On the Role of Neural Collapse in Transfer Learning

Tomer Galanti

Google DeepMind

October 20, 2021

Motivation

- In **transfer learning** [Car95, Ben12, YCBL14]: train a small network on top of a freezed pretrained model (e.g., ResNet-50 on ImageNet).

- Effective approach for dealing with **overfitting** when the data is limited [KBZ+20, CMZ19].

Motivation

- **Foundation models** are large pretrained models that can be effectively adapted to a wide variety of tasks [BMR+20, BHA+21].

- SOTA results on **few-shot learning** datasets [DCRS20, TWK+20].

Summary

1. Neural collapse
2. Neural collapse implies few-shot learnability
3. NC generalizes to new samples
4. NC generalizes to new classes
5. Experiments

Transfer Learning with Foundation Models

Target task

- k-class classification.
- Samples $(x, y) \in \mathcal{X} \times \mathcal{Y}$ are distributed by P.
- **Train set:** $S = \{(x_i, y_i)\}_{i=1}^{n}$.
- **Model:** $h \in \mathcal{H}$.
- **Goal:** learn $h : \mathcal{X} \to \mathbb{R}^k$ that minimizes the generalization risk: $L_T(h) = \mathbb{E}_{(x,y)\sim P}[\ell(h(x), y)]$.

What if S is very small (e.g., 5 samples per-class)?

Transfer Learning with Foundation Models

- Take a pretrained feature map (foundation model) f (e.g., ResNet-50 on ImageNet).
- Train $h = g \circ f$ to minimize $L_S(h)$, while freezing f.

Hopefully, it would generalize better..

Transfer Learning

Auxiliary/source task

- l-class classification problem ($l \gg k$).
- Samples $(x, y) \in \mathcal{X} \times \mathcal{Y}$ are distributed by $\tilde{P}$.
- **Large dataset:** $\tilde{S}$.
- **Model:** $\tilde{h} = \tilde{g} \circ f$.
- Train $\tilde{h}$ to minimize: $L_{\tilde{S}}(\tilde{h}) = \text{Avg}_{(x,y)\in\tilde{S}}[\ell(\tilde{h}(x), y)]$.

Neural Collapse

According to Papyan et al. 2020

- Train a large overparameterized NN for classification.
- The feature embeddings belonging to the same class concentrate around their means.

Normalized Variance

- **Embeddings mean:** $\mu_f(P) = \mathbb{E}_{x \sim P}[f(x)]$.
- **Embeddings variance:** $\mathrm{Var}_f(P) = \mathbb{E}_{x \sim P}[\|f(x) - \mu_f(P)\|^2]$.
- **Normalized variance:** for two distributions P_1 and P_2

$$V_f(P_1, P_2) = \frac{\mathrm{Var}_f(P_1) + \mathrm{Var}_f(P_2)}{2\|\mu_f(P_1) - \mu_f(P_2)\|^2}$$

Neural Collapse

- **Training dataset:** $S = \cup_{c=1}^{I} S_c$ split into balanced classes.
- **Model:** $h = g \circ f$ with f being a large network and g linear.
- **NC:** $\lim_{t \to \infty} \text{Avg}_{i \neq j}[V_{f_t}(S_i, S_j)] = 0$.

Few-Shot Learning and Normalized Variance

- **Target task:** 2-class classification with classes P_1 and P_2.
- **Dataset:** S split into classes $S_c \sim P_c^{n_c}$.
- **Feature map:** $f : \mathbb{R}^d \to \mathbb{R}^p$ (e.g., pretrained).
- **Classifier:** $h_S(x) = \arg\min_{c \in \{1,2\}} \|f(x) - \mu_f(S_c)\|$.

Then,

$$\mathbb{E}_S \Pr[h_S(x) \neq y(x)] \leq 16 V_f(P_1, P_2) \cdot (1 + 2/n_c)$$

Few-Shot Learning and Normalized Variance

If the distributions $f \circ P_c$ are also spherically symmetric around their means $\mu_f(P_c)$, we have:

$$\mathbb{E}_S \Pr[h_S(x) \neq y(x)] \leq 16 V_f(P_1, P_2) \cdot (1 + 2/n_c) \cdot p^{-1}$$

Generalization to New Samples

Setting

- **Source classes:** $\tilde{P}_1, \ldots, \tilde{P}_l$.
- **Source data:** $\tilde{S} = \cup_{c=1}^{l} \tilde{S}_c$, where $\tilde{S}_c = \{\tilde{x}_{cj}\}_{j=1}^{m_c} \sim \tilde{P}_c^{m_c}$.
- **Training:** $\tilde{g} \circ f$ for classification over $\tilde{S}$.

Generalization to New Samples

Assumption: w.p. $\geq 1 - \delta$, the learning algorithm returns a function f that satisfies:

$$\left\| \mathbb{E}_{x \sim \tilde{P}_c}[f(x)] - \text{Avg}_{x \in \tilde{S}_c}[f(x)] \right\| \leq \epsilon_1^c(\delta);$$

$$\left| \mathbb{E}_{x \sim \tilde{P}_c}[\|f(x)\|^2] - \text{Avg}_{x \in \tilde{S}_c}[\|f(x)\|^2] \right| \leq \epsilon_2^c(\delta),$$

Typically bounded by Rademacher complexities.

Generalization to New Samples

Proposition (Informal)

Fix two source classes, i and j with distributions $\tilde{P}_i$ and $\tilde{P}_j$, and let $\delta \in (0,1)$. Let $\tilde{S}_c \sim \tilde{P}_c^{m_c}$ for $c \in \{i,j\}$. Then, with probability at least $1-\delta$ over the selection of $\tilde{S}$, we have

$$V_f(\tilde{P}_i, \tilde{P}_j) \leq V_f(\tilde{S}_i, \tilde{S}_j) + \frac{\text{Avg}_{c=i,j}\left[\text{poly}(\epsilon_1^c(\delta), \epsilon_2^c(\delta))\right]}{\text{poly}(\|\mu_f(\tilde{S}_i) - \mu_f(\tilde{S}_j)\|)}$$

Generalization to New Samples

$$V_f(\tilde{P}_i, \tilde{P}_j) \leq V_f(\tilde{S}_i, \tilde{S}_j) + \frac{\text{Avg}_{c=i,j}\left[\text{poly}(\epsilon_1^c(\delta), \epsilon_2^c(\delta))\right]}{\text{poly}(\|\mu_f(\tilde{S}_i) - \mu_f(\tilde{S}_j)\|)}$$

Discussion

- **NC literature:** $V_f(\tilde{S}_i, \tilde{S}_j)$ is implicitly minimized.
- $\epsilon_i^c(\delta)$ typically tend to zero as $m_c \to \infty$.
- **NC literature:** $\|\mu_f(\tilde{S}_i) - \mu_f(\tilde{S}_j)\|$ is maximized s.t. $\{\mu_f(\tilde{S}_i)\}_{i=1}^l$ all have the same norm.

Generalization to New Samples

$$V_f(\tilde{P}_i, \tilde{P}_j) \leq V_f(\tilde{S}_i, \tilde{S}_j) + \frac{\text{Avg}_{c=i,j}\left[\text{poly}(\epsilon_1^c(\delta), \epsilon_2^c(\delta))\right]}{\text{poly}(\|\mu_f(\tilde{S}_i) - \mu_f(\tilde{S}_j)\|)}$$

Discussion

- $\epsilon_1^c(\delta) \leq Rad(\mathcal{F})/m_c + \sup_f \sup_{x\in\mathcal{X}} \|f(x)\| \cdot \sqrt{\log(1/\delta)/m_c}$.
- $\|\mu_f(\tilde{S}_i) - \mu_f(\tilde{S}_j)\| \propto \|\mu_f(\tilde{S}_c)\| < \sup_f \sup_{x\in\mathcal{X}} \|f(x)\|$.
- Problem: what if $\lim_{m_c\to\infty} \|\mu_f(\tilde{S}_c)\| = 0$?

Generalization to New Samples

Conclusion: generalization to new samples holds whenever f is properly normalized, i.e., w.p. 1, $\|\mu_f(\tilde{S}_i)\| > Const > 0$.

Generalization to New Samples

Proof idea: We look at

$$\frac{1}{\|\mu_f(\tilde{P}_i)-\mu_f(\tilde{P}_j)\|^2} = \frac{1}{\|\mu_f(\tilde{S}_i)-\mu_f(\tilde{S}_j)\|^2} - \frac{\|\mu_f(\tilde{P}_i)-\mu_f(\tilde{P}_j)\|^2 - \|\mu_f(\tilde{S}_i)-\mu_f(\tilde{S}_j)\|^2}{\|\mu_f(\tilde{P}_i)-\mu_f(\tilde{P}_j)\|^2 \cdot \|\mu_f(\tilde{S}_i)-\mu_f(\tilde{S}_j)\|^2}$$

and

$$\forall c \in \{i,j\}: \ \mathrm{Var}_f(\tilde{P}_c) \leq \mathrm{Var}_f(\tilde{S}_c) + \epsilon_2^c(\delta/4) + 2\left\|\mu_f(\tilde{P}_c)\right\| \cdot \epsilon_1^c(\delta/4) + \epsilon_1^c(\delta/4)^2$$

Generalization to New Classes

Setting

- Set $\mathcal{C}$ of class-conditional distributions.
- A class $\mathcal{F}^*$ from which the algorithm selects feature maps.
- Distribution $\mathcal{D}_C$ over classes $\mathcal{C}$.
- Source classes: $\tilde{P}_1, \ldots, \tilde{P}_l \sim \mathcal{D}_C(P_1, \ldots, P_l \mid \forall i \neq j : \ P_i \neq P_j)$.
- Target class: $P_c, P_{c'} \sim \mathcal{D}_C(P_1, P_2 \mid P_1 \neq P_2)$.

Generalization to New Classes

Proposition (Informal)

Let $\mathcal{F}^* \subset \mathcal{F}$ be any set of functions and $\Delta = \inf_{f \in \mathcal{F}^*} \Delta(f) = \inf_{f \in \mathcal{F}^*} \inf_{P_c \neq P_{c'} \in \mathcal{C}} \|\mu_f(P_c) - \mu_f(P_{c'})\| > 0$. Then,

$$\mathbb{E}\left[\sup_{f \in \mathcal{F}^*}\left[\mathbb{E}_{P_c \neq P_{c'}}[V_f(P_c, P_{c'})] - \text{Avg}_{i \neq j}\left[V_f(\tilde{P}_i, \tilde{P}_j)\right]\right]\right]$$

$$\lesssim \left(1 + \frac{\sup_{f \in \mathcal{F}^*, P' \in \mathcal{C}} \text{Var}_f(P')}{\Delta}\right) \cdot \frac{Rad(\mathcal{F}^*)}{(l-1) \cdot \Delta^2}.$$

Generalization to New Classes

Proof idea: This kind of bounds typically hold for i.i.d. samples. But, we have a different kind of sampling. Therefore, we simply apply a theorem by [MP19] which covers this case. The Δ and $\sup_{f \in \mathcal{F}^*, P' \in \mathcal{C}} \mathrm{Var}_f(P')$ terms are by-product of estimating certain terms in their bound within our setting.

Generalization to New Classes

Discussion

- Generally speaking, the bound does not prove generalization.
- If $\mu_f(P_i) = \mu_f(P_j)$ then $\Delta = 0$.
- In particular, $\mathcal{F}^*$ should not include functions, such as, $f = 0$.
- Δ decreases when $|\mathcal{C}|$ increases.

Generalization to New Classes

Discussion

- If $\mu_f(P)$ are uniformly distributed in a p-dim cube: $\Delta(f) = \Omega(\sqrt{p}|\mathcal{C}|^{-2/p})$.
- Rademacher complexity scales as $\mathcal{O}(p\sqrt{l})$.
- Maximal variance is $\mathcal{O}(p)$.
- The bound scales as $\mathcal{O}\left(\frac{\sqrt{p}|\mathcal{C}|^{6/p}}{\sqrt{l}}\right)$.

Generalization to New Classes

Discussion: maybe bounding the expected normalized variance is too much?

Alternatives:

1. Bound $\mathbb{P}[V_f(P_c, P_{c'}) > \gamma]$.
2. Bound $\mathbb{P}[h(x) \neq y]$ in terms of $\text{Avg}_{i \neq j}[V_f(\tilde{P}_i, \tilde{P}_j)]$.

Experimental Setup

Phase 1 (train)

- Train $\tilde{h} = \tilde{g} \circ f$ to minimize cross-entropy classification loss on the source classes.

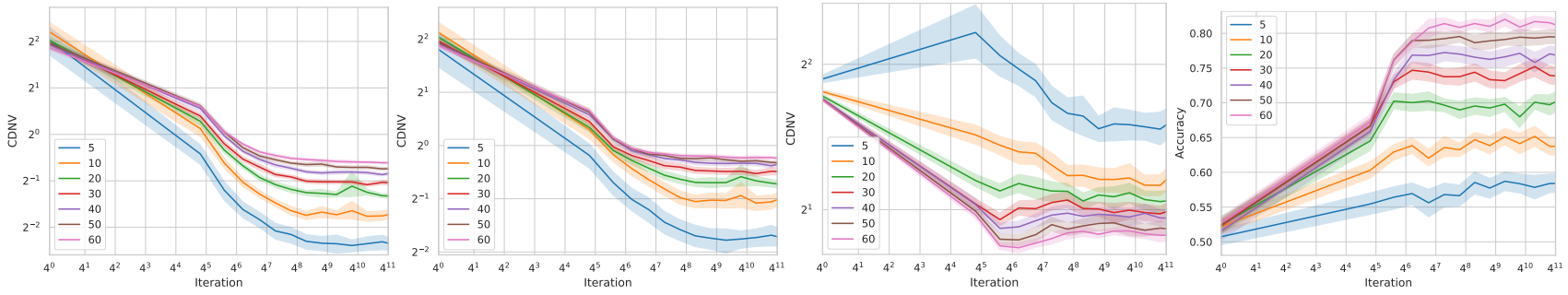
Phase 2 (eval)

- Few-shot task: 5 classes, 5 samples per-class.
- Train ridge regression on top of f using the 5x5 dataset with one-hot labels.
- Evaluate on test samples from each class.
- Average over many tasks.

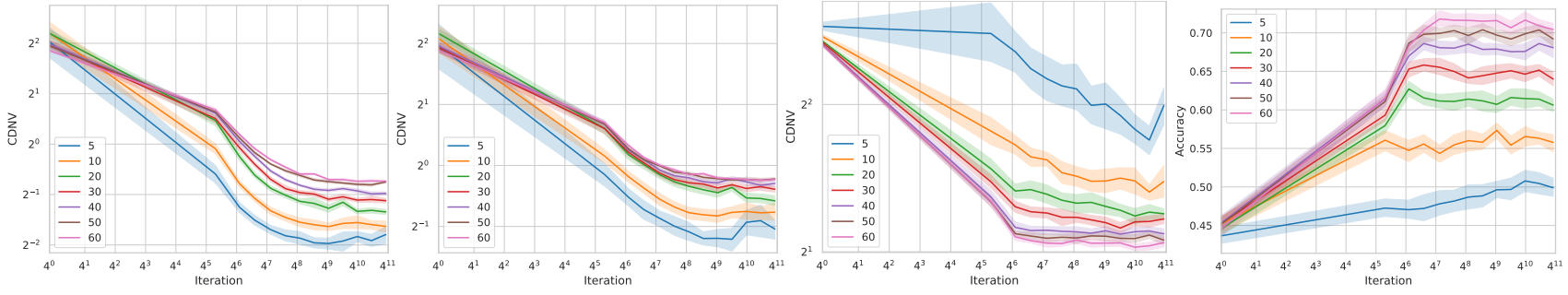
Empirical Results

Method	Architecture	Mini-ImageNet	CIFAR-FS	FC-100
Matching Networks [VBL+16]	64-64-64-64	55.31 ± 0.73	-	-
LSTM Meta-Learner [RL17]	64-64-64-64	60.60 ± 0.71	-	-
MAML [FAL17]	32-32-32-32	63.11 ± 0.92	71.5 ± 1.0	-
Prototypical Networks [SSZ17]	64-64-64-64	$68.20 \pm 0.66^{\dagger}$	72.0 ± 0.6	48.6 ± 0.6
Relation Networks [SYZ+18]	64-96-128-256	65.32 ± 0.7	69.3 ± 0.8	-
SNAIL [MRCA18]	ResNet-12	68.88 ± 0.92	-	-
TADAM [ORLL18]	ResNet-12	76.7 ± 0.3	-	56.1 ± 0.4
AdaResNet [MYMT18]	ResNet-12	71.94 ± 0.57	-	-
Dynamics Few-Shot [GK18]	64-64-128-128	73.00 ± 0.64	-	-
R2D2 [BHTV19]	96-192-384-512	68.8 ± 0.1	79.4 ± 0.1	-
SARN [HZHW19]	ResNet-101	66.16 ± 0.51	-	-
Shot-Free [RBS19]	ResNet-12	$77.64 \pm n/a$	$84.7 \pm n/a$	-
TEWAM [QSL+19]	ResNet-12	$75.90 \pm n/a$	$81.3 \pm n/a$	-
TPN [LLP+19]	ResNet-12	$75.64 \pm n/a$	-	-
MTL [SLCS19]	ResNet-12	57.6 ± 0.9	-	57.6 ± 0.9
OptNet-RR [LMRS19]	ResNet-12	77.88 ± 0.46	84.3 ± 0.5	55.3 ± 0.6
MetaOptNet [LMRS19]	ResNet-12	78.63 ± 0.46	84.2 ± 0.5	55.5 ± 0.6
Transductive Fine-Tuning [DCRS20]	WRN-28-10	78.40 ± 0.52	85.79 ± 0.50	57.57 ± 0.55
Distill [TWK+20]	WRN-28-10	82.14 ± 0.43	86.9 ± 0.5	60.9 ± 0.6
Ours	WRN-28-4	71.75 ± 0.39	82.06 ± 0.94	56.68 ± 0.51

Empirical Results



WRN-28-4 on CIFAR-FS



WRN-28-4 on MiniImageNet

(a) NV train **(b)** NV test **(c)** NV target **(d)** few-shot acc

Conclusions

- Small normalized variance implies good few-shot performance.
- Theoretical and empirical evidence on the ability of NC to generalize to new classes.

Future Work

- Improving the theoretical explanations.
- Larger scale experiments (e.g., ImageNet).
- Experiments on the effect of the embedding dimension.

Goodbye

Thanks for listening!! :-)

References

Yoshua Bengio.
Deep learning of representations for unsupervised and transfer learning.
In *Proceedings of ICML Workshop on Unsupervised and Transfer Learning*, volume 27 of *Proceedings of Machine Learning Research*, pages 17–36, Bellevue, Washington, USA, 02 Jul 2012. PMLR.

Rishi Bommasani, Drew A. Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S. Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, Erik Brynjolfsson, Shyamal Buch, Dallas Card, Rodrigo Castellon, Niladri Chatterji, Annie S. Chen, Kathleen Creel, Jared Quincy Davis, Dorottya Demszky, Chris Donahue, Moussa Doumbouya, Esin Durmus, Stefano Ermon, John Etchemendy, Kawin Ethayarajh, Li Fei-Fei, Chelsea Finn, Trevor Gale, Lauren Gillespie, Karan Goel, Noah D. Goodman, Shelby Grossman, Neel Guha, Tatsunori Hashimoto, Peter Henderson,