Anticorrelated Noise Injection for Improved Generalization

Antonio Orvieto, Hans Kersting, Frank Proske, Francis Bach, Aurelien Lucchi

Anticorrelated Noise Injection for Improved Generalization

Antonio Orvieto [*1] **Hans Kersting** [*2] **Frank Proske** [3] **Francis Bach** [2] **Aurelien Lucchi** [4]

Abstract

Injecting artificial noise into gradient descent (GD) is commonly employed to improve the performance of machine learning models. Usually, uncorrelated noise is used in such *perturbed gradient descent* (PGD) methods. It is, however, not known if this is optimal or whether other types of noise could provide better generalization performance. In this paper, we zoom in on the problem of *correlating* the perturbations of consecutive PGD steps. We consider a variety of objective functions for which we find that GD with *anticorrelated* perturbations ("Anti-PGD") generalizes significantly better than GD and standard (uncorrelated) PGD. To support these experimental findings, we also derive a theoretical analysis that demonstrates that Anti-PGD moves to wider minima, while GD and PGD remain stuck in suboptimal regions or even diverge. This new connection between anticorrelated noise and generalization opens the field to novel ways to exploit noise for training machine learning models.

1. Introduction

It is widely believed that *flat minima* generalize better than sharp minima in loss *landscapes* of overparametrized models such as deep neural networks (DNNs). This idea goes back to Hochreiter & Schmidhuber (1995; 1997) who observed that, in flat minima, it is sufficient to determine weights with low precision and conjectured that this correlates with a small generalization gap. Although it has not been proved conclusively and the debate continues, this hypothesis has been supported by increasing empirical evidence (Keskar et al., 2016; Chaudhari et al., 2017; Jiang et al., 2019).

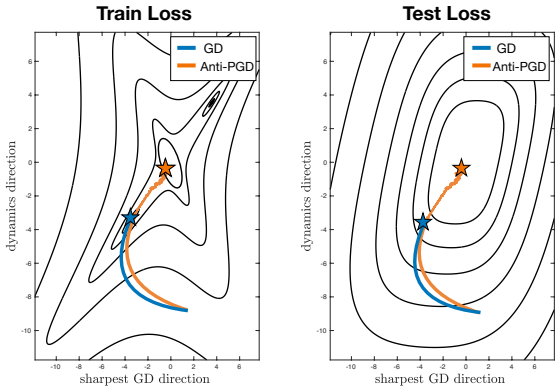

Figure 1. GD and Anti-PGD (GD with anticorrelated noise injection) on a quadratically parametrized model (details in §4) in 100 dimensions, with only 5 data points. The projection along two relevant directions is plotted. The train loss is most stable to sampling artifacts at flat minima. Indeed, the spurious minima on the south-west and north-east (left plot) are sharp. Flat minima often yield lower test losses. We prove in Thms. 2.1 & 3.1 that Anti-PGD is biased towards convergence to flat minima.

On the other hand, much work has analyzed which *optimization algorithms* yield good performance on test data. For standard stochastic gradient descent (SGD), it is a common finding (both empirically and theoretically) that its stochastic noise tends to guide the optimizer towards flat minima; see Smith et al. (2020) and the references therein. Going beyond SGD, several papers have proposed to perturb (stochastic) gradient descent methods by injecting artificial noise. So far, such perturbed gradient descent (PGD) methods have proved beneficial to quickly escape spurious local minima (Zhou et al., 2019) and saddle points (Jin et al., 2021). Based on the findings made in prior work, it seems natural to ask about the role of noise injection on the generalization performance of a model. The exact question we investigate is *whether stochastic noise can be designed to match (or even outperform) the favorable generalization properties of vanilla SGD.*

Contribution. We start from the observation that prior PGD methods rely on independent (uncorrelated) perturbations. We question whether this choice is optimal and instead study whether (anti-)correlated perturbations are

[*]Equal contribution [1]Department of Computer Science, ETH Zurich, Switzerland [2]INRIA – Ecole Normale Supérieure – PSL Research University, Paris, France [3]Department of Mathematics, University of Oslo, Norway [4]Department of Mathematics and Computer Science, University of Basel, Switzerland. Correspondence to: Antonio Orvieto <antonio.orvieto@inf.ethz.ch>, Hans Kersting <hans.kersting@inria.fr>.

Proceedings of the 39^{th} International Conference on Machine Learning, Baltimore, Maryland, USA, PMLR 162, 2022.

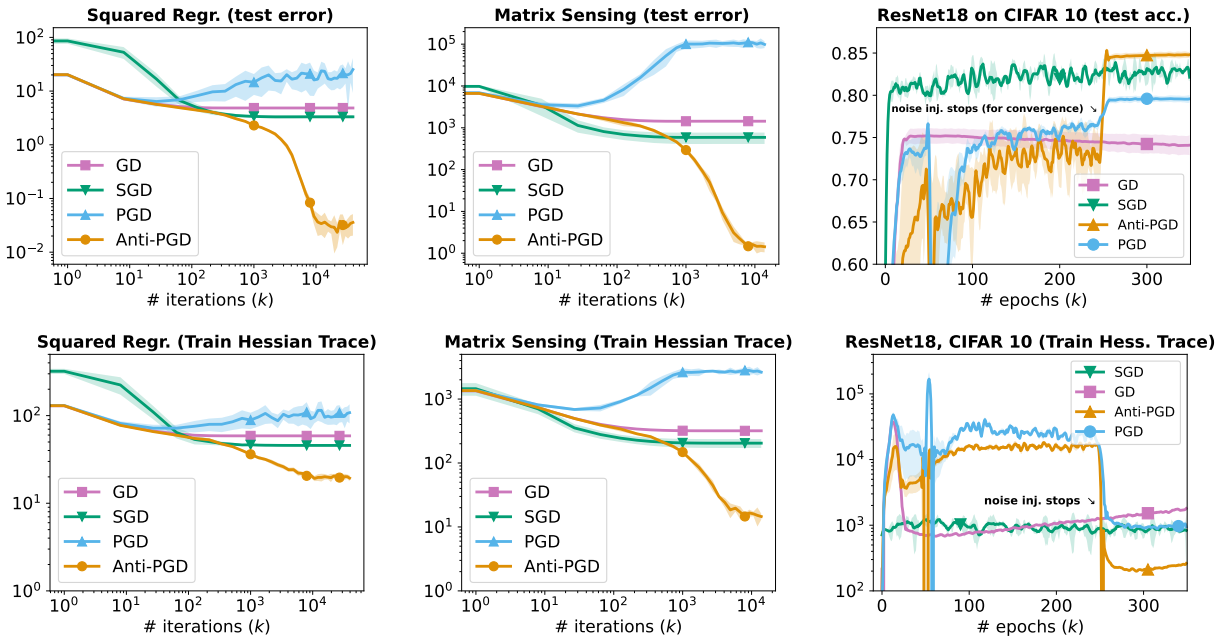

Figure 2. Effect of uncorrelated (PGD) and anticorrelated (Anti-PGD) noise injection on learning with gradient descent. Experiments are conducted on three non-convex machine learning problems with increasing complexity (details in §4). These experiments are inspired by recent literature on label noise (Blanc et al., 2020; HaoChen et al., 2021). Shown is the mean and standard deviation over several runs (5 for the first two problems, 3 for the last). Findings are robust to hyperparameter tuning (see Appendix D). All experiments suggest that Anti-PGD allows convergence to a flat minimizer (lower Hessian trace), improving generalization. In the ResNet18 experiments, the high dimensionality makes it hard to evaluate metrics under noise injection – since we converge to a neighborhood with big (dimension dependent) radius. Hence, we evaluate the accuracy and the Hessian trace after stopping noise injection, to allow exact convergence to the nearest minimizer. Note that, while SGD can temporarily be better than Anti-PGD for a small number of iterations, Anti-PGD ultimately outperforms SGD in all experiments.For more details and further investigations, see §4.

more suitable in terms of generalization. We introduce a new perturbed gradient descent method we name "Anti-PGD" whose perturbations at two consecutive steps are *anticorrelated*. We motivate this design by showing that Anti-PGD drifts (on average) to flat parts of the loss landscape. We conduct an extensive set of experiments – ranging from shallow neural networks to deep architectures with real data (e.g. CIFAR 10) – and we demonstrate that Anti-PGD, indeed, reliably finds minima that are both flatter and generalize better than the ones found by standard GD or PGD. We explain this observation with two theorems. Firstly, we show that Anti-PGD minimizes the trace of the Hessian – in the sense of converging (in expectation) to a minimum of a regularized loss to which the trace of the Hessian is added. Secondly, we show in the simplest possible toy model (the "widening valley") that Anti-PGD converges to the flattest minimum – while GD gets stuck in sharper minima and standard (uncorrelated) PGD diverges. In summary, these findings lead us to postulate that anticorrelated noise can be employed to improve generalization.

1.1. Related Work

The following lines of work are closely connected to our work. In particular, the connections with PAC-Bayes bounds and label noise are particularly relevant and will be further discussed later on.

Generalization measures and flat minima. Generalization measures are quantities that monotonically capture the generalization ability of a model. For example, Keskar et al. (2016) and Chaudhari et al. (2017) conducted an extensive set of experiments demonstrating that the spectrum of the Hessian of the loss $\nabla^2 L(w^*)$ computed at a minimum w^* is related to the generalization performance – in the sense that low eigenvalues of $\nabla^2 L(w^*)$ tend to indicate good generalization performance. To capture this phenomenon, several flatness (a.k.a. sharpness) measures have been proposed as generalization measures. Notably, Jiang et al. (2019) conducted a large-scale comparison of many popular generalization measures and concluded that some flatness measures are among the best performing measures.Recently, Petzka et al. (2021) connected flatness to generalization via the notion of 'relative flatness'.

We note, however, that the superiority of flat minima is contested: Dinh et al. (2017) demonstrated that sharp minima can also generalize well and point out that flatness is not invariant to reparametrization. Hence, the empirical finding of correlation between flatness and good generalization

should not necessarily be regarded as a causal relationship.

PAC-Bayes bounds. The generalization ability of a model can in theory be captured by upper bounding the generalization gap, as done by classical VC or Rademacher bounds, as well as PAC-Bayes bounds such as Langford & Caruana (2002), Neyshabur et al. (2017; 2018) and Tsuzuku et al. (2020). However, characterizing the generalization ability of deep learning models has proven to be a challenging task. Most classical bounds are vacuous when computed on modern over-parametrized networks. Encouragingly, empirical evidence – see, e.g., Dziugaite & Roy (2017) and follow-up works – has shown that PAC-Bayes bounds can be optimized to yield practically useful results. As discussed in Yang et al. (2019), PAC-Bayes bounds can also be related to flatness, and more precisely to the trace of the Hessian. The latter quantity will be key in our analysis and we will explain this connection in detail in §2.2.

The scale of noise in SGD. SGD tends to find minima that generalize surprisingly well in overparametrized models. This phenomenon has been explained from different perspectives in the literature. Focusing on the intrinsic noise of SGD, Zhang et al. (2019) and Smith et al. (2020) empirically showed that SGD generalizes well by converging to flat minima. Alternatively, Bradley & Gomez-Uribe (2021) characterized the stationary distribution of SGD, demonstrating a connection between increased levels of noise (i.e. smaller batch size or larger learning rate) and convergence to flat minima. Particularly related to our work is Wei & Schwab (2019) who showed that, in some settings, SGD decreases the trace of the Hessian in expectation.

The shape of noise in SGD. The distribution of the noise of SGD is often a subject of debate in the literature. In this regard, Simsekli et al. (2019) challenged the default assumption that SGD noise is Gaussian. In some particular settings, their work showed empirically that a heavy-tailed distribution is observed. This type of noise was then shown in Nguyen et al. (2019) to yield faster exit from sharp to flat minima. While the universality of the finding of Simsekli et al. (2019) is debated in the community (Panigrahi et al., 2019; Xie et al., 2020), the tail index of SGD has a drastic influence on its diffusion properties. For instance, heavy-tail noise leads to faster escape from *sharp minima*; see, e.g., Thm. 1 by Simsekli et al. (2019). Moreover, heavy-tail noise is provably found in simple models (e.g., linear regression on isotropic data), and in the regime of high-learning rates (Gurbuzbalaban et al., 2021) as an effect of multiplicative noise (Hodgkinson & Mahoney, 2021). Recently, Wang et al. (2022) demonstrated that truncated heavy-tailed noise can eliminate sharp minima in SGD.

Perturbed Gradient Descent (PGD). PGD is a version of (stochastic) gradient descent where artificial noise is added to the parameters after every step. Multiple PGD methods have been shown to help quickly escape spurious local minima (Zhou et al., 2019) and saddle points (Jin et al., 2021). These methods differ from our Anti-PGD in that they inject *uncorrelated* perturbations.
Instead of perturbing the parameter, one can alternatively add noise to the gradient which can improve learning for very deep networks (Neelakantan et al., 2015; Deng et al., 2021).

Label noise and implicit bias. Another way to add perturbations to SGD is to add noise to the labels of the data used for training. Recent work has demonstrated that such perturbations are indeed beneficial for generalization by implicitly regularizing the loss (Blanc et al., 2020; HaoChen et al., 2021; Damian et al., 2021). This alternative noise injection perturbs the labels *before* computing the gradient – instead of the parameter *after* a gradient-descent step, as in PGD and our Anti-PGD. (Nonetheless these approaches are closely connected, as we will further explain in §2.3.)
For the limit case of small SGD learning rate $\eta \to 0$, Li et al. (2022) introduced a general SDE framework to analyze the implicit bias in relation to flatness.

2. Finding Flat Minima by Anti-PGD

After introducing our problem setting, we provide a detailed description of *Anti-PGD* and explain how it is designed to find flat minima.

Problem setting. Let $\{(x^{(i)}, y^{(i)})\}_{i=1}^{M}$ denote a data set of M input-output pairs with $x^{(i)} \in \mathbb{R}^{d_{\text{in}}}$ and $y^{(i)} \in \mathbb{R}$. We consider a machine learning model $f_w : \mathbb{R}^{d_{\text{in}}} \to \mathbb{R}$, with parameters $w \in \mathbb{R}^d$, whose parameters are trained using empirical risk minimization. Let $L^{(i)} : \mathbb{R}^d \to \mathbb{R}$ be the loss associated with the i-th data point $(x^{(i)}, y^{(i)})$. We denote by $L(w) := \frac{1}{M}\sum_{i=1}^{M} L^{(i)}(w)$ the (full-batch) training loss, which we optimize to find the best parameters.

Anti-PGD. *Gradient descent* (GD) iteratively optimizes the loss $L(w)$ by computing a sequence of weights $\{w_n\}_{n=0}^{N}$ where $w_{n+1} = w_n - \eta\nabla L(w_n)$ with step size (a.k.a. learning rate) $\eta > 0$. *Perturbed gradient descent* (PGD) simply adds an i.i.d. perturbation to each step, i.e.

$$w_{n+1} = w_n - \eta\nabla L(w_n) + \xi_{n+1}, \tag{1}$$

where $\{\xi_n\}_{n=0}^{N}$ is a set of centered i.i.d. random variables with variance $\sigma^2 I$. Similarly, we define *anticorrelated perturbed gradient descent* (Anti-PGD) as

$$w_{n+1} = w_n - \eta\nabla L(w_n) + (\xi_{n+1} - \xi_n). \tag{2}$$

In other words, Anti-PGD replaces the i.i.d. perturbations $\{\xi_n\}_{n=0}^{N}$ in PGD (1) with their increments $\{\xi_{n+1} - \xi_n\}_{n=0}^{N-1}$. The name Anti-PGD comes from the fact that consecutive perturbations are anticorrelated:

$$\frac{\mathbb{E}\left[(\xi_{n+1} - \xi_n)(\xi_n - \xi_{n-1})^\top\right]}{2\sigma^2} \overset{\text{(iid)}}{=} -\frac{\operatorname{cov}(\xi_0)}{2\sigma^2} = -\frac{1}{2}I.$$

2.1. Regularization in Anti-PGD

While Anti-PGD (2) is defined as a modification of PGD (1), it can alternatively be viewed as a regularization (smoothing) of the loss landscape L. To see this, note that, after a change of variables $z_n := w_n - \xi_n$, the Anti-PGD step becomes

$$z_{n+1} = z_n - \eta \nabla L(z_n + \xi_n). \tag{3}$$

The corresponding loss $L(\cdot + \xi_n)$ can, in expectation, be regarded as a convolved (or smoothed) version of the original L. To see in which direction the gradients of this loss (and thus Anti-PGD) are biased, we perform a Taylor expansion of $\partial_i L(\cdot)$ around z_n:

$$z^i_{n+1} = z^i_n - \eta \partial_i L(z_n) - \eta \sum_j \partial^2_{ij} L(z_n) \xi^j_n - \underbrace{\frac{\eta}{2} \sum_{j,k} \partial^3_{ijk} L(z_n) \xi^j_n \xi^k_n}_{= \frac{\eta}{2} \partial_i \sum_{jk} \partial^2_{jk} L(z_n) \xi^j_n \xi^k_n} + O(\eta \|\xi_n\|^3), \tag{4}$$

where the term under the brace is due to Clairaut's theorem (assuming that L has continuous fourth-order partial derivatives). By exploiting that ξ_n has mean zero and covariance $\sigma^2 I$, we can express the conditional expectation of each step as

$$\mathbb{E}\left[z_{n+1} | z_n\right] = z_n - \eta \nabla \tilde{L}(z_n) + O\left(\eta \mathbb{E}[\|\xi_n\|^3]\right), \tag{5}$$

where the *modified loss* $\tilde{L}$ is given by

$$\tilde{L}(z) := L(z) + \frac{\sigma^2}{2} \operatorname{Tr}(\nabla^2 L(z)), \tag{6}$$

where $\operatorname{Tr}(A)$ denotes the trace of a square matrix A. (In Appendix A, we also compute the conditional variance of Anti-PGD.) The conditional mean, Eq. (5), highlights the *motivation* for Anti-PGD: When expressed in terms of the variable z_n, Anti-PGD in expectation (modulo the impact of the third moment of the noise) takes steps in the direction of a loss which is regularized by adding the trace of the Hessian. The higher the noise variance σ^2, the stronger is the influence of the (trace of the) Hessian on Anti-PGD. This is related to how stochastic gradient noise smoothes the loss in standard SGD (Kleinberg et al., 2018), with the difference that, here, we inject artificial noise that *explicitly* regularizes the trace of the Hessian. A discussion on the smoothing literature is postponed to §2.4.

In the next theorem, we analyze the case where the noise ξ_n follows a symmetric Bernoulli distribution. We find that, indeed, Anti-PGD (on average) minimizes the regularized loss $\tilde{L}$ – in the sense that the regularized gradient converges.

Theorem 2.1 (Convergence of the regularized gradients)**.** *Let $L : \mathbb{R}^d \to \mathbb{R}$ be lower bounded with continuous fourth-order partial derivatives and β-Lipschitz continuous third-order partial derivatives, for some constant $\beta > 0$. Consider the iterates $\{z_n\}_{n=0}^{N-1}$ computed by Anti-PGD as in (3) with $\eta \leq 1/\beta$, where for each n the noise coordinate ξ^n_i follows a symmetric centered Bernoulli distribution with variance σ^2 (i.e., σ and $-\sigma$ have probability $1/2$). Let $\epsilon > 0$. If we set $\eta = O(\epsilon/\sigma^3)$ and $N = \Theta(\epsilon^{-1})$, then it holds true that*

$$\mathbb{E}\left[\frac{1}{N} \sum_{n=0}^{N-1} \|\nabla \tilde{L}(z_n)\|^2\right] \leq \epsilon. \tag{7}$$

For a proof, see Appendix B. Now that we have seen how exactly anticorrelated perturbations lead to a reduction of the trace of the Hessian appearing in (6), we connect this finding to previous work relating to regularizing by the trace of the Hessian.

2.2. Connection with PAC-Bayes Bounds

PAC-Bayes bounds can be interpreted as bounds on the average loss over a posterior distribution Q. These bounds connect to the curvature of the loss through the concept of expected sharpness. The following theorem makes this connection precise.

Theorem 2.2 ((Neyshabur et al., 2017; Tsuzuku et al., 2020))**.** *Let $Q(w|w^*)$ be any distribution over the parameters, centered at the solution w^* found by a gradient-based method. For any non-negative real number λ, with probability at least $1 - \delta$ one has*

$$L_{\text{true}}(Q(w|w^*)) \leq L(w^*) + \frac{\lambda}{2M} + \frac{1}{\lambda} \ln\left(\frac{1}{\delta}\right) + \underbrace{L(Q(w|w^*)) - L(w^*)}_{\textit{expected sharpness}} + \frac{1}{\lambda} \operatorname{KL}[Q(w|w^*)||P(w)],$$

where L_{true} is the generalization loss; $L(Q) := \mathbb{E}_{w\sim Q} L(w)$ and $L_{\text{true}}(Q) := \mathbb{E}_{w\sim Q} L_{\text{true}}(w)$; P is a distribution over parameters; and KL *denotes the Kullback-Leibler divergence.*

For a proof, see Tsuzuku et al. (2020). In the setting of this theorem, by picking Q to be Gaussian with variance s^2, one obtains the following approximation of the expected sharpness

$$L(Q(w|w^*), w^*) - L(w^*) \approx \frac{s^2}{2} \operatorname{Tr}(\nabla^2 L(w^*)). \tag{8}$$

Thus, by minimizing the trace of the Hessian, Anti-PGD is expected to also reduce the PAC-Bayes bound from Thm. 2.2. In fact, the reasoning behind the bound in Thm. 2.2 has motivated researchers to find an explicit link between stochastic gradient noise and the trace of the Hessian at the solution found by SGD. Empirically, these quantities have a high correlation in many settings (Yao et al., 2020; Smith et al., 2021): usually, the lower the trace (i.e., the flatter the minima), the higher is the test accuracy.Similar bounds involving the trace of the Hessian are also discussed by (Dziugaite & Roy, 2018; Wang et al., 2018).

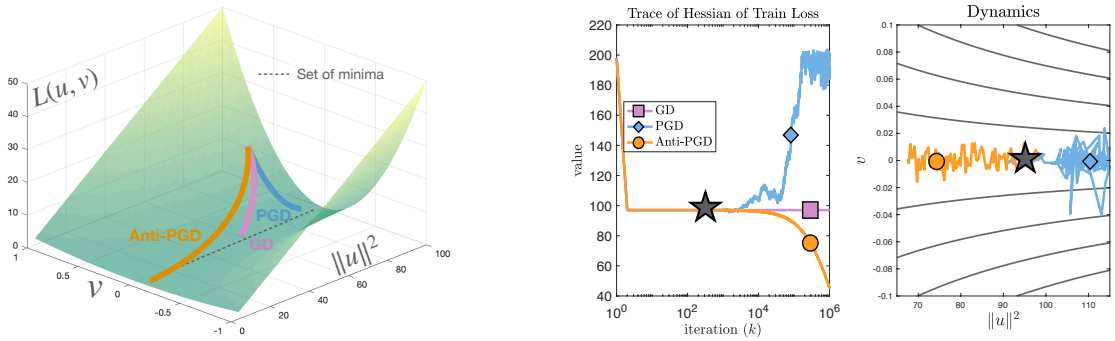

Figure 3. **Left**: Illustration of the *widening valley* loss L, Eq. (10). A valley of minima with loss $L(u, v) = 0$ for all (u, v) with $v = 0$; the smaller $\|u\|$, the flatter the minimum. GD gets stuck where it first touches this valley. PGD diverges to sharp regions (with high $\|u\|$). Anti-PGD converges to a flat minimum (with small $\|u\|$). **Right**: Simulation of the considered algorithms on the widening valley. After convergence of GD (black star), we start injecting uncorrelated and anticorrelated noise. We choose $\eta = 0.01$, and $\sigma = 0.005$ – yet the findings generalize to all sets of stable parameters. The observed behavior is supported by Thm. 3.1. The plot looks similar for both Gaussian and Bernoulli noise injection.

2.3. Comparison with Label Noise

Instead of perturbing w as in PGD, label-noise methods perturb the label $y^{(i)}$ of the data. If we denote $f_w(x)$ as the output of our model for input x, the label-noise update in the full-batch setting with squared loss is $w_{n+1} = w_n - \eta \nabla \bar{L}(w_n)$, with $\bar{L}(w) = \frac{1}{2}\sum_{i=1}^{M}\left[f_w(x^{(i)}) - y^{(i)} + \xi_{n+1}\right]^2$, for a set of random perturbations $\{\xi_n\}_{n=0}^{N}$. It is instructive to compare the label-noise loss $\bar{L}$ with the Anti-PGD loss $L(\cdot + \xi_n)$ from Eq. (3). The above formula gives the gradient as

$$\nabla \bar{L}(w) = \nabla L(w) + \sum_{i=1}^{M} \nabla f_w(x^{(i)})\xi_{n+1}. \tag{9}$$

Hence, while label noise was observed to yield an improvement in terms of generalization (Blanc et al., 2020; HaoChen et al., 2021; Damian et al., 2021), its effect (in general) is highly dependent on the model and on the data. Instead, the noise injection we propose is both data and model independent, as can be seen from the regularization in Eq. (5).

2.4. Connection to Smoothing

Eq. (6) shows that Anti-PGD amounts to optimizing a regularized loss, which we can also interpret as a smoothing of the original objective function. Smoothing is of course not a new concept in the field of optimization as it is often used to regularize non-differentiable functions in order to compute approximate derivatives (Nesterov & Spokoiny, 2017), or to obtain faster rates of convergence (Lin et al., 2018).

In the context of deep learning, noise injection (or even stochastic gradient noise) is often linked to smoothing (Kleinberg et al., 2018; Stich & Harshvardhan, 2021; Bisla et al., 2022). As we saw in Eq. (3), anticorrelated noise injection is equivalent to smoothing after a change of variables – this property was crucial in deriving the trace regularizer. We are not aware of any similar explicit regularization result in the smoothing literature (most work focuses on the resulting landscape properties and convergence guarantees). Even though Anti-PGD is linked to smoothing, it is much more convenient to analyze: $\nabla f(x + \xi)$ follows a data-dependent distribution that is complex to characterize. Instead, in Anti-PGD, the smoothing effect comes from adding — this is a *linear* operation — anticorrelated random variables. This is very convenient and will be leveraged in the proof for the next result, Thm. 3.1.

3. Convergence in Widening Valleys

We have seen above that Anti-PGD acts as a regularizer on the trace of the Hessian. In this section, we will analyze the dynamics of Anti-PGD in more detail on the "widening valley" – the simplest possible loss landscape with a changing trace of the Hessian. In the following subsections, we will introduce this model (§3.1), demonstrate with experiments that Anti-PGD successfully finds flat minima in this model (§3.2), prove this behaviour theoretically (§3.3), and explain how the widening valley relates to more realistic problems like sparse regression (§3.4).

3.1. The Widening Valley Landscape

The *widening valley* is defined as the loss function

$$L(u, v) = \tfrac{1}{2}v^2\|u\|^2, \tag{10}$$

where $\|\cdot\|$ is the Euclidean norm, $v \in \mathbb{R}$, and $u \in \mathbb{R}^d$; see Fig. 3. The gradient and Hessian of L are given by

$$\nabla L(u, v) = \begin{bmatrix} v^2 \cdot u \\ \|u\|^2 v \end{bmatrix}, \quad \nabla^2 L(u, v) = \begin{bmatrix} v^2 I_d & 2vu \\ 2vu^\top & \|u\| \end{bmatrix}. \tag{11}$$

The trace of the Hessian is thus

$$\mathrm{Tr}(\nabla^2 L(u, v)) = dv^2 + \|u\|^2. \tag{12}$$

We consider L as a suitable problem to analyze the dynamics of GD and Anti-PGD as it has a relatively simple structure consisting of a valley of minima with monotonously changing flatness (as measured by the trace of the Hessian): All (u, v) with $v = 0$ are minima, but we also require $\|u\|$ to be minimized as well to get a small trace of the Hessian.

The widening valley can also be seen as a simplified local model of the landscape close to a minimizer. Indeed, Draxler et al. (2018) showed that minimizers in neural networks are often connected by a path where the loss is exactly zero: no jumping is required for an optimizer to gradually increase the solution flatness. While these valleys are not straight in general and the flatness might not change monotonously, our straight valley (10) with monotonously changing flatness serves as a first simplified model. We will link it to a more realistic regression problem in §3.4.

3.2. Empirical Demonstration

When optimizing the widening valley (10), GD will get stuck in any of the global minima $(u, v = 0)$, regardless of their flatness. In particular, if the dimension $d \gg 1$, the path of GD will be biased towards making v small and not optimizing u (since the direction along v is the most curved). As a result, the final Hessian trace will be $\|u_0\|^2$. Improving this by injecting noise is challenging: when adding stochastic perturbations, one has to balance perturbing v away from zero – to get a gradient (11) to reduce $\|u\|$ – while preventing $\|u\|$ from growing too much.

We find empirically that Anti-PGD succeeds to do this and moves to flat parts of the valley, while PGD does not; see Fig. 3. This means that Anti-PGD converges to flat parts of the valley, while PGD diverges to sharper regions; see Fig. 4.

3.3. Theoretical Analysis

The following theorem proves what we empirically demonstrated in the preceding section.

Theorem 3.1 (Widening Valley)**.** *Let $L : \mathbb{R}^{d+1} \to \mathbb{R}$ be the* widening valley *loss from Eq. (10). We start optimizing from a point $w_0 = (u_0, 0)$, where $\|u_0\|^2 = D \gg 1$ (e.g. the solution found by gradient descent), around which we consider the domain $\mathcal{D}_\alpha := \{(u, v) \in \mathbb{R}^{d+1} : \|u\|^2 \in (\alpha D, D/\alpha)\}$ for some fixed $\alpha \in (0, 1)$. We want to compare the long-term stochastic dynamics of PGD and Anti-PGD, as defined in Eqs. (1) and (2), in terms of where they exit $\mathcal{D}_\alpha$. As a noise model, we assume that the i.i.d. perturbations ξ_n are distributed according to a symmetric centered Bernoulli distribution (i.e., σ and $-\sigma$ have probability $1/2$) whose variance σ^2 is upper bounded by $\sigma^2 \in \left(0, \min\left\{\frac{\alpha^3 D}{2}, \frac{D}{8\alpha}\right\}\right]$. As a step size, we set $\eta = \frac{\alpha}{2D}$ which, for both methods, leads to stable dynamics inside of $\mathcal{D}_\alpha$. We find that (on average) PGD and Anti-PGD exit through different sides of $\mathcal{D}_\alpha$:*

1. ***In high dimensions, PGD diverges away from zero.*** *If $d \geq \frac{2}{\alpha^2}$, then it holds for any admissible σ^2 that*

$$\lim_{n\to\infty} \mathbb{E}\left[\|u_n\|^2\right] \geq D/\alpha, \tag{13}$$

   where u_n are the first d coordinates of w_n computed by PGD as in (1).

2. ***Independent of dimensions, Anti-PGD goes to zero.*** *For any $d \in \mathbb{N}$, if we choose any admissible σ^2 such that $\sigma^2 \leq \frac{\alpha D}{2d}$, then*

$$\lim_{n\to\infty} \mathbb{E}\left[\|u_n\|^2\right] \leq \alpha D, \tag{14}$$

   where u_n are the first d coordinates of w_n, computed by Anti-PGD as in (2).

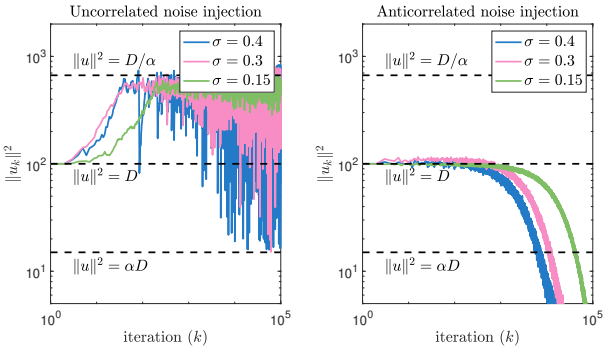

Figure 4. Numerical illustration and verification of Thm. 3.1. Performance of PGD (**left**) and Anti-PGD (**right**) on the widening valley in Eq. (10). The setting and the notation is as described in Thm. 3.1, and the simulation confirms the result: that is, Anti-PGD effectively decreases $\|u\|^2$ below αD, where for this plot we consider $\alpha = 0.25$, $\eta = \alpha/D$ and $d = 100$. Instead, the high problem dimensionality $d \geq 2/\alpha^2 = 32$ induces an increase in $\|u\|^2$ for standard PGD, which gets bigger than D/α.

As expected, this theorem implies that, as $n \to \infty$, Anti-PGD reduces the trace of Hessian while PGD increases it. For a proof, see Appendix C.

Corollary 3.1 (The trace of the Hessian in the widening valley)**.** *In the same setting as Thm. 3.1, let $\eta = \frac{\alpha}{2D}$, $\sigma^2 \in \left(0, \min\left\{\frac{\alpha^3 D}{2}, \frac{D}{8\alpha}, \frac{\alpha D}{2d}\right\}\right]$ and $d \geq \frac{2}{\alpha^2}$. If $\alpha \ll 1$, then*

$$\lim_{n\to\infty} \mathbb{E}[\mathrm{Tr}(\nabla^2 L(w_n^{\mathrm{anti}}))] \leq 16\alpha D \ll \mathbb{E}[\mathrm{Tr}(\nabla^2 L(w_0))]$$
$$\lim_{n\to\infty} \mathbb{E}[\mathrm{Tr}(\nabla^2 L(w_n^{un}))] \geq D/\alpha \gg \mathbb{E}[\mathrm{Tr}(\nabla^2 L(w_0))],$$

where $w_n^{un} = (u_n, v_n)$ and $w_n^{\mathrm{anti}} = (u_n, v_n)$ are the weights computed by Anti-PGD and PGD respectively.

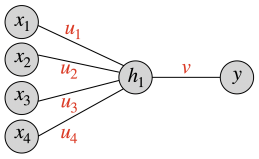

Figure 5. Pictorial illustration of the network (linear activations, one hidden unit) we study in §3.4. The associated loss function, Eq. (18), has striking similarities to the widening valley, Eq. (10).

3.4. Relation to Linear Networks with One Hidden Unit

In this section, we explain how widening valleys, similar to our model (10), might appear in more realistic learning problems. To this end, consider sparse regression with M input-output pairs $\{(x^i, y^i)\}_{i=1}^M$, where $x^i \in \mathbb{R}^{m+d}$, $d, m > 1$, and $y^i \in \mathbb{R}$ for all $i \in [M]$. To induce sparseness, we consider the setting where only the first m features of each x^i, i.e., $(x^i_1, x^i_2, \dots, x^i_m)$ are relevant predictors, while the other features $(x^i_{m+1}, x^i_{m+2}, \dots, x^i_{m+d})$ are uncorrelated from the target. Further, we assume that the input has isotropic standardised distribution. As predictor, we consider a neural network with one hidden neuron and standard square loss

$$L(u, v) = \frac{1}{2M} \sum_{i=1}^{M} \left(y^i - v \cdot u^\top x^i\right)^2 . \tag{15}$$

Such a loss is highly non-convex, due to the non-linear interaction between v and u. By expanding the square, we obtain

$$2L(u, v) = \mathbb{E}_i[(y^i)^2] - 2v \cdot u^\top \mathbb{E}[y^i x^i] + v^2 \mathbb{E}_i[(u^\top x^i)^2].$$

We can drop the first term since it is irrelevant for optimization. Further, the last term can be written as

$$v^2 \mathbb{E}_i[(u^\top x^i)^2] = v^2 \mathbb{E}_i[\mathrm{Tr}(u^\top x^i (x^i)^\top u)]. \tag{16}$$

Using the cyclic property of the trace and the assumption $\mathbb{E}_i[x^i (x^i)^\top] = I$, we get

$$v^2 \mathbb{E}_i[(u^\top x)^2] = v^2 \,\mathrm{Tr}(uu^\top) = v^2 \|u\|^2. \tag{17}$$

Therefore, we obtain $L(u, v) = \frac{1}{2} v^2 \|u\|^2 - 2v \cdot u^\top \mathbb{E}[y^i x^i]$. We now use the sparseness assumption: since the features $(x^i_{m+1}, x^i_{m+2}, \dots, x^i_{m+d})$ are uncorrelated from the target, we have

$$L(u, v) = \frac{1}{2} v^2 \|u\|^2 - 2v \cdot u^\top_{1:m} \mathbb{E}[y^i x^i_{1:m}]. \tag{18}$$

This loss (18) is very similar to the widening valley (10): For good generalization, the weights relative to the spurious coordinates $(u_{m+1}, u_{m+2}, \dots, u_{m+d})$ have to be set to zero. Unfortunately, the solution of gradient descent (without further regularization), in general does not have this property (see §3.2). This fact motivates us to look at the dynamics in the space $(u_{m+1}, u_{m+2}, \dots, u_{m+d}, v)$, ignoring the dynamics on the space $(u_1, u_2, \dots, u_m)$. In the spurious subspace of the parameter space, the last term in the last equation is a constant, and therefore the effective loss becomes $L(u, v) = \frac{1}{2} v^2 \|u\|^2$, where $u \in \mathbb{R}^d$ denotes the vector $(u_{m+1}, \dots, u_{m+d})$.

4. Additional Experiments and Details

We demonstrated the validity of our theoretical findings in Fig. 2 by showcasing the performance of Anti-PGD on the three different problems (see the next three paragraphs). Finally, in another experiment on CIFAR 10, we will show that Anti-PGD can recover from a sharp minimum.

Quadratically-parametrized linear regression. For a data matrix $X \in \mathbb{R}^{n \times d}$ ($d = 100, n = 40$) and targets $y \in \mathbb{R}^d$, this loss is $L(w) = \frac{1}{4n} \|X(w \odot w) - y\|^2$, where $\odot$ denotes the element-wise product. While the expressive power of the underlying model is limited, it has a few interesting features which make it a compelling case study (see discussion by HaoChen et al. (2021)). First, note that the nonlinear parametrization makes the loss non-convex: to see this simply note that changing the sign in any weight does not change the loss. Furthermore, inspecting the Hessian (see §D.2) one can easily see that minima with different curvature exist. On this problem, Anti-PGD (with Gaussian perturbations) is able to find a flat minimum; a pictorial illustration of the corresponding dynamics can be found in Fig. 1. In §D.2 we show that this good performance is robust to different choices of hyperparameters: we found no setting where well-tuned PGD (with Gaussian perturbations) outperforms Anti-PGD.

Matrix sensing. The corresponding loss function has some similarities with quadratically-parametrized linear regression, and was considered by Blanc et al. (2020) to study label noise. All the findings of the above paragraph hold true in this setting as well. Details on the experimental setup and hyperparameter tuning can be found in §D.3.

CIFAR10 on ResNet 18. We consider training a ResNet18-like architecture (He et al., 2016) with batch normalization. Architecture details are provided in §D.4. The performance on this network greatly depends on careful hyperparameter tuning, algorithmic choices (e.g., adaptive step sizes), schedulers, etc. Here, to keep things simple, we train with a simple SGD optimizer (with momentum 0.9), and select a learning rate of 0.05. To approximate full-batch gradient descent we use a very large batch size of 7500 samples (i.e. until saturation of 5 GPUs). To isolate the effect of noise injection, without mixing it with mini-batch noise, we also run PGD and Anti-PGD (with Gaussian perturbations) in this high batch regime (1/7 of the dataset). For SGD, we instead select a batch size of 128, and keep the learning rate at 0.05. For convergence of the test accuracy and the

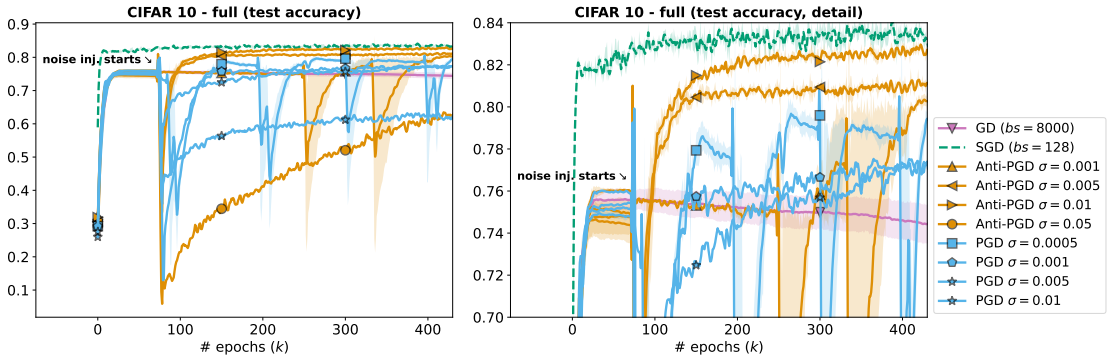

Figure 6. Anti-PGD, PGD and GD on CIFAR 10, using a ResNet18-like architecture (details and further plots in the Appendix). Mean and one standard deviation are plotted. We train all algorithms with a step size of 0.05, which leads to fast convergence of both GD and SGD (batch size 128) in 70 iteration. Compared to Fig. 1, we start the noise injection at epoch 75. Results are discussed in the main text.

Hessian trace, it is convenient to kill the noise injection after 250 epochs – so that the optimizer converges to the nearest minimum. Again, we see that (compared with PGD) Anti-PGD find a flatter minimum that generalizes better. Crucially, we show in Appendix §D.4 that – as in the problems above – no tuning of σ (noise injection level) can help PGD to reach the generalization performance of Anti-PGD. This is confirmed by the results in the next paragraph.

Recovering from a sharp minimum. In this experiment, we keep the parameter settings as in the last paragraph, but instead consider injecting noise only after 75 epochs – i.e., after convergence of full-batch gradient descent. As a result, Anti-PGD and PGD are trapped in the minimum found by GD, until noise injection starts. The behavior we observe (Fig. 6) after noise injection resembles our widening valley model in Fig. 3: noise injection makes the dynamics suddenly unstable, and we observe several points during training where the algorithm is (probably) switching between minima. Note that this behavior is very different from the one observed in Fig. 2, where we start noise injection from the very beginning – and then kill the noise injection at the end. In contrast, we here do the reverse: we first initialize at a bad minimum and then inject noise to recover. We observe that Anti-PGD is able to recover from the bad initialization much better than PGD. Interestingly, while stopping noise injection at the end of training was needed for good accuracy in Fig. 2, here no such step is needed for Anti-PGD to get an accuracy close to SGD (i.e., we directly recover from a bad minimum). We postulate that this difference comes from the different landscape properties when (a) close to initialization or (b) close to a local minimizer. For the last two experiments, we provide further plots (e.g. test-train loss) in Appendix D.4. In summary, we found that Anti-PGD reliably finds flat minima that generalize well – as predicted by the theory in §2 and §3.

Anti-SGD. In the appendix, we show that a combination of mini-batch noise and anti-correlated noise is able to further improve on the performance (see Fig. 17).

5. Conclusion and Future Work

Motivated by recent findings on the correlation of the flatness of minima with their generalization performance, we demonstrated that anticorrelated noise injection can improve the generalization of machine learning models – by biasing the optimization toward flatter minima. To this end, we replaced the i.i.d. perturbations in perturbed gradient descent with anticorrelated ones. We proved that the resulting method *Anti-PGD* regularizes the trace of the Hessian, a common measure of flatness. In order to provide further theoretical justification, we introduced the *widening valley* model and proved that Anti-PGD converges to the flat part of the valley – while GD and standard PGD remain in sharper regions. In realistic experiments with real data (e.g. CIFAR 10), we likewise observed that Anti-PGD converges to flat minima that generalize well (compared with GD and standard PGD).

These discoveries lead us to hypothesize that anticorrelated noise can improve the generalization performance of a model – which opens up several directions to investigate. First of all, the range from uncorrelated to anticorrelated perturbations should be explored further, and combined with different noise distributions. Since uncorrelated noise can help to quickly exit saddle points (Jin et al., 2021), a compromise (or adaptive schedule) between uncorrelated and anticorrelated might be beneficial.

Moreover, it seems worthwhile to explore the implications of our findings for standard SGD. Unlike common noise injection techniques, the noise of SGD is data-dependent and its magnitude is determined by the ratio of step size and batch size. One could, however, modify the selection of the batches to (negatively) correlate the stochastic gradient noise of subsequent steps. One could also add anticorrelated noise on top of the existing noise in SGD, or inject it only after the test loss of SGD (or another optimizer) plateaus. Both theoretical and empirical results are likely to provide novel insights about the importance of noise in optimization.

References

Bisla, D., Wang, J., and Choromanska, A. Low-pass filtering SGD for recovering flat optima in the deep learning optimization landscape. *arXiv preprint arXiv:2201.08025*, 2022.

Blanc, G., Gupta, N., Valiant, G., and Valiant, P. Implicit regularization for deep neural networks driven by an Ornstein-Uhlenbeck like process. In *Conference on learning theory*, pp. 483–513, 2020.

Bradley, A. V. and Gomez-Uribe, C. A. How can increased randomness in stochastic gradient descent improve generalization? *arXiv preprint arXiv:2108.09507*, 2021.

Chaudhari, P., Choromanska, A., Soatto, S., LeCun, Y., Baldassi, C., Borgs, C., Chayes, J., Sagun, L., and Zecchina, R. Entropy-SGD: Biasing gradient descent into wide valleys. In *International Conference on Learning Representations*, 2017.

Damian, A., Ma, T., and Lee, J. Label noise SGD provably prefers flat global minimizers. *arXiv preprint arXiv:2106.06530*, 2021.

Deng, Z., Huang, J., and Kawaguchi, K. How shrinking gradient noise helps the performance of neural networks. In *2021 IEEE International Conference on Big Data (Big Data)*, pp. 1002–1007, 2021.

Dinh, L., Pascanu, R., Bengio, S., and Bengio, Y. Sharp minima can generalize for deep nets. In *International Conference on Machine Learning*, pp. 1019–1028, 2017.

Draxler, F., Veschgini, K., Salmhofer, M., and Hamprecht, F. Essentially no barriers in neural network energy landscape. In *International conference on machine learning*, pp. 1309–1318, 2018.

Dziugaite, G. K. and Roy, D. M. Computing nonvacuous generalization bounds for deep (stochastic) neural networks with many more parameters than training data. In *Uncertainty in Artificial Intelligence*, 2017.

Dziugaite, G. K. and Roy, D. M. Data-dependent PAC-Bayes priors via differential privacy. *arXiv preprint arXiv:1802.09583*, 2018.

Gurbuzbalaban, M., Simsekli, U., and Zhu, L. The heavy-tail phenomenon in SGD. In *International Conference on Machine Learning*, pp. 3964–3975, 2021.

HaoChen, J. Z., Wei, C., Lee, J., and Ma, T. Shape matters: Understanding the implicit bias of the noise covariance. In *Conference on Learning Theory*, pp. 2315–2357, 2021.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Hochreiter, S. and Schmidhuber, J. Simplifying neural nets by discovering flat minima. In *Advances in Neural Information Processing Systems*, 1995.

Hochreiter, S. and Schmidhuber, J. Flat minima. *Neural Computation*, 9:11–42, 1997.

Hodgkinson, L. and Mahoney, M. Multiplicative noise and heavy tails in stochastic optimization. In *International Conference on Machine Learning*, pp. 4262–4274, 2021.

Jiang, Y., Neyshabur, B., Mobahi, H., Krishnan, D., and Bengio, S. Fantastic generalization measures and where to find them. In *International Conference on Learning Representations*, 2019.

Jin, C., Netrapalli, P., Ge, R., Kakade, S. M., and Jordan, M. I. On nonconvex optimization for machine learning: Gradients, stochasticity, and saddle points. *J. ACM*, 68 (2), 2021.

Keskar, N. S., Mudigere, D., Nocedal, J., Smelyanskiy, M., and Tang, P. T. P. On large-batch training for deep learning: Generalization gap and sharp minima. *arXiv preprint arXiv:1609.04836*, 2016.

Kleinberg, B., Li, Y., and Yuan, Y. An alternative view: When does SGD escape local minima? In *International Conference on Machine Learning*, 2018.

Langford, J. and Caruana, R. (Not) bounding the true error. In *Advances in Neural Information Processing Systems*, 2002.

Li, Z., Wang, T., and Arora, S. What happens after SGD reaches zero loss? – A mathematical framework. In *International Conference on Learning Representations*, 2022.

Lin, H., Mairal, J., and Harchaoui, Z. Catalyst acceleration for first-order convex optimization: from theory to practice. *Journal of Machine Learning Research*, 18(1): 7854–7907, 2018.

Neelakantan, A., Vilnis, L., Le, Q. V., Sutskever, I., Kaiser, L., Kurach, K., and Martens, J. Adding gradient noise improves learning for very deep networks. *arXiv preprint arXiv:1511.06807*, 2015.

Nesterov, Y. *Lectures on Convex Optimization*. Springer, 2nd edition, 2018.

Nesterov, Y. and Spokoiny, V. Random gradient-free minimization of convex functions. *Foundations of Computational Mathematics*, 17(2):527–566, 2017.

Neyshabur, B., Bhojanapalli, S., McAllester, D., and Srebro, N. Exploring generalization in deep learning. In *Advances in Neural Information Processing Systems*, 2017.

Neyshabur, B., Bhojanapalli, S., and Srebro, N. A PAC-Bayesian approach to spectrally-normalized margin bounds for neural networks. In *International Conference on Learning Representations*, 2018.

Nguyen, T. H., Simsekli, U., Gurbuzbalaban, M., and Richard, G. First exit time analysis of stochastic gradient descent under heavy-tailed gradient noise. In *Advances in Neural Information Processing Systems*, 2019.

Panigrahi, A., Somani, R., Goyal, N., and Netrapalli, P. Non-Gaussianity of stochastic gradient noise. *arXiv preprint arXiv:1910.09626*, 2019.

Petzka, H., Kamp, M., Adilova, L., Sminchisescu, C., and Boley, M. Relative flatness and generalization. In *Advances in Neural Information Processing Systems*, 2021.

Simsekli, U., Sagun, L., and Gurbuzbalaban, M. A tail-index analysis of stochastic gradient noise in deep neural networks. In *International Conference on Machine Learning*, 2019.

Smith, S., Elsen, E., and De, S. On the generalization benefit of noise in stochastic gradient descent. In *International Conference on Machine Learning*, pp. 9058–9067, 2020.

Smith, S. L., Dherin, B., Barrett, D. G., and De, S. On the origin of implicit regularization in stochastic gradient descent. *arXiv preprint arXiv:2101.12176*, 2021.

Stich, S. and Harshvardhan, H. Escaping local minima with stochastic noise. *Neurips Workshop on Optimization for Machine Learning*, 2021.

Tsuzuku, Y., Sato, I., and Sugiyama, M. Normalized flat minima: Exploring scale invariant definition of flat minima for neural networks using PAC-Bayesian analysis. In *International Conference on Machine Learning*, pp. 9636–9647, 2020.

Wang, H., Keskar, N. S., Xiong, C., and Socher, R. Identifying generalization properties in neural networks. *arXiv preprint arXiv:1809.07402*, 2018.

Wang, X., Oh, S., and Rhee, C.-H. Eliminating sharp minima from SGD with truncated heavy-tailed noise. In *International Conference on Learning Representations*, 2022.

Wei, M. and Schwab, D. J. How noise affects the Hessian spectrum in overparameterized neural networks. *arXiv preprint arXiv:1910.00195*, 2019.

Xie, Z., Sato, I., and Sugiyama, M. A diffusion theory for deep learning dynamics: Stochastic gradient descent exponentially favors flat minima. *arXiv preprint arXiv:2002.03495*, 2020.

Yang, J., Sun, S., and Roy, D. M. Fast-rate PAC-Bayes generalization bounds via shifted Rademacher processes. *Advances in Neural Information Processing Systems*, 2019.

Yao, Z., Gholami, A., Keutzer, K., and Mahoney, M. W. Pyhessian: Neural networks through the lens of the Hessian. In *2020 IEEE International Conference on Big Data (Big Data)*, pp. 581–590, 2020.

Zhang, G., Li, L., Nado, Z., Martens, J., Sachdeva, S., Dahl, G., Shallue, C., and Grosse, R. B. Which algorithmic choices matter at which batch sizes? Insights from a noisy quadratic model. *Advances in neural information processing systems*, pp. 8196–8207, 2019.

Zhou, M., Liu, T., Li, Y., Lin, D., Zhou, E., and Zhao, T. Toward understanding the importance of noise in training neural networks. In *International Conference on Machine Learning*, pp. 7594–7602, 2019.

A. Computation of Conditional Variance

We know from Eq. (5) that the *conditional mean* of an Anti-PGD step is

$$\mathbb{E}\left[z_{n+1}|z_n\right] = z_n - \eta\nabla\tilde{L}(z_n) + O\left(\eta\mathbb{E}[\|\xi_n\|^3]\right), \tag{19}$$

where the *modified loss* $\tilde{L}$ is given by

$$\tilde{L}(z) := L(z) + \frac{\sigma^2}{2}\operatorname{Tr}(\nabla^2 L(z)). \tag{20}$$

This stands in contrast to a standard PGD step (Eq. (1)) whose conditional mean coincides with gradient descent (i.e. includes no implicit bias):

$$\mathbb{E}\left[w_{n+1}|w_n\right] = w_n - \eta\nabla L(w_n). \tag{21}$$

In this section, we additionally compute the conditional variance of Anti-PGD and PGD. We start with PGD. There, we obtain by a second-order Taylor expansion of $\partial_i L(\cdot)$ around z_n in Eq. (1) that

$$w^i_{n+1} = w^i_n - \eta\partial_i L(w_n) - \eta\sum_{j=1}\partial^2_{ij}L(w_n) - \frac{\eta}{2}\partial_i\sum_{j,k}\partial^2_{j,k}L(w_n) + \xi^i_{n+1}. \tag{22}$$

Hence, the conditional variance for PGD is simply

$$\operatorname{var}\left[w^{(i)}_{n+1}|w_n\right] = \operatorname{var}\left[\xi^i_{n+1}\right] = \sigma^2. \tag{23}$$

For Anti-PGD on the other hand, we compute

$$\begin{aligned}
\operatorname{var}\left[z^{(i)}_{n+1}|z_n\right] &\overset{Eq.\ (4)}{=} \mathbb{E}\Big[\Big[\eta\partial_i L(z_n) + \eta\sum_j\partial^2_{ij}L(z_n)\xi^j_n + \frac{\eta}{2}\sum_{j,k}\partial^3_{ijk}L(z_n)\xi^j_n\xi^k_n + O(\eta\|\xi_n\|^3)\Big]^2|z_n\Big] \\
&= \eta^2[\partial_i L(z_n)]^2 + \eta^2\sigma^2\sum_{j=1}^d[\partial^2_{ij}L(z_n)]^2 + \frac{\eta^2\sigma^4}{4}\sum_{j\neq k}\partial^3_{ijk}L(z_n) + \frac{\eta^2}{4}\sum_{j=1}^d\partial_{ijj}L(z_n)\mathbb{E}[(\xi^j_n)^4] \\
&\quad + \frac{\eta^2\sigma^4}{4}\sum_{j\neq k}[\partial_{ijj}L(z_n)]\cdot[\partial_{ikk}L(z_n)] + \eta^2 O(\mathbb{E}[\|\xi_n\|^6]) \\
&\quad + 2\Bigg[\Big(0 + 0 + \frac{\eta^2\sigma^2}{2}\sum_{j=1}^d\partial_{ijj}L(z_n) + \eta^2\partial_i L(z_n)O(\mathbb{E}[\|\xi_n\|^3])\Big) + \Big(0+0+0+0\Big) + \Big(0+0\Big) \\
&\qquad + \frac{\eta^2\sigma^2}{2}\left[\sum_{j=1}^d\partial_{ijj}L(z_n)\right]O(\mathbb{E}[\|\xi_n\|^3])\Bigg]
\end{aligned}$$

By rearranging the summands, we obtain

$$\begin{aligned}
&\operatorname{var}\left[z^{(i)}_{n+1}|z_n\right] = \\
&\eta^2[\partial_i L(z_n)]^2 + \eta^2\sigma^2\sum_{j=1}^d[\partial^2_{ij}L(z_n)]^2 + \frac{\eta^2\sigma^4}{4}\sum_{j\neq k}\left(\partial^3_{ijk}L(z_n) + [\partial_{ijj}L(z_n)]\cdot[\partial_{ikk}L(z_n)]\right) + \eta^2\sigma^2\sum_{j=1}^d\partial_{ijj}L(z_n) \\
&+ \left[\frac{1}{2}\partial_i L(z_n) + \frac{\sigma^2}{2}\sum_{j=1}^d\partial_{ijj}L(z_n)\right]O(\mathbb{E}[\eta^2\|\xi_n\|^3]) + \left[\frac{1}{4}\sum_{j=1}^d\partial_{ijj}L(z_n)\right]O(\mathbb{E}[\eta^2\|\xi_n\|^4]) + O(\eta^2\mathbb{E}[\|\xi_n\|^6]).
\end{aligned} \tag{24}$$

The above expression is the asymptotic (second order) expansion of the conditional variance of Anti-PGD, as $\sigma \to 0$. It consists of two parts: Its first line contains summands which come from the first two moments of the distribution of ξ. Its second line contains the summands from the third, fourth, and sixth moment of the distribution of ξ. The precise size of the conditional variance will therefore depend on the first six moments of the noise distribution. Without assuming more on the noise distribution, it is thus difficult to make further statements about the conditional variance. By eyeballing Eq. (24), it however seems likely that this variance is larger than the one from PGD; cf. (23).

B. Proof of Theorem 2.1

Theorem 2.1 (Convergence of the regularized gradients)**.** *Let $L : \mathbb{R}^d \to \mathbb{R}$ be lower bounded with continuous fourth-order partial derivatives and β-Lipschitz continuous third-order partial derivatives, for some constant $\beta > 0$. Consider the iterates $\{z_n\}_{n=0}^{N-1}$ computed by Anti-PGD as in (3) with $\eta \leq 1/\beta$, where for each n the noise coordinate ξ_i^n follows a symmetric centered Bernoulli distribution with variance σ^2 (i.e., σ and $-\sigma$ have probability $1/2$). Let $\epsilon > 0$. If we set $\eta = O(\epsilon/\sigma^3)$ and $N = \Theta(\epsilon^{-1})$, then it holds true that*

$$\mathbb{E}\left[\frac{1}{N}\sum_{n=0}^{N-1}\|\nabla\tilde{L}(z_n)\|^2\right] \leq \epsilon. \tag{7}$$

Proof. First, we observe that Eq. (4) implies – in the considered case of Bernoulli perturbations ξ_n with $(\xi_i^n)^2 = \sigma^2$ a.s. – that

$$z_{n+1}^i - z_n^i = \underbrace{-\partial_i\eta L(z_n) - \partial_i\eta\frac{\sigma^2}{2}\sum_j \partial_{jj}^2 L(z_n)}_{=\eta\partial_i\tilde{L}(z_n)\text{ (Regularized Gradient)}} \underbrace{-\eta\sum_j \partial_{ij}^2 L(z_n)\xi_n^j - \frac{\eta}{2}\partial_i\sum_{j\neq k}\partial_{jk}^2 L(z_n)\xi_n^j\xi_n^k}_{\text{Mean-zero Perturbation}} + \underbrace{O(\eta\sigma^3)}_{\text{Expansion error (small)}}. \tag{25}$$

Since we assumed that L has β-Lipschitz continuous third-order partial derivatives, Theorem 2.1.5 from Nesterov (2018) implies that $\tilde{L}$ is β-smooth. Hence, we have

$$\begin{aligned}\tilde{L}(z_{n+1}) &\leq \tilde{L}(z_n) + \langle\nabla\tilde{L}(z_n), z_{n+1} - z_n\rangle + \frac{\beta}{2}\|z_{n+1} - z_n\|^2 \\ &\overset{(25)}{=} \tilde{L}(z_n) - \eta\langle\nabla\tilde{L}(z_n), \nabla\tilde{L}(z_n) + B(z_n)\odot\xi_n + O(\sigma^3)\rangle + \frac{\beta\eta^2}{2}\|\nabla\tilde{L}(z_n) + B(z_n)\odot\xi_n + O(\sigma^3)\|^2,\end{aligned}$$

for some tensor B which captures all summands from the mean-zero perturbation of Eq. (25). Taking the expectation, most of the terms cancel out and we get

$$\mathbb{E}[\tilde{L}(z_{n+1})] \leq \mathbb{E}[\tilde{L}(z_n)] - \left(\eta - \frac{\beta\eta^2}{2}\right)\mathbb{E}[\|\nabla\tilde{L}(z_n)\|^2] + O(\eta^2\sigma^2) + O(\eta\sigma^3). \tag{26}$$

Using the assumption $\eta \leq \frac{1}{\beta}$, this implies

$$\mathbb{E}[\|\nabla\tilde{L}(z_n)\|^2] \leq 2\beta\mathbb{E}[\tilde{L}(z_n)] - 2\beta\mathbb{E}[\tilde{L}(z_{n+1})] + O(\eta^2\sigma^2) + O(\eta\sigma^3). \tag{27}$$

From this it follows that

$$\mathbb{E}\left[\frac{1}{N}\sum_{n=0}^{N-1}\|\nabla\tilde{L}(z_n)\|^2\right] \leq \frac{\beta[\tilde{L}(z_0) - \mathbb{E}\tilde{L}(z_N)]}{N} + O(\eta^2\sigma^2) + O(\eta\sigma^3) \tag{28}$$

$$= O(N^{-1}) + O(\eta^2\sigma^2) + O(\eta\sigma^3) \tag{29}$$

Note that $\tilde{L}(z_N) \geq \tilde{L}^* = \min_z \tilde{L}(z) > -\infty$. This is because L is lower bounded and also $tr(\nabla^2 L)$ is lower bounded since we assumed L has Lipschitz gradients. Finally, exploiting the assumed $\eta = O(\epsilon/\sigma^3)$ and $N = \Theta(\epsilon^{-1})$ concludes the proof. □

C. Proof of Theorem 3.1

To proof Theorem 3.1, we first need some preliminary preparation; in doing so, we will also provide some intuition for the reader. The main proof follows afterwards, in §C.2.

Consider the problem of minimizing the cost function

$$L(u, v) = \frac{1}{2}v^2\|u\|^2, \tag{30}$$

where $\|\cdot\|$ is the Euclidean norm, $v \in \mathbb{R}$, and $u \in \mathbb{R}^d$. To minimize the loss, we use perturbed gradient descent (PGD), with noise injection. Note that any point where $v = 0$ or $\|u\|^2 = 0$ minimizes the loss. By the considerations in the section above, we want to find a solution (u, v) where $\|u\|$ is small (i.e. a solution with low curvature). We show that, while standard noise injection does not necessarily induce this bias on the dynamics, injection of anticorrelated noise does. This show that anticorrelated noise effectively minimizes the trace of the Hessian:

$$\mathrm{Tr}(\nabla^2 L(u, v)) = dv^2 + \|u\|^2. \tag{31}$$

Preliminary considerations. Let us start by writing down the update in discrete-time. Recall that the gradient is $(v^2 u, \|u\|^2 v)$, hence:

$$u_{k+1} = (1 - \eta v_k^2) \cdot u_k + \varepsilon_k^u \tag{32}$$

$$v_{k+1} = (1 - \eta \|u_k\|^2) \cdot v_k + \varepsilon_k^v \tag{33}$$

where $\varepsilon_k^u \in \mathbb{R}^d$ and $\varepsilon_k^v \in \mathbb{R}$ are the noise variables.

1. For stability (in the noiseless setting), we need $\eta \leq \frac{2}{\max\{v_k^2, \|u_k\|^2\}}$.
2. Starting from a big $\|u\|$ and any v, under noiseless GD, since $d \gg 1$, we converge to $(u_0, 0)$, with $\|u_0\| := D \gg 1$

The key to the proof of effectiveness of anticorrelated noise, compared to uncorrelated noise, relies on the following observation:

Empirical Observation: for the widening valley $L(u, v) = \frac{1}{2} v^2 \|u\|^2$, if we only perturb the v coordinate with *any noise* then we get to a wide minimum.

Why? Intuition behind the proof. Well, of course this is the case! If one knows is advance which direction to move in order to pick up a signal, then *les jeux sont faits*. The problem is that in order to perturb this direction — we need to perturb *all directions*, and this leads to "getting lost" if the noise is not controlled (i.e. does not have an attraction force to the origin). This also motivates why the effect gets more intense as the dimension d increases: there is a lot of bias added, which drives us away from good minima.

Plan: To show the result, we follow the following procedure:

1. Starting from $(u_0, 0)$, we start injecting noise and want to reach $(\tilde{u}, \tilde{v})$ such that $\|\tilde{u}\|^2 = \alpha D$ and $0 < \alpha < 1$. We want to show here a difference in behavior under different noise correlation.
2. We proceed by contradiction: starting from $(\tilde{u}, \tilde{v})$, we assume that $\|u_k\|^2 \geq \alpha D$ for all $k \geq 0$ ($\alpha \in (0, 1)$). Under injection of anticorrelated noise, we show that *this leads to a contradiction* — i.e. that the dynamics substantially decreases the trace of the Hessian: $\|u_\infty\|^2 < \alpha D$ (worst-case upper bound). Crucially, we also show that the hypothesis does not lead to any contradiction under standard noise injection — i.e. without anticorrelation we do not significantly decrease the trace of the Hessian. More specifically, we show that $\lim_{n\to\infty} \mathbb{E}\left[\|u_n\|^2\right] \geq D/\alpha$ under uncorrelated noise injection (worst-case lower bound).
3. To simplify the computations, assume that coordinate-wise the noise is a result of a Bernoulli$(1/2)$ perturbation $(\xi_k)_i \in \{-\sigma, \sigma\}$. The injected noise is then either $\varepsilon_k = \xi_k$ or $\varepsilon_k = \xi_k - \xi_{k-1}$, for PGD and Anti-PGD respectively.

C.1. Some Useful Lemmata

This section is pretty technical, hence the reader can skip the proof on a first read. *The meaning behind the bounds we derive and a numerical verification can be found in Figure 7.*

We start by recalling the variation of constants formula, which we will heavily use along the proof. We also extend this to the anticorrelated setting.

Lemma C.1 (Variations of constants formula)**.** *Let $w \in \mathbb{R}^d$ evolve with time-varying linear dynamics $w_{k+1} = A_k w_k + \varepsilon_k$, where $A_k \in \mathbb{R}^{d\times d}$ and $\varepsilon_k \in \mathbb{R}^d$ for all k. Then, with the convention that $\prod_{j=k+1}^{k} A_j = 1$,*

$$w_{k+1} = \left(\prod_{j=0}^{k} A_j\right) w_0 + \sum_{i=0}^{k} \left(\prod_{j=i+1}^{k} A_j\right) \varepsilon_i. \tag{34}$$

Proof. For $k = 1$ we get $w_1 = A_0 w_0 + \varepsilon_0$. The induction step yields

$$w_{k+1} = A_k \left(\left(\prod_{j=0}^{k-1} A_j\right) w_0 + \sum_{i=0}^{k-1} \left(\prod_{j=i+1}^{k-1} A_j\right) \varepsilon_i \right) + \varepsilon_k. \tag{35}$$

$$= \left(\prod_{j=0}^{k} A_j\right) w_0 + \sum_{i=0}^{k-1} A_k \left(\prod_{j=i+1}^{k-1} A_j\right) \varepsilon_i + \varepsilon_k. \tag{36}$$

$$= \left(\prod_{j=0}^{k} A_j\right) w_0 + \sum_{i=0}^{k-1} \left(\prod_{j=i+1}^{k} A_j\right) \varepsilon_i + \left(\prod_{j=k+1}^{k} A_j\right) \varepsilon_k. \tag{37}$$

$$= \left(\prod_{j=0}^{k} A_j\right) w_0 + \sum_{i=0}^{k} \left(\prod_{j=i+1}^{k} A_j\right) \varepsilon_i. \tag{38}$$

This completes the proof of the variations of constants formula. □

We extend this formula to the anticorrelated case, where ε_k has some additional structure.

Corollary C.2 (Anticorrelated variations of constants formula)**.** *Under the same setting of Lemma C.1, if there exist a family of vectors $\{\xi_k\}$ such that $\varepsilon_0 = \xi_0$ and $\varepsilon_k = \xi_k - \xi_{k-1}$, then*

$$w_{k+1} = \left(\prod_{j=0}^{k} A_j\right) w_0 + \xi_k + \sum_{i=0}^{k-1} (A_{i+1} - I) \left(\prod_{j=i+2}^{k} A_j\right) \xi_i. \tag{39}$$

Proof. We have, by direct computation:

$$w_{k+1} = \left(\prod_{j=0}^{k} A_j\right) w_0 + \left(\prod_{j=1}^{k} A_j\right) \xi_0 + \sum_{i=1}^{k} \left(\prod_{j=i+1}^{k} A_j\right) \xi_i - \sum_{i=1}^{k} \left(\prod_{j=i+1}^{k} A_j\right) \xi_{i-1} \tag{40}$$

$$= \left(\prod_{j=0}^{k} A_j\right) w_0 + \left(\prod_{j=1}^{k} A_j\right) \xi_0 + \sum_{i=1}^{k-1} \left(\prod_{j=i+1}^{k} A_j\right) \xi_i + \xi_k - \sum_{i=0}^{k-1} \left(\prod_{j=i+2}^{k} A_j\right) \xi_i \tag{41}$$

$$= \left(\prod_{j=0}^{k} A_j\right) w_0 + \xi_k + \sum_{i=0}^{k-1} \left(\prod_{j=i+1}^{k} A_j\right) \xi_i - \sum_{i=0}^{k-1} \left(\prod_{j=i+2}^{k} A_j\right) \xi_i \tag{42}$$

$$= \left(\prod_{j=0}^{k} A_j\right) w_0 + \xi_k + \sum_{i=0}^{k-1} \left[\left(\prod_{j=i+1}^{k} A_j\right) - \left(\prod_{j=i+2}^{k} A_j\right) \right] \xi_i. \tag{43}$$

□

Remark C.3. If $A_i = I$ for all i, then the last summand is zero. This showcases the effect of anticorrelation: noise cancellation under noise accumulation.

C.1.1. Expectation Quantities under Deterministic ρ_k

Using the variation of constants formula, we can write the dynamics of the second moment of stochastic linear time-varying dynamical systems, with either standard or anticorrelated noise.

Proposition C.4 (An Itô-like formula)**.** *Let $w \in \mathbb{R}^d$ evolve with time-varying linear dynamics $w_{k+1} = A_k w_k + \varepsilon_k$, where $A_k \in \mathbb{R}^{d\times d}$ and $\varepsilon_k \in \mathbb{R}^d$ for all k. Let $\{\xi_k\}$ be a family of uncorrelated zero-mean d-dimensional random variables with variance $\mathbb{E}[\|\xi_k\|^2] = d\sigma^2$ (dependency on the dimension because additivity of squared norm). Consider $\varepsilon_0 = \xi_0$ and $\varepsilon_k = \xi_k - \xi_{k-1}$ for all $k \geq 1$. Further, assume that $A_k = \rho_k I$ for all k (i.e. A_k is a multiple of the identity), with $\rho_k \in \mathbb{R}$ a deterministic quantity. Then, with the convention that $\prod_{j=k+1}^{k} A_j = 1$, we have*

$$\mathbb{E}[\|w_{k+1}\|^2] = \left(\prod_{j=0}^{k} \rho_j^2\right) \|w_0\|^2 + \left(1 + \sum_{i=0}^{k-1} \left[(1-\rho_{i+1})^2 \prod_{j=i+2}^{k} \rho_j^2\right]\right) d\sigma^2. \tag{44}$$

Instead, if $\varepsilon_k = \xi_k$ for all k (standard noise injection) we have

$$\mathbb{E}[\|w_{k+1}\|^2] = \left(\prod_{j=0}^{k} \rho_j^2\right) \|w_0\|^2 + \sum_{i=0}^{k} \left(\prod_{j=i+1}^{k} \rho_j^2\right) d\sigma^2. \tag{45}$$

Proof. Using independence of the $\{\xi_k\}$ family, we obtain for the anticorrelated case:

$$\mathbb{E}[w_{k+1}^\top w_{k+1}] = \left(\prod_{j=0}^{k} \rho_j\right)^2 \|w_0\|^2 + \mathbb{E}[\|\xi_k\|^2] + \sum_{i=0}^{k-1} (\rho_{i+1}-1)^2 \left(\prod_{j=i+2}^{k} \rho_j\right)^2 \mathbb{E}[\|\xi_i\|^2] \tag{46}$$

$$= \left(\prod_{j=0}^{k} \rho_j\right)^2 \|w_0\|^2 + \sigma^2 + \sum_{i=0}^{k-1} (\rho_{i+1}-1)^2 \left(\prod_{j=i+2}^{k} \rho_j^2\right) \sigma^2, \tag{47}$$

where we used the fact that the ξ_k are not correlated. The case $\varepsilon_k = \xi_k$ is similar and therefore left to the reader. □

Corollary C.5. *In the setting of Proposition C.4, assume $\rho_j = \rho \in (0,1)$ is constant for all j. If $\varepsilon_0 = \xi_0$ and $\varepsilon_k = \xi_k - \xi_{k-1}$ for all $k \geq 1$ then*

$$\mathbb{E}[\|w_{k+1}\|^2] = \rho^{2(k+1)}\|w_0\|^2 + \left(1 + \frac{(1-\rho)^2(1-\rho^{2(k+1)})}{1-\rho^2}\right) d\sigma^2 \quad \xrightarrow{\infty} \quad \frac{2}{1+\rho} d\sigma^2. \tag{48}$$

Instead, if $\varepsilon_k = \xi_k$ for all k (standard noise injection) we have

$$\mathbb{E}[\|w_{k+1}\|^2] = \rho^{2(k+1)}\|w_0\|^2 + \frac{1-\rho^{2(k+1)}}{1-\rho^2} d\sigma^2 \quad \xrightarrow{\infty} \quad \frac{1}{1-\rho^2} d\sigma^2. \tag{49}$$

Proof. Simple application of the formula for geometric series. Numerical verification in Figure 7. □

Remark C.6. Note that the corollary has a clear interpretation: if ρ is between zero and one, we experience striking difference between uncorrelated and anticorrelated noise. If ρ increases, the total accumulated anticorrelated noise decreases.[1] This trend is reversed for normal noise injection: as $\rho \to 1$ the total accumulated variance explodes. Numerical verification can be found in Figure 7.

[1] $\frac{2}{1+\rho}$ is a decreasing function or ρ, while $1/(1-\rho^2)$ is increasing.

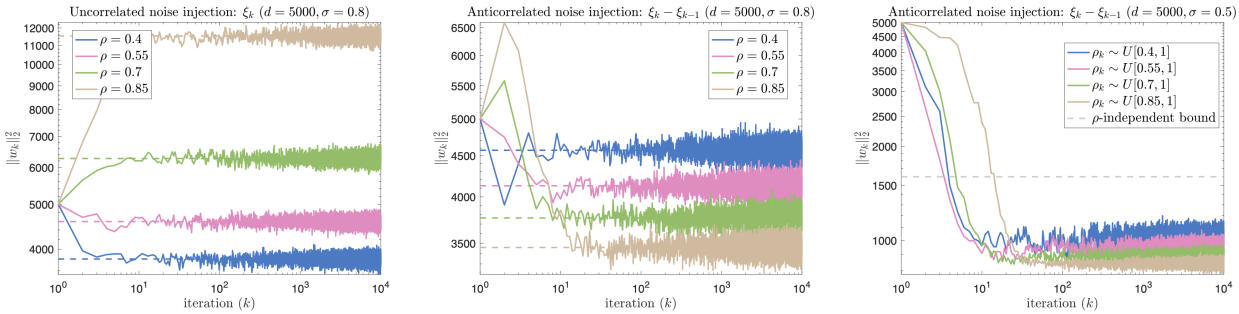

Figure 7. Numerical verification of our final result that we will use in the proof, i.e. Corollary C.5 (first and second panel) and Proposition C.7 (last panel). The dashed lines indicate our predicted value (in expectation) by the theory. In the right-most plot, we sample ρ_k at each iteration uniformly on an interval.

C.1.2. Dealing with Potential Stochasticity in the ρ_k

For the proof in the next subsection, we need to deal with stochastic ρ_k, which are only specified up to an interval.

Proposition C.7 (Limit bound on second moment for anticorrelated noise)**.** *Let $w \in \mathbb{R}^d$ evolve with time-varying linear dynamics $w_{k+1} = A_k w_k + \varepsilon_k$, where $A_k \in \mathbb{R}^{d\times d}$ and $\varepsilon_k \in \mathbb{R}^d$ for all k. Let $\{\xi_k\}$ be a family of uncorrelated zero-mean d-dimensional random variables with variance $\mathbb{E}[\|\xi_k\|^2] = d\sigma^2$ (dependency on the dimension because additivity of squared norm). Consider $\varepsilon_0 = \xi_0$ and $\varepsilon_k = \xi_k - \xi_{k-1}$ for all $k \geq 1$. Further, assume that $A_k = \rho_k I$ for all k (i.e. A_k is a multiple of the identity) and that $\rho_k \in [0, 1]$ for all k. Assume that the probability of $\rho_k < 1$ is non-zero, i.e. that $\rho_k \neq 1$ with non-vanishing probability. Then, we have*

$$\lim_{k\to\infty} \mathbb{E}[\|w_{k+1}\|^2] \leq 2d\sigma^2. \tag{50}$$

Proof. The proof is based on an induction argument, starting from the equation in Proposition C.4:

$$\mathbb{E}[\|w_{k+1}\|^2] = \left(\prod_{j=0}^{k} \rho_j^2\right) \|w_0\|^2 + \left(1 + \sum_{i=0}^{k-1} \left[(1-\rho_{i+1})^2 \prod_{j=i+2}^{k} \rho_j^2\right]\right) d\sigma^2. \tag{51}$$

First, note that by assumption on ρ_k the first term vanishes as $k \to \infty$. We just have to deal with the second term. Specifically, we want to show that for whatever sequence $\rho_k \in (0, 1)$ we have

$$\nu_k = \sum_{i=0}^{k-1} (1-\rho_{i+1})^2 \left(\prod_{j=i+2}^{k} \rho_j^2\right) \leq 1, \quad \forall k \geq 0. \tag{52}$$

A fundamental observation, is that the term can be written in a recursive form. Indeed,

$$\begin{aligned}
\nu_k &= \sum_{i=0}^{k-1} \left[(1-\rho_{i+1})^2 \left(\prod_{j=i+2}^{k} \rho_j^2\right)\right] && (53)\\
&= \sum_{i=0}^{k-2} \left[(1-\rho_{i+1})^2 \left(\prod_{j=i+2}^{k} \rho_j^2\right)\right] + (1-\rho_k)^2 && (54)\\
&= \rho_k \sum_{i=0}^{k-2} \left[(1-\rho_{i+1})^2 \left(\prod_{j=i+2}^{k-1} \rho_j^2\right)\right] + (1-\rho_k)^2 && (55)\\
&= \rho_k^2 \nu_{k-1} + (1-\rho_k)^2, && (56)
\end{aligned}$$

where the second equality follows from the fact that, as previously noted, our notation implies $\prod_{j=k+1}^{k} \rho_k^2 = 1$, for all k. Let us now proceed again by induction to show that $v_k \in (0, 1)$ for all $k \geq 0$. Note that trivially $v_0 = 0$. Let's proceed with the inductive step:

$$\nu_k = \rho_k^2 \nu_{k-1} + (1-\rho_k)^2 = (\nu_{k-1} + 1)\rho_k^2 - 2\rho_k + 1. \tag{57}$$

This quantity is less then one if and only if

$$(\nu_{k-1}+1)\rho_k^2 \leq 2\rho_k. \tag{58}$$

Note that this is satisfied since $\nu_{k-1}+1 \leq 2$, and $\rho_k^2 \leq \rho_k$ since $\rho_k \in (0,1)$. The result follows. □

A numerical verification of this result can be found in Figure 7.

C.2. Proof of the Main Result

Using the results from the last subsection, we are now ready to show the main theorem for optimization of the widening valley under noise injection.

Theorem 3.1 (Widening Valley)**.** *Let $L : \mathbb{R}^{d+1} \to \mathbb{R}$ be the* widening valley *loss from Eq. (10). We start optimizing from a point $w_0 = (u_0, 0)$, where $\|u_0\|^2 = D \gg 1$ (e.g. the solution found by gradient descent), around which we consider the domain $\mathcal{D}_\alpha := \{(u,v) \in \mathbb{R}^{d+1} : \|u\|^2 \in (\alpha D, D/\alpha)\}$ for some fixed $\alpha \in (0,1)$. We want to compare the long-term stochastic dynamics of PGD and Anti-PGD, as defined in Eqs. (1) and (2), in terms of where they exit $\mathcal{D}_\alpha$. As a noise model, we assume that the i.i.d. perturbations ξ_n are distributed according to a symmetric centered Bernoulli distribution (i.e., σ and $-\sigma$ have probability $1/2$) whose variance σ^2 is upper bounded by $\sigma^2 \in \left(0, \min\left\{\frac{\alpha^3 D}{2}, \frac{D}{8\alpha}\right\}\right]$. As a step size, we set $\eta = \frac{\alpha}{2D}$ which, for both methods, leads to stable dynamics inside of $\mathcal{D}_\alpha$. We find that (on average) PGD and Anti-PGD exit through different sides of $\mathcal{D}_\alpha$:*

1. ***In high dimensions, PGD diverges away from zero.*** *If $d \geq \frac{2}{\alpha^2}$, then it holds for any admissible σ^2 that*

$$\lim_{n\to\infty} \mathbb{E}\left[\|u_n\|^2\right] \geq D/\alpha, \tag{13}$$

   where u_n are the first d coordinates of w_n computed by PGD as in (1).

2. ***Independent of dimensions, Anti-PGD goes to zero.*** *For any $d \in \mathbb{N}$, if we choose any admissible σ^2 such that $\sigma^2 \leq \frac{\alpha D}{2d}$, then*

$$\lim_{n\to\infty} \mathbb{E}\left[\|u_n\|^2\right] \leq \alpha D, \tag{14}$$

   where u_n are the first d coordinates of w_n, computed by Anti-PGD as in (2).

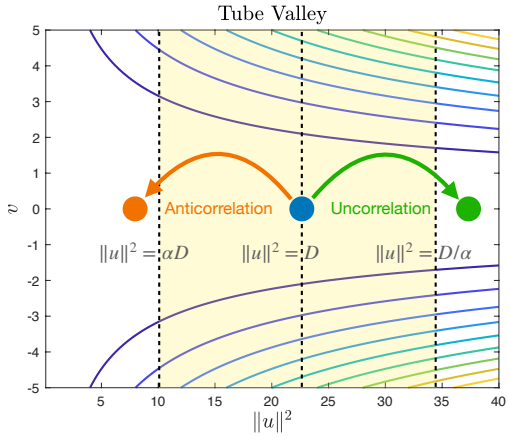

Figure 8. The sketch on the left illustrates the intuition behind the result in 3.1.

Proof. As above, we denote the perturbations by ε_k; i.e., $\varepsilon_k = \xi_k$ for PGD and $\varepsilon_k = \xi_{k+1} - \xi_k$ for Anti-PGD. Let us start by inspecting the equation

$$u_{k+1} = (1 - \eta v_k^2) \cdot u_k + \varepsilon_k^u, \tag{59}$$

where $\varepsilon_k^u \in \mathbb{R}^d$ is the projection of the noise ε_k to the first d coordinates. It is clear that the optimal strategy of making $\|w\|$ small is to increase $|v|$, so to sample nearby points and pick up the gradient. The greater v is in norm, the better. We can increase the norm of v by heavy noise injection (second equation). However, too much noise also increases ε_k^u, which acts adversarially to the decrease of $\|w\|$ (error accumulation increases the Euclidean norm in expectation).

Choice of stepsize and operating region. We start by motivating the choice of stepsize $\eta = \frac{\alpha}{2D}$. Starting from the point $(u_0, 0)$ with $\|u_0\|^2 = D > 0$, we consider the operating landscape region $\alpha D < \|u_k\|^2 < D/\alpha$, with $\alpha \in (0, 1)$. We want to show that while standard noise injection makes the process exit the region from the right (D/α side, see Figure 8), anticorrelated noise injection makes the process exit the region from the left (αD side). In this region, named $\mathcal{D}_\alpha$, the maximal allowed learning rate is $\eta \leq \frac{2}{\max_{\mathcal{D}_\alpha}\{v^2, \|u\|^2\}}$. Since v stays small (we are going to check this later in great detail), we select the stepsize $\eta \leq \frac{1}{2\max_{\mathcal{D}_\alpha}\|u\|^2} = \frac{\alpha}{2D}$ — which guarantees stability in expectaction, i.e. without noise injection (even allowing for some slack).

Lower bound for uncorrelated noise ($\boldsymbol{\varepsilon_k = \xi_k}$). For this case, we have to show that standard noise injection cannot possibly work for reaching αD, therefore we have to put ourselves in *the best case scenario* for PGD: that is, we have to provide an uniform upper bound for v_k under the second equation (i.e. the equation for v) and show that this is not enough for a substantial decrease in $\|u\|$. In the next paragraph (anticorrelated noise), we instead have to put ourselves in the *worst case scenario* — i.e. a lower bound for $|v|$— and show that this is still enough for anticorrelated noise to yield a substantial decrease in $\|u\|$.

To start, let us then look at the second equation:

$$v_{k+1} = (1 - \eta\|u_k\|^2) \cdot v_k + \varepsilon_k^v, \tag{60}$$

where ε_k^v is the $(d+1)$-th component of ε_k. Since we start from $v_0 = 0$, the equation is completely dominated by noise, and is strongly mean reverting (i.e. v is effectively bounded). Indeed, since $\eta = \frac{\alpha}{2D}$ and $\|u_k\|^2 \in (\alpha D, D/\alpha)$ by assumption, we have

$$|v_{k+1}| \leq \max\left\{1 - \frac{\alpha^2}{2}, \frac{1}{2}\right\} \cdot |v_k| + \sigma = \left(1 - \frac{\alpha^2}{2}\right)|v_k| + \sigma. \tag{61}$$

where we used the fact that $|\varepsilon_k^v| = \sigma$ and that $\alpha^2 \in (0, 1)$. By induction, the last inequality yields that, starting from $v_0 = 0$, we have

$$|v_k| \leq v_{\max} := \frac{2\sigma}{\alpha^2}, \quad \forall k \geq 0. \tag{62}$$

Hence, we found the "best case scenario" for the w equation: $|v_k| = \frac{2\sigma}{\alpha^2}$, for all k. This gives w the best decrease rate possible.[2] However, we need to check this value $v_{\max}$ is such that the equation for w is indeed stable (we promised this to the reader in the last paragraph). We recall that this equation is $w_{k+1} = (1 - \eta v_k^2) \cdot w_k + \varepsilon_k^u$. Let us require $(1 - \eta v_k^2) \in (0, 1)$, for this we need $1/v_{\max}^2 > \eta = \frac{\alpha}{2D}$. Therefore, we need

$$\frac{1}{v_{\max}^2} = \frac{\alpha^4}{4\sigma^2} \geq \eta = \frac{\alpha}{2D} \quad \implies \quad \sigma^2 \leq \alpha^3 D/2. \tag{63}$$

This is guaranteed by assumption. To proceed, we substitute $v_{\max}$ into the first equation to get

$$u_{k+1} = (1 - \eta\frac{4\sigma^2}{\alpha^4}) \cdot u_k + \varepsilon_k^u = \frac{\alpha^3 D - 2\sigma^2}{\alpha^3 D} u_k + \varepsilon_k^u. \tag{64}$$

Let us call $\rho := \left(\frac{\alpha^3 D - 2\sigma^2}{\alpha^3 D}\right) \in (0, 1)$ the (best case) shrinking factor. Since ε_i^u is zero-mean, computing the expected value of $\|u_k\|^2$ leads to the following limit by Corollary C.5:

$$\lim_{k\to\infty} \mathbb{E}[\|u_k\|^2] = \frac{d\sigma^2}{1 - \rho^2} = \frac{dD^2\alpha^6}{2D\alpha^3 - 2\sigma^2}. \tag{65}$$

where we assumed σ^2 strictly positive. Note that this limit is a monotonically increasing function of $\sigma^2 \in (0, D\alpha^3/2)$. Hence, we get

$$\lim_{k\to\infty} \mathbb{E}[\|u_k\|^2] \in \left(\frac{dD\alpha^3}{2}, dD\alpha^3\right) \tag{66}$$

[2] Note that noise injection in u is independent of v, therefore to minimize $\|u\|$ we need the shrinking factor to be as large as possible. We note that using this bound is precise: with probability one we are in the best scenario (we are finding a lower bound).

Remark C.8 (Phase transition). Note that, for σ exactly 0, the limit is instead $\|u_0\|^2 = D$. Instead, for any small noise the process will grow up until at least $dD\alpha^2/2$. This might seem weird at first — but recall that there is an interaction between noise scale and our best-case scenario bound for v: they both depend on σ. This causes a cancellation effect and a transition in behavior at $\sigma = 0$.

Last, we need to show that this lower bound on $\lim_{k\to\infty} \mathbb{E}[\|u_k\|^2]$ coincides with (or is bigger than) the right boundary of the operating region in Figure 8. To do this we set:

$$\frac{dD\alpha^3}{2} \geq \frac{D}{\alpha} \quad \Longrightarrow \quad d \geq \frac{2}{\alpha^4}. \tag{67}$$

This concludes the proof of Eq. (13).

Upper bound for anticorrelated noise ($\boldsymbol{\varepsilon_k = \xi_k - \xi_{k-1}}$). We consider anticorrelated noise injection $(\xi_k)_i \in \{-\sigma, \sigma\}$, and $\varepsilon_k = (\varepsilon_k^u, \varepsilon_k^v) = \xi_k - \xi_{k-1}$. Again, let us first look at the second equation:

$$v_{k+1} = (1 - \eta\|u_k\|^2) \cdot v_k + \varepsilon_k^v. \tag{68}$$

Since by hypothesis $\eta = \frac{\alpha}{2D}$ and $\alpha D \leq \|u_k\|^2 \leq D/\alpha$, we have $(1 - \eta\|u_k\|^2) \in \left(1 - \frac{\alpha^2}{2}, \frac{1}{2}\right)$. Clearly, we have that, for any $k \geq 1$ v_k^2, is non-zero with a non-vanishing probability. For (noiseless) stability, we also need an upper bound on $|v_k|$. An easy (yet absolutely not tight) upper bound is the following:

$$|v_{k+1}| \leq \frac{1}{2}|v_k| + 2\sigma\varepsilon_k^v. \tag{69}$$

where we simply used the absolute value subadditivity and the fact that $|\varepsilon_k^v| \leq |\xi_k| + |\xi_{k-1}| = 2\sigma$. Note that the equation directly yields by induction $|v_k| \leq 4\sigma$ for all $k \geq 0$.

Let us now deal with the equation for w.

$$u_{k+1} = (1 - \eta v_k^2) \cdot u_k + \varepsilon_k^u \tag{70}$$

For this equation, we would want all the coefficients $\rho_k := 1 - \eta v_k^2$ to be between 0 and 1 — i.e. we need to check that v is indeed not too big. Since $|v_k| \leq 4\sigma$ for all $k \geq 0$, we have the requirement $1 - \frac{\alpha}{2D} 16\sigma^2 > 0$, which implies $\sigma^2 \leq \frac{D}{8\alpha}$ — that satisfies our hypothesis.

So, to sum it up, we are in operating regime of Proposition C.7: anticorrelated noise, $\rho_k < 1$ with non-vanishing probability and ρ_k always between 0 and 1. Hence, we get that

$$\lim_{k\to\infty} \mathbb{E}[\|u_k\|^2] \leq 2\sigma^2 d. \tag{71}$$

Hence, for σ^2 small enough, the value αD is reached. This directly implies the missing Eq. (14). The proof is thereby complete. □

C.3. Proof of Corollary 3.1

Corollary C.9 (The trace of the Hessian in the widening valley)**.** *In the same setting as Thm. 3.1, let* $\eta = \frac{\alpha}{2D}$, $\sigma^2 \in \left(0, \min\left\{\frac{\alpha^3 D}{2}, \frac{D}{8\alpha}, \frac{\alpha D}{2d}\right\}\right]$ *and* $d \geq \frac{2}{\alpha^2}$. *If* $\alpha \ll 1$, *then*

$$\lim_{n\to\infty} \mathbb{E}[\mathrm{Tr}(\nabla^2 L(w_n^{\mathrm{anti}}))] \leq 16\alpha D \ll \mathbb{E}[\mathrm{Tr}(\nabla^2 L(w_0))]$$
$$\lim_{n\to\infty} \mathbb{E}[\mathrm{Tr}(\nabla^2 L(w_n^{un}))] \geq D/\alpha \gg \mathbb{E}[\mathrm{Tr}(\nabla^2 L(w_0))],$$

where $w_n^{un} = (u_n, v_n)$ *and* $w_n^{\mathrm{anti}} = (u_n, v_n)$ *are the weights computed by Anti-PGD and PGD respectively.*

Proof. Recall from Eq. (12) that $\mathrm{Tr}(\nabla^2 L(u, v)) = dv^2 + \|u\|^2$.

For the first two inequalities in the corollary, recall from Eq. (69) of the proof of Theorem 3.1 that $|v_n| \leq 4\sigma$ for all n, almost surely. The first two inequalities in the Corollary follow now by Eq. (14).

For the last two inequalities in the Corollary, we lower bound the trace by $\|u\|^2$ and make use of Eq. (13). □

D. Additional Experimental Evidence

D.1. Details for Figure 2

- **Squared regression**: Problem definition, loss function, gradient and Hessian provided in §D.2. We run GD, PGD and anti-PGD with full batch (40 datapoints in 100 dimensions), while for SGD we select a batch size of 1. All algorithms except SGD run with a constant learning rate of $\eta = 0.1$. For SGD, to improve generalization at such a small batch size, we instead select a slightly smaller learning rate $\eta = 0.01$. Perturbations in PGD and anti-PGD have parameter $\sigma = 0.05$. Findings are robust to changing these hyperparameters, as shown in § D.2. All plots also show one standard deviation for all measures.

- **Matrix sensing**: Problem definition, loss function, gradient and Hessian provided in §D.3. We run GD, PGD and anti-PGD with full batch (100 datapoints in 400 dimensions), while for SGD we select a batch size of 10. All algorithms run with a constant learning rate of $\eta = 0.001$. Perturbations in PGD and anti-PGD have parameter $\sigma = 0.1$. Findings are robust to changing these hyperparameters, as shown in § D.3. For better visualization, here all plots also show two standard deviation for all measures.

- **ResNet on CIFAR10.** Details in §4. Further supporting experiments in §D.4.

D.2. Quadratically Parametrized Model

Problem Definition. Consider the standard linear regression setting in d dimensions with M datapoints. The design matrix is $X \in \mathbb{R}^{M\times d}$. We assume there exist sparse (we also study the effect of sparseness) vector $w \in \mathbb{R}^d$ such that targets y are perfectly predicted as $y = Xw^{\odot 2}$, where $w^{\odot 2} \in \mathbb{R}^d$ is the element-wise product of w. This parametrization, also studied in (HaoChen et al., 2021; Blanc et al., 2020) in the context of label noise, makes the landscape highly non-linear, with minimizers that achieve different generalization properties.

For this loss we have

$$L(w) = \frac{1}{4n}\|Xw^{\odot 2} - y\|^2, \quad \nabla L(w) = \frac{1}{n}\cdot[X^T(Xw^{\odot 2} - y)] \odot w. \tag{72}$$

$$\nabla^2 L(w) = \frac{2}{n}\cdot \text{diag}(w)\cdot X^T X \cdot \text{diag}(w) + \frac{1}{n}\cdot \text{diag}(X^T(Xw^{\odot 2} - y)) \tag{73}$$

Precise setting. As in (HaoChen et al., 2021), we consider the case $d = 100$ and $M = 40$, and generate X_{train} at random. We consider $w^* = (1, 1, \ldots, 1, 0, 0, \ldots, 0)$ — where only 10 elements are non-zero, and generate $y = X_{\text{train}}(w^* \odot w^*)$. For testing, we use instead 100 datapoints. We test three different learning rates in Figure 9, 10, 11, and for each 3 values of noise injection variance. For SGD, since we choose a very small (i.e. unit) batch size, the learning rate is scaled down by a factor of 10, to provide stability.

Findings. We found that anti-PGD always provides the best test accuracy, and minimizes the trace of the Hessian as well. This finding is quite robust in terms of hyperparameter tuning. Further, we found that the performance is always drastically different from the one of PGD. An explanation of this phenomenon is provided in Theorem 3.1, in the main paper. Mini-batch SGD improves the final test loss if the stepsize is small enough. Larger stepsizes are unstable for SGD.

D.3. Matrix Sensing

Problem Definition. This setting is inspired by the experiment of (Blanc et al., 2020) on label noise. Let X^* be an unknown rank-r symmetric positive semidefinite (PSD) matrix in $\mathbb{R}^{n\times n}$ that we aim to recover. Assume this has unit 2-norm. Let $A_1, \ldots, A_M \in \mathbb{R}^{n\times n}$ be M given (wlog) symmetric measurement matrices. We assume that the label vector $y \in \mathbb{R}^M$ is generated by linear measurements $y_i = \langle A_i^\top, X^*\rangle = tr(A_i^T X^*)$. We want to minimize the loss

$$L(U) = \frac{1}{M}\sum_{i=1}^{M} L_i(U), \quad L_i(U) = \frac{1}{2}(y_i - \langle A_i, UU^\top\rangle)^2, \tag{74}$$

where $U \in \mathbb{R}^{n\times n}$ in general achieves a good test accuracy if has small rank.

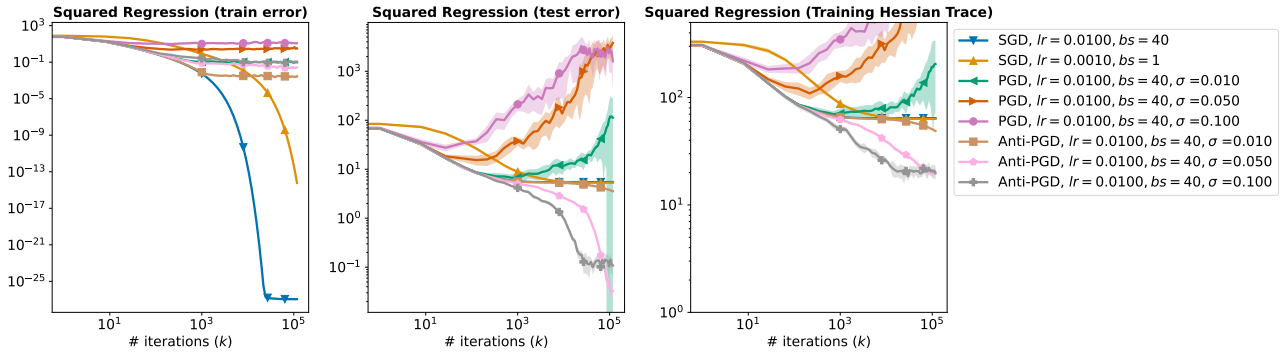

Figure 9. Performance of anti-PGD on **quadratically parametrized linear regression**, for **low learning rate** and different values of noise injection standard deviation. Plotted is also the error bar relative to 1 standard deviation (10 runs).

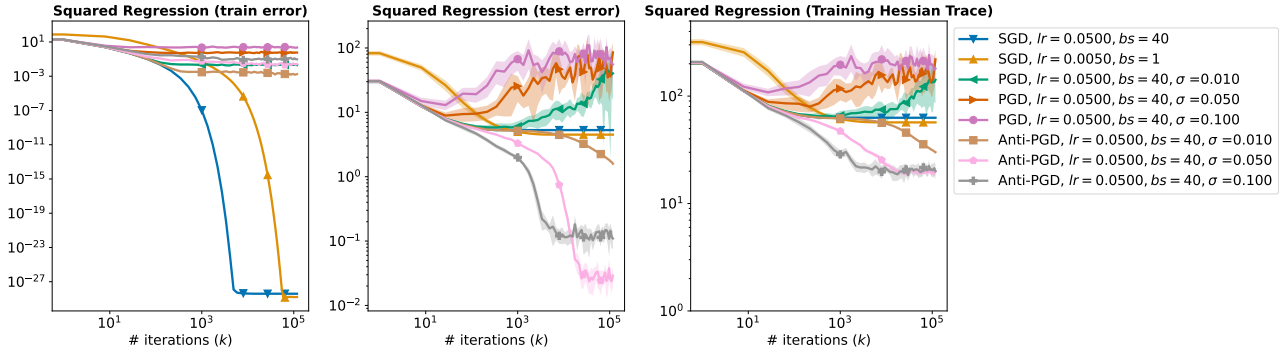

Figure 10. Performance of anti-PGD on **quadratically parametrized linear regression**, for **moderate learning rate** and different values of noise injection standard deviation. Plotted is also the error bar relative to 1 standard deviation (10 runs).

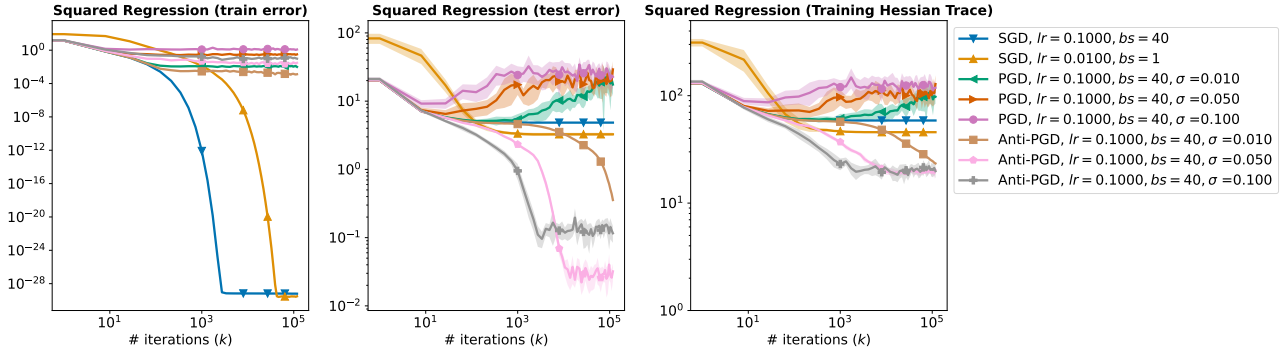

Figure 11. Performance of anti-PGD on **quadratically parametrized linear regression**, for **high learning rate** and different values of noise injection standard deviation. Plotted is also the error bar relative to 1 standard deviation (10 runs).

Precise setting. Our setting is similar to (Blanc et al., 2020). We consider the case $n = 20$, and generate a random $X^* = V^*(V^*)\top$, of rank 5, by picking $V \in \mathbb{R}^{n\times 5}$ with standard Gaussian entries. Also all the $A_i \in \mathbb{R}^{n\times n}$ are sampled at random with standard Gaussian distributed independent entries. We consider learning from 100 training examples (corrupted by a small Gaussian noise). At test time, we evaluate the solution against 100 newly sampled measurements. We test three different learning rates in Figure 12, 13, 14, and for each 3 values of noise injection variance.

Findings. We found that anti-PGD always provides the best test accuracy, and minimizes the trace of the Hessian as well. This finding is quite robust in terms of hyperparameter tuning. Further, we found that the performance is always drastically different from the one of PGD. An explanation of this phenomenon is provided in Theorem 3.1, in the main paper. Mini-batch SGD improves the final test loss if the stepsize is small enough, but gets unstable for big stepsizes.

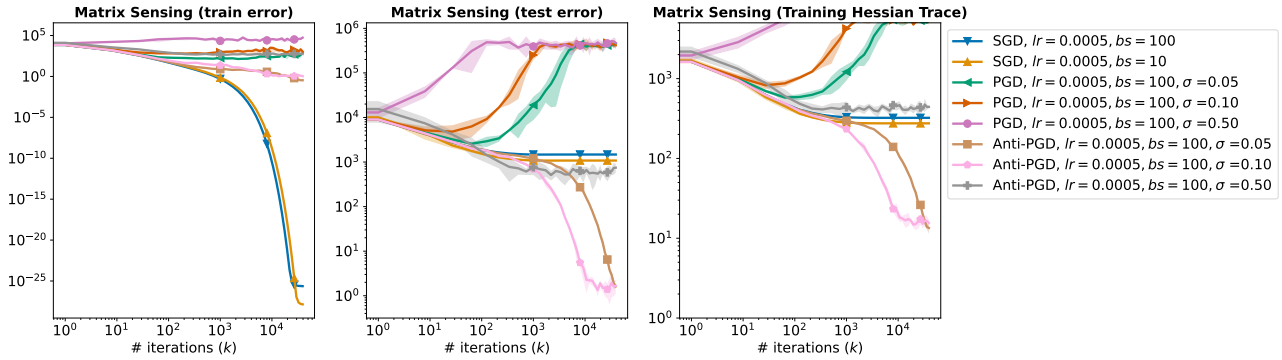

Figure 12. Performance of anti-PGD on **matrix sensing**, for **low learning rate** and different values of noise injection standard deviation. Plotted is also the error bar relative to 2 standard deviation (5 runs).

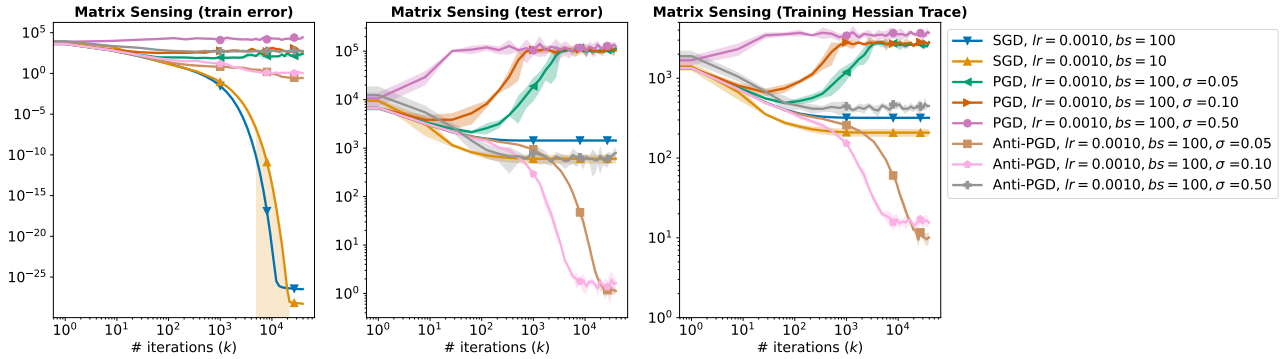

Figure 13. Performance of anti-PGD on **matrix sensing**, for **moderate learning rate** and different values of noise injection standard deviation. Plotted is also the error bar relative to 2 standard deviation (5 runs).

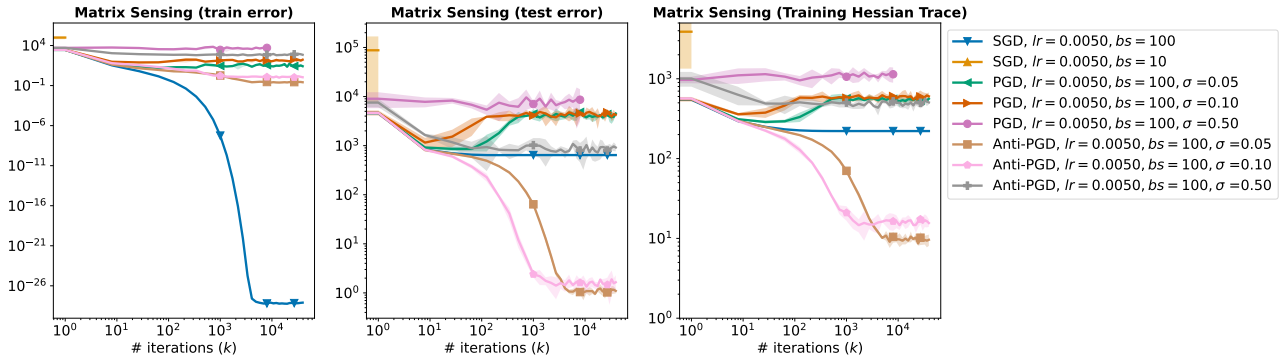

Figure 14. Performance of anti-PGD on **matrix sensing**, for **high learning rate** and different values of noise injection standard deviation. Plotted is also the error bar relative to 2 standard deviation (5 runs). SGD with batch size 10 is unstable at this learning rate.

D.4. CIFAR 10 ResNet 18

We use the implementation of ResNet18 provided by `https://github.com/kuangliu/pytorch-cifar`). Details about the corresponding experiments can be found in §4.
Takeaway: even after heavy tuning, AntiPGD performs better than standard noise injection.

D.5. Performance of Anti-SGD

We test the performance of anticorrelated noise when injected on top of mini-batch SGD. We consider the two non-toy settings of Fig. 2 and report results in Fig. 17. For matrix sensing, injecting anticorrelated noise to SGD gives a substantial improvement, with a slight edge over Anti-PGD. For the ResNet18 experiment, we compared batch-sizes of 128 or 1024:

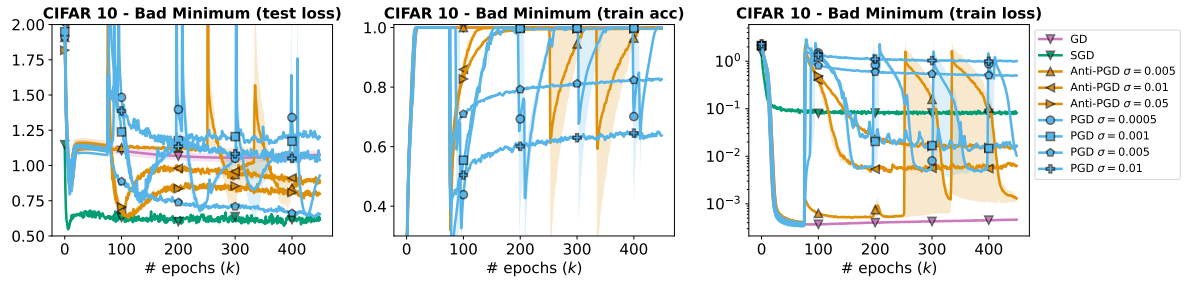

Figure 15. Additional plot for the experiment in Figure 6.

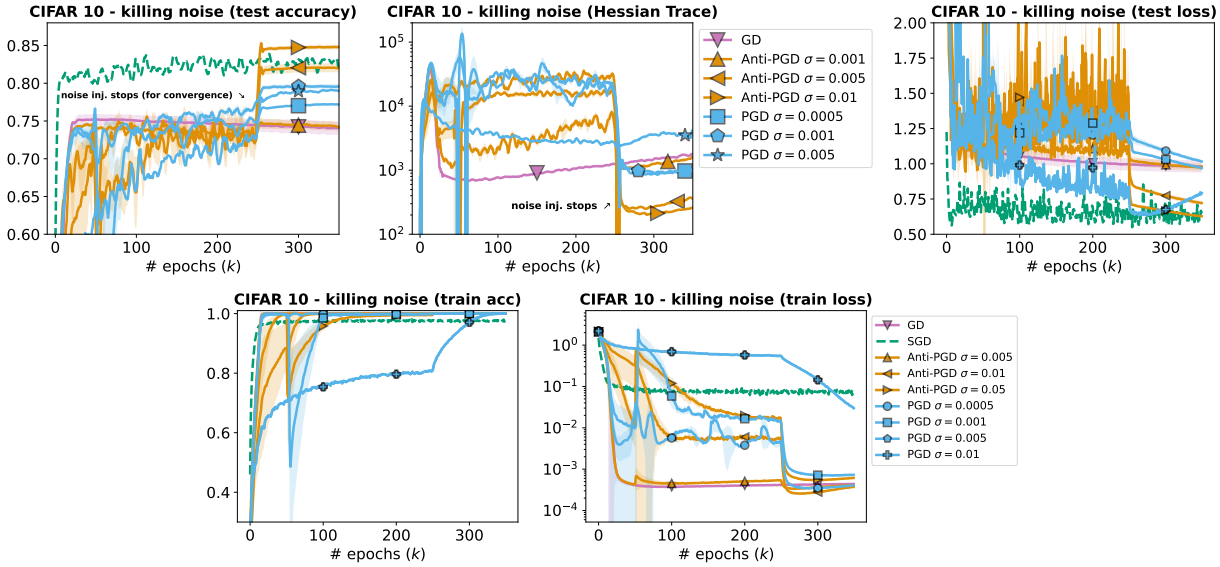

Figure 16. Additional plot for the experiment in Figure 2.

noise injection works best at moderate batch sizes. We hypothesize this is due to the high stochastic nature of SGD at low batch-sizes, which dominates over injected noise. Plots for the Hessian follow the same trend (highest test, lowest trace).

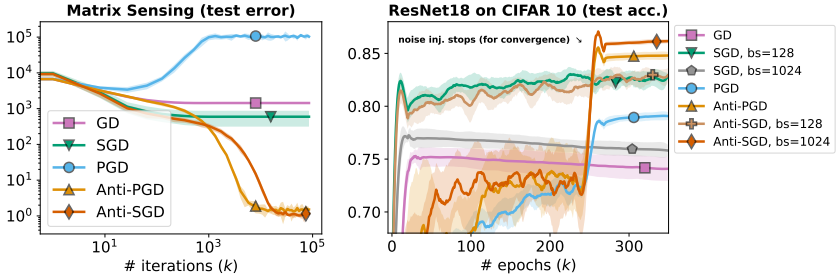

Figure 17. Performance of Anti-SGD in the same settings as Figure 2