Before we begin, we highlight one of our key contributions: **Algorithm 2**. It showcases a remarkable *interplay between statistics and optimization*: the *increasing* step sizes scheme (required for computational optimality) only works because we rely on early stopping and do not aim to fully optimize the training objective. In contrast, most results in optimization literature consider gradient descent with constant or *decreasing* step sizes to ensure convergence to the objective.

Providing more intuition (R#1 and R#2). For gradient descent (GD), our reparameterization turns *additive* updates into *multiplicative* updates (see lines 69-74). As a result, the scale of the parameter can be understood as inertia – the small parameters have a tendency to remain almost unchanged, while the larger parameters are more sensitive to the gradient size with respect to the standard parameterization $\mathbf{w}$. Sparsity is induced with reparameterization *together* with small initialization size (one without the other doesn't work). For more intuition see the proof sketch (Theorem 4), simulations and appendices A and B. Finally, previous work in the literature shows that GD implicitly regularizes the ℓ_2 norm. This corresponds to minimizing ℓ_1 norm of $\mathbf{w}$ in our parameterization on $\mathbf{u}, \mathbf{v}$ (see lines 255-256).

Choice of hyperparameters (R#1 and R#2). As discussed in lines 201-208, we only need to know $w^\star_{\max}$ up to multiplicative factors to properly initialize α and η. Theorem 2 shows how to obtain such an estimate. Hence we only need to tune the stopping time, which can be done by cross-validation.

Response to Reviewer #1.

1. See our paragraphs on intuition and hyperparameters above.

2. Most work on implicit ℓ_2 regularization focus on GD with a constant step size usually stopping at $\Theta(\sqrt{n})$ iterations; $(\eta t)^{-1}$ corresponds to the Ridge regression λ. In lines 195-200 we discuss connections to Thm 1 and 3. On the other hand, we are not aware of other work on implicit regularization achieving computational optimality via an increasing step sizes scheme (Alg. 2). We highlight that in our case implicit regularizer is not strictly ℓ_1 norm (see lines 125-126 and 334-339) and our work, to the best of our knowledge, is first to induce sparsity *implicitly* in a general noisy setting.

3. This remains an open question not considered in our paper. We believe that a good starting point would be to experiment with individualized initialization sizes and step sizes among each dimension/group.

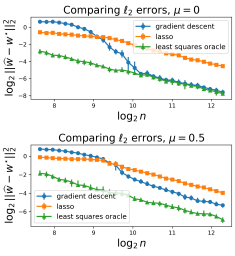
Improvements section. We agree with the suggestion and plan to add an additional paragraph to the related literature section, expanding on the second point above. Subject to space considerations we will also expand on intuition.

Response to Reviewer #2.

1. For sparsity of the optimization path see proof sketch (Thm 4), simulations (lines 314-316) and the main proofs. For sparsity at the stopping time, see the ℓ_∞ bound on S^c in Thm 1.

2. $w^\star_{\max}$ is defined in line 84. Line 151 refers to table of notation.

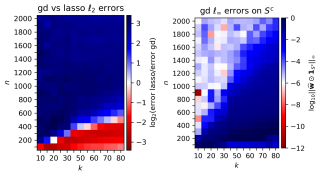
3. Since $\mathbf{X}$ needs to only satisfy RIP, n depends only logarithmically on d.



Improvements section. For intuition and hyperparameters, see the two paragraphs at the top of the rebuttal. To address the concerns on RIP assumption being too strong, we have performed additional situations when RIP assumption fails. Consider the setting given in lines 322-332, with rows of $\mathbf{X}$ now sampled from $N(\mathbf{0}, \Sigma)$, with $\Sigma = (1-\mu)\mathbf{I} + \mu \mathbf{1}\mathbf{1}^\mathsf{T}/d$. On the right, we plot simulation results with $\mu = 0$ (RIP holds) and $\mu = 0.5$ (RIP fails). We see that even when RIP fails, our method still exhibits correct rates and outperforms the lasso when the phase transition happens. The gap between gradient descent and the oracle method is visible due to the $\log k$ factor in Corollary 3, suggesting also that the rate given in Corollary 3 could be tight. We will address the reviewers concerns by adding a section on potential improvements with an expansion of the above discussion. We will also compare and contrast RIP and RE assumptions. If space permits, we will also slightly expand on the intuition.

Response to Reviewer #3.

We have previously attributed the quadratic sample complexity in k to our bounds being ℓ_∞ (which is harder) rather than ℓ_2. Our focus has been on *minimax-rates and dimension-independent rates* with *optimal computational complexity*. Also, while there is loss in sample complexity, there is gain in performance that is impossible to be achieved by the lasso (see Corollary 3 and lines 334-339).



We stick to the simulation setting described in lines 308-313 and 322-328, with $d = 5000$. The left figure on the right compares ℓ_2 error ratios for gradient descent and lasso. The blue region corresponds to our method achieving lower error, while the red region corresponds to the lasso achieving lower error than gradient descent. This plot strongly suggests, that sample complexity linear in k should indeed be enough to match/exceed performance of the lasso. The question remains, whether the ℓ_∞ bounds in Theorems 1 and 3 (in particular for stopping time t, $||\mathbf{w}_t \odot \mathbf{1}_{S^c}||_\infty \leq \sqrt{\alpha}$) require sample complexity quadratic in k? The figure on the right side suggests that the sample complexity linear in k is enough to satisfy even the ℓ_∞ bounds. We expect this sample complexity gap to be addressed in future work.

Given the results above, we absolutely agree with the suggestion to include a discussion on sub-optimal sample complexity in our revision. We plan to do so in an extra section on potential improvements (see also response to R#2).