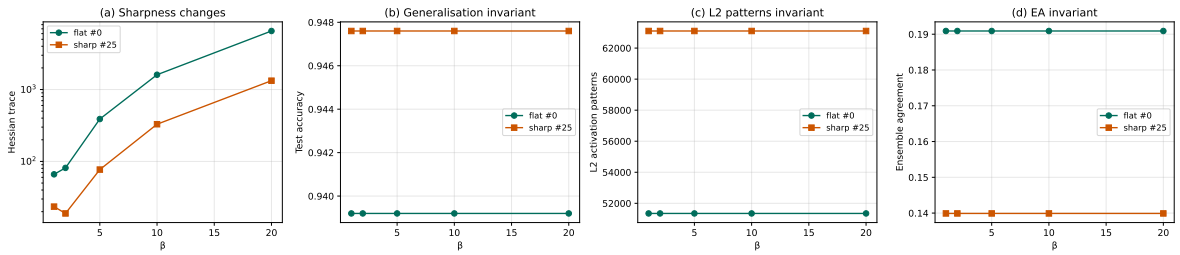

Figure 1. Reparameterisation invariance under T_β at 6,000 training points. (a) Hessian trace changes by two orders of magnitude. (b) Test accuracy is exactly invariant. (c) Layer-2 activation pattern count is exactly invariant. (d) Ensemble agreement is exactly invariant.

5.4. Correlations with Generalisation

Table 3 presents Spearman rank correlations between each measure and test accuracy across all 50 networks at each data scale.

Table 3. Spearman correlations (ρ) with test accuracy. L1 and L2 are layer-1 and layer-2 activation pattern counts on unseen data. EA is ensemble agreement. Bold indicates $|\rho| > 0.7$. The unseen set comprises leftover training data plus the test set at 500–12,000, leftover training data only at 24,000, and the test set only at 60,000.

n	Hessian	L1	L2	EA
500	$-$**0.75**	$+$0.11	$+$**0.76**	$-$**0.85**
2,000	$-$**0.82**	$+$0.08	$+$**0.73**	$-$**0.88**
6,000	$-$**0.78**	$+$0.40	$+$**0.78**	$-$**0.86**
12,000	$-$**0.84**	$+$0.22	$+$**0.77**	$-$**0.90**
24,000	$+$0.01	$+$0.06	$+$0.39	$-$0.40
60,000	$+$0.05	—	$+$0.05	$-$**0.81**

Hessian trace and L2 activation pattern count both predict generalisation at $|\rho| \approx 0.75$ across 500–12,000. At 24,000, the Hessian collapses ($\rho = +0.01$) while L2 remains significant ($\rho = +0.39$, $p = 0.005$). L1 saturates. EA is confounded by error count.

The key comparison is between the Hessian and L2 columns. Both predict. But the Hessian is not reparameterisation-invariant (Table 2). L2 is exactly invariant. Two quantities predict equally well. One is an artefact of the parameterisation. The other is a property of the function. Figure 2 shows the scatter plots at $n = 500$.

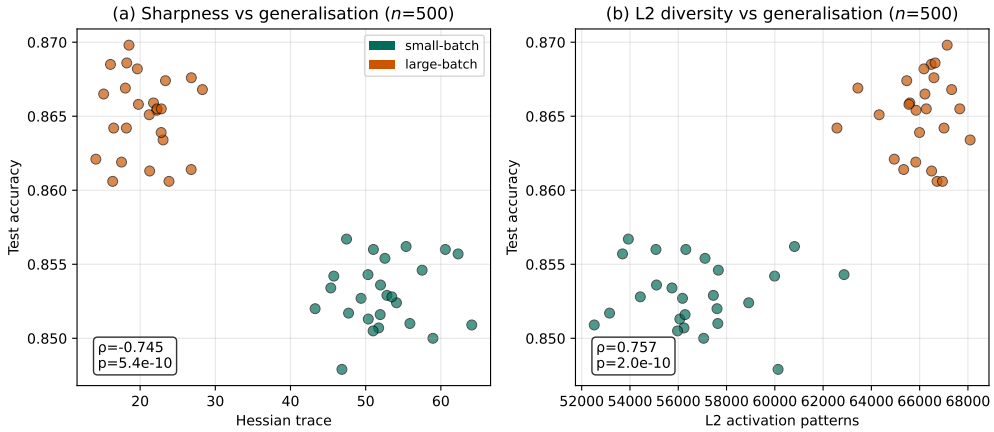

Figure 2. Scatter plots of test accuracy against (a) Hessian trace and (b) L2 activation pattern count at $n = 500$. Both correlate with generalisation at similar strength. Only L2 is reparameterisation-invariant.

 doi:10.20944/preprints202603.1737.v1

5.5. Connecting L2 Diversity to Weakness

L2 activation pattern diversity is not, on its face, a weakness measure. More patterns means more commitments at layer 2, not fewer. The connection to weakness runs through the training data and the formal construction in Appendix H.

Freeze layers 1 and 2. This fixes the L2 partition of input space. Within each L2 region, $h(x)$ is affine in x, so the output $W_3h(x) + b_3$ is also affine in x. Different inputs in the same region can in general receive different classifications. The region-class vocabulary in Appendix H approximates this by treating each region as assigning a single class. This approximation is exact when each region contains at most one data point. In the cross-regime experiments at smaller scales, L2 regions outnumber training points but not unseen points. At 60,000, L2 regions (approximately 9,500) are far fewer than training points. The approximation is therefore inexact in all experiments reported here. Define a *region-class vocabulary* (Definition A1) where programs are of the form "region r is classified as class c." The embodied language is the set of partial functions from regions to classes (Proposition A3).

An L2 region is *free* if it contains unseen data but no training data. In these regions, the network has committed to a classification, but the child task provides no constraint on what that classification should be. Theorem A2 proves that the weakness of the training-label policy is $w(\pi) = 11^k$ where k is the number of free regions. Under the region-class vocabulary (an approximation), every partial classification of free regions extends the training-label policy without contradiction. In the actual network, the shared weight matrix W_3 couples the classifications across regions, so the choices are not independent. See Section H.6 for details.

At 6,000 training points, the small-batch networks have ∼54,000 L2 regions and the large-batch networks have ∼63,000. With 6,000 points spread across this many regions, the vast majority of regions contain zero training points. The large-batch networks have ∼9,000 more L2 regions. Under the region-class approximation, more L2 regions means more free regions and therefore higher weakness (Corollary A1). The cross-regime L2 diversity correlation ($\rho = 0.73$–0.78, Table 3) is consistent with this prediction, though L2 pattern count and free-region count are not identical quantities.

There is a caveat (Section H.6). The region-class vocabulary treats each free region independently. In reality, the shared weight matrix W_3 creates correlations between regions. The vocabulary overstates the freedom of the network. Whether the ranking by free regions is preserved under the shared-weight constraint is an open question that the experiments test empirically.

5.6. Model Selection by Weakness

The cross-regime results in Tables 1–3 compare networks from different training regimes. A stronger test is model selection *within* a single regime. I train 300 networks with the same architecture (784→64→16→10), the same 250 training points, the same batch size, the same learning rate, and different random seeds. All memorise the training data. I measure six quantities per network. Three are not reparameterisation-invariant (Hessian trace, weight L2 norm, weight L1 norm). Three are invariant (L2 unseen patterns, free parameters (defined in Table 4 caption), unseen-only regions $= k_{\text{free}}$).

Unseen-only regions (k_{free}) is statistically significant ($\rho = +0.117$, $p = 0.043$). This is the quantity for which Theorem A2 proves the weakness is 11^k under the region-class vocabulary (an approximation to the actual network; see Section H.5). The sufficiency theorem predicts that weaker policies generalise better within a fixed vocabulary. The data confirms this prediction empirically, though comparing across networks involves different vocabularies (see Appendix H for discussion). L2 unseen patterns is also significant ($p = 0.050$). No sharpness or simplicity measure predicts generalisation in this setting.

The effect is small ($\rho \approx 0.12$). But the direction is correct, the quantity has a theorem, and sharpness has nothing.

 doi:10.20944/preprints202603.1737.v1

Table 4. Model selection within a single regime ($n = 300$ networks, same config, different seeds). Spearman ρ with test accuracy. Bold indicates statistically significant. The two significant predictors are both reparameterisation-invariant. No sharpness or simplicity measure predicts. $*$Free parameters $= \sum_r \max(0, D_2 \cdot 10 + 10 - n_r)$ where n_r is the number of training points in region r.

Measure	ρ	p	Invariant?
Hessian trace	-0.079	0.174	No
Weight L2 norm	-0.061	0.293	No
Weight L1 norm	$+0.011$	0.845	No
L2 unseen patterns	$+\mathbf{0.113}$	**0.050**	Yes
Free parameters*	$+0.067$	0.249	Yes
Unseen-only regions (k_{free})	$+\mathbf{0.117}$	**0.043**	Yes

5.7. Pair Proxy

A separate experiment tests the pair proxy directly. I train 100 networks (784→64→8→10, 250 training points, same batch size, different seeds). For each network, I freeze layers 1–2 and solve a linear feasibility problem for each of 100 unseen inputs $\times$ 10 classes. The linear program asks whether there exists a setting of layer-3 weights that correctly classifies all training data AND assigns the unseen input to the candidate class. The count of feasible pairs is the pair proxy.

Table 5. Pair proxy experiment ($n = 100$ networks, 784→64→8→10, 250 training points). Weakness outperforms sharpness. Simplicity does not predict.

Measure	ρ	p	Invariant?
Hessian trace	-0.226	0.024	No
Weight norm	$+0.019$	0.848	No
Pair proxy	$+\mathbf{0.374}$	$\mathbf{1.2 \times 10^{-4}}$	Yes

The pair proxy is highly significant ($\rho = +0.374$, $p = 0.00012$). It outperforms sharpness ($\rho = -0.226$, $p = 0.024$) . Simplicity does not predict at all ($p = 0.848$). The pair proxy is reparameterisation-invariant. The Hessian trace is not.

This is a head-to-head comparison on the same 100 networks. Same architecture, same data, same training, different random seeds. The answer is weakness.

6. Where μ-Weakness Sits

Generalisation theory has produced many complexity measures. Most of them answer the wrong question, or the right question at the wrong level.

VC dimension (Vapnik and Chervonenkis 1971) measures the expressiveness of a hypothesis *class*. Weakness measures the committedness of a single *hypothesis*. VC dimension cannot distinguish two hypotheses within the same class.

Algorithmic stability (Bousquet and Elisseeff 2002) measures sensitivity of the learning *algorithm* to perturbation of a single training point. Weakness is a property of the output, not the algorithm that produced it.

PAC-Bayes bounds (McAllester 1999) operate at the right level. Theorem 6 shows the KL term is a monotone function of weakness among correct policies. The difference is that weakness is defined in terms of the embodied language, which makes reparameterisation invariance obvious.

Minimum description length (Rissanen 1978) is a property of form, not function. Description length is encoding-dependent and can be gamed. Weakness cannot. If a persisting object happens to be simple, that is incidental unless it is also weak.

 doi:10.20944/preprints202603.1737.v1

Sharpness (Hochreiter and Schmidhuber 1997; Keskar et al. 2017) is a geometric property of weight space. It is not reparameterisation-invariant. The vanishing advantage (Section 5.2) shows it is not even a consistent correlate of generalisation.

AIXI (Hutter 2005) selects policies by minimising description length. Weakness maximisation is Bayes optimal under exchangeable task growth (Bennett 2024, 2025). AIXI's Occam rule is not.

Maximum entropy. Under a uniform prior over parent tasks, weakness maximisation is optimal (Proposition A1). A uniform prior over tasks is a maximum-entropy prior over which parent will appear. So weakness maximisation is the optimal response to a maximum-entropy assumption about tasks. But tasks and outputs are different things. A uniform prior over tasks implies a uniform distribution over outputs only if every combination of output constraints is satisfiable, which requires $L_{\mathfrak{v}} = 2^{\mathfrak{v}}$. This is the degenerate case where weakness collapses to a cardinality proxy and maximum entropy over outputs coincides with weakness maximisation (Bennett 2024). In the non-degenerate case ($L_{\mathfrak{v}} \neq 2^{\mathfrak{v}}$), the satisfiability constraint filters the uniform prior over tasks into a non-uniform distribution over achievable outputs. Maximum entropy over outputs ignores this geometry. Weakness maximisation respects it. They come apart whenever the vocabulary has non-trivial satisfiability structure, which is always in practice.

7. Discussion

Limitations. The pair proxy outperforms sharpness but explains a modest fraction of within-regime variance ($\rho = 0.374$). The region-class vocabulary is an approximation. The shared weight matrix W_3 couples regions (Section H.6), and formalising the exact shared-weight vocabulary is the most important open problem. Isolated networks did not reach exactly 100% training accuracy (see Table 1 caption). The experiments use MLPs on MNIST. The continuous weakness theory requires a reference measure μ not present in the finite case.

Future work. Formalising the shared-weight vocabulary and extending the sufficiency theorem to account for extension quality (not just count) are the main open problems. Repeating the experiments on CNNs, transformers, and larger datasets is the empirical priority.

Appendix A. Technical Assumption

Throughout the proofs below, fix a measurable vocabulary $(\mathfrak{v}, \mathcal{A}_L, \mu)$ with $0 < \mu(U) < \infty$. Assume $(U, \mathcal{A}_L|_U, \mu|_U)$ is an *atomless standard measure space*, meaning it is isomorphic (as a measure space) to a Lebesgue interval. This guarantees the existence of measure-preserving bijections between measurable subsets of equal measure, by Carathéodory's isomorphism theorem. The atomless requirement excludes pathological cases where atoms of equal measure but different demand probability would violate sufficiency. This assumption is satisfied by Lebesgue measure on $\mathbb{R}^d$, which covers all cases of practical interest for neural networks. The finite case, where counting measure has atoms, is handled separately by the finite proofs in Appendix B.

Appendix B. Finite Sufficiency and Necessity (Prior Results)

These results are from Bennett (2023) and Bennett (2025). I include the proofs for self-containment.

Proposition A1 (Finite sufficiency). *Let $\alpha \sqsubset \omega$ with the parent's additional demands drawn from the maximally uninformative extension model (uniform over $S \subseteq U$ where $U = L_{\mathfrak{v}} \setminus \mathrm{Ext}(I_\alpha)$), and let $\pi \in \Pi_\alpha$. Then $P(\pi \in \Pi_\omega) = 2^{|B_\pi|}/2^{|U|}$ where $B_\pi = \mathrm{Ext}(\pi) \cap U$.*

Proof. The parent adds demands by selecting $S \subseteq U$ uniformly from 2^U. The policy π survives if and only if $S \subseteq B_\pi$. There are $2^{|U|}$ equally likely subsets of U and $2^{|B_\pi|}$ of them lie inside B_π. For correct policies, $|B_\pi| = |\mathrm{Ext}(\pi)| - |O_\alpha|$ and $|O_\alpha|$ is fixed, so maximising $|\mathrm{Ext}(\pi)|$ is equivalent to maximising $|B_\pi|$ and therefore maximises the generalisation probability. □

Proposition A2 (Finite necessity). *If $|\mathrm{Ext}(\pi_1)| > |\mathrm{Ext}(\pi_2)|$, then $P(\pi_1 \in \Pi_\omega) > P(\pi_2 \in \Pi_\omega)$.*

 doi:10.20944/preprints202603.1737.v1

Proof. Immediate from Proposition A1. □

Theorem A1 (Finite exchangeable generalisation)**.** *Under any exchangeable prior over parent tasks,* $|\mathrm{Ext}(\pi_1)| \geq |\mathrm{Ext}(\pi_2)|$ *implies* $P(\pi_1 \in \Pi_\omega) \geq P(\pi_2 \in \Pi_\omega)$*. Under nondegeneracy, the inequality is strict when the extension sizes differ.*

Proof. Under exchangeability, the prior treats all completions symmetrically. A larger extension captures at least as many completions under any exchangeable measure. Nondegeneracy guarantees that the extra completions in the larger extension have positive probability. □

Appendix C. Proof of Theorem 1 (Sufficiency Under μ-Exchangeable Demands)

Proof. Let $\mu(B_{\pi_1}) \leq \mu(B_{\pi_2})$.

Step 1. Choose a measurable $B_2' \subseteq B_{\pi_2}$ with $\mu(B_2') = \mu(B_{\pi_1})$ (exists by the atomless standard measure space assumption, Appendix A). Build a measure-preserving bijection $\sigma_1 \colon B_{\pi_1} \to B_2'$ and a measure-preserving bijection $\sigma_2 \colon U \setminus B_{\pi_1} \to U \setminus B_2'$ (both pairs have equal measure). Define $\sigma = \sigma_1$ on B_{π_1} and $\sigma = \sigma_2$ on $U \setminus B_{\pi_1}$. Then $\sigma \colon U \to U$ is a measure-preserving bijection with $\sigma(B_{\pi_1}) = B_2' \subseteq B_{\pi_2}$.

Step 2. If $S \subseteq B_{\pi_1}$, then $\sigma(S) \subseteq \sigma(B_{\pi_1}) = B_2' \subseteq B_{\pi_2}$.

Step 3. By μ-exchangeability, $P(S \subseteq B_{\pi_1}) = P(S \subseteq \sigma(B_{\pi_1})) = P(S \subseteq B_2')$. Since $B_2' \subseteq B_{\pi_2}$, $P(S \subseteq B_2') \leq P(S \subseteq B_{\pi_2})$.

Step 4. Therefore $w_P(\pi_1) \leq w_P(\pi_2)$. Set $f(t) = w_P(\pi)$ for any π with $\mu(B_\pi) = t$. □

Appendix D. Proof of Theorem 2 (Strict Necessity)

Proof. Step 1. Since $\mu(B_{\pi_1}) > \mu(B_{\pi_2})$, choose $B_1' \subseteq B_{\pi_1}$ with $\mu(B_1') = \mu(B_{\pi_2})$. Let $D = B_{\pi_1} \setminus B_1'$, so $\mu(D) > 0$. By the two-piece construction in Theorem 1, any two buffers of equal measure give equal generalisation probability. So $P(S \subseteq B_{\pi_2}) = P(S \subseteq B_1')$.

Step 2. Since $B_1' \subsetneq B_{\pi_1}$,

$$P(S \subseteq B_{\pi_1}) = P(S \subseteq B_1') + P(S \subseteq B_{\pi_1},\, S \cap D \neq \emptyset).$$

Step 3. The second term is at least $P(\emptyset \neq S \subseteq D)$. By nondegeneracy and $\mu(D) > 0$, this is strictly positive. Therefore $P(S \subseteq B_{\pi_1}) > P(S \subseteq B_1') = P(S \subseteq B_{\pi_2})$, giving $w_P(\pi_1) > w_P(\pi_2)$. □

Appendix E. Proof of Theorem 3 (Unified Optimality)

Proof. Part (1) is Theorem 1. Part (2) is Theorem 2. For Part (3), suppose $\pi_1, \pi_2 \in \Pi_\alpha$ both maximise $G(\pi, P)$ over Π_α for every nondegenerate $P \in \mathcal{E}$. If $\mu(B_{\pi_1}) \neq \mu(B_{\pi_2})$, say $\mu(B_{\pi_1}) > \mu(B_{\pi_2})$, then Part (2) gives $G(\pi_1, P) > G(\pi_2, P)$ for every nondegenerate P, so π_2 does not maximise $G(\cdot, P)$. Contradiction. So $\mu(B_{\pi_1}) = \mu(B_{\pi_2})$. □

Appendix F. Proof of Theorem 4 (Minimax Optimality)

Proof. By Theorem 3(1), $G(\pi^*, P) \geq G(\pi, P)$ for all π and all $P \in \mathcal{E}$. So $\max_\pi G(\pi, P) = G(\pi^*, P)$ for every P, giving

$$\inf_{P \in \mathcal{E}} \max_\pi G(\pi, P) = \inf_{P \in \mathcal{E}} G(\pi^*, P).$$

Sufficiency (Theorem 3(1)) gives $G(\pi^*, P) \geq G(\pi, P)$ for all π and all $P \in \mathcal{E}$, so

$$\inf_{P \in \mathcal{E}} G(\pi^*, P) \geq \inf_{P \in \mathcal{E}} G(\pi, P) \quad \text{for all } \pi.$$

Therefore π^* achieves $\max_\pi \inf_P G(\pi, P)$. The minimax equality follows from the chain

$$\max_\pi \inf_P G(\pi, P) \geq \inf_P G(\pi^*, P) = \inf_P \max_\pi G(\pi, P) \geq \max_\pi \inf_P G(\pi, P).$$

 doi:10.20944/preprints202603.1737.v1

□

Appendix G. Reparameterisation Results

Below are representative reparameterisation invariance results. In every case, test accuracy, L1, L2, and EA are exactly invariant under both T_β and T_γ. Only the Hessian trace changes. Table 2 in the main text shows the full detail for Net 0 at 6,000 training points.

500 training points. Net 0 (small-batch, test=0.8536). Hessian at $\beta = 1$: 48.9. Hessian at $\beta = 20$: 4,751 (97×). Test accuracy: 0.8536 at all β. Net 25 (large-batch, test=0.8619). Hessian at $\beta = 1$: 22.4. Hessian at $\beta = 20$: 1,787 (80×). Test accuracy: 0.8619 at all β.

2,000 training points. Net 0 (small-batch, test=0.9050). Hessian at $\beta = 1$: 61.9. Hessian at $\beta = 20$: 5,789 (93×). Test accuracy: 0.9050 at all β. Net 25 (large-batch, test=0.9160). Hessian at $\beta = 1$: 24.5. Hessian at $\beta = 20$: 1,671 (68×). Test accuracy: 0.9160 at all β.

6,000 training points. See Table 2 in the main text for Net 0. Net 25 (large-batch, test=0.9476). Hessian at $\beta = 1$: 23.5. Hessian at $\beta = 20$: 1,328 (56×). Test accuracy: 0.9476 at all β. Net 30 (large-batch, test=0.9498). Hessian at $\beta = 1$: 22.6. Hessian at $\beta = 20$: 1,318 (58×). Test accuracy: 0.9498 at all β.

Appendix H. Weakness for Frozen-Partition ReLU Networks

This appendix constructs an approximate vocabulary for a ReLU network with frozen early layers and derives the weakness of the training-label policy under that vocabulary. The approximation is exact when each L2 region contains at most one data point and inexact otherwise (see Section H.5 for details).

Appendix H.1. Setup

Fix a 3-layer ReLU MLP $f_\theta(x) = W_3\sigma(W_2\sigma(W_1x + b_1) + b_2) + b_3$ where σ is the ReLU activation. Partition the parameters as $\theta = (\theta_{1:2}, \theta_3)$ where $\theta_{1:2} = (W_1, b_1, W_2, b_2)$ are the first two layers and $\theta_3 = (W_3, b_3)$ is the third.

Freeze $\theta_{1:2}$. This fixes the *L2 partition*. The function $h(x) = \sigma(W_2\sigma(W_1x + b_1) + b_2)$ maps each input x to a D_2-dimensional hidden representation. The binary activation pattern $\mathbf{1}[W_2\sigma(W_1x + b_1) + b_2 > 0]$ determines a *region* of input space within which h is affine in x. Within a single region, the output $W_3h(x) + b_3$ is also affine in x, so different inputs in the same region can in general receive different classifications. The region-class vocabulary defined below is therefore an *approximation* that treats each region as assigning a single class. This is exact when there is at most one data point per region and approximate otherwise. In all experiments reported here, multiple data points share L2 regions, so the approximation is inexact.

Let $\mathcal{R} = \{r_1, \ldots, r_R\}$ be the set of L2 regions that contain at least one data point (training or unseen).

Appendix H.2. The Vocabulary

Definition A1 (Region-class vocabulary)**.** *Define the environment $\Phi = \{0, \ldots, 9\}^{\mathcal{R}}$, the set of all total classification functions from regions to classes. For each region $r \in \mathcal{R}$ and class $c \in \{0, \ldots, 9\}$, define the program*

$$p_{r,c} = \{g \in \Phi : g(r) = c\}.$$

The vocabulary is $\mathfrak{v}_{\text{net}} = \{p_{r,c} : r \in \mathcal{R}, c \in \{0, \ldots, 9\}\}$.

Each program says "region r is classified as c." For a given region, the 10 programs $p_{r,0}, \ldots, p_{r,9}$ are mutually exclusive. No region can be assigned two classes. This is the satisfiability constraint that makes $L_{\mathfrak{v}_{\text{net}}} \neq 2^{\mathfrak{v}_{\text{net}}}$.

 doi:10.20944/preprints202603.1737.v1

Proposition A3 (Structure of the embodied language). *The embodied language $L_{\mathfrak{v}_{\text{net}}}$ is the set of all partial functions from $\mathcal{R}$ to $\{0,\ldots,9\}$, encoded as consistent subsets of $\mathfrak{v}_{\text{net}}$. Its cardinality is $|L_{\mathfrak{v}_{\text{net}}}| = 11^R$ (10 classes plus "unspecified" for each of R regions).*

Proof. A subset $l \subseteq \mathfrak{v}_{\text{net}}$ is consistent (i.e., $\bigcap_{p\in l} p \neq \emptyset$) if and only if no region is assigned two distinct classes. This is exactly the condition for l to encode a partial function from $\mathcal{R}$ to $\{0,\ldots,9\}$. Each of the R regions is either assigned to one of 10 classes or left unspecified, giving 11^R elements. This count includes the empty set $\emptyset \in L_{\mathfrak{v}_{\text{net}}}$ (no region classified). □

Appendix H.3. Policy and Weakness

Let $X_{\text{train}} = \{x_1,\ldots,x_n\}$ be the training inputs with labels $y_1,\ldots,y_n$. Let $\mathcal{R}_{\text{train}} \subseteq \mathcal{R}$ be the set of regions containing at least one training point. For each $r \in \mathcal{R}_{\text{train}}$, let y_r be the majority training label among points in r. When the region contains a single training point, y_r is that point's label. When it contains multiple training points with possibly different labels, the region-class vocabulary is an approximation. In the model-selection experiments (784→64→16→10, 250 training points), many regions contain multiple points, so the approximation is inexact. The pair proxy experiment (Section 5.7) avoids this issue by using per-input linear feasibility rather than the region-class vocabulary.

Definition A2 (Training-label policy). *The training-label policy is $\pi = \{p_{r,y_r} : r \in \mathcal{R}_{\text{train}}\}$. It assigns each training region to the label y_r and is silent on all other regions. When a region contains training points with conflicting labels, y_r is the majority label and the policy is an approximation of the network's actual behaviour in that region.*

Definition A3 (Free regions). *Let $\mathcal{R}_{\text{free}} = \mathcal{R} \setminus \mathcal{R}_{\text{train}}$ be the set of L2 regions containing unseen data but no training data. Write $k = |\mathcal{R}_{\text{free}}|$.*

These are the regions where the network has committed (it classifies inputs landing there) but the training data provides no constraint on what the classification should be. The network's behaviour in these regions is determined by the specific setting of θ_3, not by the child task.

Theorem A2 (Weakness of the training-label policy). *Under the vocabulary $\mathfrak{v}_{\text{net}}$, the weakness of the training-label policy π is*

$$w(\pi) = |\text{Ext}(\pi)| = 11^k,$$

where $k = |\mathcal{R}_{\text{free}}|$ is the number of free regions.

Proof. The extension $\text{Ext}(\pi) = \{l \in L_{\mathfrak{v}_{\text{net}}} : \pi \subseteq l\}$ consists of all partial classifications that agree with π on training regions. For each training region $r \in \mathcal{R}_{\text{train}}$, the class is fixed to y_r. For each free region $r \in \mathcal{R}_{\text{free}}$, there are 11 independent choices (assign to one of 10 classes, or leave unspecified). So $|\text{Ext}(\pi)| = 11^k$. □

Corollary A1 (Weakness ranking by free regions). *Consider two frozen-partition networks that both memorise the same training data. Let k_1 and k_2 be their respective numbers of free L2 regions. If $k_1 > k_2$, then $w(\pi_1) > w(\pi_2)$.*

Proof. $11^{k_1} > 11^{k_2}$ if and only if $k_1 > k_2$. □

The quantity $k = |\mathcal{R}_{\text{free}}|$ is the "unseen-only regions" measure from the experiments in Section 5. It is reparameterisation-invariant by Theorem 5, because the L2 activation patterns do not change under function-preserving reparameterisation T_β or T_γ.

 doi:10.20944/preprints202603.1737.v1

Appendix H.4. Connection to Generalisation

The sufficiency theorem (Theorem 1) and the finite sufficiency proposition (Proposition A1) predict that policies with higher weakness generalise better. Corollary A1 says that networks with more free regions have higher weakness. Combining these yields the prediction that networks with more free regions generalise better.

There is a subtlety. The sufficiency theorem compares policies *within the same vocabulary*. Two networks with different L2 partitions define different region-class vocabularies. In the strictest reading, the theorem does not directly compare policies across different vocabularies.

The experimental test in Section 5 bypasses this subtlety. It measures k_{free} for each network and correlates it with test accuracy across 300 networks. The result ($\rho = +0.117$, $p = 0.043$) confirms that the prediction holds empirically even across vocabularies. The pair proxy experiment in Section 5.7 provides a stronger test ($\rho = +0.374$, $p = 0.00012$) using linear feasibility rather than region counting.

Appendix H.5. Approximation Caveats

The single-class-per-region approximation. The region-class vocabulary assigns one class per region. In reality, $W_3 h(x) + b_3$ is affine in x within a region, so different inputs in the same region can be classified differently. The vocabulary would be exact if each region contained at most one data point. In all experiments reported here, multiple data points share regions, so the approximation is inexact. The pair proxy experiment (Section 5.7) avoids this issue entirely by checking per-input linear feasibility rather than assigning one class per region.

Appendix H.6. The Shared-Weight Caveat

Theorem A2 counts the extension in the region-class vocabulary $\mathfrak{v}_{\text{net}}$, where each free region's classification is an independent choice. But the actual network shares a single weight matrix W_3 across all regions. A setting of W_3 that correctly classifies training regions also determines the classification of free regions. The achievable total classifications of free regions are the cells of a hyperplane arrangement in the polytope of valid W_3 values. These cells are not independent across regions.

This means the region-class vocabulary overstates the freedom of the network. A "shared-weight vocabulary" (where programs encode constraints on W_3 rather than per-region classifications) would give a smaller extension.

Formalising a shared-weight vocabulary and computing its extension is the main open problem for the operationalisation of weakness in neural networks.

References

Bennett, Michael Timothy. 2023. The optimal choice of hypothesis is the weakest, not the shortest. In *16th International Conference on Artificial General Intelligence*, pp. 42–51. Springer.

Bennett, Michael Timothy. 2024. Is complexity an illusion? In *17th International Conference on Artificial General Intelligence*. Springer.

Bennett, Michael Timothy. 2025. *How To Build Conscious Machines*. Ph. D. thesis, The Australian National University. https://doi.org/10.25911/ta58-p428.

Bousquet, Olivier and André Elisseeff. 2002. Stability and generalization. *Journal of Machine Learning Research* 2, 499–526.

Dinh, Laurent, Razvan Pascanu, Samy Bengio, and Yoshua Bengio. 2017. Sharp minima can generalize for deep nets. In *International Conference on Machine Learning*, pp. 1019–1028. PMLR.

Foret, Pierre, Ariel Kleiner, Hossein Mobahi, and Behnam Neyshabur. 2021. Sharpness-aware minimization for efficiently improving generalization. In *International Conference on Learning Representations*.

Hochreiter, Sepp and Jürgen Schmidhuber. 1997. Flat minima. *Neural Computation* 9(1), 1–42.

Hutter, Marcus. 2005. *Universal Artificial Intelligence: Sequential Decisions Based on Algorithmic Probability*. Springer.

Keskar, Nitish Shirish, Dheevatsa Mudigere, Jorge Nocedal, Mikhail Smelyanskiy, and Ping Tak Peter Tang. 2017. On large-batch training for deep learning: Generalization gap and sharp minima. In *International Conference on Learning Representations*.

 doi:10.20944/preprints202603.1737.v1

Kwon, Jungmin, Jeongseop Kim, Hyunseo Park, and In Kwon Choi. 2021. ASAM: Adaptive sharpness-aware minimization for scale-invariant learning of deep neural networks. In *International Conference on Machine Learning*. PMLR.

McAllester, David A. 1999. Pac-Bayesian model averaging. In *Proceedings of the Twelfth Annual Conference on Computational Learning Theory*, pp. 164–170. ACM.

Petzka, Henning, Michael Kamp, Linara Adilova, Cristian Sminchisescu, and Mario Boley. 2021. Relative flatness and generalization. In *Advances in Neural Information Processing Systems*, Volume 34.

Rissanen, Jorma. 1978. Modeling by shortest data description. *Automatica 14*(5), 465–471.

Tsuzuku, Yusuke, Issei Sato, and Masashi Sugiyama. 2020. Normalized flat minima: Exploring scale invariant definition of flat minima with first order methods. In *International Conference on Machine Learning*. PMLR.

Vapnik, Vladimir N. and Alexey Ya. Chervonenkis. 1971. On the uniform convergence of relative frequencies of events to their probabilities. *Theory of Probability and its Applications 16*(2), 264–280.