2 Papers Summary: SGD

Paper 1: Stochastic Gradient Algorithm

The central idea of the paper is to view stochastic gradient descent (SGD) not as a sequence of discrete updates, but as the discretization of an underlying continuous-time stochastic system. By moving to this viewpoint, the authors gain access to tools from stochastic calculus and optimal control, which can be used to understand algorithm behavior and to design principled hyperparameter policies.

From Discrete Steps to Dynamics: Instead of thinking of a discrete update $\rightarrow$ tune by heuristic, the SME framework promotes continuous dynamics $\rightarrow$ analyze motion $\rightarrow$ compute good controls.

The resulting stochastic differential equation (SDE) does not reproduce every sample path of SGD, but it accurately captures its distributional behavior and dominant trends.

What the Continuous View Reveals: The approximation highlights a fundamental feature of training: two competing forces.

- A deterministic drift that pushes parameters downhill.
- A stochastic forcing term coming from gradient noise.

These produce two natural regimes:

1. **Descent phase:** far from the optimum, signal dominates $\rightarrow$ aggressive movement is beneficial.
2. **Fluctuation phase:** near the optimum, noise dominates $\rightarrow$ large steps mainly increase variance.

The SDE makes this tradeoff transparent and even allows estimation of when the transition should occur.

Once SGD is written as a dynamical system, selecting learning rates or momentum becomes a question from control theory:

What choice of time-varying parameters minimizes the expected objective at the end of training?

Solving this produces feedback laws rather than fixed schedules.

In words, the prescriptions typically become:

- large steps when far away,
- smaller steps near minima,
- automatic adaptation to curvature and gradient variability.

Resulting Algorithms: Turning these ideas into practice leads to adaptive procedures such as **cSGD** and **cMSGD**. They estimate local curvature and noise levels online using exponential moving averages, requiring little additional computation and no Hessian information.

Empirical Message: Across standard benchmarks, the proposed methods achieve performance comparable to optimizers such as Adam or Adagrad, while showing substantially reduced sensitivity to initial hyperparameter choices.

Conceptual Takeaway: The broader contribution is methodological. Hyperparameter schedules need not be guessed or prescribed in advance; they can be derived from an approximate understanding of the dynamics governing the optimization process.

Paper 2: 3 Factors

Big-Picture Idea: This paper studies how the noise inherent in stochastic gradient descent (SGD) influences *which* minima the algorithm ultimately finds. By approximating SGD with a stochastic differential equation and analyzing its equilibrium behavior, the authors argue that the geometry of the final solution is largely governed by a single effective quantity: the ratio of learning rate to batch size.

The three factors control the trade-off between the depth and width of the minima found by SGD (with wider minima favoured by a higher ratio of learning rate to batch size):

- learning rate
- batch size
- variance of the loss gradients

Noise as the Governing Quantity: The key modeling step is to treat SGD as a noisy dynamical system whose long-run behavior resembles sampling from a Boltzmann-type distribution. Under simplifying assumptions (most importantly isotropic gradient noise), the stationary distribution depends on the loss scaled by a "temperature" proportional to learning rate / batch size.

Thus, rescale learning rate and batch size together → keep noise constant → expect similar endpoints.

What Determines Which Minimum Wins: The theory predicts that the probability of landing near a particular minimum depends on two features:

- its depth (value of the loss), and
- its width (captured by curvature / Hessian determinant).

Higher noise increases the relative importance of width.

In other words, more noise → flatter minima become more likely, less noise → deeper but sharper minima become competitive.

Practical Interpretation: Batch size alone is not the decisive factor. Instead, it is the ratio η/S that controls how strongly SGD prefers flat regions.

This provides a theoretical explanation for empirical observations such as:

- why increasing batch size can hurt generalization,
- why scaling the learning rate with batch size can recover performance,
- and why noisier training often resists memorization.

Empirical Message: Across multiple architectures and datasets, the experiments show:

- increasing η/S correlates with flatter solutions,
- flatter solutions correlate with better validation accuracy,
- and training behavior is often similar when learning rate and batch size are scaled together.

They also demonstrate limits: if the learning rate becomes too large, the continuous-time approximation breaks and the invariance no longer holds.

Conceptual Takeaway: SGD is not merely descending the loss; it is navigating a noise-shaped landscape. The level of noise, largely controlled by the ratio of learning rate to batch size, determines whether optimization prioritizes depth or breadth of minima, with broader minima typically associated with better generalization.