DeepProb Meeting

Andrew Campbell and Cong Lu

13^{th} October 2021

Table of Contents

1 Paper 1

2 Paper 2

Small Data, Big Decisions: Model Selection in the Small-Data Regime

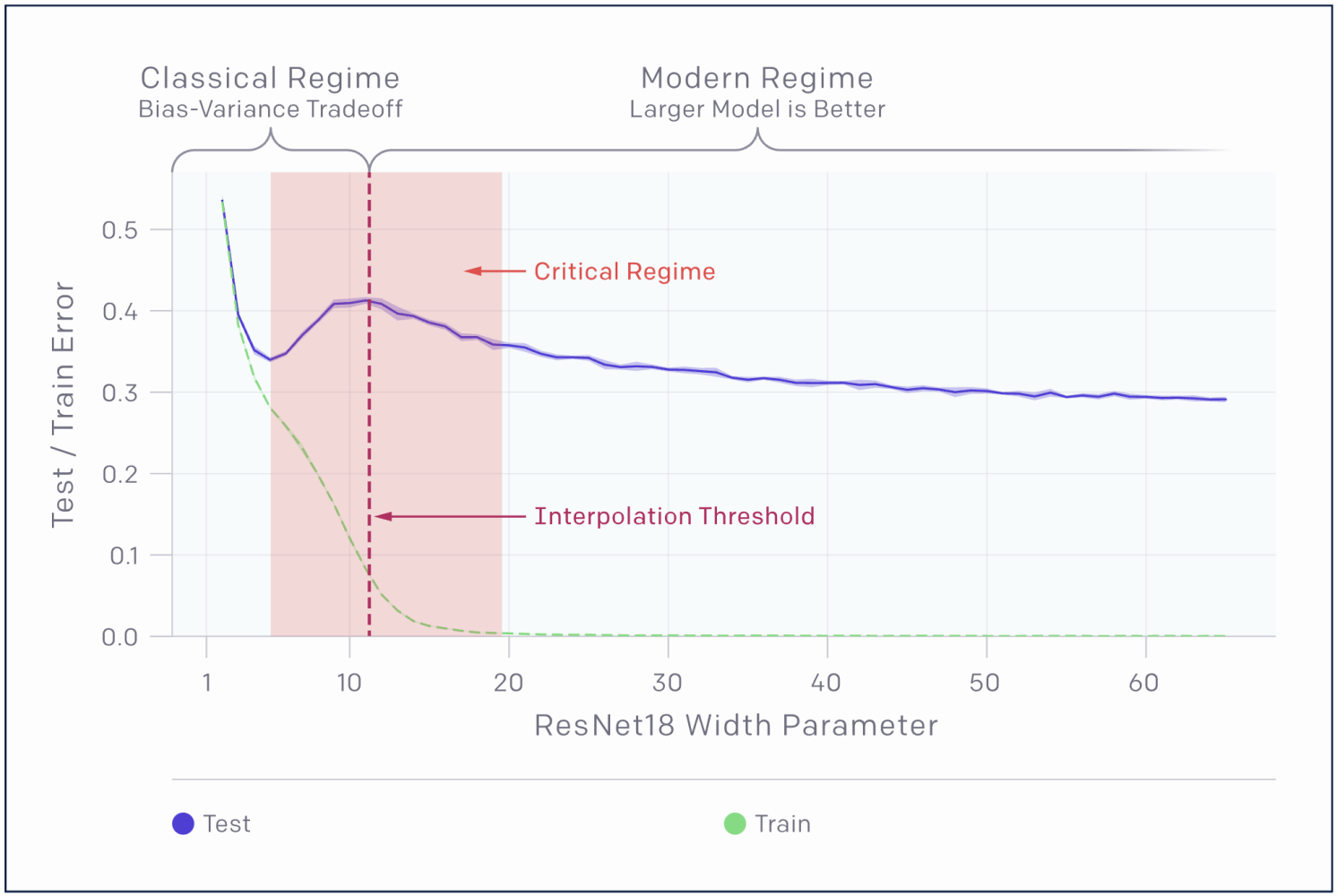
Jörg Bornschein [1] **Francesco Visin** [1] **Simon Osindero** [1]

Overview

- Small Data, Big Decisions: Model Selection in the Small-Data Regime. From Jorg Bornschein, Francesco Visin, Simon Osindero at ICML 2020
- Highly overparameterized neural networks display surprisingly good generalization performance
- Previous work has studied this from the perspective of model size
- This paper, on the other hand, is an extensive empirical study looking at generalization as a function of training data size

Background

"Double Descent" pattern moving out of the classical bias-variance regime (Nakkiran et al., 2019) Interpolation threshold = when model+training can just fit the training set.



Main Findings

1. Provides an extensive collection of training curves (as a function of dataset size) for a wide range of architectures on ImageNet, CIFAR10, MNIST and EMNIST
2. Overparameterized model architectures maintain their relative ranking in terms of generalization performance, when trained on arbitrarily small subsets of the training set. Useful for NAS...
3. It is possible to avoid some overfitting by choosing an optimal softmax temperature on a small heldout dataaet

Experimental Setups

1. Adam (Kingma & Ba, 2014) with fixed learning rates $\left\{10^{-4}, 3 \cdot 10^{-4}, 10^{-3}\right\}$ and 50 epochs.
2. Momentum SGD with initial learning rates $\left\{10^{-4}, 3 . 10^{-4}, \cdots, 10^{-1}\right\}$ cosine-decaying over 50 epochs down to 0 (0.9 momentum and $\epsilon = 10^{-4}$).
3. RMSProp + cosine schedule (Tieleman & Hinton, 2012) with initial learning rates of $\{0.03, 0.1, 0.3\}$ and cosine-decaying to 0 over 50 epochs.

Temperature Calibration

- Overparameterized models are typically trained by minimizing either a regression loss or a categorical cross-entropy loss. However, their generalization performance is then evaluated using the error-rate. This is because the generalization cross-entropy can overfit severely for models that are sufficiently big.
- One can avoid the negative effects of overfitting by choosing an optimal temperature by calibrating the network on a heldout dataset.

Temperature Calibration

- In softmax models, for each input x_i, given logits z_i, we output class prediction $\hat{y}_i$ and confidence $\hat{p}_i$:

$$\sigma_{\mathrm{SM}}\left(\mathrm{z}_i\right)^{(k)} = \frac{\exp\left(z_i^{(k)}\right)}{\sum_{j=1}^{K}\exp\left(z_i^{(j)}\right)}, \quad \hat{p}_i = \max_k \sigma_{\mathrm{SM}}\left(\mathrm{z}_i\right)^{(k)}$$

- Guo et al. (2017) proposes to soften this by:

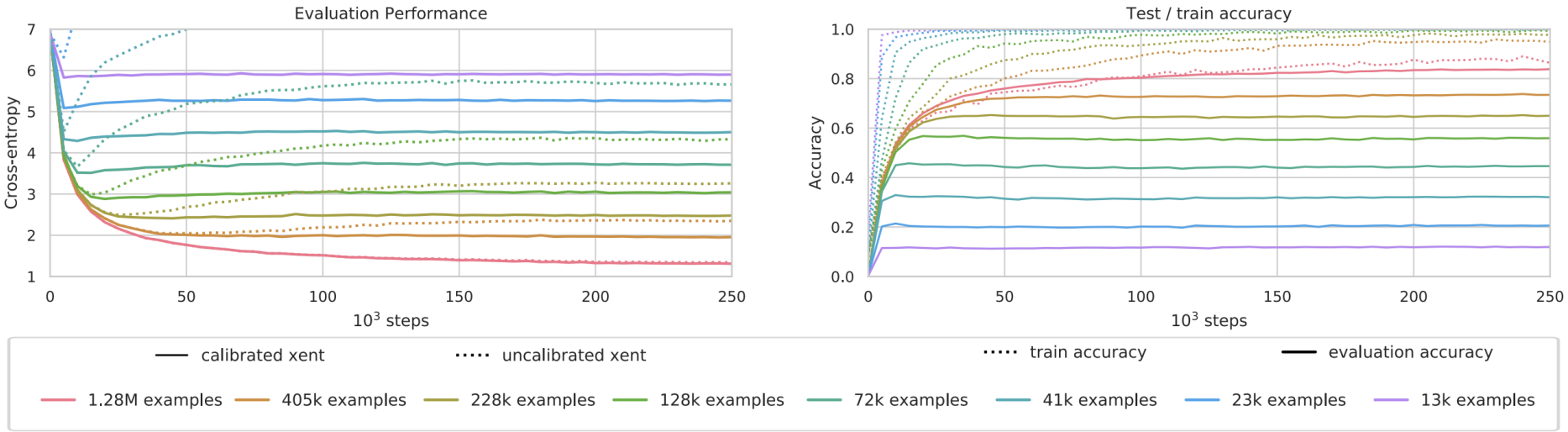
$$\hat{p}_i = \max_k \sigma_{\mathrm{SM}}\left(\mathrm{z}_i / T\right)^{(k)}$$

  for $T > 0$

- This is tuned in practice by gradient descent on a small heldout dataset (e.g. 10% of the training set). This is done alongside regular model training steps. Also allows us to assess generalisation cross-entropy during training

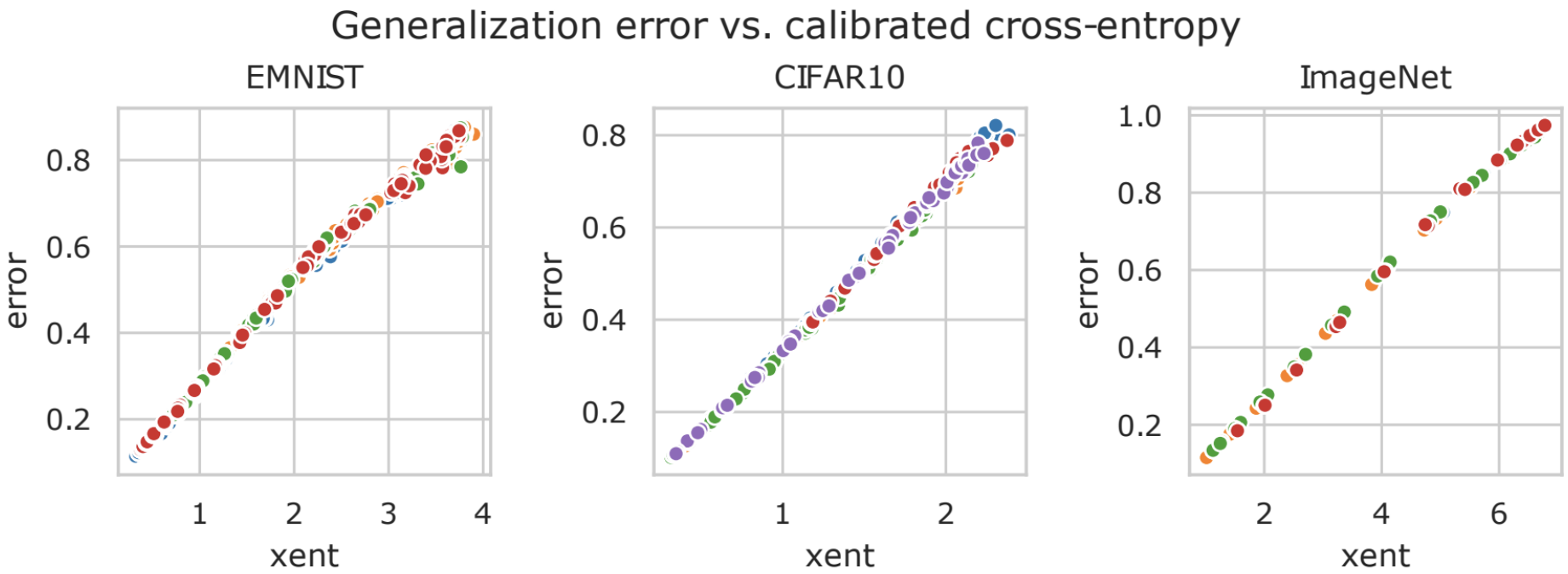
Temperature Calibration

The calibrated generalisation cross-entropy does not exhibit the same signs of overfitting. (ResNet-101 model with cosine-decayed RMSProp on subsets of ImageNet.) Smaller training datasets are always sampled i.i.d.



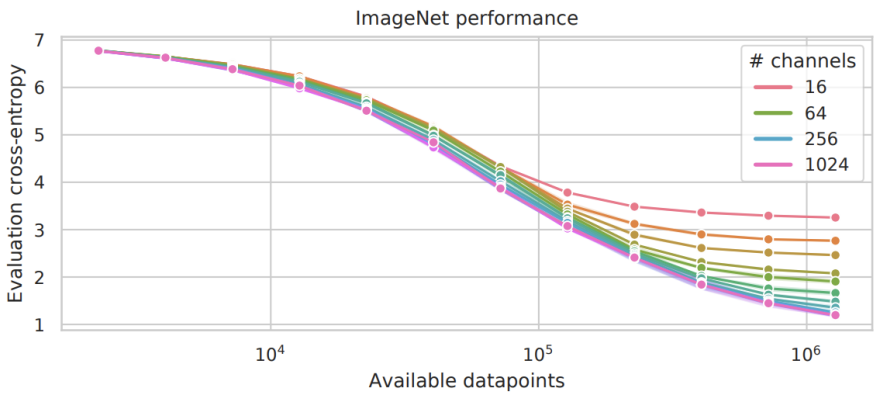
Temperature Calibration

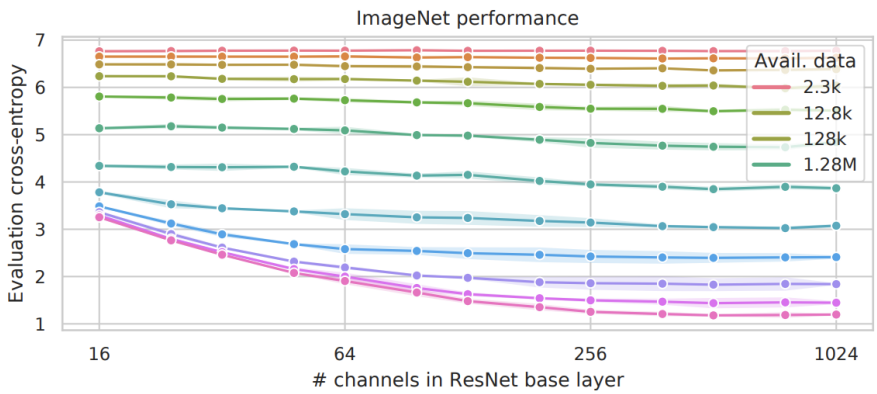
We see the post-convergence generalization error and generalization cross-entropy are well correlated for a range of model architectures and training set sizes. This is desirable as this is the loss we optimize for. This is the metric we measure from now on.



Varying dataset + model size

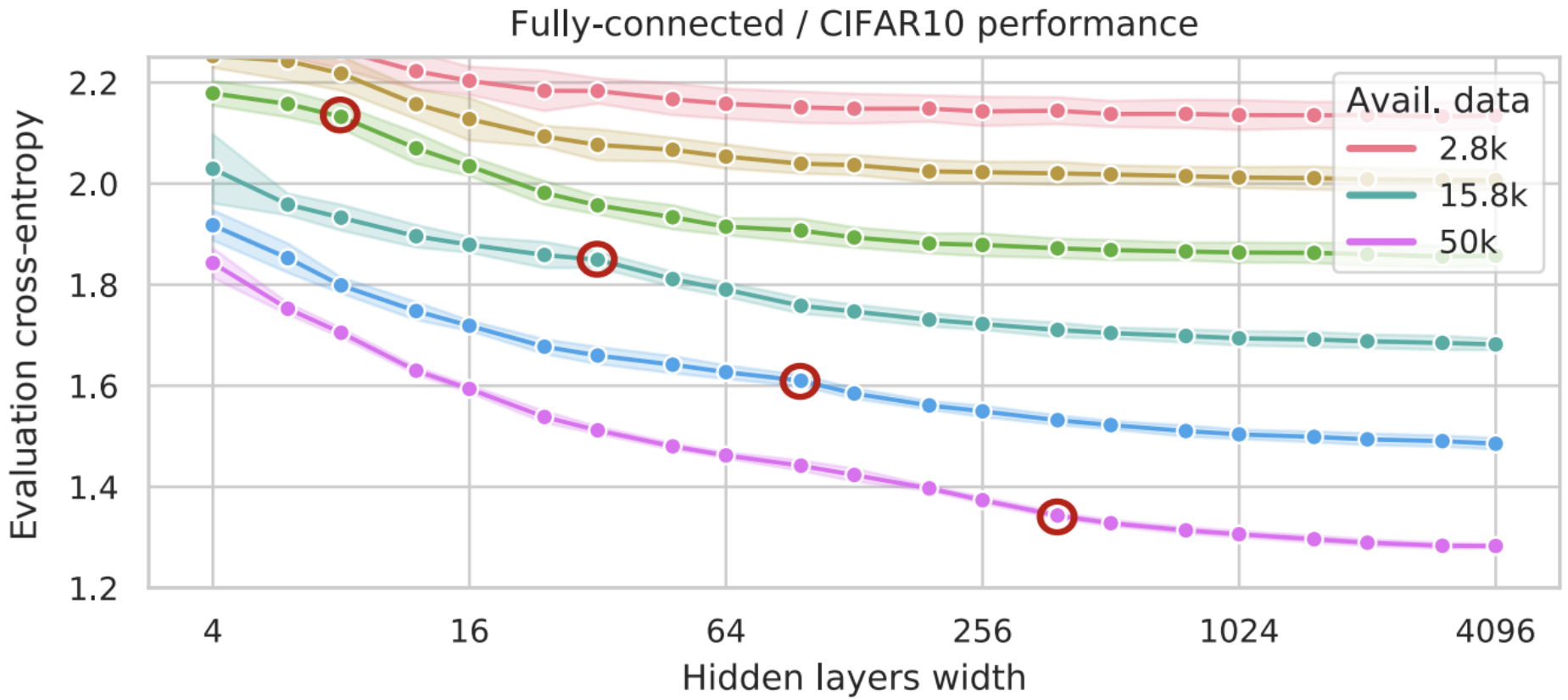
Cross-entropy performance profiles for the ResNet-101 architecture on ImageNet when trained with RMSProp. Even with as little as ≈ 2.3 images per class, there is no harm in using a ResNet model with 4 $\times$ more channels (16$\times$ more parameters) than the standard ResNet. How many channels are required to get optimal performance is roughly consistent with dataset size (right).





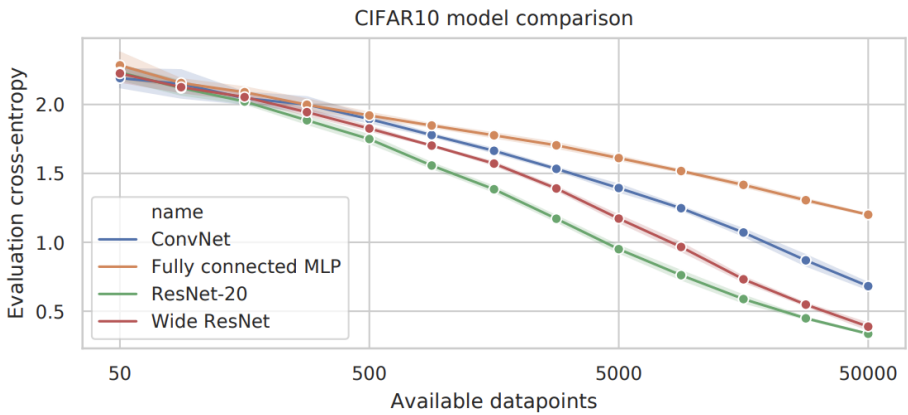
Varying dataset size

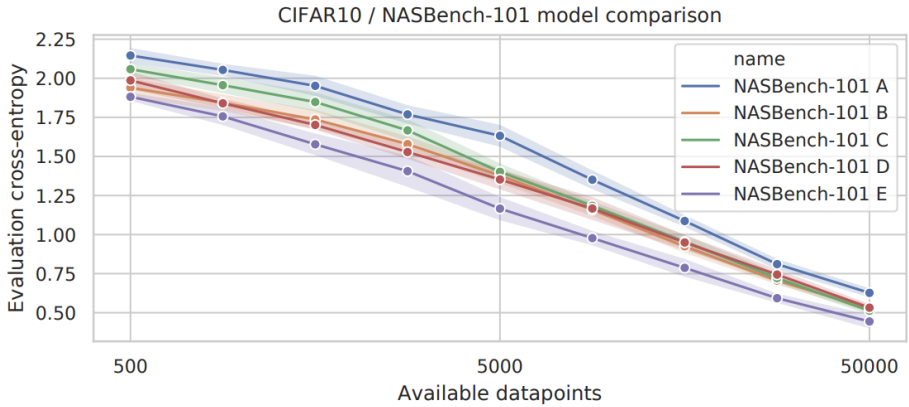
Performance profiles for a fully connected MLP with 3 hidden layers on CIFAR10 as a function of the hidden layer size. Red points mark the smallest models that approach a close to zero training error-rate. Again, around 2k width is optimal for all.



Consistent model ranking

Models either canonical or selected from NASBench-101, uncertainty bands from 30 seeds. The model architectures all maintain their relative rank. Due to the noise, one may also quantify the reliability of the performance gap between pairs with the signal-to-noise ratio: $SNR = \sqrt{\frac{\Delta^2}{\text{Var}[\Delta]}}$ with $\Delta = \mathcal{L}_A - \mathcal{L}_B$, where $\mathcal{L}_X$ is the final generalization cross-entropy after training model X.





MDL and Bayes Factor Estimation

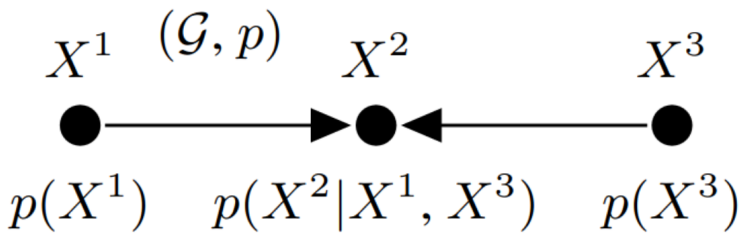
- One could also consider approaches, for example, Bayesian model selection or the Minimum Description Length (MDL) principle to select between two model architectures, the next paper will explore this idea more...
- The use of calibrated generalisation cross entropy facilitates this, see paper for more details!

Prequential MDL for Causal Structure Learning with Neural Networks

Jörg Bornschein **Silvia Chiappa** **Alan Malek** **Rosemary Nan Ke**
DeepMind, London
{bornschein,csilvia,alanmalek,nke}@deepmind.com

Bayesian Network

A directed acyclic graph (DAG) $\mathcal{G}$ whose nodes $X^1, \ldots, X^D$ represent random variables and links express statistical dependencies among them.



Characterized by its joint distribution

$$p(X^1, \ldots, X^D|\mathcal{G}) = \prod_{d=1}^{D} p(X^d|\text{pa}(X^d))$$

Ranking graphs

Given dataset $\mathcal{D} = \{x_i := (x_i^1, \ldots, x_i^D)\}_{i=1}^n$

Naive maximum likelihood score:
$\log p(\mathcal{D}|\hat{\theta}^{\text{MLE}}(\mathcal{D}), \mathcal{G})$ where $\hat{\theta}^{\text{MLE}}(\mathcal{D}) := \arg\max_\theta \log p(\mathcal{D}|\theta, \mathcal{G})$.

No measure of model complexity *as the distribution over $\mathcal{D}$ is not normalized*

Minimum Description Length

Pick graph $\mathcal{G}$ that assigns the shortest code to the observed dataset $\mathcal{D}$

This requires a normalized distribution over possible datasets

Code length for $\mathcal{D}$ will be $-\log \bar{p}(\mathcal{D}|\mathcal{G})$

How to define $\bar{p}(\mathcal{D}|\mathcal{G})$?

Minimum Description Length

$$\bar{p}(\cdot|\mathcal{G}) = \operatorname*{argmax}_{q} \min_{\mathcal{D}\in\mathcal{X}^n} \left(\log q(\mathcal{D}) - \log p(\mathcal{D}|\hat{\theta}^{\mathrm{MLE}}(\mathcal{D}), \mathcal{G}) \right)$$

.

$\log \bar{p}(\mathcal{D}|\mathcal{G})$ has the same shape as $\log p(\mathcal{D}|\hat{\theta}^{\mathrm{MLE}}(\mathcal{D}), \mathcal{G})$ but with a constant offset to ensure normalization.

Minimum Description Length

Observed dataset $\mathcal{D}^*$

- Complex $\mathcal{G} \rightarrow \log p(\mathcal{D}|\hat{\theta}^{\text{MLE}}(\mathcal{D}), \mathcal{G})$ high for many $\mathcal{D} \rightarrow \bar{p}(\mathcal{D}|\mathcal{G})$ is spread out $\rightarrow \bar{p}(\mathcal{D}^*|\mathcal{G})$ is low (long code length)

Minimum Description Length

Observed dataset $\mathcal{D}^*$

- Complex $\mathcal{G} \rightarrow \log p(\mathcal{D}|\hat{\theta}^{\text{MLE}}(\mathcal{D}), \mathcal{G})$ high for many $\mathcal{D} \rightarrow \bar{p}(\mathcal{D}|\mathcal{G})$ is spread out $\rightarrow \bar{p}(\mathcal{D}^*|\mathcal{G})$ is low (long code length)
- Simple $\mathcal{G}$ but bad fit for $\mathcal{D}^* \rightarrow \log p(\mathcal{D}|\hat{\theta}^{\text{MLE}}(\mathcal{D}), \mathcal{G})$ high for only a few $\mathcal{D}$ not including $\mathcal{D}^* \rightarrow \bar{p}(\mathcal{D}^*|\mathcal{G})$ is low (long code length)

Minimum Description Length

Observed dataset $\mathcal{D}^*$

- Complex $\mathcal{G} \rightarrow \log p(\mathcal{D}|\hat{\theta}^{\text{MLE}}(\mathcal{D}), \mathcal{G})$ high for many $\mathcal{D} \rightarrow \bar{p}(\mathcal{D}|\mathcal{G})$ is spread out $\rightarrow \bar{p}(\mathcal{D}^*|\mathcal{G})$ is low (long code length)
- Simple $\mathcal{G}$ but bad fit for $\mathcal{D}^* \rightarrow \log p(\mathcal{D}|\hat{\theta}^{\text{MLE}}(\mathcal{D}), \mathcal{G})$ high for only a few $\mathcal{D}$ not including $\mathcal{D}^* \rightarrow \bar{p}(\mathcal{D}^*|\mathcal{G})$ is low (long code length)
- **Simple $\mathcal{G}$ and good fit for $\mathcal{D}^* \rightarrow \log p(\mathcal{D}|\hat{\theta}^{\text{MLE}}(\mathcal{D}), \mathcal{G})$ high for only a few $\mathcal{D}$ including $\mathcal{D}^* \rightarrow \bar{p}(\mathcal{D}^*|\mathcal{G})$ is high (short code length)**

Prequential MDL

How to approximate $\log \bar{p}(\mathcal{D}|\mathcal{G})$?

Want same shape as $\log p(\mathcal{D}|\hat{\theta}^{\text{MLE}}(\mathcal{D}), \mathcal{G})$ but normalized

$$\begin{aligned}\log p(\mathcal{D}|\hat{\theta}^{\text{MLE}}(\mathcal{D}), \mathcal{G}) &= \log p(x_1, \ldots, x_n|\hat{\theta}^{\text{MLE}}(x_1, \ldots, x_n), \mathcal{G}) \\ &= \log \prod_{i=1}^{n} p(x_i|\hat{\theta}^{\text{MLE}}(x_1, \ldots, x_n), \mathcal{G})\end{aligned}$$

Prequential MDL

How to approximate $\log \bar{p}(\mathcal{D}|\mathcal{G})$?

Want same shape as $\log p(\mathcal{D}|\hat{\theta}^{\text{MLE}}(\mathcal{D}), \mathcal{G})$ but normalized

$$\begin{aligned}\log p(\mathcal{D}|\hat{\theta}^{\text{MLE}}(\mathcal{D}), \mathcal{G}) &= \log p(x_1, \ldots, x_n|\hat{\theta}^{\text{MLE}}(x_1, \ldots, x_n), \mathcal{G}) \\ &= \log \prod_{i=1}^{n} p(x_i|\hat{\theta}^{\text{MLE}}(x_1, \ldots, x_n), \mathcal{G})\end{aligned}$$

Normalized version:

$$\log p_{\text{preq}}(\mathcal{D}|\mathcal{G}) = \log \prod_{i=1}^{n} p(x_i|\hat{\theta}^{\text{MLE}}(x_{<i}), \mathcal{G})$$

Prequential MDL - Practicalities

$$\log p_{\text{preq}}(\mathcal{D}|\mathcal{G}) = \log \prod_{i=1}^{n} p(x_i|\hat{\theta}^{\text{MLE}}(x_{<i}), \mathcal{G})$$

Use set of increasing split points $\{s_k\}_{k=1}^{K}$ with $s_k \in [2, \ldots, n]$.

$$\log p_{\text{preq}}(\mathcal{D}|\mathcal{G}) \approx \log \prod_{k=1}^{K-1} \prod_{j=s_k}^{s_{k+1}-1} p(x_j|\hat{\theta}^{\text{MLE}}(x_{<s_k}), \mathcal{G})$$

Calibrate softmax temperature on a validation set in the small data regime to mitigate extreme overfitting.

Experiments

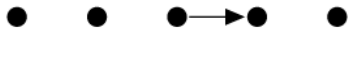
$\mathcal{E}$	$\mathcal{H}$	Inferred DAG (DAG-GNN)	Ground-Truth DAG	Inferred DAG (**ours**)	$\mathcal{H}$	$\mathcal{E}$
✗	2				0	✓
✗	4				0	✓
✓	0				0	✓
✗	5				1	✗
✓	0				0	✓
✓	0				1	✓
✓	0				1	✓
✗	2				1	✓
✓	0 ;				3	✗
✓	0				0	✓

Table 1: Results on nonlinearities from Yu et al. [2019]. We list the ground-truth DAGs and the DAGs inferred by DAG-GNN and by prequential scoring. We also report whether the inferred DAGs are in the Markov equivalence class $\mathcal{E}$ of the ground-truth DAGs and the structural Hamming distance $\mathcal{H}$.