高维情形下线性模型的泛化误差研究

2024 春季学期 week3,4,5 工作

Yukun Dong

目录

任务:

- 仔细阅读 Bartlett, Montanari, and Rakhlin (2021) Deep learning: a statistical viewpoint，讲的是深度学习的统计思想，包含隐正则化，min-norm least squares，gradient flow 以及 RKHS，值得细品。有待阅读的章节：1 Introduction，2 Generalization and uniform convergence，3 Implicit regularization，4 Benign overfitting，6 Generalization in the linear regime。
- 阅读 Ali, Dobriban, and Tibshirani (2020) 。The Implicit Regularization of Stochastic Gradient Flow for Least Squares。
- 阅读 Belkin (2021) 。Fit without fear: remarkable mathematical phenomena of deep learning through the prism of interpolation。
- 阅读 Belkin, Ma, and Mandal (2018) ，To Understand Deep Learning We Need to Understand Kernel Learning。

文章阅读：Deep learning: a statistical viewpoint

ULLN & empirical processes

“经验最小化的理论” 必须满足两个数学要求：

ULLN: uniform law of large numbers。

CC: Capacity Control。

全局大数定律（ULLN）表明对于假设空间中的任何假设 H，训练数据上的损失可以预测未来的期望损失：

$$\text{ULLN: } \forall f \in \mathcal{H} \quad R(f) = \mathbb{E}_{(x,y)}[l(f(x), y)] \geq R_{\text{emp}}(f)$$

我们通常期望 $R(f) \geq R_{\text{emp}}(f)$，这允许将 ULLN 写成单边不等式，通常是以下形式：

$$\forall f \in \mathcal{H} \quad R(f) - R_{\text{emp}}(f) < O^* \left(\sqrt{\frac{\text{cap}(\mathcal{H})}{n}} \right) \quad (2)$$

这里 $\text{cap}(\mathcal{H})$ 是空间 $\mathcal{H}$ 容量的度量，比如它的 Vapnik-Chervonenkis 维数，而 O^* 可以包含对数项和其他低阶项。上述不等式在数据样本选择的高概率上是成立的。

等式 (2) 是 ULLN 条件的数学实例，并直接推导出

$$R_{\text{emp}} - \min_{f \in \mathcal{H}} R(f) < O^* \left(\sqrt{\frac{\text{cap}(\mathcal{H})}{n}} \right)$$

这保证了 f_{emp} 的真实风险几乎对于 $\mathcal{H}$ 中的任何函数都是最优的，只要 $\text{cap}(\mathcal{H}) \ll n$。

implicit regularization and self-induced regularization

ANN as Gaussian process

Wide NNs perform like linear models

- ...

6.2 Ridge regression in the linear regime

我们考虑两层神经网络的情况：

$$f(x; \theta) := \frac{\alpha}{\sqrt{m}} \sum_{j=1}^{m} b_j \sigma(\langle w_j, x \rangle), \quad \theta = (w_1, \dots, w_m).$$

一些神经网络的泛化结果都是基于下面的假设：通过梯度下降训练的神经网络将被模型 $f_{lin}(\hat{a}) = f(\theta_0) + Df(\theta_0)\hat{a}$ 很好地近似，其中 $\hat{a}$ 在经验风险最小的同时最小化了 $||\hat{a}||_2$：

$$\hat{a} := \arg\min_{a\in\mathbb{R}^p}\{||a||_2 : y_i = f_{lin}(x_i; a) \text{ 对所有} i \le n\}.$$

我们将 (6.1) 的最小范数过程推广到考虑岭回归 estimator：

$$\hat{a}(\lambda) := \arg\min_{a\in\mathbb{R}^p}\left\{\frac{1}{n}\sum_{i=1}^{n}(y_i - f_{lin}(x_i; a))^2 + \lambda||a||_2^2\right\},$$

其中，

$$f_{lin}(x_i; a) := \langle a, Df(x_i; \theta_0)\rangle.$$

函数类 $\{f_{lin}(x_i; a) := \langle a, Df(x_i; \theta_0)\rangle : a \in \mathbb{R}^p\}$ 是一个线性空间，它由 a 线性参数化。我们考虑两个特定的例子，**Random Feature(RF)** 和 **Neural Tangent(NT)** 模型，这些例子是通过线性化两层神经网络获得的，但线性化方法不一样：

$$\mathcal{F}_{RF}^m := \{f_{lin}(x; a) = \sum_{i=1}^{m} a_i\sigma(\langle w_i, x\rangle) : a_i \in \mathbb{R}\},$$

$$\mathcal{F}_{NT}^m := \{f_{lin}(x; a) = \sum_{i=1}^{m}\langle a_i, x\rangle\sigma'(\langle w_i, x\rangle) : a_i \in \mathbb{R}^d\}.$$

即，$\mathcal{F}_{RF}^m$（RF 代表 '随机特征'）是通过对第二层权重线性化并保持第一层固定而获得的函数类，而 $\mathcal{F}_{NT}^m$（NT 代表 '神经切线'）是通过对第一层线性化并保持第二层固定而获得的函数类。

下面就 RF 和 NT 分别考察高维情形的泛化性能。

6.3 Random features model

在此语境当中，一个密切相关的方法是随机子集选择 (randomized subset selection)，也被称为 Nyström 方法。这里考虑高维情形 $m, n, d \to \infty$。在第

6.3.1 节中，讨论在更粗糙的尺度上的行为，即当 m 和 n 随着 d 多项式规模变化时：这种类型的分析为如何增大 m 以接近 $m = \infty$ 的极限提供了一个简单的定量答案。接下来，在第 6.3.2 节中，考虑成比例情形 $m \approx n \approx d$。这允许我们更精确地探索从欠参数化到过参数化过渡中发生的情况。

6.3.1 Polynomial scaling

此情形下有如下定理：

Theorem 6.1. Fix an integer $\ell > 0$. Let the activation function $\sigma : \mathbb{R} \to \mathbb{R}$ be independent of d and such that: (i) $|\sigma(x)| \le c_0 \exp(|x|^{c_1})$ for some constants $c_0 > 0$ and $c_1 < 1$, and (ii) $\langle \sigma, q \rangle_{L^2(\gamma)} \neq 0$ for any non-vanishing polynomial q, with $\deg(q) \le \ell$. Assume $\max((n/m), (m/n)) \ge d^{\delta}$ and $d^{\ell+\delta} \le \min(m, n) \le d^{\ell+1-\delta}$ for some constant $\delta > 0$. Then, for any $\lambda = O_d((m/n) \vee 1)$, and all $\eta > 0$,

$$L_{\mathsf{RF}}(\lambda) = \|\mathsf{P}_{>\ell} f^*\|_{L^2}^2 + o_d(1)(\|f^*\|_{L^2}^2 + \|\mathsf{P}_{>\ell} f^*\|_{L^{2+\eta}}^2 + \tau^2). \tag{6.18}$$

In words, as long as the number of parameters m and the number of samples n are well separated, the test error is determined by the minimum of m and n.

这表明测试误差在过参数化之前持续下降，在过参数化时，再增加 network size(m) 就不会对测试误差有显著提升。因此此情形下的回归模型和一般的 KRR 模型有着非常相似的数学性质。

具体的解释涉及到 feature matrix 的各个成分的分解：

considering the feature matrix $\mathbf{\Phi} \in \mathbb{R}^{n \times m}$:

$$\mathbf{\Phi} := \begin{bmatrix} \sigma(\langle \boldsymbol{x}_1, \boldsymbol{w}_1 \rangle) & \sigma(\langle \boldsymbol{x}_1, \boldsymbol{w}_2 \rangle) & \cdots & \sigma(\langle \boldsymbol{x}_1, \boldsymbol{w}_m \rangle) \\ \sigma(\langle \boldsymbol{x}_2, \boldsymbol{w}_1 \rangle) & \sigma(\langle \boldsymbol{x}_2, \boldsymbol{w}_2 \rangle) & \cdots & \sigma(\langle \boldsymbol{x}_2, \boldsymbol{w}_m \rangle) \\ \vdots & \vdots & & \vdots \\ \sigma(\langle \boldsymbol{x}_n, \boldsymbol{w}_1 \rangle) & \sigma(\langle \boldsymbol{x}_n, \boldsymbol{w}_2 \rangle) & \cdots & \sigma(\langle \boldsymbol{x}_n, \boldsymbol{w}_m \rangle) \end{bmatrix}. \tag{6.19}$$

可以把上述的 Φ 分解为高频和低频成分：

$$\mathbf{\Phi} = \mathbf{\Phi}_{\le \ell} + \mathbf{\Phi}_{>\ell}, \tag{6.21}$$

$$\mathbf{\Phi}_{\le \ell} = \sum_{j=0}^{k(\ell)} s_j \boldsymbol{\psi}_j \boldsymbol{\phi}_j^{\top} = \boldsymbol{\psi}_{\le \ell} \boldsymbol{S}_{\le \ell} \boldsymbol{\phi}_{\le \ell}^{\top}, \tag{6.22}$$

where $\boldsymbol{S}_{\le \ell} = \mathrm{diag}(s_1, \ldots, s_{k(\ell)})$, $\boldsymbol{\psi}_{\le \ell} \in \mathbb{R}^{n \times k(\ell)}$ is the matrix whose jth column is $\boldsymbol{\psi}_j$, and $\boldsymbol{\phi}_{\le \ell} \in \mathbb{R}^{m \times k(\ell)}$ is the matrix whose jth column is $\boldsymbol{\phi}_j$.

总结来说，相对于随机特征 $\sigma(\langle w_j, \cdot \rangle)$ 的回归实际上等同于相对于一个多项式核的岭回归，这个多项式核的度数 ℓ 取决于样本大小和网络大小中较小的一个。激活函数中的高阶部分实际上在回归器中表现为噪声。

文章阅读：The Implicit Regularization of Stochastic Gradient Flow for Least Squares

在一些梯度下降的隐正则化结果之上，很自然提出 SGD 对应的隐正则化。本文考察最小二乘场景下步长为定值的 SGD，给出了基于 SGD 的 excess risk 在时间 t 下的 upper bound，并且是以 $\lambda = 1/t$ 为参数的岭回归作为参照的。这个 bound 可以拆分成三个部分：第一项是 ridge 导致的方差；第二项是 price of stochasticity，随着 t 增大递减；第三项与 stochastic gradient flow 的优化误差有关，插值情形这一项为 0，反之为正，这表明 SGD 的结果会在最小二乘解附近跳动。

考虑常见的最小二乘回归问题：

$$\min_{\beta \in \mathbb{R}^p} \frac{1}{2n} \|y - X\beta\|^2 \tag{1}$$

其中 $y \in \mathbb{R}^n$ 是响应变量，$X \in \mathbb{R}^{n \times p}$ 是数据矩阵。minibatch SGD 迭代如下：

$$\begin{aligned} \beta^{(k)} &= \beta^{(k-1)} - \frac{\varepsilon}{m} \sum_{i \in I_k} (y_i - x_i^T \beta^{(k-1)}) x_i \\ &= \beta^{(k-1)} + \frac{\varepsilon}{m} \cdot X_{I_k}^T (y_{I_k} - X_{I_k} \beta^{(k-1)}) \end{aligned} \tag{2}$$

对于 $k = 1, 2, 3, ...$，其中 $\varepsilon > 0$ 是固定步长，m 是小批量大小，$I_k \subseteq \{1, ..., n\}$ 表示第 k 次迭代的小批量，且 $|I_k| = m$，对所有 k 成立。

这可以写为梯度下降，加上 m 个独立同分布随机变量样本平均与它们均值之间的偏差：

$$d\beta(t) = -\frac{1}{n} X^T (y - X\beta(t)) dt + Q_e(\beta(t))^{1/2} dW(t), \tag{3}$$

其中 $\beta(0)=0$。这里，$W(t)$ 是标准的 p 维布朗运动。我们将扩散系数表示为

$$Q_e(\beta)=\varepsilon\cdot \mathrm{Cov}_I\left(\frac{1}{m}X_I^T(y_I-X_I\beta)\right), \tag{4}$$

随机性是由于 $I\subseteq\{1,\dots,n\}$ 产生的。我们称扩散过程 (3) 为随机梯度流 (Stochastic Gradient Flow, SGF)。注意 SGD 和 SGF 要区分开来。

文章阅读：Fit without fear: remarkable mathematical phenomena of deep learning through the prism of interpolation

the blessing of dimension

利用 Gaussian 或 Laplacian kernel 的插值分类器，甚至在标签受到污染的情况下也可以达到接近 Bayes optimal 的效果。在标签污染率 q 给定时，为了说明核方法的泛化性能，需要证明：

$$\frac{q}{2}\le O^*\left(\sqrt{\frac{\mathrm{cap}(\mathcal{H},X)}{n}}\right)\le\frac{1}{2} \tag{5}$$

因为在这里，1/2 是 Random guess 的 Bayes risk。左边的不等式较为显然，而右边的不等式就显得不是很平凡。维数的增大可能也是一种 blessing。考虑单纯形插值器，对于大的 d，函数 f_{simp} 的超额风险随维度减少：

$$R(f_{\mathrm{simp}})-R(f^*)=O\left(\frac{1}{\sqrt{d}}\right).$$

这个现象的直观解释是，影响预测的数据点在 noisy training points 附近才会受到影响，而这个邻域随着维数增大，占的体积是在逐渐变小的。

Random Fourier Features

随机傅里叶特征（*RFF*）。随机傅里叶特征空间 $\mathcal{H}_m$ 由 m 个复值参数构成的函数集 $f:\mathbb{R}^d\to\mathbb{C}$，这些函数的形式为

$$f(w,x)=\sum_{k=1}^{m} w_k e^{-\sqrt{-1}(v_k,x)}$$

其中向量 $v_1,\dots,v_m$ 是固定权重，其值独立地从 $\mathbb{R}^d$ 上的标准正态分布中采样。向量 $w=(w_1,\dots,w_m)\in\mathbb{C}^m\approx\mathbb{R}^{2m}$ 包含了可训练的参数。$f(w,x)$ 可以被看作是一个隐藏层大小为 m 并且第一层权重固定的神经网络。

给定数据 $\{x_i,y_i\},i=1,\dots,n$，我们可以通过线性回归在系数 w 上拟合 $f_m\in\mathcal{H}_m$。在过参数化范式下，线性回归通过最小化在插值约束条件下的范数给出

$$f_m(x)=\arg\min_{f\in\mathcal{H}_m, f(w,x_i)=y_i}||w||.$$

有结果：

$$\lim_{m\to\infty} f_m(x)=\arg\min_{f\in S\subseteq\mathcal{H}_k}||f||_{\mathcal{H}_k}=f_{\text{kern}}(x)$$

这里 $\mathcal{H}_k$ 是对应于高斯核 $K(x,z)=\exp(-||x-z||^2)$ 的 RKHS，而 $S\subseteq\mathcal{H}_k$ 是 $\mathcal{H}_k$ 中插值函数的流形。

论文阅读：Belkin, Ma, and Mandal (2018) , To Understand Deep Learning We Need to Understand Kernel Learning。

现有的 bound 在 kernel learning 中并不能很好地解释

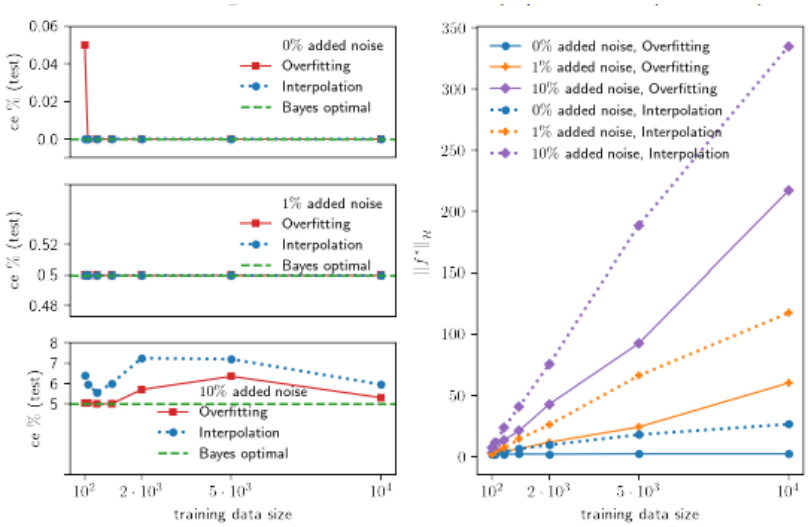
文中提到，现有的有关 kernel learning 的 bounds 必须依赖于 RKHS 范数的多项式速度增长，然而，对于 Interpolated Kernel Classifiers，文中 Theorem 1 指出该分类器的范数是呈指数增长的，因此现有的 bounds 并不奏效。

Theorem 1. *Let $(\boldsymbol{x}_i, y_i), i = 1, \ldots, n$ be data sampled from P on $\Omega \times \{-1, 1\}$. Assume that y is not a deterministic function of x on a subset of non-zero measure. Then, with high probability, any h that t-overfits the data, satisfies*

$$\|h\|_{\mathscr{H}} > Ae^{Bn^{1/d}}$$

for some constants $A, B > 0$ depending on t.

事实上，overfitted 分类器（用优化方法得到）比 Interpolated 分类器具有更小的测试误差和更小的 RKHS 范数。



Gaussian and Laplacian kernels

高斯核相比于 laplace 核需要更多的 epochs，而 laplace 核在高维情形下展现出了类似于 ReLU 的性质。

待解决的问题

- 待复现：Belkin, Ma, and Mandal (2018)，To Understand Deep Learning We Need to Understand Kernel Learning。该文章给出了一些实验的详细细节，涉及到 kernel 的编程，值得复现。

- 和老师讨论技术文档的具体内容，寻找合适的 idea。

参考文献

Ali, Alnur, Edgar Dobriban, and Ryan J. Tibshirani. 2020. “The Implicit Regularization of Stochastic Gradient Flow for Least Squares.” https://arxiv.org/abs/2003.07802.

Bartlett, Peter L., Andrea Montanari, and Alexander Rakhlin. 2021. “Deep Learning: A Statistical Viewpoint.” https://arxiv.org/abs/2103.09177.

Belkin, Mikhail. 2021. “Fit Without Fear: Remarkable Mathematical Phenomena of Deep Learning Through the Prism of Interpolation.” https://arxiv.org/abs/2105.14368.

Belkin, Mikhail, Siyuan Ma, and Soumik Mandal. 2018. “To Understand Deep Learning We Need to Understand Kernel Learning.” In *Proceedings of the 35th International Conference on Machine Learning*, edited by Jennifer Dy and Andreas Krause, 80:541–49. Proceedings of Machine Learning Research. PMLR. https://proceedings.mlr.press/v80/belkin18a.html.