Imperial College
London

IMPERIAL COLLEGE LONDON

DEPARTMENT OF COMPUTING

Predicting Grokking Onset from Intrinsic Training Metrics

Author:
Felix Baastad Berg

Supervisor:
Tolga Birdal

Submitted in partial fulfillment of the requirements for the MSc degree in Type of Degree of Imperial College London

Abstract

Grokking is a training phenomenon in which neural networks overfit for a prolonged period before abruptly transitioning to strong generalization. This project studies whether the onset of grokking can be anticipated *without monitoring test error*, by tracking intrinsic properties of a model's internal representations during training. Using a modular addition benchmark with a two-hidden-layer MLP, we measure several intrinsic signals—including correlation dimension, PCA participation ratio/effective rank, persistent-homology–derived quantities, and minimum spanning tree (MST)–based fractal dimension—across layers and throughout optimization. We quantify how these signals relate to delayed generalization via time-resolved lead–lag correlation analysis, and assess robustness across hyperparameter sweeps. We find that MST-based dimension is the most practically informative signal in our setting: the first hidden-layer MST dimension correlates strongly with the generalization gap at substantial positive lead times (on the order of 1000 epochs). Building on these observations, we introduce a simple parametric onset predictor that fits a logistic transition model to intrinsic trajectories and calibrates an offset to estimate grokking time from intrinsic dynamics alone. Across runs for the second hidden-layer MST dimension, predicted onset times align closely with observed grokking times ($R^2 \approx 0.98$; RMSE ≈ 1637 epochs), supporting the view that grokking corresponds to a measurable internal phase transition that can be detected before test accuracy reveals it.

Acknowledgments

I thank Tolga Birdal for proposing a project closely aligned with my research interests and for supporting me in pursuing it. I am grateful for his guidance throughout the work, including providing resources and access to the codebase, and for helpful discussions.

Contents

1 Introduction

1.1 Research Motivation

Grokking is the phenomenon where a neural network suddenly transitions from memorizing its training data to perfectly generalizing long after the point of overfitting [12]. In other words, training accuracy reaches 100%, yet the validation accuracy remains at chance level for an extended period before abruptly jumping to near-perfect generalization performance. This striking delay in generalization defies the expectations of classical learning theory. Understanding when and why a network will grok is of both theoretical and practical interest. If we could predict the onset of grokking without directly observing test error, it would shed light on the internal mechanisms enabling generalization and could inform training protocols (e.g. early stopping or regularization scheduling) for tasks where extended training eventually yields improved generalization.

Several hypotheses have been proposed to explain grokking's delayed generalization. One view is that heavy regularization (particularly weight decay) biases the network toward simpler, more generalizable solutions that are harder to find [12, 9]. In this view, the network might be slowly working toward the general solution under the hood, even though the outward transition in error is sudden [2]. Another perspective situates grokking as a transition between two regimes of training dynamics: a "lazy" regime near initialization and a "rich" regime where the network's weights move significantly to capture structure [5, 10]. Empirical and theoretical work supports that grokking corresponds to the network escaping the lazy regime and undergoing a sharp change in its internal representations [11].

Our study is motivated by recent evidence that grokking is accompanied by qualitative changes in model complexity and representations. For example, Power et al. (2022) originally observed grokking in small algorithmic tasks (like modular arithmetic) under strong weight decay [12]. Subsequent analyses found that when grokking occurs, the network's representations become simpler: the intrinsic dimension of the last-layer activations drops concurrently with the jump in test accuracy. This suggests that the network's final representation manifold compresses when it discovers a generalizable pattern. Similarly, DeMoss et al. (2025) showed that properly regularized networks exhibit a sharp complexity phase transition at grokking: a custom complexity measure (based on Kolmogorov complexity via lossy compression) rises during memorization and then falls as the network discovers a simpler underlying solution [6]. Taken together, these findings hint that certain intrinsic metrics – e.g. measures of dimensionality or complexity of internal states – might serve as "order parameters" for the grokking transition. If so, monitoring these metrics during training could allow us to predict the grokking time (the onset of generalization) without observing the test error.

1.2 Objectives and Contributions

This project investigates whether intrinsic signals derived from a model's internal representations can provide indicators of grokking in a controlled setting. The objectives are to:

- reproduce grokking on a modular arithmetic prediction task;
- track intrinsic metrics across layers and training epochs;
- analyze whether these signals exhibit systematic changes during the generalization transition;
- evaluate robustness across runs and hyperparameter settings.

To address these objectives, we conduct a systematic empirical study of intrinsic signals during training in a grokking-prone setting. We implement and benchmark a broad suite of intrinsic-dimension and complexity measures drawn from the literature—including correlation dimension, minimum spanning tree (MST) fractal dimension, PCA-based effective dimension/participation ratio, persistent-homology–derived dimensionality estimates, and an algorithmic complexity proxy—and track them across layers and epochs for a modular arithmetic prediction task.

Beyond collecting these metrics, we contribute two complementary analysis approaches. First, we study how intrinsic signals relate to generalization using correlation-based analyses that explicitly account for temporal structure, evaluating multiple correlation formulations and a range of lead–lag offsets to test whether intrinsic trajectories provide strong associations with generalization gap during grokking. Second, we propose a grok-time prediction method that fits a sigmoid transition model to intrinsic trajectories and uses the fitted parameters to estimate the onset time directly from intrinsic dynamics. We quantify uncertainty using bootstrap confidence intervals and assess robustness via augmented results across hyperparameter settings. Together, these experiments characterize which intrinsic measures are most informative, how early they provide predictive signal, and how stable these conclusions are across training conditions.

1.3 Research Approach and Outlines

We begin by reviewing the relevant background on grokking and its interpretation as a phase transition. We then introduce the concept of intrinsic and fractal dimensions in neural networks, laying out the metrics we will employ. Next, we describe our methodology – the experimental setup and how each metric is computed and tracked. We then present our results, examining the dynamics of the most important metrics and their relationship to test error, including visualizations of their trajectories. Finally, we discuss the implications of our findings.

2 Grokking Background

2.1 Delayed Generalization

Grokking was first reported by Power et al. (2022) in small-scale algorithmic learning tasks and refers to a delayed transition from a memorising solution to a rule-based solution that generalises [12]. It is most easily observed when the training set covers only a subset of the full input–output space, so that memorisation and rule learning are both viable ways to achieve low training loss. Training performance saturates early, while validation/test per- formance remains near chance for many additional epochs before rapidly improving.

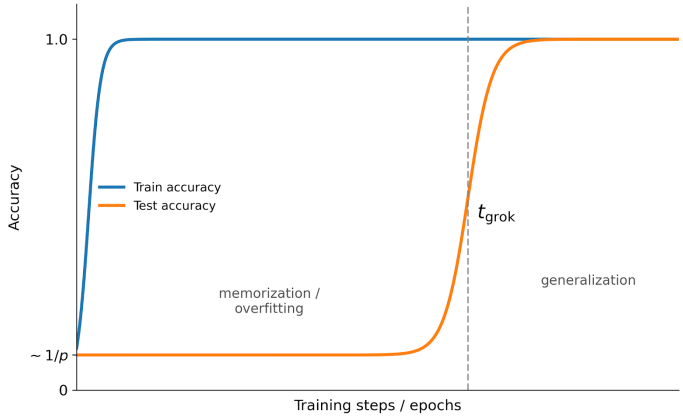


Figure 2.1: Schematic illustration of grokking dynamics: training performance saturates early while validation/test performance remains near chance before abruptly improving at the grokking onset time t_{grok} (not to scale).

This chapter reviews the canonical grokking setup and the main explanatory perspectives proposed in the literature. These perspectives motivate our focus on intrinsic signals—quantities computed from internal representations during training—as potential early indicators of the generalization transition.

2.1.1 Canonical Grokking Setup in Algorithmic Tasks

A prototypical grokking benchmark is modular arithmetic, such as learning addition modulo a prime p. Inputs consist of pairs (a, b), and the target is defined by

$$y = (a + b) \bmod p, \quad a, b \in \{0, \ldots, p - 1\}. \tag{2.1}$$

The dataset is constructed so that only a subset of all possible input pairs is included in training, while evaluation is performed on the held-out pairs. This split creates two qualitatively different solution strategies: a memorization solution that interpolates the training set, and an algorithmic solution that captures the underlying

modular rule and extrapolates to unseen combinations.

When training a neural network on such tasks, grokking manifests as a characteristic two-stage learning curve. First, the model quickly achieves near-perfect training accuracy while test accuracy remains near chance, indicating that the learned function does not generalize. Second, after a prolonged plateau, test accuracy rises sharply to near-perfect levels while training accuracy remains high. The key feature is the delay between the epoch at which the training objective is essentially solved and the epoch at which generalization emerges; we refer to this delayed transition as the grokking onset and formalize it later when defining our prediction targets.

In this report, we refer to the time at which validation/test performance rapidly improves as the grokking onset, denoted t_{grok} in Figure 2.1. A precise operational definition used for measurement is given later in the methodology section.

2.1.2 Regularization and Training Conditions

A prominent observation in early work is the apparent role of regularization, particularly weight decay. Power et al. report that grokking occurs reliably under sufficiently strong weight decay, whereas in the absence of weight decay the same architectures often remain in a memorizing solution for the duration of training [12]. One interpretation is that weight decay introduces a simplicity bias: among functions that fit the training set, optimization is gradually nudged toward solutions with smaller norm and greater regularity, which may correspond to the underlying algorithmic rule. Under this view, the extended plateau reflects slow movement in parameter space toward a flatter region of the loss landscape until the model transitions into the basin of a generalizing solution.

At the same time, subsequent studies and variants of the experimental setup suggest that weight decay is not the only route to grokking-like delayed generalization. In our experiments, inspired by recent work [13], we observe grokking behaviour even without explicit weight decay under certain training conditions; we return to this point when describing the benchmark and hyperparameter settings used in this report.

2.2 Explanatory Perspectives and Implications for Intrinsic Signals

Several lines of work interpret grokking as a qualitative transition in training dynamics rather than a smooth continuation of ordinary generalization. Although these perspectives differ in emphasis, they share a common implication for our goal: grokking is associated with a reorganisation of internal representations and/or parameter structure.

2.2.1 Lazy-to-Rich Dynamics and Feature Learning

One prominent account frames grokking as a transition from an initial *lazy* regime to a later *feature-learning* (or *rich*) regime. In the lazy regime, the learned function remains close to its linearisation around initialization, resembling kernel regression; later in training, parameter updates become large enough that the network begins to learn task-relevant features [5, 10]. Kumar et al. (2023) formalise grokking as the point at which this shift in implicit bias occurs, showing that a sharp change in training dynamics can induce delayed generalisation [10]. From the perspective of our study, the key takeaway is that such a regime change should be reflected in internal geometry: if representations transition from "memorisation-like" to "rule-like", measures of effective dimension or complexity of activations may exhibit a corresponding shift.

2.2.2 Delayed Compression and Representation Geometry

A complementary viewpoint emphasises two-stage representation dynamics: an early fitting phase followed by a later compression or organisation phase. In modular arithmetic settings, Sakamoto & Sato (2025) report a delayed compression of representations analogous to an information-bottleneck-like effect, and connect grokking to the emergence of neural-collapse-like geometry via contraction of within-class variance [14]. This line of work provides a direct motivation for intrinsic dimension metrics: if late-stage generalisation coincides with representational compression and increased organisation, then intrinsic dimensionality estimates of activation manifolds may decrease around the transition—and potentially begin changing before test accuracy improves.

2.2.3 Statistical-Physics Analogies and Order Parameters

Statistical physics analogies treat grokking as a phase transition between a "disordered" memorising phase and an "ordered" rule-encoding phase [7]. This framing highlights metastability (long plateaus) and abrupt transitions, and suggests the existence of an *order parameter*—a measurable quantity that changes sharply when the transition occurs. While test accuracy is an obvious order parameter, our interest is in internal order parameters that can be monitored during training and that may provide advance warning.

2.3 Comparison of explanations

Although the literature proposes different mechanisms for grokking, the accounts share two broad points of agreement. First, grokking is typically characterised by a prolonged period in which training loss/accuracy saturates while test performance remains poor [12]. Second, multiple papers report that the transition to generalisation coincides with a qualitative reorganisation of internal structure (representations

and/or parameter geometry), motivating intrinsic signals as candidate internal "order parameters" [10, 14, 6].

At the same time, there are important differences that affect what should be measured and how general any predictor can be. Power et al. report that sufficiently strong weight decay reliably produces grokking in modular arithmetic tasks and interpret this as an optimisation bias toward simpler solutions [12]. In contrast, subsequent work reports grokking-like transitions under other training conditions, suggesting that weight decay is not the only route to delayed generalisation [13]. This matters for our benchmark because an early-warning signal should ideally be robust across regimes; a signal that appears only under weight decay may reflect norm control rather than a general mechanism of representation change.

Statistical-physics accounts emphasise metastability and an abrupt transition between a memorising phase and a rule-encoding phase, implying internal order parameters that change sharply near the transition [7]. By contrast, lazy-to-rich/feature-learning accounts describe a shift from near-linearised dynamics around initialization to substantial feature learning later in training [5, 10]. This distinction affects what "early warning" should look like: a phase-transition view suggests rapid changes near t_{grok}, whereas a feature-learning view suggests gradual drift that may precede t_{grok}.

Representation-geometry accounts report late-stage organisation of activations (including neural-collapse-like effects), implying that activation-manifold metrics such as intrinsic dimension, MST/PH0 summaries, and variance-based effective rank should change as grokking approaches [14, 10]. Other perspectives emphasise shifts in implicit bias and parameter structure, motivating complementary weight-spectrum measures (effective rank, spectral entropy) [12].

Taken together, these agreements and differences imply that (i) multiple internal signals are plausible candidates, (ii) their usefulness may depend on training regime, and (iii) the key empirical question is not only whether signals change *around* grokking but whether they change *before* test accuracy does.

However, existing studies typically consider a limited set of metrics and do not systematically test whether changes in these signals precede improvements in validation/test performance. In this project, we implement a suite of intrinsic dimension and complexity measures within a single pipeline and benchmark them on the same grokking-prone task, explicitly analysing temporal relationships (lead–lag structure) to assess actionable early warning of grokking onset. The next chapter introduces the intrinsic metrics used in our benchmark and discusses their theoretical and practical properties.

3 Intrinsic Metrics

Chapter 2 reviewed perspectives that interpret grokking as a qualitative transition in training dynamics and representation geometry. This motivates tracking *intrinsic signals*—quantities computed from activations or weights during training—as candidate early-warning indicators of grokking onset. Prior work supports this framing: modular-arithmetic studies report late-stage representational compression [14], and intrinsic-dimension analyses suggest that late-layer activation manifolds can become effectively lower-dimensional around the grokking transition [10]. Rather than proposing a single metric, we benchmark a suite of intrinsic measures and test whether any provide predictive signal, in particular a measurable lead time before the grokking onset t_{grok}.

3.1 What We Measure: Activations and Weights

We compute intrinsic signals from two complementary sources: (i) *activation manifolds*, i.e. point clouds formed by layer activations over a fixed input set; and (ii) *weight structure*, i.e. layer weight matrices and their spectra. Activation-based metrics probe how inputs are represented internally, whereas weight-based metrics capture global properties of the learned parameters. Because grokking is often described as a transition in representation quality, we primarily focus on activation-derived signals and analyse them layer-by-layer.

3.2 Activation-Manifold Intrinsic Dimension

Intrinsic dimension (ID) quantifies the effective degrees of freedom of a point cloud or manifold embedded in a high-dimensional ambient space. In neural networks, activations often lie on low-dimensional structures despite high ambient dimension, and this has been linked to generalisation [4, 3]. In grokking settings, a transition from memorisation to rule learning is hypothesised to coincide with representational simplification; consequently, activation-manifold IDs are natural candidates for internal order parameters.

3.2.1 Related work and estimator considerations

Intrinsic-dimension estimators have been used to study representation complexity in deep networks and its relationship to generalisation [4, 3]. In grokking-like settings, prior studies report that late-layer representations can become effectively lower-dimensional around the transition, consistent with representational simplification [10]. However, absolute ID values are known to depend on estimator choice, sample size, distance concentration effects in high dimensions, and preprocessing

such as centering and normalisation [3]. For this reason, we track multiple complementary estimators (fractal, neighbour-based, and linear-variance-based) and focus primarily on within-run temporal trends and cross-run consistency rather than any single absolute ID value.

3.2.2 Correlation Dimension

We measure correlation dimension (CorrDim), a fractal dimension estimator based on how the number of close point pairs scales with radius. Fractal dimensions generalise the notion of dimension to non-integer values and can capture nonlinear manifold structure that is not well-described by linear subspaces [3]. In our context, CorrDim provides a scale-sensitive estimate of how "space-filling" the activation manifold is at different stages of training.

3.2.3 Two-Nearest-Neighbors Intrinsic Dimension (TwoNN)

To complement CorrDim, we also consider a nearest-neighbour based estimator (TwoNN). TwoNN estimates intrinsic dimension from the ratio of distances to the first and second nearest neighbours and is often used as a lightweight ID estimator for high-dimensional point clouds. Including multiple ID estimators reduces the risk that conclusions are artefacts of a single estimator's assumptions.

3.2.4 PCA Participation Ratio / Effective Dimension

Not all notions of dimension are fractal. We also track the PCA participation ratio (sometimes called an effective rank), defined from the eigenvalues of the activation covariance. This captures how many linear directions carry substantial variance and is commonly used as a simple proxy for representational compression. While PCA-based measures can miss nonlinear structure, they are computationally stable and provide a useful baseline for comparison [3].

3.3 Graph-Based Representations

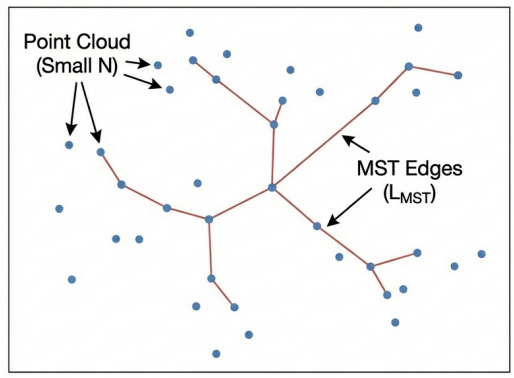
While intrinsic dimension estimators summarise manifold complexity via scaling or variance, graph- and topology-based measures provide a complementary view by explicitly encoding the geometry of the activation point cloud. This family is particularly relevant to our goal because it yields signals that are interpretable as changes in global connectivity structure, and in our experiments MST-based measures proved especially descriptive of grokking dynamics.

3.3.1 Related work and motivation for MST/PH0 summaries

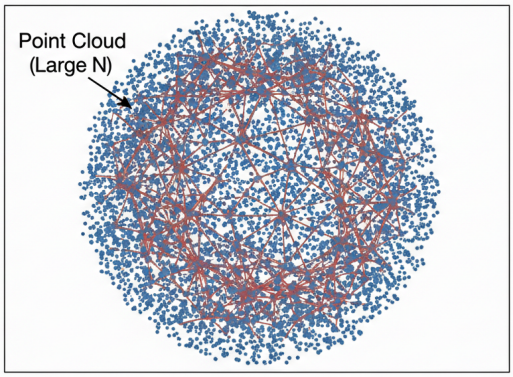
Graph- and topology-based summaries provide a complementary view of representation geometry by capturing global connectivity structure rather than only variance or local scaling. MST-based statistics in particular are closely connected to intrinsic-dimension estimation via scaling laws for Euclidean functionals [3]. Persistent homology extends this idea by tracking how connectivity changes across distance scales; we focus on PH0 because it is tightly linked to MST structure and yields stable, interpretable summaries (e.g. merge scales and long-edge structure) that can be monitored throughout training [3].

3.3.2 Minimum Spanning Tree (MST) Dimension

Given a set of activation vectors, the minimum spanning tree (MST) connects all points with minimal total edge length. For random samples from a d-dimensional space, the total MST length (and related functionals) scales with sample size in a way that depends on d, which motivates MST-based estimators of fractal/intrinsic dimension [3]. Intuitively, if representations occupy a higher-dimensional region, the MST must span space more extensively; if representations collapse onto a simpler structure, the MST geometry and derived dimension estimates change accordingly. We compute MST-based dimension estimates (including the MST exponent / parameterisation used in our implementation). These quantities provide a tractable summary of activation geometry and, importantly for our analysis, produce smooth trajectories over training that can be examined for lead–lag structure relative to t_{grok}.



(a) MST edges on a smaller activation subset.



(b) MST edges on a larger activation subset.

Figure 3.1: Illustration of MST construction on an activation point cloud. As sample size increases, the MST captures geometry at finer scales, which motivates MST-based scaling approaches to intrinsic-dimension estimation.

3.3.3 Persistent Homology (PH0) Features

Persistent homology (PH) tracks the birth and death of topological features across a filtration of the point cloud. For order-0 persistent homology (PH0), the relevant

events are merges of connected components as the connectivity radius increases. PH0 is closely related to MST structure, since the MST encodes the critical edges at which components merge [3].

To test whether the predictive utility of MST signals generalises to a broader topological summary, we track PH0-derived features including total persistence, persistence entropy, the maximum edge, and counts of long edges (e.g. above a percentile threshold). These features provide interpretable markers of connectivity structure and potential "re-organisation" in representations.

3.3.4 Energy Functionals E_α and Edge-Magnitude Summaries

Beyond dimension-like summaries, we consider additional graph-based functionals derived from MST/PH0 structure. In particular, we track E_α energies for multiple values of α, as well as edge-magnitude summaries (e.g. Euclidean magnitude variants) computed from row-normalised activations. These provide a controlled way to test whether grokking signatures arise from a specific dimensional estimator or from a broader change in geometric scaling.

3.4 Weight-Spectrum Measures

Because grokking is sometimes associated with shifts in implicit bias and optimisation regime, we also track weight-spectrum measures that summarise parameter structure. For each weight matrix, we compute: (i) an effective rank (a participation-ratio-like quantity derived from singular values) and (ii) spectral entropy, which measures how concentrated or diffuse the singular-value distribution is. These measures provide a complementary lens on complexity and can help distinguish representation-driven effects from purely parameter-norm effects.

3.5 Summary of Tracked Signals

Overall, our benchmark includes (i) activation-manifold intrinsic dimension estimates (CorrDim, TwoNN, PCA participation ratio), (ii) graph/topology-based geometry signals (MST-based measures, PH0 features, E_α energies), and (iii) weight-spectrum measures (effective rank, spectral entropy). Our goal is to evaluate which signals change systematically around grokking and, crucially, whether any provide advance warning of grokking onset.

The following methodology chapter describes how these quantities are computed consistently across runs and layers, and how we evaluate predictive utility via lead–lag correlation analyses and parametric grok-time estimation with uncertainty quantification.

4 Methodology

4.1 Experimental Setup and Definitions

This chapter describes the experimental protocol used to benchmark intrinsic training signals and evaluate their relationship to delayed generalisation. We train neural networks on a canonical grokking task (modular addition), log internal states throughout optimisation, compute intrinsic metrics defined in Chapter 3, and evaluate two analysis tracks: (i) time-resolved lead–lag association between intrinsic signals and generalisation error, and (ii) a parametric onset predictor that estimates grokking time from intrinsic dynamics alone.

4.1.1 Task and Model

We adopt a grokking-capable modular arithmetic benchmark used in prior work [12, 10, 13]. The learning problem is modular addition modulo a fixed modulus $p = 113$:

$$y \;=\; (a+b) \bmod p, \qquad a, b \in \{0, 1, \ldots, p-1\}. \tag{4.1}$$

The full dataset contains all ordered input pairs (a, b), yielding p^2 examples. We construct a training set by sampling a fraction f of these pairs. Each integer a and b is represented by a one-hot vector such that the network input is

$$x \;=\; [\mathrm{onehot}(a),\, \mathrm{onehot}(b)] \in \{0,1\}^{2p}, \tag{4.2}$$

and the target is a p-class label $y \in \{0, \ldots, p-1\}$. We train using standard cross-entropy loss with a two-layer multilayer perceptron (MLP) with 200 and ReLU activations. Optimisation is performed with Adam for $80,000$ epochs.

4.1.2 Run grid and training protocol

We run a grid of training configurations over three hyperparameters:

$$\begin{aligned} \text{learning rate } \eta &\in \{0.005,\ 0.01,\ 0.015,\ 0.02\}, \\ \text{weight decay } \lambda &\in \{0,\ 0.03,\ 0.003,\ 0.0003\}, \\ \text{training fraction } f &\in \{0.39,\ 0.40,\ 0.42,\ 0.45\}. \end{aligned}$$

This yields $4 \times 4 \times 4 \;=\; 64$ runs in total. Each run is trained for a fixed number of epochs T (chosen to extend beyond the delayed generalisation regime when grokking occurs - $T = 80,000$ in our case).

At regular checkpoints during training (every 1000 epochs), we record: (i) scalar training/held-out metrics (loss and accuracy), and (ii) internal states needed to compute intrinsic signals, including layer activations and weight matrices. These are the metrics described in 3. This protocol yields time series $m(t)$ for each metric, layer, and run, which are then analysed using the procedures described later in this section.

4.2 Correlation Measures

For each run, let $m(t)$ denote a scalar intrinsic metric (e.g. MST dimension) evaluated at training epoch t, and let $A(t)$ denote held-out accuracy. We also define held-out error $E(t) = 1 - A(t)$.

A central question is whether intrinsic dynamics lead or follow generalisation dynamics. We therefore compute lead–lag association curves between $m(t)$ and $E(t)$. The association at lead Δ is then

$$r(\Delta) \;=\; \mathrm{assoc}(\{m(t)\}_{t\in\mathcal{T}_\Delta},\; \{E(t+\Delta)\}_{t\in\mathcal{T}_\Delta})\,. \tag{4.3}$$

A positive lead Δ means the intrinsic signal is evaluated *earlier* than the generalisation target; thus a strong association at $\Delta > 0$ indicates potential predictive value. The association measueres we will use is pearson r_P, spearman ρ_S and kendall τ, which all are sensetive to different kinds of relationships.

In addition, we will vary hyperparameters in a granulated analysis for robustness of our results [8, 1]. Using multiple measures helps distinguish relationships that depend on scale (Pearson) from those that are monotonic but non-linear (Spearman/Kendall).

4.2.1 Aggregation and Uncertainty Across runs

All lead–lag curves $r(\Delta)$ are first computed per run r_i. For each metric (and optionally layer), we compute the mean association curve

$$\bar{r}(\Delta) \;=\; \frac{1}{N}\sum_{i=1}^{N} r_i(\Delta),$$

where $r_i(\Delta)$ is the run-level curve and N is the number of included runs. We also report the distribution of $\Delta_i^\star$ across runs as a measure of lead-time stability.

To quantify uncertainty in across-run summaries, we use bootstrap resampling over runs. Specifically, we draw B bootstrap samples of the N runs (sampling with replacement), recompute $\bar{r}^{(b)}(\Delta)$ and $\Delta^{\star(b)}$, and report pointwise confidence bands for $\bar{r}(\Delta)$ (e.g. 2.5–97.5 percentiles) as well as an interval summary for $\Delta^\star$.

4.3 Parametric Onset Prediction

In addition to lead-lag association curves, we introduce a parametric approach that converts intrinsic trajectories into *event-time* features and uses these to predict grokking onset from intrinsic dynamics alone. We define the target (the onset time of grokking) as T_{grok} as the epoch where the test accuracy becomes 0.5. The goal of this section is to predict T_{grok} with intrinsic measures.

4.3.1 Event-time features from intrinsic trajectories

Given an intrinsic metric time series $m(t)$ (for a fixed metric and layer), we fit the logistic form

$$\hat{m}(t) \;=\; \ell_m \;+\; \frac{h_m - \ell_m}{1 + \exp\big(-k_m\,(t - t_{0,m})\big)}, \tag{4.4}$$

and normalizes it between 0 and 1 to the function $p(t)$. For any $q \in (0,1)$, the timing marker t_q is the epoch at which the fitted intrinsic trajectory reaches fraction q of its transition:

$$t_q \;:=\; \{t:\; p(t) = q\} \;=\; t_{0,m} + \frac{1}{k_m}\ln\!\left(\frac{q}{1-q}\right), \qquad 0 < q < 1. \tag{4.5}$$

Small q values correspond to earlier parts of the intrinsic transition, while $q \approx 0.5$ corresponds to the intrinsic midpoint.

4.3.2 Offset calibration and onset prediction

Across all runs, we compute the offset between intrinsic timing and grokking onset:

$$\Delta t_q^{(i)} \;=\; t_q^{(i)} - T_{\text{grok}}^{(i)}. \tag{4.6}$$

We select a quantile $q^\star$ by sweeping a grid of candidate values and choosing the one that yields the most stable offset across runs:

$$q^\star \;=\; \arg\min_{q \in \mathcal{Q}} \operatorname{std}(\Delta t_q)\,, \tag{4.7}$$

where $\mathcal{Q}$ is a fixed grid ($\{0.05, 0.10, \ldots, 0.95\}$ for us). Given $q^\star$, we define the calibrated offset μ_{q^*} as the mean of Δt_{q^*}, and define the onset predictor for a new run as

$$\hat{T}_{\text{grok}} \;=\; t_{q^\star} - \mu_{q^\star}. \tag{4.8}$$

This predictor deliberately has low capacity (one timing feature q^*), reducing overfitting risk in the small-N regime. Instead of overfitting a large model, we use priors of the shape of the grokking phenomenon to make a model that is less prone to overfitting.

4.3.3 Cross-validations and hyperparameters

We evaluate predictive performance using leave-one-out cross-validation (LOOCV). For each held-out run i, we predict

$$\hat{T}_{\text{grok}}^{(i)} \;=\; t_{q^\star}^{(i)} - \mu_{q^\star}^{(-i)}, \quad \mu_{q^\star}^{(-i)} \;=\; \operatorname{mean}\Big(\Delta t_{q^\star}^{(j)}\Big)_{j \neq i}. \tag{4.9}$$

The per-run prediction error is $e^{(i)} \;=\; \hat{T}_{\text{grok}}^{(i)} - T_{\text{grok}}^{(i)}$. We summarize errors using RMSE, and the 95th percentile of $|e|$. We also report an empirical 95% prediction interval using the 2.5th and 97.5th percentiles of the LOOCV error distribution. We do the similar analysis on granules over different hyperparameters to check robustness.

5 Experimental Results

This chapter evaluates whether intrinsic training metrics provide (i) interpretable signatures of grokking, (ii) measurable lead-time relative to the generalization transition, and (iii) accurate onset-time prediction across runs. We first summarize the training regimes observed in the modular addition benchmark, then benchmark intrinsic signals via lead–lag association analysis, and finally present onset prediction results based on parametric timing markers extracted from intrinsic trajectories.

5.1 Training Regimes in Modular Addition

5.1.1 Regimes Induced by Training Fraction

Training fraction controls the constraint density of the learning problem: smaller fractions admit many functions that fit the training set, increasing the chance that optimization initially settles into a memorizing solution. In contrast, larger fractions more strongly constrain the target function and tend to induce earlier generalization. In initial sweeps where we varied training fraction, we consistently observed three qualitative regimes: (i) *early generalization*, where held-out accuracy rises quickly; (ii) *grokking*, where held-out accuracy remains near chance for a prolonged period before a sharp transition; and (iii) *persistent overfitting*, where the model fits training data but does not recover held-out performance within the training horizon.

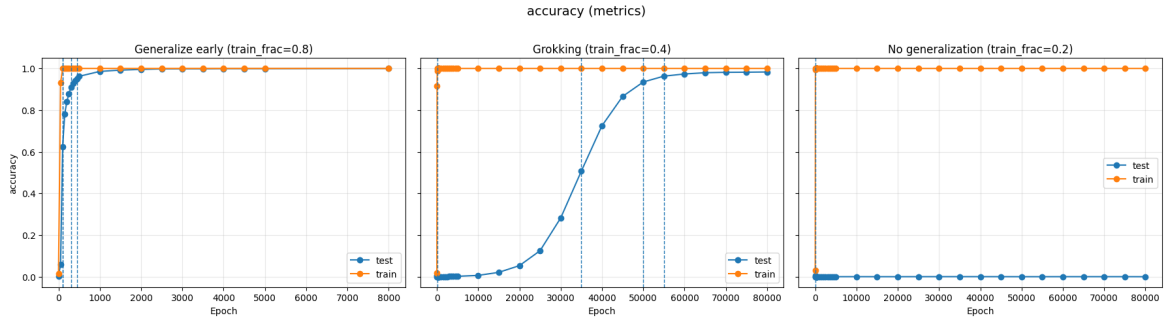


Figure 5.1: Accuracy trajectories illustrating early generalization, grokking, and persistent overfitting. Vertical markers indicate the grokking onset time t_{grok} for grokking runs (when defined).

Figure 5.1 shows the canonical grokking signature: training accuracy saturates early, while held-out accuracy remains near chance for many epochs before abruptly rising to near-perfect performance.

In Figure 8.1, the corresponding loss is plotted. Loss trajectories provide a complementary view of these regimes. In grokking and persistent overfitting, the model typically enters a phase where held-out loss increases (or remains high) even after training loss becomes small, reflecting the generalization gap. In grokking runs, this

behavior is transient: the loss eventually decreases again around the onset, whereas in persistent overfitting the loss fails to recover.

Among all tracked intrinsic quantities, MST-based measures exhibited one of the most visually distinct and structured behavior around grokking. Figure 5.2 shows the evolution of MST alpha across layers. A notable feature is layer dependence: different layers exhibit different degrees of transition-localized structure, suggesting that grokking corresponds to a reorganization that is not uniform throughout the network.

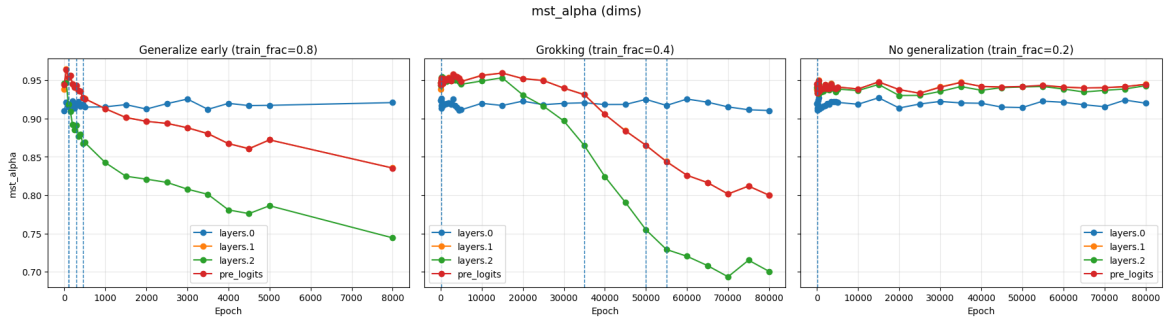


Figure 5.2: MST alpha across layers in representative grokking runs. MST-based signals change markedly around the transition, with layer-specific structure that motivates layerwise benchmarking.

Informally, MST alpha often begins changing shortly before the held-out accuracy jump, consistent with the hypothesis that internal geometry reorganizes prior to externally visible generalization. The next sections quantify this intuition.

The remainder of this chapter focuses on runs that exhibit grokking-like delayed generalization. Runs that overfit of generelizes immediately are being left out.

5.2 Lead–Lag Association Benchmark

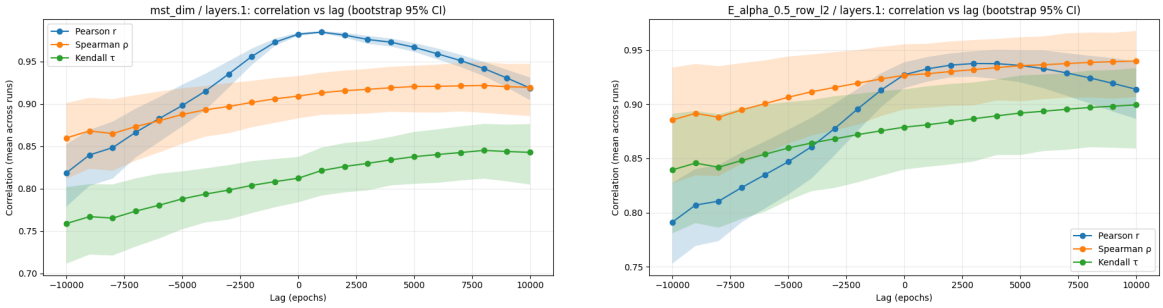
We now benchmark intrinsic metrics by measuring time-resolved lead–lag association with generalization dynamics. The aim is to answer: *which intrinsic signal–layer combinations track the generalization gap most strongly, and at what lead times?*

5.2.1 Lag Convention and Target Signal

Throughout this section, the target is the generalization gap $\mathrm{gen\_gap}(t)$, computed per epoch t (as defined in the methodology chapter). For each intrinsic time series $m(t)$ and integer lag Δ, we compute association between $m(t)$ and $\mathrm{gen\_gap}(t + \Delta)$. Under this convention, $\Delta > 0$ indicates that the intrinsic metric *leads* the generalization gap (early-warning), while $\Delta < 0$ indicates that the intrinsic metric *lags* behind it.

5.2.2 Top-Ranked Metric–Layer Combinations

To identify candidate signals, we compute lead–lag association curves per run and summarize by the *per-run mean* correlation across time. Table 8.1 reports the top-performing metric–layer combinations for Pearson, Spearman, and Kendall criteria. MST Dimension measured at hidden layer 1 scored the best for pearson, while E_α with $\alpha = 0.5$ for layer 1 scored the highest spearman and kendall correlation.



(a) **(b)**

Figure 5.3: Lead–lag correlations (mean $\pm$ bootstrap 95% CI) for two top metric–layer pairs. Panel (a) shows `mst_dim` (layer 1), which maximizes Pearson r and closely tracks generalization-gap magnitude. Panel (b) shows an E_α metric, which scores highly under rank correlations (Spearman/Kendall), indicating monotonic co-trending but weaker magnitude alignment.

Although Table 8.1 ranks candidates separately under Pearson, Spearman, and Kendall criteria, these criteria measure different notions of "tracking". Rank correlations are sensitive primarily to monotonic co-movement: they are high when a metric changes in the same direction as the generalization gap. Pearson correlation is stricter: it rewards both correct direction and *magnitude/shape* alignment. This distinction is visible in Figure 5.3: the Pearson-optimal signal (`mst_dim` in layer 1) exhibits a sharp, high Pearson peak and visually matches the gap trajectory, while the rank-optimal signal maintains strong ρ/τ but shows weaker Pearson alignment, consistent with tracking only the ordering of changes rather than their scale. Because our downstream goal is onset prediction from intrinsic trajectories, we report all three criteria but prioritize Pearson when selecting signals intended to *numerically predict* gap magnitude or event timing.

Figure 5.4 illustrates a representative alignment of normalized MST dimension with the generalization gap at the best-performing positive lead. The near-linear tracking highlights why MST-based measures appear as dominant candidates in the ranking: they provide high signal-to-noise, smooth trajectories, and consistent alignment across the plateau and transition.

For comparison, Figure 8.2 shows an E_α energy summary plotted against the generalization gap. It focuses more on monotonicity of the intrinsic metric and not necessary how closely it aligns with the generalization gap. For that reason, we will focus more on the pearson correlation for the rest of this section.

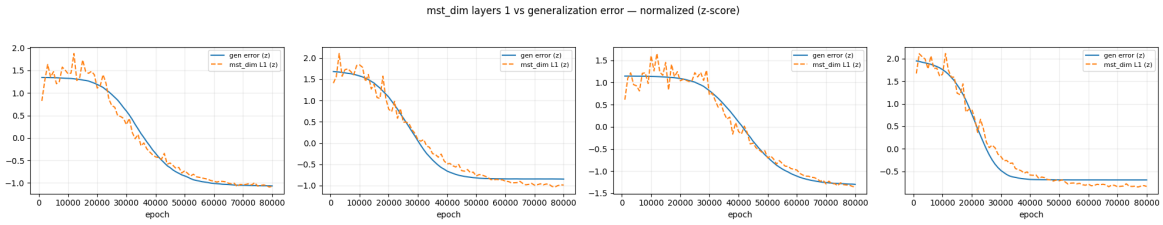
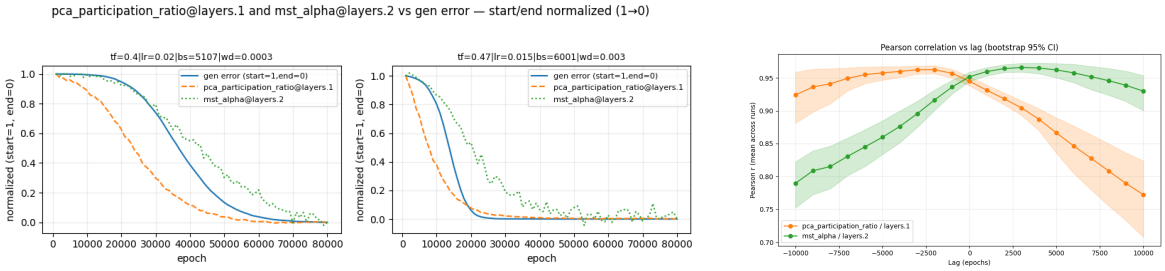


Figure 5.4: Representative alignment of normalized `mst_dim` (layer 1) with the generalization gap under the best-performing lead.

5.2.3 Which Metric Dominates at Which Lead?

Different intrinsic signals peak at different lead ranges. Some metrics are best at long positive leads, while others peak closer to $\Delta \approx 0$ (useful for event localization). Table 8.3 summarizes the best-performing (metric, layer) winner as a function of lag, collapsing contiguous lag ranges where the same winner dominates.



(a) **(b)**

Figure 5.5: Early-warning vs post-transition intrinsic signals. (a) Normalized trajectories comparing generalization error with PCA participation ratio (layer 1) and MST alpha (layer 2). (b) Correlation versus lag (mean $\pm$ bootstrap 95% CI), showing PCA leads the transition (negative lag peak) while MST alpha lags (positive lag peak).

Figure 5.5 provides representative lead–lag curves for two winner metrics from Table 8.3, illustrating how different metrics dominate at different lag ranges. PCA participation ratio at layer 1 has the highest pearson correlation 10,000 epochs before the generalisation error, while MST alpha at the second layer has the highest correlation 10,000 epochs afer.

A plausible interpretation is that these two metrics probe different stages of the same phenomenon. PCA participation ratio captures a gradual reduction in *linear* degrees of freedom that can begin during the memorisation plateau, consistent with the idea that internal structure starts forming before test accuracy reveals it. MST alpha, by construction, is sensitive to global connectivity and scaling properties of the activation geometry. It can change most strongly when the representation manifold has already reorganized into a more coherent structure. This separation motivates reporting both types of signals.

5.2.4 Heterogeneity Across Hyperparameters

Because grokking time and intrinsic dynamics depend on optimization settings, we analyze heterogeneity across training fraction, learning rate, and weight decay. This serves two purposes: (i) it checks whether a signal remains informative across conditions, and (ii) it motivates conditioned evaluation for onset prediction.

Figure 8.3 visualizes run-level alignment of MST dimension with generalization gap at a fixed lead, with point style indicating hyperparameters. The scatter reveals systematic structure: some hyperparameter combinations yield smoother transitions and tighter alignment, while others yield more spiky or delayed transitions where different metrics may dominate.

Figure 5.6 further stratifies runs by varying one hyperparameter at a time. The plot shows that while some dimensions are strong for certain grokking behaviour, MST Dimension layer 1, always show high and consistent corrlation. This solidifies our belief that MST can be used as a trusworthy correlator.

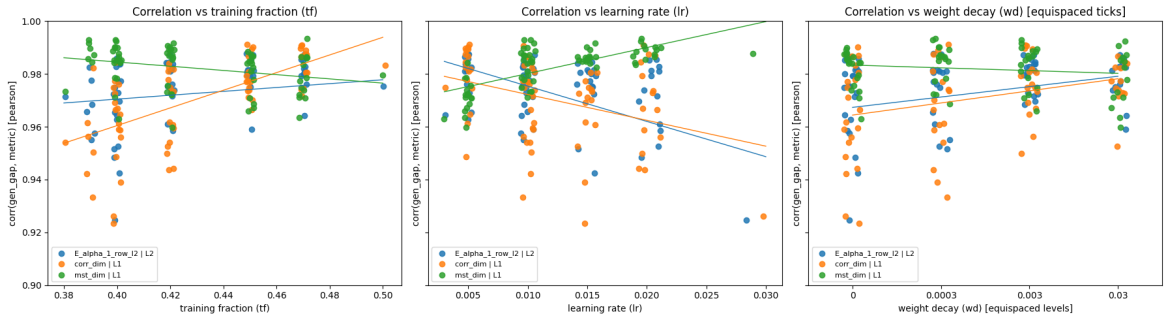


Figure 5.6: Run-level scatter diagnostics stratified by training fraction, learning rate, and weight decay.

5.3 Intrinsic Onset Prediction

We now evaluate whether intrinsic dynamics can predict the grokking onset time. We use the logistic timing marker method introduced in Section 4.3, which extracts an intrinsic event time t_q from a fitted sigmoid and maps it to a predicted onset time $\hat{T}_{\text{grok}}$ via an offset calibration.

5.3.1 Stable Timing Markers from Logistic Fits

Table 8.2 summarizes timing offsets $\Delta t = t_q - T_{\text{grok}}$ for candidate (metric, layer) combinations, after selecting q to minimize the across-run spread. Low $\text{std}(\Delta t)$ indicates that the intrinsic timing marker is consistently aligned to the onset across runs, which is precisely what is needed for a stable onset predictor.

Across candidates, MST-derived signals yield some of the tightest timing distributions, and their best-performing layer depends on whether the goal is (i) long lead-time early warning or (ii) accurate onset localization.

5.3.2 Predicting T_{grok} Across Runs

We next evaluate prediction quality by comparing predicted onset time $\hat{T}_{\text{grok}}$ to the observed onset time T_{grok}. Figure 5.7 shows predicted onset times across the hyperparameter grid, and Figure 8.4 shows predicted versus observed onset times.

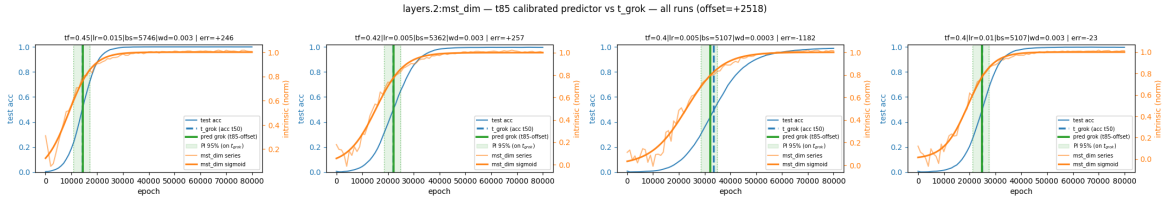


Figure 5.7: Predicted grokking onset times with uncertainty intervals based on sigmoid curve fits of MST-derived intrinsic dynamics with optimal $q = 0.85$.

Across runs, the relationship is strongly linear, with $R^2 \approx 0.98$ and RMSE ≈ 1637 epochs for the best-performing MST-based predictor in our setting. This supports the interpretation of grokking as a detectable internal transition whose timing is encoded in intrinsic trajectories.

5.3.3 Prediction Intervals and Conditioning on Hyperparameters

Finally, Table 5.1 reports empirical 95% prediction intervals for the onset prediction error $e = \hat{T}_{\text{grok}} - T_{\text{grok}}$, both pooled across runs and under conditioned evaluation that controls for hyperparameter confounds.

Table 5.1: 95% prediction intervals (PI) for onset prediction error $e = \hat{T}_{\text{grok}} - T_{\text{grok}}$. "All runs" denotes the pooled estimate across runs, while "conditioned" denotes stratified evaluation where all but one hyperparameter are held fixed and results are aggregated across granules.

Condition	n_{runs}	n_{granules}	$\text{PI}_{2.5\%}$	$\text{PI}_{97.5\%}$
All runs (pooled)	–	–	-2744.23	3582.21
Conditioned on varying train_fraction	65	15	-2101.35	2148.45
Conditioned on varying lr	61	15	-1244.20	1184.58
Conditioned on varying weight_decay	67	17	-1483.15	1218.03

Conditioning consistently tightens prediction intervals relative to the pooled estimate, indicating that a substantial component of apparent prediction uncertainty is attributable to systematic hyperparameter-dependent shifts rather than irreducible noise. Among the stratified settings, varying `train_fraction` yields the widest interval, while varying `lr` yields the tightest in our experiments.

6 Conclusion

This project asked whether grokking onset can be predicted from *intrinsic* training signals rather than held-out performance. On a modular addition benchmark, we benchmarked a suite of activation- and weight-based geometry/complexity measures using lead–lag correlation analysis. Across grokking runs, MST-derived metrics were the most consistently informative, showing strong layer-dependent alignment with the generalization gap and yielding usable positive-lead signal. Using a simple logistic timing-marker model fit to intrinsic trajectories, we could accurately estimate T_{grok} across runs, with tighter uncertainty once hyperparameter confounds were controlled. Overall, the results support the interpretation of grokking as an internal geometric reorganization that is detectable before test accuracy improves, while leaving open how broadly these signals transfer beyond small algorithmic tasks.

7 Declaration & Sustainability

Declaration

I declare that this report is my own work and has not been submitted in substantially the same form for the award of a higher degree elsewhere. Where material has been used from other sources, it has been properly acknowledged through citations and references.

Some illustrative figures in this report were generated using Google Gemini. These figures are used for visualization and communication purposes only; the experimental design, methodology, implementation, and analysis presented in this report were developed by me.

ChatGPT was used as a writing aid throughout the project, primarily to improve grammar, clarity, and overall language delivery. The scientific content, structure, and claims in the report remain my own responsibility.

In addition, parts of the experimental code were written with the assistance of AI-based autocompletion in the Cursor editor. As a result, a substantial portion of the codebase contains AI-suggested code fragments. I reviewed, adapted, and validated this code for correctness and consistency with the experimental goals, and I take responsibility for the final code and results reported here.

Sustainability

This project is computational in nature and therefore has an environmental footprint primarily due to energy use during model training and metric computation. To reduce unnecessary computation, we used a small, controlled benchmark (modular arithmetic with a compact MLP), performed structured hyperparameter sweeps with a limited grid size, and logged intrinsic metrics at discrete intervals rather than every training step. Expensive procedures (e.g., compression-based complexity estimation where applicable) were run at reduced checkpoint frequency.

A longer-term motivation of this work is also efficiency: if grokking onset can be anticipated from intrinsic signals, one could terminate unpromising runs earlier, schedule regularization adaptively, or allocate compute more effectively—reducing trial-and-error training and potentially lowering overall energy usage in future experimentation.

Bibliography

[1] Rayna Andreeva et al. “Topological Generalization Bounds for Discrete-Time Stochastic Optimization Algorithms”. In: *Advances in Neural Information Processing Systems (NeurIPS)*. arXiv:2407.08723. 2024 (cit. on p. 12).

[2] Boaz Barak et al. “Hidden progress in deep learning: SGD learns parities near the computational threshold”. In: *Advances in Neural Information Processing Systems* 35 (2022) (cit. on p. 1).

[3] Tolga Birdal et al. “Intrinsic Dimension, Persistent Homology and Generalization in Neural Networks”. In: *Advances in Neural Information Processing Systems (NeurIPS)*. Vol. 34. 2021, pp. 6776–6789 (cit. on pp. 7–10).

[4] Bradley CA Brown et al. “Relating Regularization and Generalization through the Intrinsic Dimension of Activations”. In: *NeurIPS 2022 Workshop on Has it Trained Yet?* 2022. URL: https://openreview.net/forum?id=3ajyK7Mvl7 (cit. on p. 7).

[5] Lenaic Chizat, Edouard Oyallon, and Francis Bach. “On lazy training in differentiable programming”. In: *Advances in Neural Information Processing Systems* 32 (2019) (cit. on pp. 1, 5, 6).

[6] Branton DeMoss et al. “The complexity dynamics of grokking”. In: *Physica D: Nonlinear Phenomena* 482 (2025), p. 134859. ISSN: 0167-2789. DOI: 10.1016/j.physd.2025.134859 (cit. on pp. 1, 6).

[7] Yizhong He and Haiping Huang. “Grokking as a First Order Phase Transition in Two Layer Networks”. In: *arXiv preprint arXiv:2310.03789* (2023). URL: https://arxiv.org/abs/2310.03789 (cit. on pp. 5, 6).

[8] Yiding Jiang et al. “Fantastic Generalization Measures and Where to Find Them”. In: *arXiv preprint arXiv:1912.02178* (2020). arXiv: 1912.02178 [cs.LG] (cit. on p. 12).

[9] Alon Kuznets-Etzioni and Nimrod Libman. “Weight Decay is All You Need to Grok”. In: *arXiv preprint arXiv:2301.05179* (2023) (cit. on p. 1).

[10] Ziming Liu, Eric J Michaud, and Max Tegmark. “Omnigrok: Grokking is ubiquitous in deep learning”. In: *arXiv preprint arXiv:2210.01117* (2022) (cit. on pp. 1, 5–7, 11).

[11] William Merrill, Nikolaos Tsilivis, Amanand Pantazis, et al. “The Logic of Grokking: Emergence of Structure in Transformers”. In: *Advances in Neural Information Processing Systems (NeurIPS)*. 2023. URL: https://arxiv.org/abs/2310.05740 (cit. on p. 1).

[12] Alethea Power et al. “Grokking: Generalization beyond overfitting on small algorithmic datasets”. In: *arXiv preprint arXiv:2201.02177* (2022) (cit. on pp. 1, 3–6, 11).

[13] Lucas Prieto et al. “Grokking at the Edge of Numerical Stability”. In: *The Thirteenth International Conference on Learning Representations (ICLR)*. 2025. URL: https://openreview.net/forum?id=TvfkSyHZRA (cit. on pp. 4, 6, 11).

[14] Keitaro Sakamoto and Issei Sato. “Explaining Grokking and Information Bottleneck through Neural Collapse Emergence”. In: *arXiv preprint arXiv:2509.20829* (2025). URL: https://arxiv.org/abs/2509.20829 (cit. on pp. 5–7).

8 Appendix

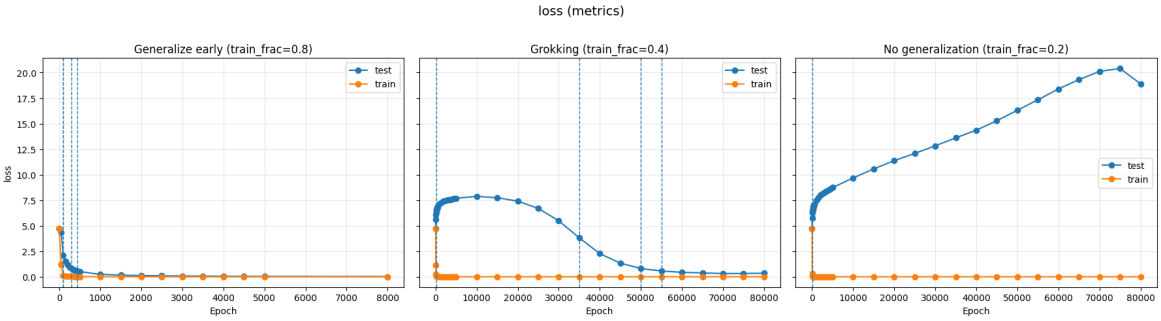


Figure 8.1: Loss trajectories for early generalization, grokking, and persistent overfitting. Grokking runs exhibit an extended plateau/high-loss phase followed by a late decrease, whereas persistent overfitting does not.

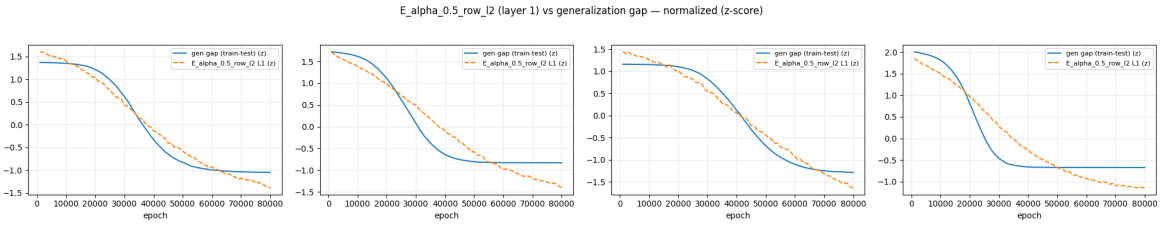


Figure 8.2: Representative alignment of E_α (here $E_{\alpha=0.5}$) with the generalization gap.

Table 8.1: Top-10 dimension–generalization-gap correlations ranked by *per-run mean* correlation. Lag is defined by correlating $\mathrm{gen\_gap}(t)$ with $\dim(t + \mathrm{lag})$.

Pearson (mean across runs: `pearson_r_run_mean`)				
Rank	ID	Layer	Lag	Mean
1	mst_dim	layers.1	1000	0.984571
2	corr_dim	layers.1	-1000	0.975415
3	E_alpha_1_row_l2	all_layers_concat	-1000	0.974372
4	ph0_total_persistence_row_l2	all_layers_concat	-1000	0.974372
5	ph0_total_persistence_row_l2	layers.2	-1000	0.974363
6	E_alpha_1_row_l2	layers.2	-1000	0.974363
7	E_alpha_2_row_l2	layers.1	1000	0.973317
8	E_alpha_0.5_row_l2	all_layers_concat	0	0.973248
9	E_alpha_0.5_row_l2	layers.2	0	0.973209
10	weight_effective_rank	layers.1.weight	0	0.972185

Spearman (mean across runs: `spearman_r_run_mean`)				
Rank	ID	Layer	Lag	Mean
1	E_alpha_0.5_row_l2	layers.1	10000	0.939930
2	E_alpha_1_row_l2	layers.1	10000	0.939190
3	ph0_total_persistence_row_l2	layers.1	10000	0.939190
4	ph0_persistence_entropy_row_l2	layers.1	8000	0.936715
5	E_alpha_2_row_l2	layers.1	10000	0.935393
6	mag_euclidean_s1_row_l2	layers.0	10000	0.925730
7	pmag_euclidean_s1_row_l2	layers.0	10000	0.924895
8	ph0_persistence_entropy_row_l2	layers.2	5000	0.922715
9	mst_dim	layers.1	8000	0.921752
10	mst_alpha	layers.1	8000	0.921752

Kendall (mean across runs: `kendall_tau_run_mean`)				
Rank	ID	Layer	Lag	Mean
1	E_alpha_0.5_row_l2	layers.1	10000	0.899467
2	E_alpha_1_row_l2	layers.1	10000	0.897918
3	ph0_total_persistence_row_l2	layers.1	10000	0.897918
4	E_alpha_2_row_l2	layers.1	10000	0.892162
5	ph0_persistence_entropy_row_l2	layers.1	8000	0.876967
6	mag_euclidean_s1_row_l2	layers.0	10000	0.867416
7	pmag_euclidean_s1_row_l2	layers.0	10000	0.865341
8	ph0_persistence_entropy_row_l2	all_layers_concat	6000	0.859152
9	ph0_persistence_entropy_row_l2	layers.2	5000	0.858851
10	weight_spectral_entropy	layers.1.weight	-3000	0.855467

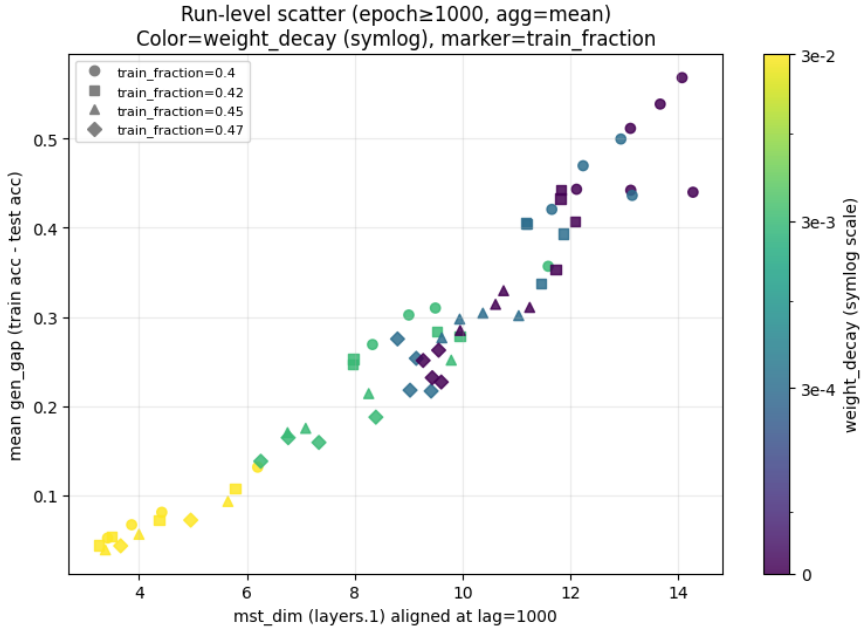


Figure 8.3: Run-level alignment of `mst_dim` with the generalization gap at a fixed lead, with point styling indicating training fraction and weight decay.

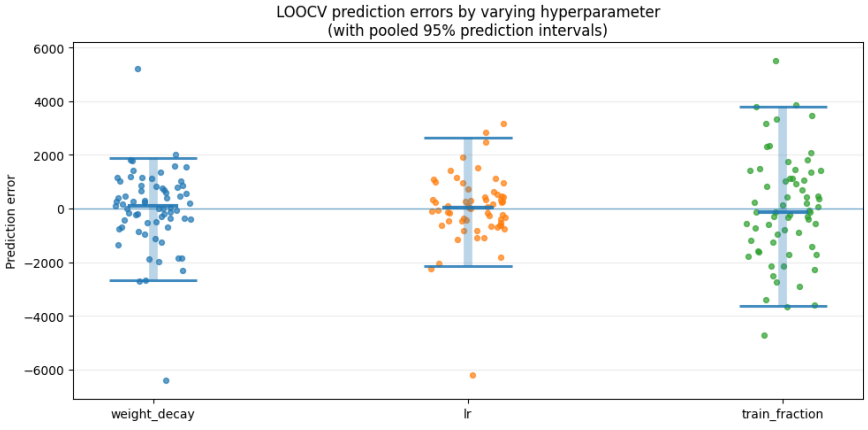


Figure 8.4: Predicted onset time $\hat{T}_{\text{grok}}$ versus observed onset time T_{grok} across runs.

Layer	Metric	q	mean Δt	std Δt
layers.2	mst_dim	0.85	2517.768	1638.614
layers.1	mst_dim	0.50	1847.864	1866.279
layers.2	mst_alpha	0.35	-132.378	1985.494
all_layers_concat	mst_alpha	0.35	287.961	2027.331
layers.1	corr_dim	0.75	51.785	2742.804
layers.2	E_alpha_1_row_l2	0.75	2684.687	3315.142
layers.2	ph0_total_persistence_row_l2	0.75	2684.687	3315.142
layers.2	E_alpha_0.5_row_l2	0.65	2059.686	3422.284
all_layers_concat	E_alpha_1_row_l2	0.85	-3255.878	3987.835
all_layers_concat	E_alpha_0.5_row_l2	0.70	-8215.278	5933.536
layers.1	E_alpha_2_row_l2	0.50	-13751.653	10468.012

Table 8.2: Summary statistics (sorted by std Δt).

Lag range	Best (metric, layer)	$\bar{r}$ (mean Pearson)
-10000	`pca_participation_ratio`, `layers.1`	0.924
-9000 to -5000	`E_alpha_2_row_l2`, `layers.2`	0.936–0.960
-4000 to -3000	`mst_dim`, `layers.2`	0.963–0.968
-2000 to -1000	`corr_dim`, `layers.1`	0.974–0.975
0 to 6000	`mst_dim`, `layers.1`	0.959–0.985
7000 to 10000	`mst_alpha`, `layers.2`	0.930–0.952

Table 8.3: Best-performing (metric, layer) as a function of lag (winner is the combo with maximum per-run mean Pearson correlation at each lag). Contiguous lags with the same winner are collapsed into ranges; $\bar{r}$ shows the observed range within each segment.