Pillars of Statistical Learning Theory

Setting

* Input space $\mathcal{X} \subseteq \mathbb{R}^k$
* Output space $\mathcal{Y}$ ($\mathbb{R}$ for regression or $\{-1, 1\}$ for classification)
* Distribution (unknown) D over $\mathcal{X} \times \mathcal{Y}$
* Loss function $\ell : \mathcal{Y} \times \mathcal{Y} \to [0, 1]$
* Training set $S = \{(x_i, y_i)\}_{i=1}^m$ drawn i.i.d. from D

Task

Find a hypothesis/predictor $h : \mathcal{X} \to \mathcal{Y}$ that minimizes the population loss:

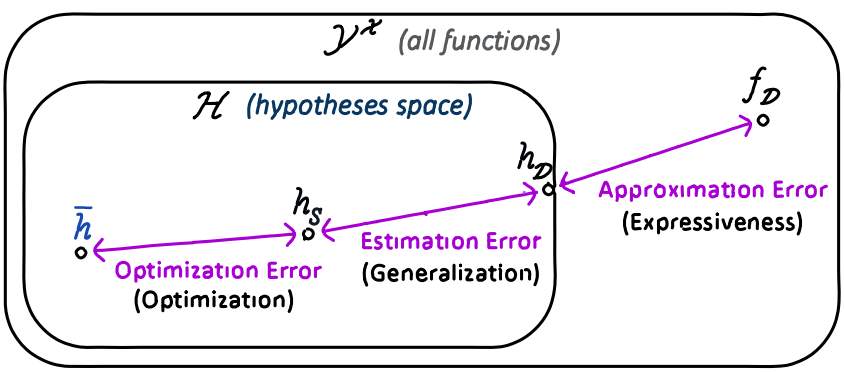
$$L_D(h) := \mathop{\mathbb{E}}_{(x,y)\sim D} [\ell(y, h(y))]$$

Typical approach: Select a hypothesis class $\mathcal{H} \subseteq \mathcal{Y}^{\mathcal{X}}$ and minimize training loss:

$$\operatorname*{argmin}_{h \in \mathcal{H}} L_S(h) := \frac{1}{m} \sum_{i=1}^{m} \ell(y_i, h(x_i))$$

Statistical learning rests on three fundamental pillars

Optimization	Generalization	Expressiveness
Ability to minimize L_S	Performance on unseen data (i.e., on D)	Which functions are included in $\mathcal{H}$



$f_D := \text{argmin}_{f \in Y^X} L_D(f)$ — ground truth (minimizes L_D over all funcs)

$h_D := \text{argmin}_{h \in \mathcal{H}} L_D(h)$ — best hypothesis (minimizes L_D over $\mathcal{H}$)

$h_S := \text{argmin}_{h \in \mathcal{H}} L_S(h)$ — empirically optimal hypothesis (minimizes L_S over $\mathcal{H}$)

$\bar{h}$ — The hypothesis returned by our algorithm

Categorization to optimization, generalization, and expressiveness is useful

* For theory: analyze learning algorithms
* For Practice: e.g., when debugging poor performance it can help pinpoint the underlying issue.

  high training loss $\Rightarrow$ optimization or expressiveness issue

  low training loss but high population loss $\Rightarrow$ generalization issue

Our Focus: Mystery of Generalization in Deep Learning

Generalization theory strives to derive bounds of the form:

$$\forall \delta \in (0,1) \text{ w.p. } \geq 1-\delta \text{ over sampling of } S:$$
$$\Delta_S(\bar{h}) := L_D(\bar{h}) - L_S(\bar{h}) \leq \underbrace{g(m, \delta, \mathcal{H}, \bar{h}, S)}_{\text{should} \to 0 \text{ when } m \to \infty}$$

We want the bound to be:

1) Tight
2) Insightful so that we can use it for designing neural network architectures and training algorithms

An example for a tight but uninsightful bound is the loss over a validation set

Beyond uniform convergence

Last lecture we saw uniform convergence bounds of the form:

$\forall D, \delta \in (0,1)$ w.p. $\geq 1-\delta$ over sampling of S from D:

$\forall h \in H \quad L_S(h) \leq \underbrace{g(m, \delta, H)}_{\text{does not depend on learned hypothesis or the data!}} \approx \sqrt{\frac{C(H) + \ln\frac{1}{\delta}}{m}}$, where $C(H) = \ln|H|$ if H is finite or $\ln(\text{size of cover for } H)$

Limitations of uniform convergence for explaining generalization in deep learning (based on empirical evidence, e.g., from "Understanding Deep Learning Requires Rethinking Generalization"; Zhang et al. 2017)

1) Neural networks (NNs) generalize well in practice even when # learned parameters $\gg$ # training examples (e.g., ResNet50 over CIFAR10). In this case, some parameter assignments (i.e., $h \in H$) that minimize L_S generalize poorly ($L_D(h)$ is high) while others generalize well ($L_D(h)$ is low).

$\Downarrow$

Need bounds that depend on the learned hypothesis h

2) The same NN h can fit both the original training set (e.g., CIFAR10) and a set of the same size with random data/labels, while achieving a far better than trivial population loss over the original distribution D. On the other hand, the population loss over random data is of course trivial.

$\Downarrow$

Need bounds that depend on the dataset S or distribution D

Hypothesis dependence: compression-based bounds

Compression bounds are based on the premise that the learned hypothesis $\bar{h}$ can be approximated by a hypothesis from a much simpler class $\mathcal{H}'$. For example, $\mathcal{H}'$ can contain NNs with significantly fewer parameters than those in $\mathcal{H}$. In this case, $\bar{h}$ can inherit the generalization properties of $\mathcal{H}'$.

To be concrete, denote: $d(h, \mathcal{H}') := \min_{h' \in \mathcal{H}'} \sup_{x \in \mathcal{X}} \|h(x) - h'(x)\|$

reflects the extent to which h can be compressed into $\mathcal{H}'$

Theorem

Assume that the loss ℓ is ρ-Lipschitz and that $\mathcal{H}'$ has the following generalization guarantee.

$\forall \delta \in (0,1)$ w.p. $\geq 1-\delta$ over sampling of S from D:

$\forall h' \in \mathcal{H}' \quad |\Delta_S(h')| \leq \sqrt{\frac{C(\mathcal{H}') + \ln(1/\delta)}{m}}$, where $C(\mathcal{H}')$ is some complexity measure of $\mathcal{H}'$

Then, $\forall \delta \in (0,1)$ w.p. $\geq 1-\delta$ over S: $|\Delta_S(\bar{h})| \leq \sqrt{\frac{C(\mathcal{H}') + \ln(1/\delta)}{m}} + 2\rho \cdot d(\bar{h}, \mathcal{H}')$

Example: $\mathcal{H}'$ consists of hypotheses that can be represented using b bits and $C(\mathcal{H}') = \ln|\mathcal{H}'| = b \cdot \ln 2$.

Proof

Let $\bar{h}' := \operatorname{argmin}_{h' \in \mathcal{H}'} \sup_{x \in \mathcal{X}} \|\bar{h}(x) - h'(x)\|$. Thus, $\sup_{x \in \mathcal{X}} \|\bar{h}(x) - \bar{h}'(x)\| = d(\bar{h}, \mathcal{H}')$.

It holds that: $|L_S(\bar{h}) - L_S(\bar{h}')| = \left| \frac{1}{m}\sum_{i=1}^{m} \ell(y_i, \bar{h}(x_i)) - \frac{1}{m}\sum_{i=1}^{m} \ell(y_i, \bar{h}'(x_i)) \right|$

$$\leq \frac{1}{m}\sum_{i=1}^{m} \left| \ell(y_i, \bar{h}(x_i)) - \ell(y_i, \bar{h}'(x_i)) \right|$$

ℓ is ρ-Lipschitz $\longrightarrow$ $\leq \frac{1}{m}\rho \sum_{i=1}^{m} \underbrace{\|\bar{h}(x_i) - \bar{h}'(x_i)\|}_{\leq d(\bar{h}, \mathcal{H}')}$

$$\leq \rho \cdot d(\bar{h}, \mathcal{H}')$$

Similarly can show that $|L_D(\bar{h}) - L_D(\bar{h}')| \leq \rho \cdot d(\bar{h}, \mathcal{H}')$

Thus:

$$|\Delta_S(\bar{h})| = |L_D(\bar{h}) - L_S(\bar{h})| \leq |L_D(\bar{h}) - L_D(\bar{h}')| + |\Delta_S(\bar{h}')| + |L_S(\bar{h}') - L_S(\bar{h})|$$
$$\leq |\Delta_S(\bar{h}')| + 2\rho \cdot d(\bar{h}, \mathcal{H}')$$

Combining this with the generalization bound for $\mathcal{H}'$ concludes the proof. □

Example

Consider a fully connected NN with input, hidden, and output dimensions all equal to k and depth L.

$$\mathcal{H} = \left\{ x \mapsto W_L \sigma(W_{L-1} \sigma(\cdots \sigma(W_1 x) \cdots)) : W_1, \ldots, W_L \in \mathbb{R}^{k \times k} \right\}$$

We omit biases for simplicity and assume the element-wise activation $\sigma(\cdot)$ is γ-Lipschitz and satisfies $\sigma(0) = 0$. For example, $\sigma(\cdot)$ can be the ReLU activation, in which case $\gamma = 1$.

Let $\mathcal{H}'$ be the hypothes class corresponding to the same NN with parameter matrices constrained to be rank 1.

$$\mathcal{H}' = \left\{ x \mapsto u_L v_L^T \sigma\left(u_{L-1} v_{L-1}^T \sigma\left(\cdots \sigma\left(u_1 v_1^T x\right)\cdots\right)\right) : u_1,\ldots,u_L, v_1,\ldots,v_L \in \mathbb{R}^k \right\}$$

The # of parameters used to represent $\mathcal{H}'$ is $2kL$, as opposed to k^2L for $\mathcal{H}$. The generalization bound for $\mathcal{H}'$ based on quantized parameters is much smaller than that for $\mathcal{H}$. A NN $h \in \mathcal{H}$ can inherit the bound for $\mathcal{H}'$ if $d(h, \mathcal{H}')$ is small. Denote by $W_1,\ldots,W_L$ the parameter matrices of h. Let $W_1',\ldots,W_L'$ be their closest rank 1 approximations and denote by h' the resulting hypothesis. It can be shown that:

$$d(h,\mathcal{H}') \le \sup_{x\in X} \|h(x)-h'(x)\| \le \gamma^{L-1} \sum_{i=1}^{L} \prod_{j=1}^{L} \|W_j\|_{\text{spectral}} \cdot \|W_i - W_i'\|_{\text{spectral}} \cdot \sup_{x\in X} \|x\|$$

Thus, the closer $W_1,\ldots,W_L$ are to rank 1, the lower our compression bound error will be.

Hypothesis and data dependence through PAC-Bayes bounds

In the PAC-Bayes approach, rather than deriving generalization bounds for individual hypotheses, one considers distributions over $\mathcal{H}$. Let Q be such a distribution. We define its population and training losses by:

$$L_D(Q) := \mathop{\mathbb{E}}_{h\sim Q}[L_D(h)] \quad , \quad L_S(Q) := \mathop{\mathbb{E}}_{h\sim Q}[L_S(h)]$$

PAC-Bayes upper bounds $\Delta_S(Q) := L_D(Q) - L_S(Q)$ according to the distance of Q from some predetermined prior distribution P over $\mathcal{H}$.

Intuitively, $\Delta_S(P)$ is typically small since P does not depend on S and if Q is close to P then $\Delta_S(Q)$ should also be small.

Theorem (Theorem 31.1 in Shalev-Shwartz & Ben-David 2014)
Let P be a prior distribution over $\mathcal{H}$ and let $\delta \in (0,1)$. Then, w.p. $\geq 1-\delta$ over sampling S from D:

$$\forall \text{ distributions } Q \text{ over } \mathcal{H} \quad \Delta_S(Q) \leq \sqrt{\frac{KL(Q\|P) + \ln(2m/\delta)}{2(m-1)}}$$

where $KL(Q\|P) := \mathbb{E}_{h\sim Q}\left[\ln \frac{Q(h)}{P(h)}\right]$ is the Kullback-Leibler divergence.

Example (based on "Computing Nonvacuous Generalization Bounds..."; Dziugaite & Roy 2017)
Suppose $\mathcal{H}$ corresponds to a class of NNs parameterized by $w \in \mathbb{R}^k$.
We take the prior P to be $N(0, \sigma^2 \cdot I)$ — Gaussian with zero mean and independent components having σ^2 variance — and Q to be $N(\bar{w}, \sigma^2 \cdot I)$, where $\bar{w} \in \mathbb{R}^k$ are the parameters returned by our learning algorithm.
Then, $KL(Q\|P) = \frac{1}{2\sigma^2}\|\bar{w}\|^2$ and the PAC-Bayes bound gives:

$$L_D(Q) \leq L_S(Q) + \sqrt{\frac{\frac{1}{2\sigma^2}\|\bar{w}\|^2 + \ln(2m/\delta)}{2(m-1)}}$$

$L_S(Q) = \mathbb{E}_{\theta \sim N(\bar{w}, \sigma^2 I)}[L_S(w)]$ — averages L_S over neighborhood of $\bar{\theta}$, can be seen as a measure of flatness

depends on the learned hypothesis

This bound depends on both the learned hypothesis and the data, and with additional tricks can give nonvacuous bounds in some settings.

Conclusion: generalization bounds

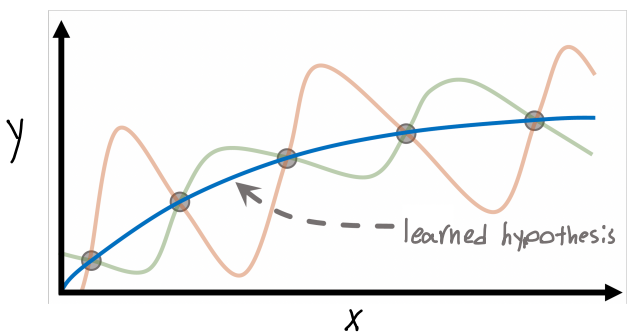
While PAC-Bayes bounds can be nonvacous:

* They are still far from tight in standard settings
* Existing complexity measures that appear in generalization bounds do not correlate well with generalization in practice, and so are unable to provide practical guidelines (see, e.g., "Fantastic Generalization Measures and Where to Find Them"; Jiang et al. 2020).

In other words, it is still unclear what is the "right" complexity measure to consider for NNs.

Explaining Generalization via Implicit Bias

When using overparameterized NNs (# learned parameters >> # training examples), there are many hypotheses with low L_S, some have low L_D as well, while others overfit. The fact that gradient-based optimization often finds hypotheses that generalize well is attributed to an implicit bias toward low complexity. Characterizing this bias is a central question in the theory of deep learning. This approach is complementary to research on generalization bounds: if we identify a complexity measure that is implicitly minimized, we can hope to derive a tight bound based on it.



Optimization method: gradient descent

NNs are typically trained using variants of gradient descent (GD). Namely, if our hypothesis is parameterized by $w \in \mathbb{R}^k$, then GD iteratively updates the parameters according to:

$$\forall t \in \{0,1,2,\ldots\} \quad w_{t+1} = w_t - \eta \cdot \nabla L_S(w_t) \quad , \quad \begin{array}{l} w_0 \in \mathbb{R}^k \text{ — initialization} \\ \eta > 0 \text{ — learning rate} \end{array}$$

As we will see, the implicit bias can depend on every aspect of the algorithm, e.g., the loss, initialization, and parameterization (i.e., NN architecture).

Overparameterized linear regression

Setting:

* $S = \{(x_i, y_i)\}_{i=1}^m \subseteq \mathbb{R}^k \times \mathbb{R}$
* $\mathcal{H} = \{x \mapsto \langle w, x \rangle : w \in \mathbb{R}^k\}$
* $L_S(w) = \frac{1}{m} \sum_{i=1}^m (\langle w, x_i \rangle - y_i)^2$

We are interested in the overparameterized setting where L_S has multiple global minima. This amounts to assuming $k > m$ (more dimensions/parameters than examples) and that $x_1, \ldots, x_m$ are linearly independent.

The solution set, i.e. global minima of L_S, is $G := \{w \in \mathbb{R}^k : \underbrace{L_S(w) = 0}_{\Leftrightarrow \forall i \ \langle w, x_i \rangle = y_i}\}$.

Question of implicit bias: when minimizing L_S via GD, which solution will we converge to?

Theorem

Suppose we minimize L_S via GD and that GD converges to a global minimum of L_S. That is, $w_\infty := \lim_{t\to\infty} w_t$ satisfies $L_S(w_\infty) = 0$. Then, w_∞ is the solution closest to w_0 in Euclidean distance: $w_\infty = \underbrace{\operatorname{argmin}_{w \in G} \|w - w_0\|^2}_{(*)}$.

Proof

At any $w \in \mathbb{R}^k$:

$$\nabla L_S(w) = \frac{2}{m}\sum_{i=1}^{m}(\langle w, x_i\rangle - y_i)\cdot x_i \in \operatorname{span}\{x_1,\dots,x_m\}$$

Thus:

$$w_t = w_0 - \sum_{t'=0}^{t-1}\eta\,\nabla L_S(w_{t'}) = w_0 + \sum_{i=1}^{m}\alpha_i x_i \in w_0 + \operatorname{span}\{x_1,\dots,x_m\}$$

$$\Uparrow$$

for some $\alpha_1,\dots,\alpha_m \in \mathbb{R}$

Since affine spaces are closed, this implies that $w_\infty \in w_0 + \operatorname{span}\{x_1,\dots,x_m\}$

Now, the optimization problem $(*)$ is strongly convex with linear constraints. Its unique minimizer w^* can be characterized using the method of Lagrange multipliers:

$$(1)\ w^* \in G \quad (\text{i.e., } \langle w^*, x_i\rangle = y_i \text{ for all } i)$$

$$(2)\ \nabla|_{w=w^*}\|w - w_0\|^2 = \nabla|_{w=w^*}\sum_{i=1}^{m}\lambda_i\left(\langle w, x_i\rangle - y_i\right)$$

$$\Leftrightarrow\ 2(w^* - w_0) = \sum_{i=1}^{m}\lambda_i x_i$$

for some $\lambda_1,\dots,\lambda_m \in \mathbb{R}$

$$\Leftrightarrow\ w^* = w_0 + \sum_{i=1}^{m}\frac{\lambda_i}{2}\cdot x_i$$

$$\Leftrightarrow\ w^* \in w_0 + \operatorname{span}\{x_1,\dots,x_m\}$$

Since w_∞ satisfies (1) and (2) we get that $w_\infty = w^*$. □

Beyond linear regression

The implicit bias in linear regression depends on the initialization w_0. In more complex settings, characterizing the implicit bias is often not as clean, but can be done (though still for simplified models). Different model parameterizations can yield different implicit biases, even when the induced hypothesis class is the same! For example, if we parameterize our linear model as $h_\theta(x) = \langle u \odot u - v \odot v, x \rangle$, where $\theta = (u, v) \in \mathbb{R}^{2k}$ and $\odot$ is the element-wise product, then depending on the initialization, the solution we get is not the closest in Euclidean distance to the initialization.

More broadly, implicit bias can also be affected by the choice of gradient-based optimization method (GD, SGD, Adam, ...) and hyperparameters such as the learning rate. See Chapter 8 of the course book for additional information and references. Next, we will see the type of implicit biases that appear in classification problems.

Linear binary classification

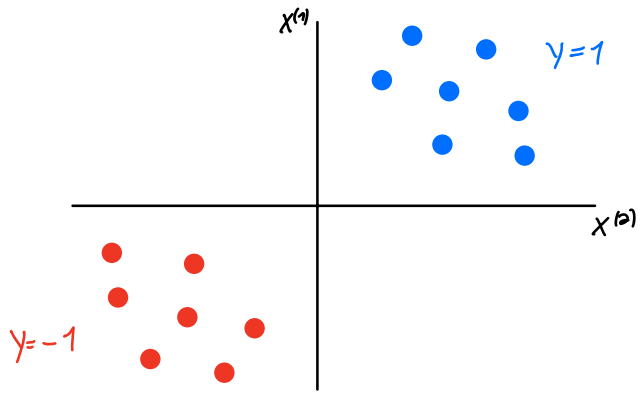
Setting:

* $S = \{(x_i, y_i)\}_{i=1}^m \subseteq \mathbb{R}^k \times \{-1, 1\}$
* $\mathcal{H} = \{x \mapsto \text{sign}(\langle w, x \rangle) : w \in \mathbb{R}^k\}$
* $L_S(w) = \frac{1}{m} \sum_{i=1}^m \ell(y_i, \langle w, x_i \rangle)$, where ℓ is the $\begin{cases} \text{logistic loss } \ell(y, a) = \ln(1 + \exp(-y \cdot a)) \\ \text{exp loss } \ell(y, a) = \exp(-y \cdot a) \end{cases}$

* We assume S is linearly separable: $\exists u \in \mathbb{R}^k$ s.t. $\forall i \in \{1, \ldots, m\}$ $y_i \langle u, x_i \rangle > 0$

In this setting, there exist infinitely many linear predictors that correctly classify the training examples.

2D illustration:



The max-margin predictor is:

$$w^* := \underset{w \in \mathbb{R}^k \setminus \{0\}}{\operatorname{argmax}} \min_{i \in \{1,\ldots,m\}} \frac{y_i \langle w, x_i \rangle}{\|w\|} = \underset{w \in \mathbb{R}^k}{\operatorname{argmin}} \|w\|^2 \text{ s.t. } \forall i \; y_i \langle w, x_i \rangle \geq 1$$

Facts:

* There exist $\alpha_1, \ldots, \alpha_m \in \mathbb{R}^m_{\geq 0}$ s.t. $w^* = \sum_{i=1}^m \alpha_i y_i \cdot x_i$
* For all $i \in \{1, \ldots, m\}$, if $\alpha_i > 0$ then $y_i \langle w^*, x_i \rangle = 1$ (such x_i are the "support vectors")
* w^* is the only vector that can be represented in this way while satisfying $y_i \langle w, x_i \rangle \geq 1$ for all i.

Theorem ("The Implicit Bias of GD on Separable Data"; Soudry et al. 2018)
Suppose we minimize L_S via GD and that $\lim_{t \to \infty} L_S(w_t) = 0$. Then:

$$\lim_{t \to \infty} \frac{w_t}{\|w_t\|} = \frac{w^*}{\|w^*\|}$$

That is, GD converges in direction to the max-margin predictor.

Note: To minimize L_S to zero $\|w\|$ must go to infinity, so there is no finite minimizer of L_S. Thus, we look at convergence in direction, which fully specifies the decision rule of the predictor in our setting.

Proof Sketch (for the exp loss)

For any $t \in \{0,1,\dots\}$:

$$-\nabla L_S(w_t) = \frac{1}{m}\sum_{i=1}^{m} \exp(-y_i \langle w_t, x_i \rangle) \cdot y_i x_i$$

To minimize L_S it is necessary that $y_i \langle w_t, x_i \rangle \xrightarrow[t\to\infty]{} \infty$ for all $i \in \{1,\dots,m\}$.
Thus, as L_S is minimized, only examples with the smallest $y_i \langle w_t, x_i \rangle$ will contribute non-negligibly to $\nabla L_S(w_t)$. These examples are precisely the "support vectors". As $t \to \infty$ they will dominate $\nabla L_S(w_t)$ and w_t, which will converge in direction to a non-negative linear combination of the support vectors. By the facts mentioned above, this implies convergence in direction to the max-margin predictor w^*.

Extension to ReLU networks

It is possible to prove an analogous result for more advanced NNs such as fully connected NNs with ReLU activations (but no bias terms). In this case the max-margin predictor is defined by:

$$\underset{\theta}{\operatorname{argmin}} \|\theta\|^2 \text{ s.t. } \forall i \; y_i \cdot h_\theta(x_i) \geq 1,$$ where θ is the parameters of the NN h_θ

Though here the guarantee is that GD (with a small learning rate) converges in direction to a critical point (KKT point) of this objective, which is not necessarily its global minimizer. For details see Lyu & Li 2020 ("GD Maximizes the Margin of Homogeneous NNs").

Can Implicit Bias Be Explained by Norms?

In the settings that we saw (and many other beyond our scope), implicit bias minimizes a norm of the parameters.

> Question: does implicit bias simply amount to norm minimization?

Counter examples:

* Razin & Cohen 2020 ("Implicit Regularization in Deep Learning May Not Be Explainable by Norms"): For certain NNs shows that no norm is being minimized.

* Vardi & Shamir 2021 ("Implicit Regularization in ReLU Networks with the Square Loss"): For a single ReLU neuron, implicit bias cannot be framed as the exact minimization of any non-trivial (continuous) function of the parameters.

These results suggest that it may be impossible to get such a clean formalization of implicit bias in deep learning. For example, may need to consider approximate minimization of complexity measures.

Implicit Bias in the Age of Language Models

In the initial wave of deep learning:

* NNs often had # parameters >> # training examples
* NNs were often trained until the training loss was completely minimized

In the context of language models:

* # parameters $\lesssim$ # training examples (at least in pretraining)
* Training loss is not completely minimized (usually only a few passes over each data point)

Thus:

* Generalization in-distribution is perhaps not that surprising
* Less relevant to discuss minimal complexity subject to fitting the data
* However, implicit bias in terms of the optimization trajectory of the learning algorithm is still relevant! For any given loss, there can be many parameter assignments attaining that loss, and they can differ in how they generalize, especially to downstream tasks.

$\Rightarrow$ Implicit bias still matters, but the question is messier and we still lack a clean formalization for modern language model settings.