Deep Nets Don't Learn via Memorization

David Krueger*, Nicolas Ballas*, Stanislaw Jastrzebski*,
Devansh Arpit*, Maxinder Kanwal, Tegan Maharaj,
Emmanuel Bengio, Asja Fischer, Aaron Courville

MILA

1

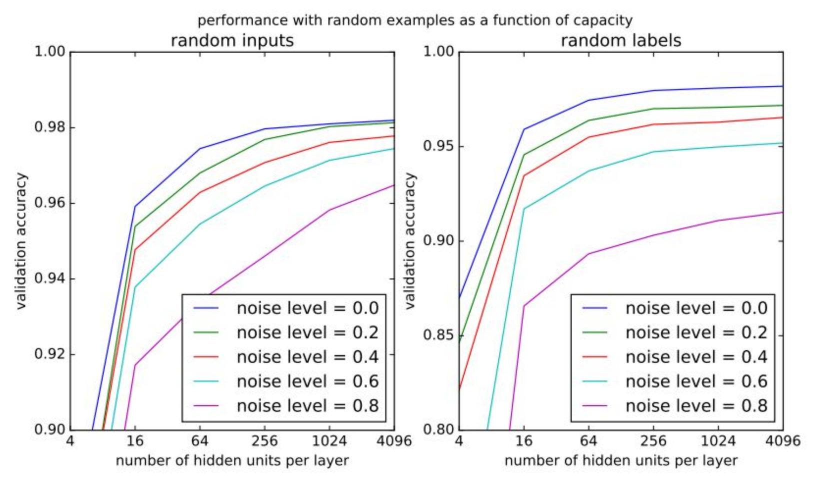

Fig. 1 Performance as a function of capacity for different levels of noise in 2-layer MLPs (real data = **blue**). Random inputs (**left**) is percentage of examples replaced with noise, (**right**) is random labels). For real data, performance is already very close to maximal with 4096 hidden units, but as noise is increased, higher capacity is needed to achieve maximal performance.

2

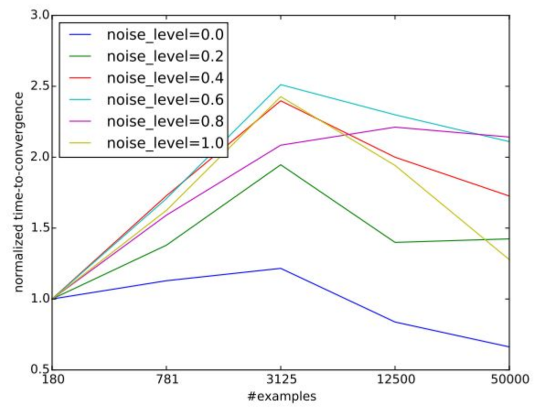

Fig. 2 Change in normalized time to convergence as a function of dataset size, with capacity fixed at 4096 units. Because there are patterns underlying real data, increasing dataset size doesn't increase training time for real data as much as it does for noise.

Related work & Conclusions

Zhang et al. [1] raise questions about memorization and generalization in deep networks. We address these questions by providing insight on learning behaviour of deep nets. Comparing our work with [1]:

Our work	Zhang et al. [1]
• Focuses on **differences** in learning noise/data	• Focuses on **similarities**
• **Conclude** DNNs don't just memorize real data	• **Suggests** DNNs might use memorization
• Training time is more sensitive to capacity and #examples on noise	• Training time increases by a constant factor on noise
• Regularization can target memorization	• Regularization doesn't explain generalization

Goodfellow et al. [2] explain that a model's representational capacity (~#parameters) is limited by (1) learning algorithm and (2) regularization, to become the **effective capacity**, and suggest learning_rate*#iterations as a measure. They note understanding effective capacity is difficult without understanding non-convex optimization.

We demonstrate that the data distribution is also an important consideration, which our proposed of **critical sample ratio** depends on. Understanding generalization requires thinking about how **data**, **learning**, and **regularization** affect capacity, and each other.

3

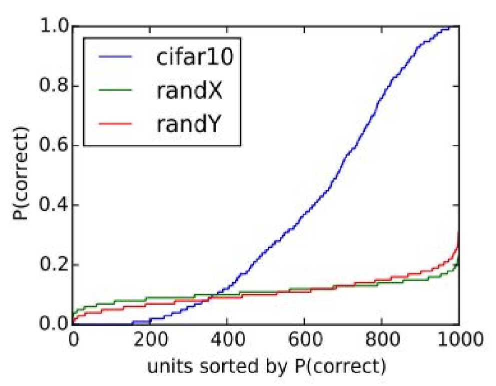

Fig. 3.1 Average (100 experiments) misclassification rate for each of 1000 examples after one epoch of training, for real data (cifar10, **blue**), random noise 'images' (randX, **green**) and random labels (randY, **red**). Easiness of examples (i.e. probability of being correctly classified after 1 epoch of training) varies much more for real data.

$$\bar{g}_x = \frac{\sum_t \left\| \partial \mathcal{L} / \partial x \right\|_1}{t}$$

We compute loss-sensitivity as the partial derivative of the loss L wrt example x, averaged over training iterations t.

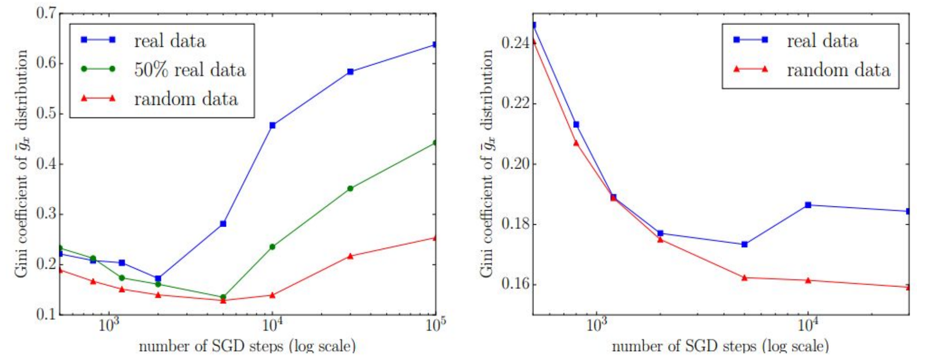

Fig. 3.2 Gini coefficient (a measure of roughness/disparity over categories) of the average loss-sensitivity over the course of training, on a 1000-example real dataset (14x14 MNIST) (**blue**) versus noise data (**red**) and 50% noise (**green**). On the **left**, the target is the normal class label; on the **right**, there are as many classes as examples. Disparity (of loss-sensitivity, between different examples, over the course of training) is higher for real data in both cases.

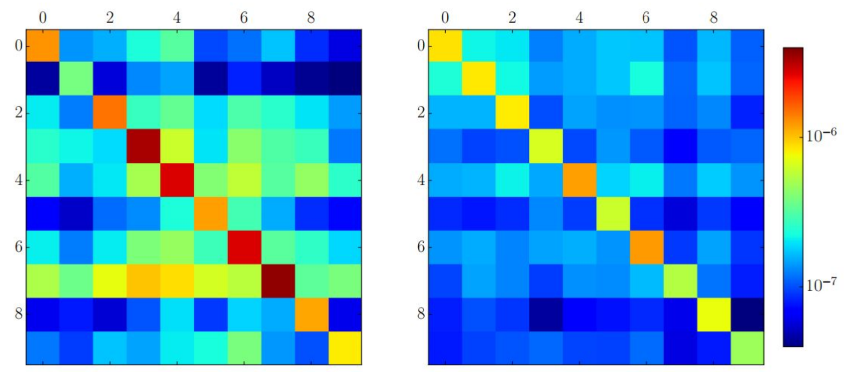

Fig. 3.3 Per-class loss-sensitivity;, a cell i,j represents the average loss-sensitivity of examples of class i w.r.t. training examples of class j. **Left** is real data, **right** is random data. Loss-sensitivity is more highly class-correlated for real data.

4

We define a critical sample as an example which has a nearby adversarial (differently classified) example. The ratio of critical samples is the proportion of examples for which a critical sample is found in radius r. This gives an idea of the number of decision boundaries in the function a network computes; i.e. how complicated that function is.

$$\arg\max_i f_i(\mathbf{x}) \neq \arg\max_j f_j(\hat{\mathbf{x}}) \qquad \|\mathbf{x} - \hat{\mathbf{x}}\|_\infty \leq r$$

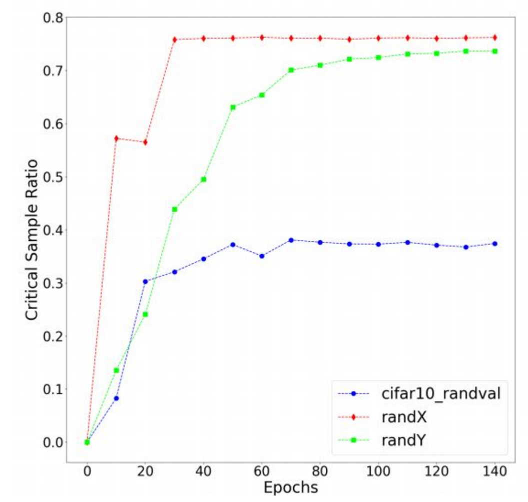

Fig. 4.1 Critical sample ratio for randomly chosen examples over the course of training on CIFAR-10, for noise input (randX, **red**) and noise labels (randY, **green**), and real data (**blue**). As measured by critical sample ratio, function complexity increases very rapidly for noise data (red), increases eventually, to almost the same level, for noise labels (green).

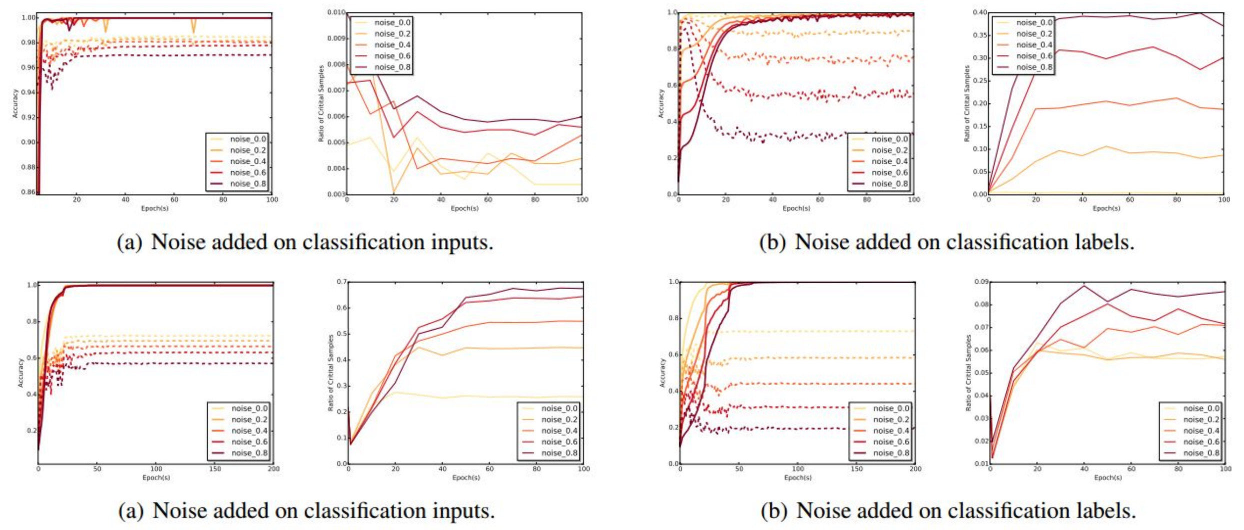

Fig. 4.2 Accuracy (**left** in each pair, **solid** is train, **dotted** is validation) and Critical sample ratios (**right** in each pair) for MNIST (**top** row) and CIFAR-10 (**bottom** row) for different types of noise (inputs: **left columns**, labels: **right columns**), for different amounts of noise (real data is **yellow**, increasing noise is **red**). Critical samples provide a good basis for assessing generalization across tasks and data types.

5

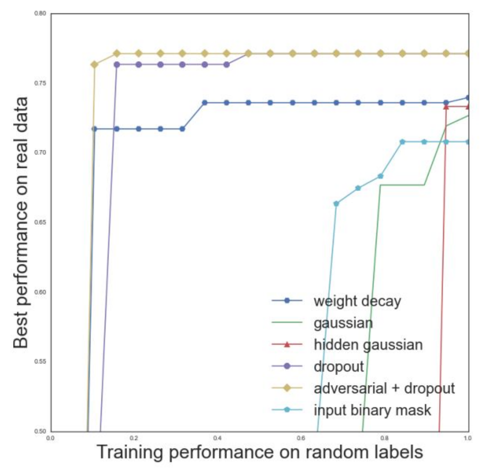

Fig. 5.1 Best validation performance on real data vs. training performance on noise labels for the same model, for different regularizers. Flatter curves indicate that memorization (as indicated by noise performance) can be capped without sacrificing generalization (on real data).

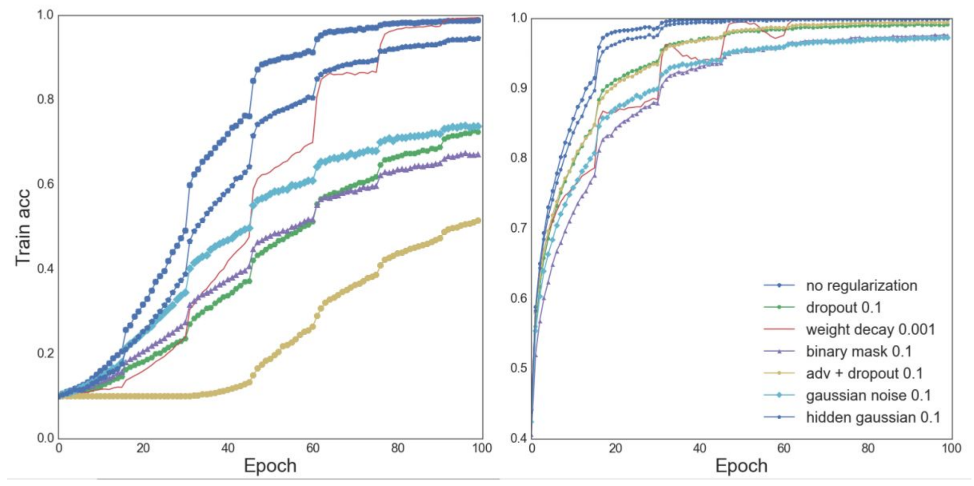

Fig.5.2 Training accuracy over time (epochs), on noise labels (**left**) and real labels (**right**) data. Regularization can slow down memorization behaviour.

What is memorization?

Behaviour on random noise is a useful operational definition of memorization.

Deep nets can achieve 0 training error on datasets of random noise; does this mean their learning strategy is to memorize everything?

We perform a thorough empirical investigation of behaviour on real vs. noise data, and show this is not the case.

We show that for deep nets:

1. Fitting noise requires more effective capacity
2. Training on noise gets harder, faster, when the dataset grows
3. On real data, some examples are always/never fit immediately, and some examples have more/less impact on training (not so for noise)
4. Simple patterns are learned first., before memorizing
5. Regularization can effectively reduce memorization

References

Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, Oriol Vinyals. **Understanding Deep Learning Requires Rethinking Generalization**. ICLR 2017. [1]

Ian Goodfellow, Yoshua Bengio, Aaron Courville. **Deep Learning**. MIT Press 2016. [2]