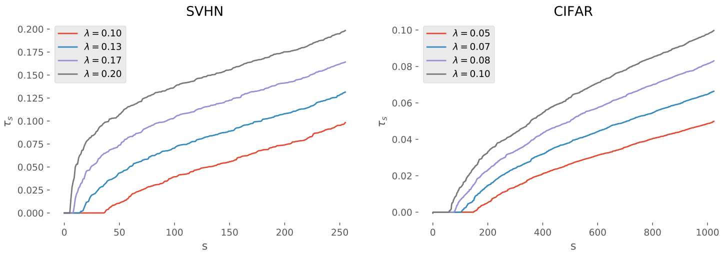
We thank the reviewers for their encouraging and instructive comments, and the AC for guiding the review process.

Encoder Gap and Further Numerical Evidence (R1, R2 and R3) The encoder gap can be viewed simply as a measure of maximal energy along any dictionary atom that is not in the support of an input vector. For example, if we assume that the dictionary forms an orthogonal basis for principal subspace of the data, and the encoder is linear (e.g., as in PCA), then the expected encoder gap would be $\lambda - (s+1)^{th}$ largest eigenvalue of the covariance matrix $\mathbf{C}_{\mathbf{xx}} = \mathbf{E}[\mathbf{xx}^\top]$ (assuming zero-mean data). More generally, it is the $(s+1)^{th}$ entry of the vector $\lambda\mathbf{1} - |\langle \mathbf{D}, \mathbf{x} - \mathbf{D}\varphi_{\mathbf{D}}(\mathbf{x})\rangle|$ (ordered in increasing manner) as we state on line 123. The formal max-min definition follows the previous work of Mehta and Gray (2013), and may look a bit too complicated. We will add a remark in line with our comment above.

Note that the assumption on encoder gap is very mild. Intuitively, if a dictionary $\mathbf{D}$ provides quickly decaying approximation error as a function of the cardinality, then a positive encoder gap exists for some s. Importantly, the cardinality s provides a knob for our results as one can always consider larger s to guarantee that $\tau_s(\mathbf{x}) > 0$, at the expense of the scaling of our generalization bound (through η_s). Moreover, one can still induce a larger encoder gap by increasing the regularization parameter λ, as demonstrated in Fig. 1 in the paper (and the figures in this document), which will come at the expense of accuracy (as demonstrated in Fig. 2c and 2d in the manuscript). In this way, our results guarantee that if one can achieve good accuracy with large encoder gap, then one can generalize robustly. This is reminiscent of generalization bounds for any margin based predictor (e.g., SVM): If the empirical margin loss is small and the margin achieved is large, then the hypothesis generalizes well.

Lastly, while our contribution is mostly theoretical, we also provide further numerical evidence for the encoder gap in real scenarios. We trained two models, on SVHN (with 256 atoms) and CIFAR (with 1024 atoms), with the training procedure described in our paper, and depict the value of τ_s (obtained over a collection of samples), for different values of λ. As you can see, one can easily obtain $\tau_s > 0$ for quite small values of s.



R2: ***It is not clear that sparsity-promoting encoders are the right models to be studying.*** Deploying sparse priors in the learned representations is a first-take at the analysis of non-linear and data-dependent mappings for supervised models. Ours is the first work to address this. Moreover, *parsimony* (e.g., sparse feature learning) plays an important role throughout data science and machine learning. Sparsity of learned representation can be ensured by sparsity on the weight vectors in the second layer of a neural network (the dense first layer can be viewed as learning a dictionary), and it is indeed common in practice to have an ℓ_1 penalty on weight matrices. Convolutional structures further promote sparsity. The challenge in analyzing a two hidden layer neural network (as described above) is that it is unclear what a good distributional property would be (akin to encoder gap) that will allow a trade-off between robustness and accuracy.

R2: ***For Theorem 4.1, it would be good to explain how your result compares to prior theoretical bounds on adversarially robust generalization.*** There is no direct analytical comparison since the nature of work here is quite different: none of the prior works study the role of margin and stability of the learned representation in enabling a trade-off between robustness and accuracy. We will add a qualitative comparison and discussion on these in the revised version.

R3: ***It seems that the method is limited to the linear case as well as the incoherent assumption. It is possible that the data does not satisfy the incoherent assumption or the data could not be represented by linear combinations of columns in D?*** Note that the end-to-end map $f(\mathbf{x})$ is *non-linear* in $\mathbf{x}$, as it is the composition of a linear function and a non-linear representation map. Next, the only assumption we need on the incoherence of the dictionary is that $\eta_s < 1$, which is mild. The fact that image data can be sparsified by incoherent dictionaries is well-known (e.g. the cornerstone of JPEG-2000). In practice, this is ensured during training via regularization as described in Equation (7). If you look at the statement of Theorem 4.1, there is no assumption on η_s, instead our bounds are in terms of η_s, so the sample complexity is expressed directly in terms of this quantity.

R3: ***It is weird that in Figure 2(b), the accuracy is not monotonically decreasing with regard to the adversarial budget.*** Computing adversarial perturbations requires solving a non-convex optimization problem. Since this is infeasible in general (for non-linear models), one resorts to approximations (such as those based on projected gradient descent [Madry 2018]). These approximations are not guaranteed to recover the perturbations that maximize the error, and as a result, the empirical reported accuracy is not necessarily monotonically decreasing. Note though that fluctuations are very small and within the margin of approximation.

Experimental details: We are committed to the reproducibility of our results. All code to reproduce experiments will be shared and openly available.