arXiv:2304.02641v1 [stat.ML] 5 Apr 2023

高斯过程回归与分类的自蒸馏

Kenneth Borup
数学系 奥胡斯大学
kennethborup@math.au.dk

Lars Nørvang Andersen数学系 奥胡斯大学 larsa@math.au.dk

摘要

我们提出了两种方法，将知识蒸馏的概念扩展到高斯过程回归（GPR）和高斯过程分类（GPC）：数据中心方法和分布中心方法。数据中心方法类似于当前大多数机器学习蒸馏技术：利用教师给出的确定性预测重新拟合模型；而分布中心方法则在下一次迭代中重复使用完整的概率后验。通过分析这些方法的性质，我们表明，针对高斯过程回归的数据中心方法与核岭回归自蒸馏的已知结果密切相关，而针对高斯过程回归的分布中心方法则对应于采用非常特殊的超参数选择的普通高斯过程回归。此外，我们证明，针对GPC的分布中心方法大致对应于数据复制以及对协方差进行特定缩放；而针对GPC的数据中心方法需要将模型从二项似然重新定义为连续伯努利似然，才能得到良好设定。据我们所知，我们提出的方法首次针对高斯过程模型专门构建了知识蒸馏方法。

代码：github.com/Kennethborup/gaussian_过程_自我_蒸馏

1 引言与问题设定

在本文中，我们提出高斯过程（高斯过程）背景下知识蒸馏（知识蒸馏）的概念。尽管在深度学习背景下对（扩展的）知识蒸馏有大量研究和应用，但对深度学习以外的其他机器学习方法的知识蒸馏研究却很缺乏。近期工作由 Borup & Andersen （2021年）；Mobahi 等人 （2020年）；Phuong & Lampert （ 2019 年）；Frosst & Hinton （ 2018 年）等人提供了简化设置下知识蒸馏的结果，例如（核岭）回归。这些结果的一个自然扩展是研究高斯过程的知识蒸馏，但由于在此背景下尚未开展先前工作，我们最初提出多种方法来如何定义高斯过程的知识蒸馏。在本文中，我们将分析限制在自蒸馏，即学生和教师模型属于同一类，这在实践中对应于两个模型的核函数相同。

不同于例如深度学习，在深度学习中，样本和模型在训练期间是确定性的，而高斯过程以先验和后验预测分布的形式包含随机性。在普通高斯过程回归中，人们定义一个先验，并基于一组观测值 $\mathcal{D}$，从而获得一个后验模型，该模型可用于对之前未见过的样本进行推断。我们提出了两种不同且直接的高斯过程自蒸馏方法： 数据中心 和 分布中心（见图 2）。

以数据为中心的自蒸馏。在以数据为中心的蒸馏中，我们将模型在步骤 t 时的输出预测作为模型在步骤 $t+1$时的输入。此过程与当前大多数蒸馏

预印本。审稿中。

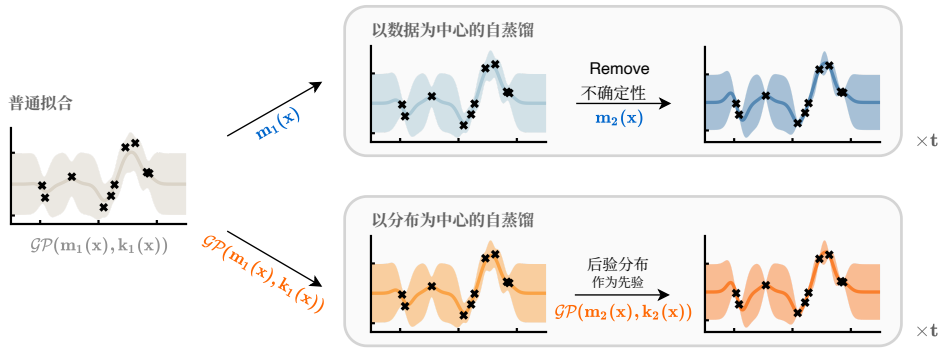


图1：我们提出的两种高斯过程蒸馏方法的概念图示。在以数据为中心的自蒸馏中，我们在下一步中重用前一解的均值函数；而在以分布为中心的自蒸馏中，我们重用完整分布。

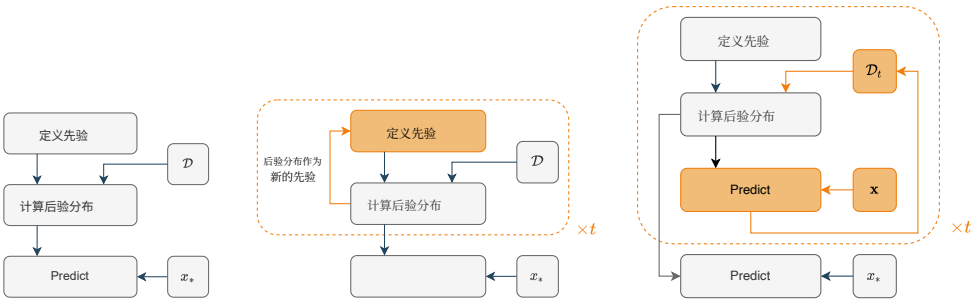
深度学习中已知的方法（Hinton et al.，2015年）。然而，与深度学习不同，我们的模型输出的不只是标量，而是分布。我们提出，在步骤 t 舍弃预测的随机性，仅保留均值预测，用于步骤 $t+1$模型的训练。我们表明，对于高斯过程回归，自蒸馏后的均值与Borup & Andersen （2021年）和Mobahi 等人 （2020年）所述的核岭回归蒸馏密切相关。此外，我们认为，在分类场景中进行蒸馏时，需要调整似然函数，使其支持连续目标，而非二元目标。我们通过使用连续伯努利分布（Loaiza-Ganem & Cunningham，2019年）解决这一问题。我们在第3.1 节研究回归的数据中心蒸馏，在第4.1节研究分类的数据中心蒸馏。

以分布为中心的自蒸馏。 在以分布为中心的蒸馏中，我们将给定步骤的先验 t 替换为前一步的后验分布 $t-1$。在高斯过程回归中，后验分布本身就是一个高斯过程，我们可以直接将其视为后续步骤的先验。这会在每个蒸馏步骤中产生一个迭代精炼且依赖于数据的先验。对于分类，由于后验分布无法解析已知，需要更加谨慎。然而，通过将后验分布的拉普拉斯近似用作潜在高斯过程模型的先验，我们也能在分类设置中定义有意义的以分布为中心的蒸馏概念。我们在第3.2 节研究回归情形，在第 4.2节研究分类情形。

结合与扩展方向。 按照文献中的常见做法，可以考虑扩展以数据为中心的蒸馏方法：我们不仅考虑前一步的后验预测，还考虑这些后验预测与原始观测的凸组合（权重为 $\alpha \in (0,1)$）。此外，由于以数据为中心和分布中心的方法并非互斥，我们可以考虑一种结合内外层蒸馏循环的组合方法。在外层循环中，先验会像分布中心蒸馏那样替换为前一步的后验；而在内层循环中，则使用以数据为中心的蒸馏，在外层循环中固定先验，并对后验进行固定步数的蒸馏。我们将这些复杂但颇具吸引力的设定留待未来研究。

我们将我们的**贡献** 总结如下：

- 我们提出了两种适用于高斯过程回归和高斯过程分类的自蒸馏方法，分别重用模型前一次迭代的均值预测（数据中心）或完整预测分布（分布中心）。
- 我们展示了高斯过程回归的以数据为中心的自蒸馏与核岭回归的自蒸馏之间的联系。此外，我们证明了高斯过程回归的以分布为中心的自蒸馏等价于采用特定超参数选择的普通高斯过程回归。
- 我们发现，直接采用以数据为中心的自蒸馏方法进行高斯过程分类，会因预测结果的连续形式而产生设定错误的模型。为此，我们提出使用连续伯努利分布来缓解这一问题。此外，我们发现



(a) 经典。 (b) 分布中心。 (c) 数据中心。

图 2：在（a）中，我们展示了将高斯过程拟合到观测数据集 $\mathcal{D} = \{(x_i,\ y_i)\}_{i=0}^n$，并为某个未见样本 x_* 计算输出分布的典型过程。在（b）中，我们展示了以分布为中心的自蒸馏过程，其中我们将后验分布重新用作先验。在（c）中，我们展示了以数据为中心的自蒸馏过程，在该过程中，我们迭代地将训练观测 $\mathcal{D}_t = \{(x_i,\ y_{t,\ i})\}_{i=0}^n$ 替换为当前模型的后验预测 $\mathcal{D}_{t+1} = \{(x_i,\ y_{t+1,\ i})\}_{i=0}^n$，并将模型重新拟合到这些样本。

表明，以分布为中心的自蒸馏可以通过对协方差函数进行特定缩放来高效近似，适用于任意数量的蒸馏步骤。

本文的结构如下。在第 2 节中，我们介绍高斯过程回归和高斯过程分类的初步结果。接下来，我们在第 3节重点讨论高斯过程回归的自蒸馏，尤其是第3.1节中的以数据为中心的蒸馏以及第 3.2节中的分布中心蒸馏。随后，在第 4节中，我们研究分类设置，尤其是第 4.1 节中的数据中心设置和第 4.2节中的分布中心设置。最后，我们在第 5节将我们的工作与现有研究联系起来。此外，我们在附录中提供了更多数学细节、实现细节、实验消融以及所有证明。

2 预备知识

我们的出发点是高斯过程回归的标准设置。这些结果是众所周知的（参见例如 Rasmussen & Williams (2006)）。

2.1 高斯过程回归

我们假设一个通用的输入-输出对 $(\mathbf{x}\ y) \in \mathbb{R}^d \times \mathbb{R}$ 通过 $y = f(\mathbf{x}) + \varepsilon$相关联，其中 $f(\cdot)$ 是一个函数，我们希望基于 f 是一个随机函数的假设来推断它，其先验分布为 $f \sim \mathcal{GP}(m, k)$，均值函数为 $m\colon \mathbb{R}^d \to \mathbb{R}$，半正定协方差函数（通常称为核函数）为 $k\colon \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$，并且 ε是一个独立的噪声项，服从 $\varepsilon \sim \mathcal{N}(0, \gamma)$。为了符号方便，我们假设输入是一元的，即 $d = 1$。对于训练集$\mathcal{D} = \{(x_i, y_i) \mid i = 1, \ldots, N\}$和测试集 $(x_{*1}, \ldots, x_{*M})$，首要任务是描述给定 $\mathcal{D}$ 时 $\mathbf{f}_* = (f(x_{*1}) \ldots, f(x_{*M}))$ 的后验分布。$\mathbf{f}_*$给定 $\mathcal{D}$ 的后验预测分布是

$$\begin{aligned}
\mathbf{f}_* \mid \mathbf{x}, \mathbf{y}, \mathbf{x}_* &\sim \mathcal{N}(\boldsymbol{\mu}_*, \boldsymbol{\Sigma}_*), \quad \text{where} \\
\boldsymbol{\mu}_* &= m(\mathbf{x}_*) + k(\mathbf{x}_*, \mathbf{x}^\intercal)(\mathbf{K} + \gamma \mathbf{I}_N)^{-1}\left(\mathbf{y} - m(\mathbf{x})\right), \\
\boldsymbol{\Sigma}_* &= k(\mathbf{x}_*, \mathbf{x}_*^\intercal) - k(\mathbf{x}_*, \mathbf{x}^\intercal)(\mathbf{K} + \gamma \mathbf{I}_N)^{-1} k(\mathbf{x}, \mathbf{x}_*^\intercal).
\end{aligned}$$

其中 $\mathbf{x}$ 和$\mathbf{x}_*$ 分别是堆叠的训练输入和测试输入。在 $\boldsymbol{\mu}_*$ 和 $\boldsymbol{\Sigma}_*$ 的表达式中，我们使用了以下符号

$$g(\mathbf{x}, \mathbf{y}^\intercal) = \{g(x_i, y_j)\}_{i,j=1}^{N,M} \in \mathbb{R}^{N \times M}$$

其中 $g\colon \mathbb{R}^2 \ni (x,y) \mapsto g(x,y) \in \mathbb{R}$，且 $\mathbf{x} = (x_1,\ldots,x_N) \in \mathbb{R}^N$，$\mathbf{y} = (y_1,\ldots,y_M) \in \mathbb{R}^M$是列向量。注意，如果任一参数是标量，则有

$$g(x,\mathbf{x}^\intercal) = (g(x,x_1),\ldots,g(x,x_N)) \in \mathbb{R}^{1\times N}$$
$$g(\mathbf{x},y) = (g(x_1,y),\ldots,g(x_N,y))^\intercal \in \mathbb{R}^{N\times 1}$$

其中 $g(x,\mathbf{x}^\intercal)$ 是行向量，$g(\mathbf{x},y)$ 是列向量。

在全文中，我们将用 $\mathbf{K}$ 表示矩阵 $\mathbf{K} = k(\mathbf{x},\mathbf{x})$，并假设该矩阵可逆。实际上，如果 $\mathbf{K}$ 不可逆，可以在对角线元素上加一个小常数。此外，为了便于阐述，我们通常会省略对$\mathbf{x}$ 和 $\mathbf{x}_*$ 的条件化。

2.2 高斯过程分类

我们遵循高斯过程分类的通常设定，其中每个输入-输出对（$\mathbf{x}$，y）都基于这样的假设相关联：条件分布 $y \mid f(\mathbf{x})$ 是伯努利分布，其概率为 $\sigma(f(\mathbf{x}))$，其中 $\sigma(\cdot)$ 是逻辑斯蒂函数，并且 f 的先验分布为 $\mathcal{GP}(m,k)$。对于训练集 $\mathcal{D} = \{(x_i,y_i) \mid i = 1,\ldots,N\}$ 我们记 $\mathbf{f} = (f_n)_{n=1}^N = (f(\mathbf{x}_1)\ldots,f(\mathbf{x}_N))$ 和 $\mathbf{y} \vDash (y_1,\ldots,y_n)$，从而

$$\mathbf{y} \mid \mathbf{f} \sim \prod_{n=1}^{N} \sigma(f_n)^{y_n}(1-\sigma(y_n))^{1-y_n}\,. \tag{1}$$

对数后验的形式为

$$\psi(\mathbf{f}) \stackrel{\text{def}}{=} \log p(\mathbf{f} \mid \mathbf{y}) \underset{\mathbf{f}}{\propto} \log p(\mathbf{f}) + \log p(\mathbf{y} \mid \mathbf{f}),$$

其中 $p(\mathbf{f})$ 是多元高斯分布的概率密度函数，由（1）可得

$$\log p(\mathbf{y} \mid \mathbf{f}) = \mathbf{y}^\intercal \mathbf{f} - \sum_{n=1}^{N} \log(1+\exp(f_n))\,.$$

与分类情况不同，后验分布 $\mathbf{f} \mid \mathbf{y}$ 无法进行解析求解，需要进行近似。这一问题并非蒸馏所特有，而是涉及高斯过程分类的任何实际应用都会出现的问题；人们已经提出了多种方法来获得后验分布的近似值。在本文中，我们将重点讨论拉普拉斯近似，其中后验分布通过

$$q(\mathbf{f} \mid \mathbf{y}) = \mathcal{N}\left(\mathbf{f} \mid \hat{\mathbf{f}}, (\mathbf{K}^{-1} + \mathbf{W})^{-1}\right)$$

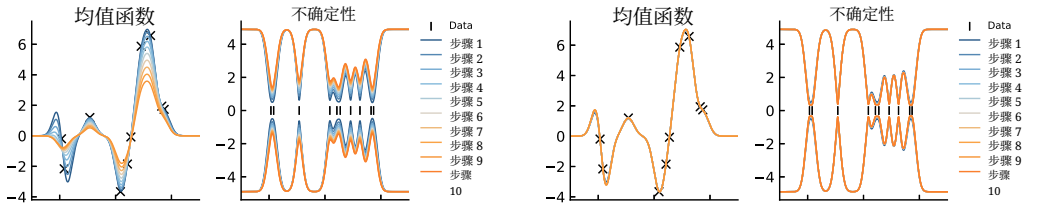
其中 $\hat{\mathbf{f}} = (\hat{f}_n)_{n=1}^N$ 是 $\psi(\mathbf{f})$ 的众数，$\mathbf{W}$ 是 $N \times N$ 矩阵 $\operatorname{diag}_n\{\sigma(\hat{f}_n)(1-\sigma(\hat{f}_n))\}$。近似后验预测分布可在 Rasmussen & Williams（2006，参见第 3.4.2 节）中找到；在 $m = 0$ 的情况下，对于一般的先验均值 m，我们有 $p(f_* \mid \mathbf{x},\mathbf{y}) \approx \mathcal{N}(\boldsymbol{\mu}_*, \boldsymbol{\Sigma}_*)$，其中

$$\begin{aligned}
\boldsymbol{\mu}_* &= \mathbb{E}_q[\mathbf{f}_* \mid \mathbf{y},\mathbf{x}_*] \\
&= m(\mathbf{x}_*) + k(\mathbf{x}_*,\mathbf{x}^\intercal)\mathbf{K}^{-1}(\hat{\mathbf{f}} - m(\mathbf{x})) \qquad (2)\\
\boldsymbol{\Sigma}_* &= \operatorname{Cov}_q(\mathbf{f}_* \mid \mathbf{y},\mathbf{x}_*) \\
&= k(\mathbf{x}_*,\mathbf{x}_*^\intercal) - k(\mathbf{x}_*,\mathbf{x}^\intercal)(\mathbf{K}+\mathbf{W}^{-1})^{-1}k(\mathbf{x},\mathbf{x}_*^\intercal). \qquad (3)
\end{aligned}$$

请参见第 A.1 ，了解有关高斯过程分类的更多详细信息。我们将在第4节回到分类设置。

3 用于高斯过程回归的自蒸馏

下面我们分别考虑以数据和分布为中心的自蒸馏在高斯过程回归（高斯过程回归）中的应用。



(a) 以数据为中心的蒸馏进行十步。　　(b) 分布中心蒸馏十步。

图 3：对于数据中心方法和分布中心方法，在固定超参数以及 $(\gamma_1, \ldots, \gamma_{10}) = (0.1, \ldots, 1)$ 的情况下进行十步自蒸馏。我们绘制每一步的不确定性估计的均值函数以及第 2.5 和第 97.5 百分位数。

3.1 以数据为中心的自蒸馏

对于以数据为中心的蒸馏的一般设置，我们考虑均值为零的先验分布，$f \sim \mathcal{GP}(0, k)$，以及在步骤 $t \geq 1$ 时关于 $\mathbf{x} = (x_1, \ldots, x_N)$ 的后验分布为 $\mathbf{f}_t \mid \mathbf{y}_{t-1}{}^{\intercal} \sim \mathcal{N}(\boldsymbol{\mu}_t, \boldsymbol{\Sigma}_t)$，其中

$$\boldsymbol{\mu}_t = \mathbf{K}(\mathbf{K} + \gamma_t \mathbf{I}_N)^{-1} \mathbf{y}_{t-1},$$
$$\boldsymbol{\Sigma}_t = \mathbf{K} - \mathbf{K}(\mathbf{K} + \gamma_t \mathbf{I}_N)^{-1} \mathbf{K},$$

并且 $\gamma_t > 0$ 对于 $t \geq 1$。以数据为中心的自蒸馏对应于使用均值预测 $\boldsymbol{\mu}_t$ 代替 $\mathbf{y}_t$ 来拟合后续步骤，从而将后续模型拟合到当前模型的均值预测。在此设置下，均值函数的行为等价于Mobahi 等人 (2020年) 和 Borup &Andersen (2021年)，如下方结果所示。

命题 3.1。 定义 $\mathbf{y}_t \overset{\text{定义}}{=} \boldsymbol{\mu}_t$ 和 $\mathbf{y}_0 \overset{\text{定义}}{=} \mathbf{y}$。那么对于任意 $t \geq 1$，均有 $\mathbf{f}_t \mid \boldsymbol{\mu}_{t-1} \sim \mathcal{N}(\boldsymbol{\mu}_t, \boldsymbol{\Sigma}_t)$，其中

$$\mathbf{y}_t = \left(\prod_{s=1}^{t} \mathbf{K}(\mathbf{K} + \gamma_s \mathbf{I}_N)^{-1} \right) \mathbf{y}, \tag{4}$$

$$\mathbb{E}[\mathbf{f}_{t,*} \mid \boldsymbol{\mu}_{t-1}] = k(\mathbf{x}_*, \mathbf{x})(\mathbf{K} + \gamma_t \mathbf{I}_N)^{-1} \mathbf{y}_{t-1}, \tag{5}$$

以及 $\boldsymbol{\Sigma}_t = \mathbf{K} - \mathbf{K}(\mathbf{K} + \gamma_t \mathbf{I}_N)^{-1} \mathbf{K}$ 以及 $\mathbf{y}_t$ 仅受 γ_t 的选择影响。附录。*(Proof in*

可以证明，$\mathbf{y}_t$ 随着 t 的增加对应于通过最大程度地收缩具有最小特征值的基函数来逐步稀疏化底层基函数，从而对预测施加正则化效应（Mobahi 等人，2020年；Borup & Andersen，2021年）。此外，由于 $\mathbf{y}_t$ 仅依赖于 t 通过 $\{\gamma_s\}_{s=1}^{t}$，因此可以高效地计算（4）对于任何选择的 t，通过使用$\mathbf{K}$的谱分解。
此外，遵循 Borup & Andersen (2021年) 结果可以扩展为包括加权（通过 α) 原始训练观测以及（通过 $1 - \alpha$) 模型先前迭代的预测。

3.2 以分布为中心的自蒸馏

分布中心方法的核心观察是，后验分布本身就是一个高斯过程，即 $\mathcal{GP}(m_*, k_*)$，其中

$$m_*(x) = m(x) + k(x, \mathbf{x}^{\intercal}) \left(\mathbf{K} + \gamma \mathbf{I}_N\right)^{-1} (\mathbf{y} - m(\mathbf{x}))$$
$$k_*(x, y) = k(x, y) - k(x, \mathbf{x}^{\intercal}) \left(\mathbf{K} + \gamma \mathbf{I}_N\right)^{-1} k(\mathbf{x}, y)\,.$$

由于后验分布是一个高斯过程，我们可以将后验分布用作初始设置中的先验分布，从而对后验分布进行“蒸馏”，并且可以将这一过程迭代任意有限次数，见图 1（b）。此外，我们可以选择噪声项 γ_t 使其不同

在每一步中，我们看到蒸馏过程会产生一个序列 的高斯过程$\mathcal{GP}(m_t,\ k_t)$，其中均值和协方差函数满足递归

$$m_{t+1}(x) = m_t(x) + k_t(x, \mathbf{x}^\intercal)[\mathbf{K}_t + \gamma_t \mathbf{I}_N]^{-1}(\mathbf{y} - m_t(\mathbf{x})) \tag{6}$$

$$k_{t+1}(x, y) = k_t(x, y) - k_t(x, \mathbf{x}^\intercal)[\mathbf{K}_t + \gamma_t \mathbf{I}_N]^{-1} k_t(\mathbf{x}, y) \tag{7}$$

其中 $\mathbf{K}_t \overset{\text{def}}{=} k_t(\mathbf{x}, \mathbf{x}), {}^\intercal m_0(x) \overset{\text{def}}{=} 0$, $k_0(x, y) \overset{\text{def}}{=} k(x, y)$ and $\gamma_t > 0, t = 0, 1, \ldots$.

上述递归关系（6）和（7）的解由下文的定理3.3 给出。我们将证明，使用惩罚参数 $\gamma_0, \gamma_1, \ldots, \gamma_{t-1}$ 递推 t 次，所得结果与使用惩罚参数 $1/\gamma_{t-1}^-$ 进行单次递推相同，其中 $\gamma_t^- \overset{\text{def}}{=} \sum_{s=0}^{t} 1/\gamma_s$，且 $\gamma_{-1}^- \overset{\text{def}}{=} 0$。为证明这一点，我们首先需要一个引理。

引理 3.2。 设 $\lambda_i^{(t)}, for\ i = 1, \ldots, N$ 是 $\mathbf{K}_t$ 的特征值。记 $\mathbf{K}_0 = \mathbf{O}\,\mathrm{diag}_i(\lambda_i^{(0)})\mathbf{O}$ 为 $\mathbf{K}_0$ 的谱分解。那么对于 $t = 0, 1, \ldots,$ 有

$$\mathbf{K}_t = \mathbf{O}\mathrm{diag}_i\left(\lambda_i^{(0)} \zeta_i^{(t)}\right)\mathbf{O}^\intercal \tag{8}$$

其中 $\zeta_i^{(t)} \overset{\text{def}}{=} \frac{1}{\lambda_i^{(0)} \gamma_{t-1}^- + 1}$。对于 $k_t(x, \mathbf{x}), {}^\intercal$以及 $m_t(\mathbf{x})$，我们有

$$k_t(x, \mathbf{x}^\intercal) = k_0(x, \mathbf{x}^\intercal)\mathbf{O}\mathrm{diag}_i\left(\zeta_i^{(t)}\right)\mathbf{O}^\intercal \tag{9}$$

$$m_t(\mathbf{x}) = \mathbf{O}\mathrm{diag}_i\left(\lambda_i^{(0)} \gamma_{t-1}^- \zeta_i^{(t)}\right)\mathbf{O}^\intercal \mathbf{y}. \tag{10}$$

（证明见附录。）

现在我们可以证明本节的主要结果。

定理 3.3。 递归(6) 和 (7) 的解为

$$m_t(x) = k_0(x, \mathbf{x}^\intercal)(\mathbf{K} + \mathbf{I}_N/\gamma_{t-1}^-)^{-1}\mathbf{y} \tag{11}$$

$$k_t(x, y) = k_0(x, y) - k_0(x, \mathbf{x}^\intercal)(\mathbf{K} + \mathbf{I}_N/\gamma_{t-1}^-)^{-1} k_0(\mathbf{x}, y)$$

对于 $t = 1, 2, \ldots$.

（证明见附录。）

由定理 3.3可知，执行多个以分布为中心的自蒸馏步骤，等价于在特定噪声参数下拟合一个普通高斯过程。

接下来，我们注意到，若假设所有蒸馏步骤中的噪声参数相同，即$\gamma_t = \gamma$对 $t = 0,\ 1, \ldots$ ，则 $\gamma_{t-1}^- = t/\gamma$这相当于对数据集 $\mathcal{D}_t$拟合一个高斯过程，其中该数据集由 t 个带噪声参数 γ的 $\mathcal{D}$ 副本组成。其含义由以下推论精确定义。

推论3.4。 设

$$\mathcal{D}_t = \{(x_{ij}, y_{ij}) \mid i = 1, \ldots, N,\quad j = 1, \ldots, t\}$$

其中，对所有 i和 j，$x_{ij} = x_i$成立，则 $\mathbf{f}$的后验分布为 $\mathcal{N}(\boldsymbol{\mu}_t, \boldsymbol{\Sigma}_t)$，其中

$$\boldsymbol{\mu}_t = \begin{pmatrix} \mathbf{K}(\mathbf{K} + \gamma/t\mathbf{I}_N)^{-1}\mathbf{y} \\ \vdots \\ \mathbf{K}(\mathbf{K} + \gamma/t\mathbf{I}_N)^{-1}\mathbf{y} \end{pmatrix} \tag{12}$$

$$\boldsymbol{\Sigma}_t = \mathbf{1}\mathbf{1}^\intercal \otimes (\mathbf{K} - \mathbf{K}(\mathbf{K} + \gamma/t\mathbf{I}_N)^{-1}\mathbf{K}) \tag{13}$$

其中 $\otimes$ 是克罗内克积。 *(Proof in appendix.)*

3.3 说明性示例

我们现在考虑一个说明性示例，其中我们假设一个真实函数 $g(z) = z\sin(z)$，10 个训练样本 $\mathbf{x}$在 0 和 10 之间等距分布，以及观测值 $y_i = g(x_i) + \varepsilon_i$，其中$\varepsilon_i$ 服从标准正态分布。我们使用具有两个超参数的缩放径向基核函数σ和 l 以及噪声参数 γ_t。参见第 B 节以获取有关设置的更多详细信息。

现在我们考虑10 个步骤，在所有步骤中使用固定的超参数，分别进行以数据为中心的自蒸馏和以分布为中心的自蒸馏。在图 3a 和图 3b 中，我们绘制了以数据为中心的蒸馏结果，其中 $(\gamma_1, \ldots\ldots, \gamma_{10}) = (0。1, \ldots\ldots, 1)$。在这里，我们观察到均值的正则化如预期般增加，并且不确定性随着每个步骤而增加，这也符合对 $(\gamma_1, \ldots\ldots, \gamma_{10})$的选择所产生的预期结果。这与我们的理论非常一致。然而，对于以分布为中心的情况，均值函数几乎不受蒸馏过程的影响，不确定性估计也没有明显变化。这是由于 $1/\gamma_t^-$在 t 从 1 变为 10 时变化相对较小，因为 $1/\gamma_1^- = 0.1$ 和$1/\gamma_{10}^- = 0.034$，所以每个蒸馏步骤产生的变化都很小。有关采用不同参数选择的其他示例，请参见附录 B 。

4 自蒸馏 for高斯过程分类

In t下面我们将分别研究用于 G 的数据中心和以分布为中心的自蒸馏。 P
clas分类。

4.1 以数据为中心的自蒸馏

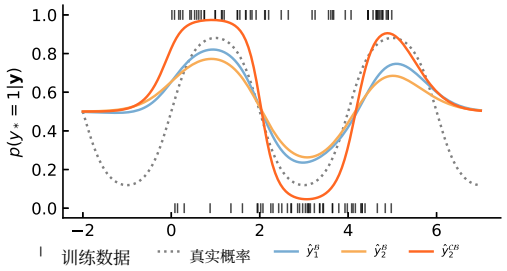


图 4：采用连续伯努利分布和普通伯努利似然的普通 GPC 模型及一步以数据为中心的蒸馏。我们注意到，伯努利似然虽然表现良好，但并不构成设定合理的模型。

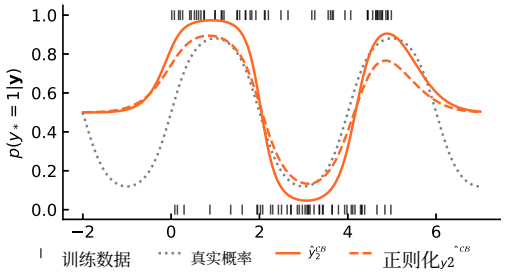


图5：如图所示，采用连续伯努利似然进行数据中心蒸馏的一步 4，其中训练观测上的噪声假设所施加的正则化既有启用也有未启用。调整噪声参数 γ_1 调整正则化。

类似于高斯过程回归的以数据为中心的方法，我们将在本节中将后验均值预测视为模型下一次迭代的确定性目标。特别地，在每一步t，我们假设 f_t 的先验为 $\mathcal{GP}(\mathbf{0}, k)$，并且对于 $t = 1$ 我们将获得预测，$\hat{}\mathbf{y}_0$ 作为基于拉普拉斯近似的平均后验预测，例如 $\hat{}\mathbf{y}_0 = \mathbb{E}_q[\sigma(\mathbf{f}_1) \mid \mathbf{y}_0]$。然后这些预测将用于替代步骤 $t = 2$的原始观测，这将导致新的预测 $\hat{}\mathbf{y}_2$, 我们可以在后续步骤中使用，依此类推。

由于原始观测$\mathbf{y}_0$, 位于 $\{0, 1\}^N$ 而预测值$\hat{}\mathbf{y}_1$, 位于 $[0, 1]^N$，因此在后续步骤中直接将预测值$\hat{}\mathbf{y}_1$, 替代观测值$\mathbf{y}_0$, 会导致模型设定错误，因为伯努利分布的支持集仅为 $\{0, 1\}$。为缓解这一模型设定错误，我们重新定义 $t \geq 2$ 的模型，假设其服从条件连续伯努利分布，而不是通常的条件伯努利分布（Loaiza-Ganem 和 Cunningham，2019年）。我们有时将此方法称为 C-GPC。在实践中，这要求将式（1）的每个元素乘以适当的归一化常数 C（参见式（20）和附录C）。也就是说，对于 $t \geq 2$ 我们假设

$$\mathbf{y}_{t-1} \mid \mathbf{f}_t \sim \prod_{n=1}^{N} C(\sigma(f_{t,n}))\sigma(f_{t,n})^{y_{t-1,n}}(1 - \sigma(f_{t,n}))^{1-y_{t-1,n}},$$

这继而会通过 $\log\ \tilde{p}(\mathbf{f}_t \mid \mathbf{y}_{t-1})$ 影响对数后验 $\log\ \tilde{p}(\mathbf{y}_{t-1} \mid \mathbf{f}_t)$，后者现在变为

$$\log \tilde{p}(\mathbf{y}_{t-1} \mid \mathbf{f}_t) = \log p(\mathbf{y}_{t-1} \mid \mathbf{f}_t) + \sum_{n=1}^{N} \log C(\sigma(f_{t,n})).$$

此外，这也会影响用于获得拉普拉斯近似的 $\log\ \tilde{p}(\mathbf{f}_t \mid \mathbf{y}_{t-1})$ 的梯度和海森矩阵，尽管$\log C(\lambda)$ 和 $\frac{d}{d\lambda} \log C(\lambda)$ 都无法得到简单的表达式，但我们展示了

命题 4.1 表明，$(\sigma(a))$ 以及导数 $\frac{d}{da}\log C(\sigma(a))$ 和 $\frac{d^2}{d^2a}\log C(\sigma(a))$ 均可表示为出人意料地简洁的标准双曲函数表达式（另见图 14，其中绘制了这些函数的图像）。

命题 4.1。 设 $C(\sigma(a))$ 按(20) 定义，且 $\lambda = \sigma(a)$，则有

$$C(\sigma(a)) = \begin{cases} 2 & \text{if } a = 0 \\ a\coth\left(\frac{a}{2}\right) & \text{otherwise,} \end{cases}$$

$$\frac{d}{da}\log(C(\sigma(a))) = \begin{cases} 0 & \text{if } a = 0 \\ \frac{1}{a} - \frac{1}{\sinh(a)} & \text{otherwise,} \end{cases}$$

$$\frac{d^2}{d^2a}\log(C(\sigma(a))) = \begin{cases} \frac{1}{6} & \text{if } a = 0 \\ -\frac{1}{a^2} + \frac{\coth(a)}{\sinh(a)} & \text{otherwise,} \end{cases}$$

其中 双曲余切函数和双曲正弦函数分别为双曲余切函数和双曲正弦函数。（证明见附录。）

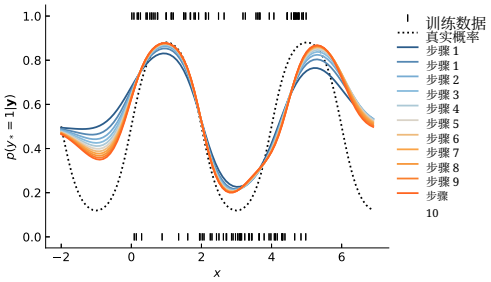
因此，如果我们记作 $\mathbf{C}_t \overset{\text{def}}{=} \sum_{n=1}^{N}\log(C(\sigma(f_{t,n})))$, 我们得到对数后验的梯度和海森矩阵变为

$$\nabla\tilde{\psi}(\mathbf{f}_t) = \mathbf{y}_{t-1} - \boldsymbol{\sigma}_{t-1} - \mathbf{K}^{-1}\mathbf{f}_t + \nabla\mathbf{C}_t,$$
$$-\nabla\nabla\tilde{\psi}(\mathbf{f}_t) = \mathbf{W}_t + \mathbf{K}^{-1} - \nabla\nabla\mathbf{C}_t,$$

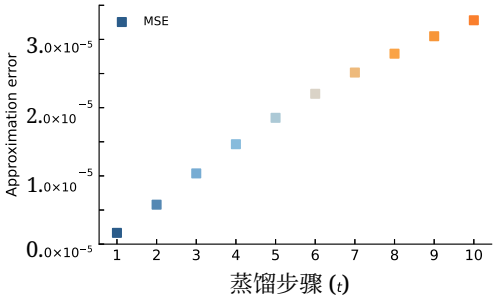
其中 $\mathbf{W}_t = \text{diag}_n\{\sigma(f_{t,n})(1-\sigma(f_{t,n}))\}$ 和 $\nabla\nabla\mathbf{C}_t = \text{diag}_n\{\frac{d^2}{d^2 f_{t,n}}\log C(\sigma(f_{t,n}))\}$，分别。这些直接影响后验分布的拉普拉斯近似，例如从图 4.

不幸的是，与高斯过程回归的以数据为中心的自蒸馏不同，由于获得后验分布（预测）所需的数值近似，我们无法获得蒸馏模型的闭式解。然而，我们确实通过经验研究了使用连续伯努利分布而非离散伯努利分布的效果，如图4。特别是，我们绘制了使用 C-GPC 进行单步蒸馏的解，以及使用离散伯努利分布的解，其中尽管伯努利分布的支撑集是离散的，我们仍使用连续观测值。我们在第4.3 节下文中对此示例进行了扩展。

4.2 以分布为中心的自蒸馏



(a) 蒸馏 10 步后的解。



(b) 迭代解与缩放解之间的近似误差。

图 6：用于高斯过程分类的以分布为中心的自蒸馏步骤，共十步。在 (a) 中，我们绘制了使用缩放协方差近似得到的解；在 (b) 中，我们绘制了迭代蒸馏过程与缩放近似在区间 $[-2, 7]$ 上 90 个等距点构成的测试集上的近似误差。

在分布中心方法中，我们按照第3.2节中的类似方式进行。首先，我们从（2）和（3）中观察到，我们的近似后验分布是一个高斯过程 $\mathcal{GP}(m_*,\ k_*)$

with

$$m_*(x) = m(\mathbf{x}) + k(\mathbf{x}, \mathbf{x}^\intercal)\mathbf{K}^{-1}(\hat{\mathbf{f}} - m(\mathbf{x}))$$
$$k_*(x, y) = k(x, y) - k(x, \mathbf{x}^\intercal)(\mathbf{K} + \mathbf{W}^{-1})^{-1}k(\mathbf{x}, y)\,.$$

这随后可用作先验分布，这再次意味着一系列高斯过程$\mathcal{GP}(m_t, k_t)$, $t = 0, 1, 2, \ldots$ 其中类似于方程 (6) 和 (7) 在这种情况下变为

$$m_{t+1}(x) = m_t(x) + k_t(x, \mathbf{x}^\intercal)\mathbf{K}_t^{-1}(\hat{\mathbf{f}}_t - m_t(\mathbf{x})) \tag{14}$$
$$k_{t+1}(x, y) = k_t(x, y) - k_t(x, \mathbf{x}^\intercal)(\mathbf{K}_t + \mathbf{W}_t^{-1})^{-1}k_t(\mathbf{x}, y). \tag{15}$$

ˆ此处，$\mathbf{f}_t$ 是 ψ_t(**f**)的众数，即第 t 次迭代的对数后验；类似地，$\mathbf{W}_t$ =ˆ ˆ diag$_n\{\sigma(f_{t,}$ $_n)(1 - \sigma(f_{t,n}))\}$。

ˆ由于$\mathbf{f}_t$ 没有解析形式，因此我们无法像在回归情形中那样求解分类情形下的上述递归关系，必须借助数值结果。不过，我们给出了一个与推论3.4 中的数据复制结果类似的结果，由此可知，在分布中心设置中，迭代蒸馏可以通过缩放第一次蒸馏步骤中的协方差来近似。这些观测值总结在以下命题中，其中 $\mathcal{D}_t$ 与推论3.4一样，由数据集 $\mathcal{D}$ 复制 t 次构成；图6a 和6b展示了一个示例。

命题*4.2*。（*A*）使用 $\mathcal{D}_t$ 和高斯先验$\mathcal{GP}$*(0,* k*)*进行单步分布中心蒸馏，与使用高斯先验 $\mathcal{GP}$ *(0,* tk*)*进行单步分布中心蒸馏得到相同的后验分布。（*B*）使用 $\mathcal{D}$ 进行 t 次分布中心蒸馏，并从初始高斯先验 $\mathcal{GP}$*(0,* k*)*开始，其结果与使用高斯先验 $\mathcal{GP}$*(0,* tk*)*进行单步分布中心蒸馏得到的后验近似相同。（证明见附录。）

4.3 说明性示例

现在我们来看一个示例，其中真实的底层函数为 $g(x)$ =2sin $\left(\frac{x\pi}{2}\right)$。我们从 $\mathcal{U}(0, 5)$ 分布中抽取训练观测 $\mathbf{x}$，并从伯努利分布中抽取二元目标，其概率由 $g(x)$ 参数化。

以数据为中心的 GPC 蒸馏 对于以数据为中心的蒸馏，我们首先使用伯努利分布，将普通高斯过程分类模型拟合到二元观测值上，再将第二个模型拟合到第一个模型的连续预测值上。在图4 中，我们绘制了初始普通 GPC 的预测结果，以及第二个模型的预测结果；对于第二个模型，我们分别使用连续伯努利分布（如第 4 节所述）和离散伯努利分布，后者有意违反了该分布的离散支撑集。我们选择模型的超参数，使其在输入数据上的负对数似然（NLL）最小。与初始拟合相比，蒸馏后的伯努利模型略显平缓，但表现良好；而 C-GPC 模型则给出了更“极端”的预测。我们推测，蒸馏后的伯努利模型因设定错误而受到的隐式正则化，在 C-GPC 模型中并不存在。此外，根据第 C 节，我们预计，与普通伯努利分布相比，连续伯努利分布会倾向于给出更极端的值，这一点在图 4 中也很明显。为了减轻 C-GPC 模型的过拟合程度，我们在训练数据的核矩阵对角线上重新引入噪声参数 $\gamma_t > 0$。我们在图5中绘制了正则化和非正则化的解，并观察到，引入噪声参数后，我们不仅得到了设定正确的模型，也能更好地控制正则化程度。更多实验请参见附录B 。

以分布为中心的 GPC 蒸馏 与以数据为中心的情形类似，我们最初拟合一个普通 GPC，但与以数据为中心的情形不同，在以分布为中心的蒸馏中，我们不会在蒸馏步骤中改变模型假设。在图 6a 中，我们绘制了蒸馏 10 个步骤后的解，并观察到其行为与以数据为中心的蒸馏相比有显著不同；蒸馏的影响要微妙得多，也更接近回归情形中的蒸馏，解逐步与原始数据拟合得越来越好。此外，在图 6b 中，我们绘制了由（6）-（7）得到的解与按照命题 4.2中的论述对核矩阵进行缩放所得到的解之间的均方近似误差。尽管误差几乎呈线性增长，但每一步的增幅不到 0.5×10^{-5}，且经过 10 个步骤后，近似误差约为 10^{-5}。因此，由于缩放近似快得多（尤其是在蒸馏步骤很多时），它是朴素计算的一种绝佳替代方法。

5 相关工作

据我们所知，目前尚无关于高斯过程模型知识蒸馏的文献。尽管如此，为求周全，下文仍将我们提出的方法与现有的知识蒸馏及高斯过程回归/分类相关文献联系起来。

知识蒸馏 知识蒸馏的思想最早可追溯至 Ba & Caruana (2014);Bucil & Caruana (2015年)，但通常最广为人知的是 Hinton 等人 (2015年) 的工作。最初，其目标是实现模型压缩：通过使用较大模型的输出训练一个较小的（按适当的模型规模指标衡量）学生模型。教师模型。

近年来，蒸馏的思想已朝着多个方向扩展，并应用于各种类型的任务。其中尤为相关的是自蒸馏，即认为学生模型与教师模型属于相同的模型类别／架构（Furlanello 等人，2018年；Zhang 等人，2019年；Allen-Zhu & Li，2020年）。蒸馏研究的其他有趣方向包括教师与学生之间规模差距的影响（Mirzadeh 等人，2019年）、匹配输出以外的其他信息（Park 等人，2019年；Romero 等人，2015年；Srinivas & Fleuret，2018年），以及多教师和多源蒸馏（Li 等人，2021年；Liu & Zenke，2020年；You 等人，2017年；Anonymous Authors，2023年）。最后，由于大多数蒸馏方法具有领域无关的特性，因此被应用于广泛的领域（Krishnan Parthasarathi & Strom，2019年；Borup 等人，2023年）。
尽管各种蒸馏技术在实证上取得了巨大成功，但我们仍缺乏对蒸馏的严谨理解。不过，近期研究已在一些简化场景中提供了见解，例如线性模型、核岭回归和决策树（Phuong & Lampert，2019年；Borup& Andersen，2021年；Frosst &Hinton，2018年）。

高斯过程 关于高斯过程模型的许多结果早已为人熟知，并汇集在一些奠基性著作中，例如 Rasmussen& Williams （2006）；Bishop （2007年）；Murphy（2013）。然而，近年来，关于拓展高斯过程模型能力，并使其适应现代计算资源和数据集规模的研究日益增多。这包括神经网络与高斯过程之间的理论联系（Lee et al.，2018年；Yang，2019年；Garriga-Alonso et al.，2019年）、随机高斯过程和稀疏高斯过程（Titsias；Hensmanet al.，2013年，2015年），以及将神经网络的优势与高斯过程相结合的可能性，例如深度核学习（Wilson et al.，2016a，b）。

6 结论

总之，我们提出了两种将知识蒸馏扩展到高斯过程回归和分类的方法：数据中心方法和分布中心方法。我们展示了其中一些方法与已知结果或超参数的某些非常特殊的选择密切相关，但也表明，有些方法需要谨慎地重新定义模型假设，才能使其适用。据我们所知，本文首次尝试专门针对高斯过程模型提出知识蒸馏，为进一步研究开辟了新的方向，并可能对机器学习产生影响。未来有趣的研究方向包括但不限于：在蒸馏步骤中同时利用原始目标和预测目标的凸组合，以及结合数据中心和分布中心的蒸馏方法。

References

Allen-Zhu, Z. 和 Li, Y. 深入理解深度学习中的集成、知识蒸馏与自蒸馏。2020年。网址 http://arxiv.org/abs/2012.09816。

匿名作者。利用多任务蒸馏从少量标签中学习高效模型。 未发表/审稿中，2023年。

Ba, L. J. 和 Caruana, R. 深度网络真的需要很深吗？ 神经信息处理系统进展, 3(一月):2654–2662, 2014. ISSN 10495258.

Bishop, C. M. 模式识别与机器学习（信息科学与统计）. Springer, 2007年. ISBN 0387310738.

Borup, K. 和 Andersen, L. N. 即使你的教师也需要指导：真实标签会减弱自蒸馏施加的正则化。载于 Ranzato, M.、Beygelzimer, A.、Dauphin, Y.、Liang, P. S. 和 Vaughan, J. W.（编），神经信息处理系统进展，第 34 卷，第 5316–5327 页。Curran Associates, Inc.，2021年。网址 http://arxiv.org/abs/2102.13088https://proceedings.neurips.cc/paper/2021/file/2adcefe38fbcd3dcd45908fbab1bf628-Paper.pdf。

Borup, K., Kidmose, P., Phan, H. 和 Mikkelsen, K. 使用患者特异性集成模型和知识蒸馏对耳部脑电数据进行自动睡眠评分。 生物医学信号处理与控制，81（2022年5月）：104496，2023年。ISSN 17468108。doi：10.1016/j.bspc.2022.104496。URLhttps://doi.org/10.1016/j.bspc.2022.104496。

Bucil, C. 和 Caruana, R. 模型压缩 Cristian。IEEE Communications Maga-zine,54(1):1–9, 2015年。ISSN 0218-0014。doi: 10.1111/j.1467-985x.2009.00624{_}10.x。URL http://users.iems.northwestern.edu/~nocedal/PDFfiles/dss.pdf%0Ahttp: //citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.67.7952&rep=rep1& amp;type=pdf% 255Cnhttp://books.nips.cc/papers/files/nips15/AA03.pdf% 250Ahttps://ieeexplore.ieee.org/document/85783。Frosst,N. 和Hinton, G. r. 将神经网络蒸馏为软决策树，2018年。ISSN 16130073。Furlanello,T.、Lipton, Z. C.、Tschannen,M.、Itti, L. 和 Anandkumar,A. 重生神经网络。载于 Dy, J.和 Krause,A.（编），第35届国际机器学习会议论文集，机器学习研究论文集第80卷，第1607–1616页，斯德哥尔摩国际博览中心，瑞典斯德哥尔摩，2018年。PMLR。Garriga-Alonso,A.、Aitchison,L.和 Rasmussen, C. E. 作为浅层高斯过程的深度卷积网络。第七届国际学习表征会议，ICLR 2019，第1–16页，2019年。Hensman,J.、Fusi,N. 和 Lawrence,N.D. 面向大数据的高斯过程。载于人工智能中的不确定性——第29届会议论文集，UAI2013，第282–290页，2013年。Hensman,J.、Matthews, A. G. 和 Ghahramani,Z. 可扩展的变分高斯过程分类。机器学习研究杂志，38:351–360，2015年。ISSN 15337928。Hinton, G.、Vinyals,O. 和Dean, J. 蒸馏神经网络中的知识。arXiv预印本 arXiv:1503.02531，第1–9页，2015年。URLhttp://arxiv.org/abs/1503.02531。KrishnanParthasarathi,S.H.和Strom, N. 从构建包含百万小时语音的声学模型中获得的经验。ICASSP，IEEE国际声学、语音与信号处理会议——论文集，2019年5月:6670–6674，2019年。ISSN 15206149。doi:10.1109/ICASSP。2019.8683690。Lee, J.、Bahri, Y.、Novak,R.、Schoenholz, S. S.、Pennington,J.和 Sohl-Dickstein, J. 作为高斯过程的深度神经网络。第六届国际学习表征会议，ICLR 2018——会议专题论文集，2018年。Li,Z.、Ravichandran, A.、Fowlkes, C.、Polito, M.、Bhotika, R.和 Soatto, S. 用于训练专家学生的表征整合，2021年。URLhttp://arxiv.org/abs/2107.08039。Liu, T. 和 Zenke, F. 通过神经切线迁移寻找可训练的稀疏网络，2020年。Loaiza-Ganem,G.和 Cunningham,J. P. 连续伯努利分布：修正变分自编码器中的普遍性错误。神经信息处理系统进展，32(NeurIPS):1–11，2019年。ISSN 10495258。Mirzadeh, S.-I.、Farajtabar, M.、Li, A.、Ghasemzadeh, H.、Levine, N.、Matsukawa,A.和 Ghasemzadeh,H.通过教师助理改进知识蒸馏：弥合学生与教师之间的差距。arXiv预印本arXiv:1902.03393，2019年。URL http: //arxiv.org/abs/1902.03393。

Mobahi, H.、Farajtabar, M. 和 Bartlett, P. L. 自蒸馏增强希尔伯特空间中的正则化。神经信息处理系统进展，2020年。Murphy, K. P. 机器学习：概率论视角。MIT出版社，2013年。ISBN 9780262018029 0262018020。Park, W.、Kim, D.、Lu, Y. 和 Cho, M. 关系知识蒸馏。发表于 *IEEE*计算机学会计算机视觉与模式识别会议论文集，2019年。ISBN 9781728132938。doi：10.1109/CVPR.2019.00409。Phuong, M. 和 Lampert, C. H. 朝着理解知识蒸馏迈进。第*36*届国际机器学习会议（*ICML 2019*），2019年6月(2014):8993–9007，2019年。Rasmussen, C. E. 和 Williams, K. 机器学习中的高斯过程。2006年。Romero, A.、Ballas, N.、Kahou, S. E.、Chassang, A.、Gatta, C. 和 Bengio, Y. FitNets：为轻量深度网络提供提示。第*3*届国际学习表征会议（*ICLR 2015*）会议论文集，第1–13页，2015年。Srinivas, S. 和 Fleuret, F. F. 通过雅可比匹配进行知识迁移。第*35*届国际机器学习会议（*ICML 2018*），11:7515–7523，2018年。Titsias, M. K. 稀疏高斯过程中诱导变量的变分学习。Michalis。Wilson, A. G.、Hu, Z.、Salakhutdinov, R. 和 Xing, E. P. 深度核学习。第*19*届人工智能与统计国际会议论文集（*AISTATS 2016*），(1998): 370–378，2016a。Wilson, A. G.、Hu, Z.、Salakhutdinov, R. 和 Xing, E. P. 随机变分深度核学习。神经信息处理系统进展，(Nips):2594–2602，2016b。ISSN 10495258。Yang, G. 权重共享的宽神经网络的缩放极限：高斯过程行为、梯度独立性与神经切线核推导，2019年。网址 http://arxiv.org/abs/1902.04760。You, S.、Xu, C.、Xu, C. 和 Tao, D. 从多个教师网络中学习。发表于第*23*届*ACM SIGKDD*知识发现与数据挖掘国际会议论文集，KDD '17，第1285–1294页，美国纽约州纽约市，2017年。计算机协会。ISBN 9781450348874。doi：10.1145/3097983.3098135。网址https://doi.org/10.1145/3097983.3098135。Zhang, L.、Song, J.、Gao, A.、Chen, J.、Bao, C. 和 Ma, K. 做自己的教师：通过自蒸馏提升卷积神经网络的性能。*IEEE*国际计算机视觉会议论文集，2019年10月：3712–3721，2019年。ISSN 15505499。doi：10.1109/ICCV.2019.00381。

A 关于高斯过程分类自蒸馏的更多细节

下面我们将更详细地重述主论文中关于高斯过程分类的一些已知结果。我们还将补充一些关于高斯过程分类自蒸馏结果的细节。

A.1 高斯过程分类

Foll按照使用高斯过程进行分类的通常设置，我们假设 $f \sim \mathcal{GP}(m,\ k)$，其中 d

$$\mathbf{y} \mid \mathbf{f} \sim \prod_{n=1}^{N} \sigma(f_n)^{t_n}(1-\sigma(f_n))^{1-t_n},$$

并将 $\mathbf{y} = (y_1, \ldots\ldots, y_n)$ 记为 $\forall y_i \in \{0,1\}$，类似地，$\mathbf{f}$表示 GP 先验在相应 x_i 值处的实现向量。于是，对数密度为

$$\psi(\mathbf{f}) \stackrel{\text{def}}{=} \log p(\mathbf{f} \mid \mathbf{y}) \propto \log p(\mathbf{f}) + \log p(\mathbf{y} \mid \mathbf{f}),$$

其中 log $p(\mathbf{f})$ 是一个多维高斯分布，并且可以证明

$$\log p(\mathbf{y} \mid \mathbf{f}) = \mathbf{y}^\intercal \mathbf{f} - \sum_{n=1}^{N} \log(1+\exp(f_n))\,,$$

因此（忽略**f**中的比例系数）：

$$\psi(\mathbf{f}) = -\frac{1}{2}(\mathbf{f}_N - \mathbf{m})^\intercal \mathbf{K}^{-1}(\mathbf{f}-\mathbf{m}) + \mathbf{y}^\intercal \mathbf{f} - \sum_{n=1}^{N} \log(1+\exp(f_n))\,. \tag{16}$$

由于后验分布是非高斯的，我们应用拉普拉斯近似以获得高斯近似。

ˆ因此，我们需要梯度和海森矩阵来获得 $\mathbf{f}$（MAP估计）和协方差矩阵。由此从（16）可得

$$\nabla\psi(\mathbf{f}) = \mathbf{y} - \boldsymbol{\sigma} - \mathbf{K}^{-1}\mathbf{f} + \mathbf{K}^{-1}\mathbf{m}$$
$$-\nabla\nabla\psi(\mathbf{f}) = \mathbf{W} + \mathbf{K}^{-1}$$

其中 ($\boldsymbol{\sigma}_n = \sigma\ f_n$ $\sum_{n=1}^{N}\ 1 +\ ^{f_n}$ $=$ $_n\{\sigma\ f_n\ \ 1-)(\ )$（该项来自 log(e)）以及 $\mathbf{W}$ diag ( )((f_n))}。

ˆ 利用这些可以使用牛顿-拉夫森法求出 $\mathbf{f}$，并且根据 ˆ 梯度的表达式可知，$\mathbf{f}$ 将是以下内容的解

$$\hat{\mathbf{f}} = \mathbf{m} + \mathbf{K}(\mathbf{y} - \boldsymbol{\sigma})\,. \tag{17}$$

对 $p(\mathbf{f} \mid \mathbf{y})$的高斯近似结果为

$$q(\mathbf{f}) = \mathcal{N}(\mathbf{f} \mid \hat{\mathbf{f}}, (\mathbf{W}+\mathbf{K}^{-1})^{-1})$$

后验预测分布计算如下

$$p(f_* \mid \mathbf{x}, \mathbf{y}, \mathbf{x}_*) = \int p(f_* \mid \mathbf{f}) p(\mathbf{f} \mid \mathbf{y}) \mathrm{d}\mathbf{f} \tag{18}$$

（参见 Bishop （2007年）），其中 $\mathcal{GP}$（m，k）先验给出了条件分布

$$p(f_* \mid \mathbf{f}) = \mathcal{N}(f_* \mid m(x) + \mathbf{k}^T\mathbf{K}^{-1}(\mathbf{f}-\mathbf{m}), k - \mathbf{k}^T\mathbf{K}^{-1}\mathbf{k})\,. \tag{19}$$

我们可以通过两种不同的方式得到预测：一种是用 q（$\mathbf{f}$）替换 p（$\mathbf{f} \mid \mathbf{y}$）见（19）（由此得到（2）和（3））， 另一种则直接将其视为均值的预测

$$\sigma(\mathbb{E}_q[\mathbf{f} \mid \mathbf{y}]) = \sigma(\hat{\mathbf{f}}).$$

我们注意到，虽然两者通常有所不同，但对于二元分类而言，决策边界是相同的(Bishop，2007年，第 10.3 节)。

A.2 以数据为中心的自蒸馏用于高斯过程分类

对于以数据为中心的蒸馏中的单步过程，我们可以选择上述任一种预测作为新的目标，并将其记为 $\mathbf{y}_1$，无论选择哪一种。不过，虽然 $\mathbf{y} \in \{0,1\}^N$，现在我们有 $\mathbf{y}_1 \in [0,1]^N$，因此不再能假设 $\mathbf{y}_1$服从条件伯努利分布。不过，通过扩展到连续伯努利分布（Loaiza-Ganem & Cunningham，2019年），我们可以引入适当的归一化常数来避免这一问题

$$C(\lambda) \overset{\text{def}}{=} \begin{cases} 2 & \text{if } \lambda = \frac{1}{2}, \\ \frac{2\tanh^{-1}(1-2\lambda)}{1-2\lambda} & \text{otherwise.} \end{cases} \tag{20}$$

See 图 14 关于归一化常数、其梯度和海森矩阵的图。现在我们假设 e
$f_1 \sim \mathcal{GP}(0, k)$ 和

$$\mathbf{y}_1 \mid \mathbf{f}_1 \sim \prod_{n=1}^{N} C(\sigma(f_{1,n}))\sigma(f_{1,n})^{y_{1,n}}(1-\sigma(f_{1,n}))^{1-y_{1,n}},$$

进而得到

$$\log p(\mathbf{y}_1 \mid \mathbf{f}_1) = \mathbf{y}_1^\intercal \mathbf{f}_1 - \sum_{n=1}^{N} \log(1+\exp(f_{1,n})) + \sum_{n=1}^{N} \log C(\sigma(f_{1,n})),$$

其中我们注意到 $\sum_{n=1}^{N} \log C(\sigma(f_{1,n})) > 0$。在命题 4.1 中，我们求得了 $\log C(\sigma(a))$ 的导数和二阶导数。因此，若我们记 $\mathbf{C}_1 \overset{\text{def}}{=} \sum_{n=1}^{N} \log(C(\sigma(f_{1,n})))$，则条件密度的梯度为

$$\nabla\psi_1(\mathbf{f}_1) = \mathbf{y}_1 - \boldsymbol{\sigma}_1 - \mathbf{K}^{-1}\mathbf{f}_1 + \nabla\mathbf{C}_1,$$

wher因此，我们可以利用上述导数计算最后一项。同样地，可以得到 t
海森
矩阵可以计算为

$$-\nabla\nabla\psi_1(\mathbf{f}_1) = \mathbf{W}_1 + \mathbf{K}^{-1} - \nabla\nabla\mathbf{C}_1,$$

其中我们注意到 $\nabla\nabla\mathbf{C}_1$ 是一个对角矩阵，并且 $\boldsymbol{\sigma}_1$ 和 $\mathbf{W}_1$ 的定义类似于经典情况。有了梯度和海森矩阵，我们可以遵循通常的设置并计算拉普拉斯近似均值的最大后验估计，并使用海森矩阵的逆作为协方差。也就是说，我们使用

$$\begin{aligned} \mathbf{f}_1^{\text{new}} &= \mathbf{f}_1 - \left(-\mathbf{W}_1 - \mathbf{K}^{-1} + \nabla\nabla\mathbf{C}_1\right)^{-1}\left(\mathbf{y}_1 - \boldsymbol{\sigma}_1 - \mathbf{K}^{-1}\mathbf{f}_1 + \nabla\mathbf{C}_1\right) \\ &= \left(\mathbf{W}_1 + \mathbf{K}^{-1} - \nabla\nabla\mathbf{C}_1\right)^{-1}\left(\left(\mathbf{W}_1 + \mathbf{K}^{-1} - \nabla\nabla\mathbf{C}_1\right)\mathbf{f}_1 + \mathbf{y}_1 - \boldsymbol{\sigma}_1 - \mathbf{K}^{-1}\mathbf{f}_1 + \nabla\mathbf{C}_1\right) \\ &= \mathbf{K}\mathbf{K}^{-1}\left(\mathbf{W}_1 + \mathbf{K}^{-1} - \nabla\nabla\mathbf{C}_1\right)^{-1}\left(\left(\mathbf{W}_1 - \nabla\nabla\mathbf{C}_1\right)\mathbf{f}_1 + \mathbf{y}_1 - \boldsymbol{\sigma}_1 + \nabla\mathbf{C}_1\right) \\ &= \mathbf{K}\left(\left(\mathbf{W}_1 - \nabla\nabla\mathbf{C}_1\right)\mathbf{K} + \mathbf{I}\right)^{-1}\left(\left(\mathbf{W}_1 - \nabla\nabla\mathbf{C}_1\right)\mathbf{f}_1 + \mathbf{y}_1 - \boldsymbol{\sigma}_1 + \nabla\mathbf{C}_1\right) \end{aligned}$$

也就是说，我们现在有了拉普拉斯近似

$$p(\mathbf{f}_1 \mid \mathbf{y}) \approx q(\mathbf{f}_1 \mid \mathbf{y}) = \mathcal{N}\left(\mathbf{f}_1 \mid \hat{\mathbf{f}}_1, \left(\mathbf{W}_1 + \mathbf{K}^{-1} - \nabla\nabla\mathbf{C}_1\right)^{-1}\right),$$

ˆ其中 $\mathbf{f}_1$ 是上述的最大后验估计，我们还可以使用拉普拉斯近似来获得ˆ边际对数似然。首先，通过在 $\mathbf{f}_1$, 处对 $\psi_1(\mathbf{f}_1)$进行泰勒近似，我们得到 $\psi_1(\mathbf{f}_1) \approx \psi_1(\hat{}\mathbf{f}_1) - \frac{1}{2}(\mathbf{f}_1 - \hat{}\mathbf{f}_1)\mathbf{H}^{-1}(\mathbf{f}_1 - \hat{}\mathbf{f}_1)$，其中$\mathbf{H}^{-1}$是 q的协方差矩阵，即 $\mathbf{H} \text{ def } = \mathbf{W}_1 + \mathbf{K}^{-1} -$

$\nabla\nabla\mathbf{C}_1$。因此，我们有

$$\begin{aligned}
\log p(\mathbf{y}_1) &= \int \psi(\mathbf{f}_1) d\mathbf{f}_1 \\
&\approx \psi_1(\hat{\mathbf{f}}_1) + \int -\frac{1}{2}(\mathbf{f}_1 - \hat{\mathbf{f}}_1)^\intercal \mathbf{H}^{-1}(\mathbf{f}_1 - \hat{\mathbf{f}}_1) d\mathbf{f}_1 \\
&= \psi_1(\hat{\mathbf{f}}_1) + \frac{N}{2}\log(2\pi) - \frac{1}{2}\log|\mathbf{H}| \\
&= \mathbf{y}_1^\intercal \hat{\mathbf{f}}_1 - \sum_{n=1}^{N} \log(1+\exp(\hat{f}_{1,n})) + \sum_{n=1}^{N} \log C(\sigma(\hat{f}_{1,n})) - \frac{1}{2}\hat{\mathbf{f}}_1^\intercal \mathbf{K}^{-1}\hat{\mathbf{f}}_1 \\
&\quad - \frac{1}{2}\log|\mathbf{K}| - \frac{N}{2}\log(2\pi) + \frac{N}{2}\log(2\pi) - \frac{1}{2}\log|\mathbf{H}| \\
&= \mathbf{y}_1^\intercal \hat{\mathbf{f}}_1 - \sum_{n=1}^{N} \log(1+\exp(\hat{f}_{1,n})) + \sum_{n=1}^{N} \log C(\sigma(\hat{f}_{1,n})) \\
&\quad - \frac{1}{2}\hat{\mathbf{f}}_1^\intercal \mathbf{K}^{-1}\hat{\mathbf{f}}_1 - \frac{1}{2}\log(|\mathbf{K}|) - \frac{1}{2}\log(|\mathbf{W}_1 + \mathbf{K}^{-1} - \nabla\nabla\mathbf{C}_1|)
\end{aligned}$$

边际对数似然可用于超参数优化。

然后，我们可以任意次数地迭代这一蒸馏过程。通常，我们认为求逆$\mathbf{K}$是拟合高斯过程计算成本最高的步骤；由于我们可以在所有步骤中重复使用$\mathbf{K}^{-1}$，因此，增加蒸馏步骤在计算上并不十分繁重。在上述 MAP 估计中，我们甚至改写了优化问题，以避免使用$\mathbf{K}$的逆矩阵。

B 玩具示例和其他结果的设置

在主论文中，我们加入了基于各种玩具示例的插图和图表。然而，所有这些说明性示例都是使用径向基核函数计算得到的

$$k(\mathbf{x}_1, \mathbf{x}_2) = \sigma_f^2 \exp\left(-\frac{\|\mathbf{x}_1 - \mathbf{x}_2\|^2}{2l}\right),$$

其中，$\sigma_f^2 > 0$ 和 $l > 0$ 都是可针对具体示例进行调节的超参数。我们还（主要针对回归设置）考虑一个噪声参数 $\gamma \geq 0$，它被添加到$\mathbf{K}$的对角线中，即假设训练观测中存在噪声。即使没有这一假设，我们通常也会使用一个较小的 $\gamma \approx 10^{-8}$，以避免回归和分类示例中的数值不稳定性。

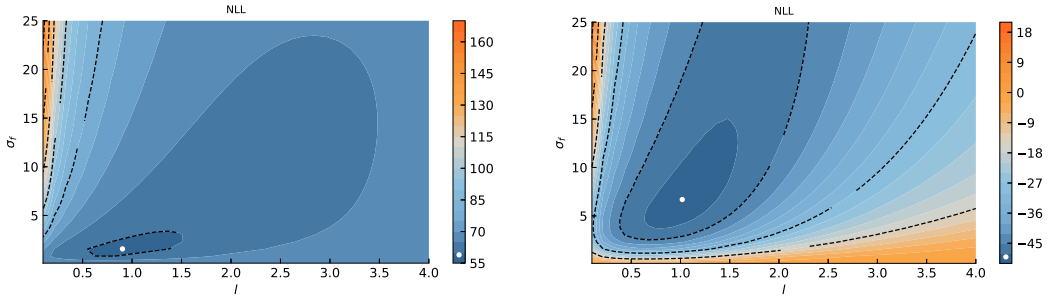
在图7中，我们绘制了σ_f 和 l 在取值网格上的负对数似然，分别针对使用普通高斯过程分类与伯努利似然的情况，以及使用连续伯努利似然的调整设置，当输入为真实连续底层函数 $g(x) = 2\sin(x\pi/2)$。

B.1 额外实验和结果

下文中，我们给出与主论文中相同示例实验的额外结果，但对例如模型参数或数据假设进行了更改。特别地，在图8 中，我们给出正文中数据中心高斯过程回归示例在改变（γ_1，…，γ_{10}）选择时的不同解。类似地，在图9 中，我们对分布中心高斯过程回归重复相同实验。最后，在图 10 中，我们绘制数据中心GPC的解，此时我们将蒸馏模型拟合到通过对初始模型在（0，1）编码目标（通过将初始模型在0.5处阈值化得到）而非初始模型的连续预测。在此设置下，使用普通伯努利假设不会导致设定错误的模型，但阈值化忽略了连续目标中包含的部分信息，并且从图10中可以看出，此情况下的蒸馏模型与针对连续目标的C-GPC大体相似。

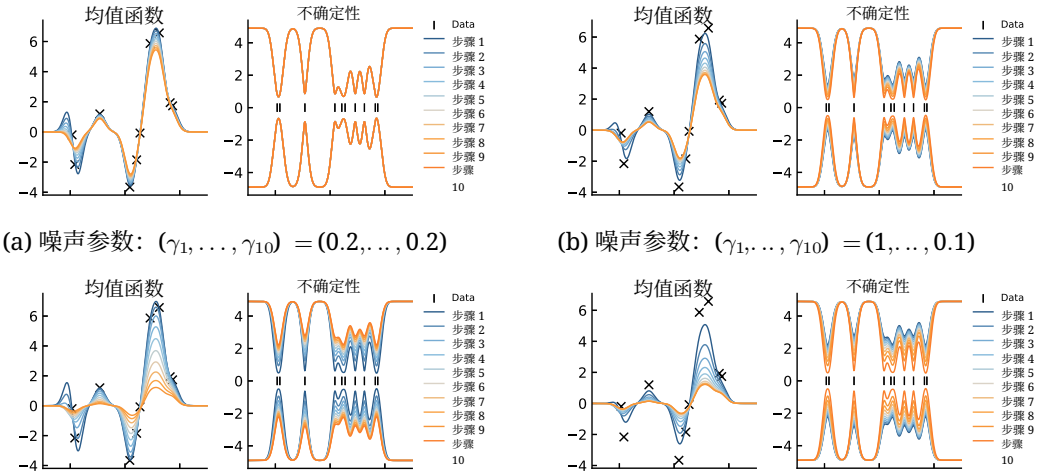
C 连续伯努利分布的一些细节

连续伯努利分布（记作 $\mathcal{CB}(\lambda)$）由 Loaiza-Ganem 和 Cunningham （2019年）提出，并处理了这样一种相对常见的情况：数据是连续值，位于



(a) 普通 GPC (b) C-GPC

图7：两种不同类型的高斯过程分类的超参数网格搜索。我们用白点标出每个网格的最小值。请注意，由于似然函数不同，两个模型之间的绝对值不可比较。



(a) 噪声参数：$(\gamma_1, \ldots, \gamma_{10}) = (0.2, \ldots, 0.2)$ (b) 噪声参数：$(\gamma_1, \ldots, \gamma_{10}) = (1, \ldots, 0.1)$

(c) 噪声参数：$(\gamma_1, \ldots, \gamma_{10}) = (0.1, \ldots, 3)$ (d) 噪声参数: $(\gamma_1, \ldots, \gamma_{10}) = (3, \ldots, 0.1)$

图 8：在固定超参数但采用不同噪声参数选择的情况下，高斯过程回归进行以数据为中心的自蒸馏的十个步骤。我们注意到，所有示例中都存在逐步增强的正则化。

$[0, 1]$ 而不是 $\{0, 1\}$中的离散值。在进行蒸馏时，确实是这种情况，而仅仅采用（隐式或显式的）伯努利分布假设并不合适。为稳妥起见，我们在下一节重复介绍 $\mathcal{CB}$ 来自 Loaiza-Ganem & Cunningham (2019年)的一些关键结果。在下文中请注意，我们将把伯努利分布变量视为可以在$[0, 1]$, 中取连续值，而这正是当前实践中经常采用的错误方法。
首先，我们回顾一下，参数为 $\lambda \in (0, 1)$ 的伯努利分布的概率密度为

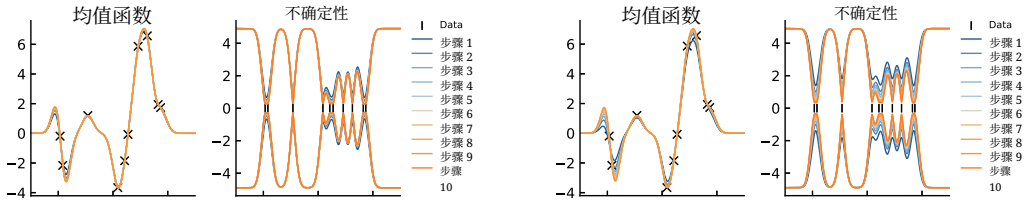
$$X \sim \mathcal{B}(\lambda) \iff \tilde{p}(x \mid \lambda) = \lambda^x (1 - \lambda)^{1-x}.$$

其次，根据 Loaiza-Ganem 与Cunningham (2019年)， 的密度与 $\mathcal{B}$ 的密度 $\mathcal{B}$仅相差一个乘法归一化常数（取决于参数 λ），记为 $C(\lambda)$。请参见图 11，其中绘制了这两个密度。特别地，对于 $\lambda \in (0,1)$，有

$$X \sim \mathcal{CB}(\lambda) \iff p(x \mid \lambda) = C(\lambda) \lambda^x (1 - \lambda)^{1-x}$$

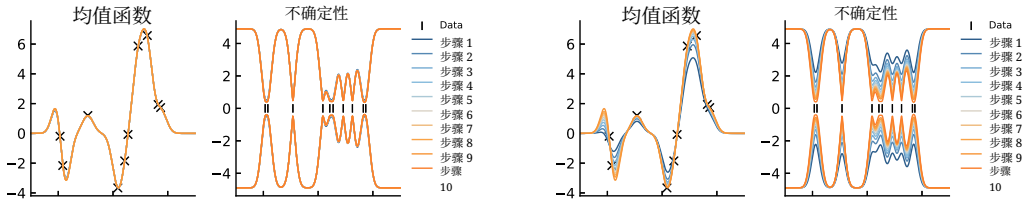
其中归一化常数可以证明为

$$C(\lambda) \stackrel{\text{def}}{=} \begin{cases} 2 & \text{if } \lambda = \frac{1}{2}, \\ \frac{2 \tanh^{-1}(1-2\lambda)}{1-2\lambda} & \text{otherwise}, \end{cases}$$



(a) 噪声参数: $(\gamma_1, \ldots, \gamma_{10}) = (0.2, \ldots, 0.2)$

(b) 噪声参数: $(\gamma_1, \ldots, \gamma_{10}) = (1, \ldots, 0.1)$



(c) 噪声参数: $(\gamma_1, \ldots, \gamma_{10}) = (0.1, \ldots, 3)$

(d) 噪声参数: $(\gamma_1, \ldots, \gamma_{10}) = (3, \ldots, 0.1)$

图 9: 在超参数固定、噪声参数选择不同的情况下，对 GPR 进行分布中心自蒸馏的十个步骤。值得注意的是，虽然 (c) 和 (d) 在第 10 步得到了相同的解，但达到该解的路径差异显著。

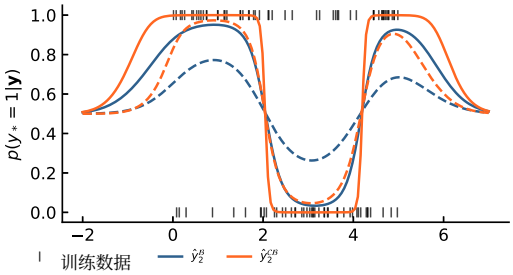


图 10: 设置与图 4相同，但我们没有将蒸馏模型拟合到第一个模型的软输出，而是还加入了（以虚线表示）普通 GPC 和 C-GPC，并将它们拟合到二元预测（以 0.5 为阈值）。我们看到，C-GPC 的过拟合现象加剧了，而普通 GPC 则与最初在软目标上训练的 C-GPC 十分相似。

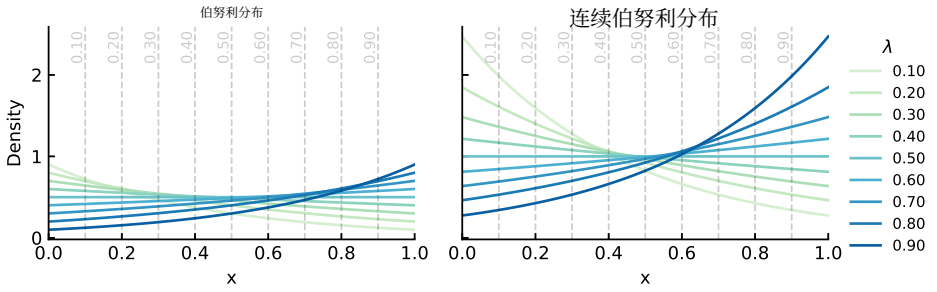


图 11: 参数 λ 取不同值时，伯努利分布和连续伯努利分布的概率密度。

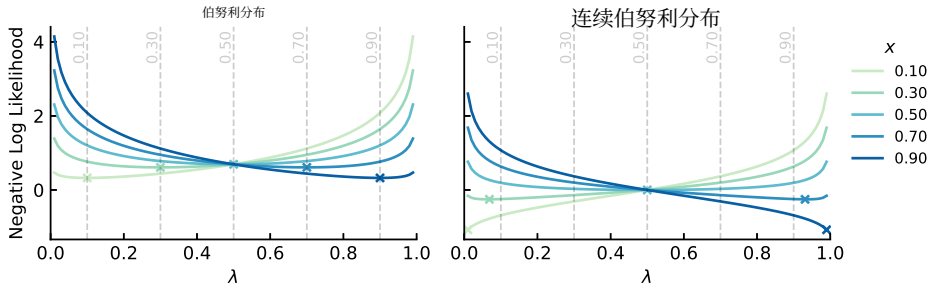


图 12：伯努利分布和连续伯努利分布在不同参数选择下的负对数似然 λ。我们用叉号标出每个 λ 的优化 x，并指出对于给定观测 x，伯努利分布的最优选择为 λ，而连续伯努利分布的最优选择是 x的非线性函数，通常偏向 x的更极端值 λ。另见图13。

$\tanh^{-1}$ 是反双曲正切函数。

在图13 中，我们绘制了普通伯努利分布（对所有 λ均等于1 ）和连续伯努利分布的归一化常数，同时观察到，由于 $\mathcal{CB}$（λ）分布变量的均值并非仅仅是 λ，因此单个观测 x 的最优 λ （即最大化密度）不同于 λ。

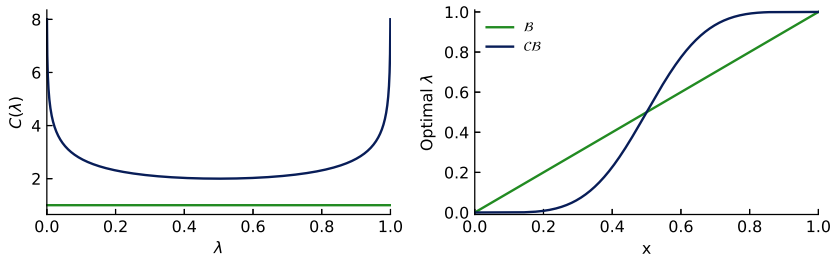


图 13：普通伯努利分布和连续伯努利分布的归一化常数及最优 λ 值（即对单个观测 x 最大化似然）。

最后，在图 14 中，我们同时绘制了对数归一化常数，以及关于 a相对于 $\lambda = \sigma(a)$ 或 a的一阶和二阶导数。推导过程见命题 4.1 。

D 时间与复杂度分析

尽管我们提出的自蒸馏方法主要旨在实现理论目标，下面我们将探讨这些方法的计算需求。

D.1（高效）实现

我们在 github.com/Kennethborup/ gaussian_process_self_distillation 上公开提供了用 Python 编写的方法实现。我们提供四个类，分别针对高斯过程回归和高斯过程分类的以数据为中心和以分布为中心的自蒸馏。这些类的名称直观明了，分别为 DataCentricGPR、DistributionCentricGPR、DataCentricGPC 和 DistributionCentricGPC。每种方法都包含 .fit 和 .predict 方法，与 scikit-learn API 类似。对于 DataCentricGPR，我们提供了朴素和高效两种实现，后者显著提升了多步蒸馏的训练速度。具体而言，受 Borup & Andersen（2021年）的启发，我们利用 K 的奇异值分解 VDV（其中 V $\in \mathbb{R}^{N\times N}$ 是一个正交矩阵，其各行为 K 的特征向量；D $\in \mathbb{R}^{N\times N}$ 是一个非负对角矩阵，其对角线上的元素为对应的特征值），将（4）重写为 $t = 1$

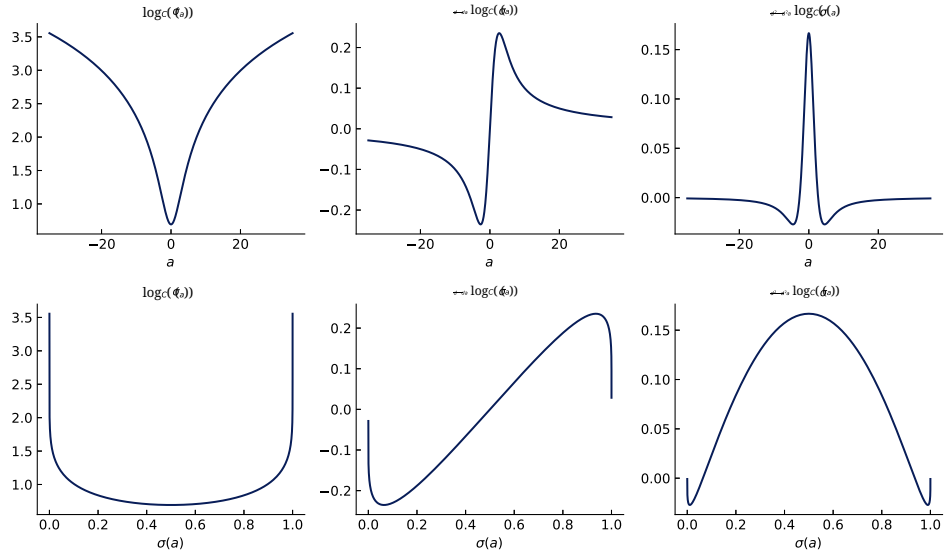


图 14：连续伯努利分布 $C(\lambda)$ 的对数归一化常数，以及关于 a 的一阶和二阶导数。

into

$$\mathbf{y}_1 = \mathbf{V}\mathbf{D}\left(\mathbf{D} + \gamma_1 \mathbf{I}_N\right)^{-1} \mathbf{V}^\intercal \mathbf{y}$$

通过迭代代入，可得到更一般的表达式

$$\begin{aligned}\mathbf{y}_t &= \left(\prod_{s=1}^{t} \mathbf{K}(\mathbf{K} + \gamma_s \mathbf{I}_N)^{-1}\right)\mathbf{y}, \\ &= \mathbf{V}\left(\prod_{s=1}^{t} \mathbf{A}_s\right)\mathbf{V}^\intercal \mathbf{y},\end{aligned}$$

其中 $\mathbf{A}_s = \mathbf{D}(\mathbf{D} + \gamma_s \mathbf{I}_N)^{-1} = \text{对角}_{n=1,\ldots,N}\left(\begin{smallmatrix} d_n \\ d_n+\gamma_s \end{smallmatrix}\right)$ 对于 $s = 1, \ldots t$ 均为对角矩阵。因此，计算出 $\mathbf{V}$ 和 $\mathbf{D}$ 后，由于 $\mathbf{A}_s$具有对角结构，后续的蒸馏步骤计算成本很低。预测公式（5）也可以类似地改写——更多实现细节请参见源代码。

D.2 时间要求

为评估我们提出的方法在实践中的影响，我们比较了拟合这些方法所需的时间（蒸馏步骤数量各不相同）与使用 `sklearn` 中高斯过程回归和高斯过程分类的标准实现进行常规拟合所需的时间；具体而言，使用 `GaussianProcessRegressor`和 `GaussianProcessClassifier`。具体来说，我们针对特定的步骤数，分别使用每种方法拟合一个模型，并将拟合时间除以对应 sklearn 常规方法的拟合时间。我们重复实验 30 次，并在图 15 中报告均值以及第 10 和第 90 经验分位数。所有实验均在 Mac M1 Pro CPU 上进行。

对于回归，我们观察到，以分布为中心的自蒸馏（正如定理3.3中所预期的）所需的计算量不超过使用 `sklearn` 实现的高斯过程回归。这是因为，对于回归而言，以分布为中心的自蒸馏始终处于高斯过程框架内。对于以数据为中心的自蒸馏，情况则有所不同；在一种朴素的迭代实现中，每个后续模型都拟合前一个模型的预测结果，其时间复杂度随蒸馏步骤数量线性增长（增长速率为 ≈ 0.11）。然而，通过对 $\mathbf{K}$ 在（4）中进行特征分解，可以重写该解，从而显著提高拟合速度。具体而言，每增加一个蒸馏步骤，仅需更新一个对角线矩阵和少量矩阵乘积（由重复使用的矩阵构成），因此计算成本低得多。因此，采用这种

通过高效实现，我们能够以恒定的时间复杂度将以数据为中心的自蒸馏拟合到任意数量的蒸馏步骤。因此，回归中的以数据为中心的自蒸馏和以分布为中心的自蒸馏，其拟合耗时都不会超过 sklearn 中普通高斯过程回归的拟合耗时。

对于分类，我们观察到，以数据为中心的自蒸馏会以 ≈ 0.26的速率随蒸馏步骤数量呈线性增长，并且在此设置下，多步自蒸馏的计算成本会迅速变得高昂。然而，使用以分布为中心的自蒸馏进行拟合所需的时间（正如预期）与蒸馏步骤数量无关。这一点显然可由命题4.2预期，并且拟合速度与`sklearn`实现一样快。

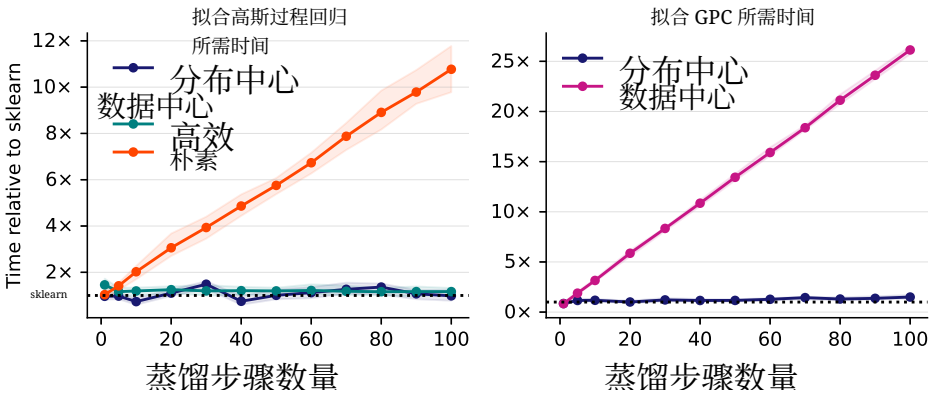


图15：与分别使用标准sklearn实现的回归和分类模型进行单次拟合相比，拟合各种自蒸馏高斯过程模型的相对时间。在（左）我们绘制了拟合回归模型的相对时间。对于以数据为中心的自蒸馏，我们使用了朴素和高效两种方法。虽然朴素方法随蒸馏步骤线性缩放，但高效方法在所有蒸馏步骤中的相对时间保持恒定。在（右）我们绘制了分类模型相对于sklearn实现的相对时间，并观察到以数据为中心的方法随蒸馏步骤线性缩放，而（如预期）分布中心方法随蒸馏步骤保持恒定。阴影区域表示每种方法和蒸馏步骤数量的30次重复的10和90经验分位数。

D.3 计算需求

接下来，我们考虑我们方法的计算需求。我们将矩阵求逆、矩阵乘积、Cholesky分解和奇异值分解都视为 $\mathcal{O}(N^3)$ 忽略常数，尽管存在一些算法可以将不同方法的计算阶数降低到$\mathcal{O}(N^\omega)$ 其中通常 $2 < \omega \leq 3$。我们的实现基于算法3.1、3.2和5.1，出自Rasmussen & Williams (2006), 这在很大程度上依赖于使用Cholesky分解进行回代。

需要指出的是，尽管我们的设置框架与普通高斯过程回归和高斯过程分类的设置有明显差异，但我们提出的方法仍然接近或处于普通高斯过程框架之内，从而简化了分析。具体而言，对于分布中心高斯过程回归，其解是一个噪声参数经过特殊缩放的普通高斯过程回归模型（见定理3.3），并且可以使用任何现有的高斯过程回归实现。因此，该方法的复杂度通常为 $\mathcal{O}(N^3)$。对于分布中心高斯过程分类，我们考虑命题 4.2 中的近似方法，并指出它同样等价于普通 GPC 模型，其复杂度为 $\mathcal{O}(N^3)$。对于数据中心高斯过程回归，朴素实现需要对普通高斯过程回归模型进行 t 次迭代拟合；但利用上述高效实现，我们以奇异值分解替代 Cholesky分解，因此可以在多个蒸馏步骤中重复使用 $\mathbf{V}$ 和 $\mathbf{D}$。因此，每增加一个蒸馏步骤，只需对对角矩阵以及 $N \times N$ 矩阵进行矩阵乘积运算，这也使该方法的复杂度为 $\mathcal{O}(N^3)$。最后，由于数据中心高斯过程分类具有迭代性质，无法得到简化的解，因此该方法的计算开销会随蒸馏步骤数量线性增长（尽管其复杂度仍为 $\mathcal{O}(N^3)$）。具体而言，对于 $t > 1$，我们将似然函数改为连续伯努利分布。鉴于命题 4.1 提供了简单的闭式解，这一改动影响甚微；但由于我们的模型没有闭式解，我们需要

以普通方式拟合每个模型，因此需要 $\mathcal{O}(tN^3)$，其中 t 表示蒸馏步骤数量。对于较大的 t，这在计算上可能开销很大。总体而言，我们发现，尽管数据中心高斯过程分类的计算开销随 t 线性增长，但我们提出的所有方法主要仍受 N 的计算开销限制。最后，由于我们的方法仅有细微改动，用于高效计算/近似$\mathbf{K}$ 的方法应该大多仍适用于我们的方法。

E 正文中的证明

下面我们重新陈述主论文中的所有结果，并给出相应的证明 s.

A 第 3 节的证明（高斯过程回归的自蒸馏）3（高斯过程回归的自蒸馏）

命题 3.1。 定义 $\mathbf{y}_t \overset{\text{定义}}{=} \boldsymbol{\mu}_t$ 和 $\mathbf{y}_0 \overset{\text{定义}}{=} \mathbf{y}$。则对于任意 $t \geq 1$ 使得 $\mathbf{f}_t \mid \boldsymbol{\mu}_{t-1} \sim \mathcal{N}(\boldsymbol{\mu}_t, \boldsymbol{\Sigma}_t)$，其中

$$\mathbf{y}_t = \left(\prod_{s=1}^{t} \mathbf{K}(\mathbf{K} + \gamma_s \mathbf{I}_N)^{-1} \right) \mathbf{y}, \tag{4}$$

$$\mathbb{E}[\mathbf{f}_{t,*} \mid \boldsymbol{\mu}_{t-1}] = k(\mathbf{x}_*, \mathbf{x})(\mathbf{K} + \gamma_t \mathbf{I}_N)^{-1} \mathbf{y}_{t-1}, \tag{5}$$

以及 $\boldsymbol{\Sigma}_t = \mathbf{K} - \mathbf{K}(\mathbf{K} + \gamma_t \mathbf{I}_N)^{-1}\mathbf{K}$ 以及 $\mathbf{y}_t$ 仅受 γ_t 的选择影响。

证明。我们假设模型 $y_{t-1} = f_t(\mathbf{x}) + \varepsilon_t$，其中 $f_t \sim \mathcal{GP}(0, k)$，且对所有$t \geq 1$，$\varepsilon_t \sim \mathcal{N}(0, \gamma_t)$。由此可得

$$\begin{aligned} p(\mathbf{f}_t) &= \mathcal{N}(\mathbf{f}_t \mid 0, \mathbf{K}) \\ p(\mathbf{y}_{t-1} \mid \mathbf{f}_t) &= \mathcal{N}(\mathbf{y}_{t-1} \mid \mathbf{f}_t, \gamma_t \mathbf{I}_N) \\ p(\mathbf{y}_{t-1}) &= \int p(\mathbf{y}_{t-1} \mid \mathbf{f}_t) p(\mathbf{f}_t) \, \mathrm{d}\mathbf{f}_t = \mathcal{N}(\mathbf{y}_{t-1} \mid 0, \mathbf{K} + \gamma_t \mathbf{I}_N), \end{aligned}$$

其中，我们对在$\mathbf{x}_1, \ldots, \mathbf{x}_N$处取值的函数使用常用粗体记号；即$\mathbf{y}_t = (y_t(\mathbf{x}_1) \ldots, y_t(\mathbf{x}_N)) \in \mathbb{R}^N$和$\mathbf{f}_t = (f_t(\mathbf{x}_1) \ldots, f_t(\mathbf{x}_N)) \in \mathbb{R}^N$。将它们合并后得到先验

$$\begin{bmatrix} \mathbf{y}_{t-1} \\ \mathbf{f}_t \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \mathbf{0} \\ \mathbf{0} \end{bmatrix}, \begin{bmatrix} \mathbf{K} + \gamma_t \mathbf{I} & \mathbf{K} \\ \mathbf{K} & \mathbf{K} \end{bmatrix} \right).$$

因此，根据条件多元高斯分布的标准结果，我们得到后验分布

$$\mathbf{f}_n \mid \mathbf{y}_{t-1} \sim \mathcal{N}\Big(\mathbf{K}(\mathbf{K} + \gamma_t \mathbf{I}_N)^{-1} \mathbf{y}_{t-1}, \; \mathbf{K} - \mathbf{K}(\mathbf{K} + \gamma_t \mathbf{I}_N)^{-1}\mathbf{K} \Big). \tag{21}$$

回忆一下，我们使用$\mathbf{y}_t = \boldsymbol{\mu}_t = \mathbb{E}[\mathbf{f}_n \mid \mathbf{y}_{t-1}] = \mathbf{K}(\mathbf{K} + \gamma_{t-1}\mathbf{I}_N)^{-1}\mathbf{y}_{t-2}$ 21 $\mathbf{y}_{t-1} = \mathbf{y}_{t-2} + \text{def}\mathbf{y}$，并通过在式 () 中反复代入$\mathbf{y}$，得到

$$\begin{aligned} \mathbf{y}_t = \mathbb{E}[\mathbf{f}_t \mid \mathbf{y}_{t-1}] &= \mathbf{K}(\mathbf{K} + \gamma_t \mathbf{I}_N)^{-1} \mathbf{y}_{t-1} \\ &= \mathbf{K}(\mathbf{K} + \gamma_t \mathbf{I}_N)^{-1} \mathbf{K}(\mathbf{K} + \gamma_{t-1} \mathbf{I}_N)^{-1} \mathbf{y}_{t-2} \\ &= \mathbf{A}_t \mathbf{A}_{t-1} \mathbf{y}_{t-2} \\ &= \left(\prod_{m=1}^{n} \mathbf{A}_m \right) \mathbf{y}_0, \end{aligned}$$

其中，我们使用记号$\mathbf{A}_i = \mathbf{K}(\mathbf{K} + \gamma_i \mathbf{I}_N)^{-1}$。注意，关于 $\mathbb{E}[f_t(x)]$ 的结论可直接由适用于任意$\mathbf{x}$的类似结果推出。 □

引理3.2。设 $\lambda_i^{(t)}$，对于 $i = 1, \ldots, N$，表示K_t的特征值。将$\mathbf{K}_0 = \mathbf{O}\,\mathrm{diag}_i(\lambda_i^{(0)})\mathbf{O}$记作 $\intercal$ K_0的谱分解。则对于 $t = 0, 1, \ldots$，有

$$\mathbf{K}_t = \mathbf{O}\underset{i}{\mathrm{diag}}\left(\lambda_i^{(0)} \zeta_i^{(t)} \right) \mathbf{O}^\intercal \tag{8}$$

其中 $\zeta_i^{(t)} \overset{\text{def}}{=} \frac{1}{\lambda_i^{(0)} \gamma_{t-1}^{-} + 1}$。对于 $k_t(x, \mathbf{x})$ 和 $m_t(\mathbf{x})$ 我们有

$$k_t(x, \mathbf{x}^\intercal) = k_0(x, \mathbf{x}^\intercal)\mathbf{O}\underset{i}{\mathrm{diag}}\left(\zeta_i^{(t)} \right) \mathbf{O}^\intercal \tag{9}$$

$$m_t(\mathbf{x}) = \mathbf{O}\underset{i}{\mathrm{diag}}\left(\lambda_i^{(0)} \gamma_{t-1}^{-} \zeta_i^{(t)} \right) \mathbf{O}^\intercal \mathbf{y}. \tag{10}$$

证明。首先我们观察到，式（7）给出了关于$\mathbf{K}_t$的以下递推关系：

$$\mathbf{K}_{t+1} = \mathbf{K}_t - \mathbf{K}_t(\mathbf{K}_t + \gamma_t \mathbf{I}_N)^{-1}\mathbf{K}_t \tag{22}$$

如果我们令 $(\lambda, \mathbf{o})$ 为 $\mathbf{K}_t$ 的一个特征值-特征向量对，由此从 (22), 得出 $(\lambda - \lambda^2/(\lambda+\gamma_t), \mathbf{o})$是 $\mathbf{K}_{t+1}$ 的一个特征值-特征向量对，因此我们有以下关于$\mathbf{K}_t$-矩阵特征值的递推关系：

$$\lambda_i^{(t+1)} = \lambda_i^{(t)} - \frac{\left(\lambda_i^{(t)}\right)^2}{\lambda_i^{(t)} + \gamma_t} = \frac{\lambda_i^{(t)}\gamma_t}{\lambda_i^{(t)} + \gamma_t} \qquad i = 1, \ldots, N\,.$$

由此可以很容易通过归纳法验证，

$$\lambda_i^{(t)} = \frac{\lambda_i^{(0)}}{\lambda_i^{(0)}\gamma_{t-1}^- + 1} \qquad i = 1, \ldots, N\,,$$

并且我们看到递推关系的解 (8) 是

$$\mathbf{K}_t = \mathbf{O}\underset{i}{\mathrm{diag}}\left(\frac{\lambda_i^{(0)}}{\lambda_i^{(0)}\gamma_{t-1}^- + 1}\right)\mathbf{O}^\intercal\,. \tag{23}$$

注意到（23）对于 $t = 0$显然成立。接下来，我们注意到（23）有以下推论：首先，由（22）可得

$$(\mathbf{K}_t + \gamma_t\mathbf{I})^{-1} = \mathbf{O}\underset{i}{\mathrm{diag}}\left(\frac{1}{\frac{\lambda_i^{(0)}}{\lambda_i^{(0)}\gamma_{t-1}^- + 1} + \gamma_t}\right)\mathbf{O}^\intercal = \mathbf{O}\underset{i}{\mathrm{diag}}\left(\frac{\lambda_i^{(0)}\gamma_{t-1}^- + 1}{\gamma_t(\lambda_i^{(0)}\gamma_t^- + 1)}\right)\mathbf{O}^\intercal\,. \tag{24}$$

其次，结合（23）和（24）可得

$$\mathbf{K}_t(\mathbf{K}_t + \gamma_t\mathbf{I})^{-1} = (\mathbf{K}_t + \gamma_t\mathbf{I})^{-1}\mathbf{K}_t = \mathbf{O}\underset{i}{\mathrm{diag}}\left(\frac{\lambda_i^{(0)}}{\gamma_t(\lambda_i^{(0)}\gamma_t^- + 1)}\right)\mathbf{O}^\intercal\,. \tag{25}$$

最后，我们看到的是

$$\begin{aligned}\mathbf{I}_N - [\mathbf{K}_t + \gamma_t\mathbf{I}_N]^{-1}\mathbf{K}_t &= \mathbf{O}\underset{i}{\mathrm{diag}}\left(1 - \frac{\lambda_i^{(0)}/\gamma_t}{\lambda_i^{(0)}\gamma_t^- + 1}\right)\mathbf{O}^\intercal \\ &= \mathbf{O}\underset{i}{\mathrm{diag}}\left(\frac{\lambda_i^{(0)}\gamma_{t-1}^- + 1}{\lambda_i^{(0)}\gamma_t^- + 1}\right)\mathbf{O}^\intercal\,.\end{aligned} \tag{26}$$

现在，我们可以将注意力转向 $k_t(x,\ \mathbf{x}^\intercal)$。这里我们有

$$\begin{aligned}k_{t+1}(x, \mathbf{x}^\intercal) =& \big(k_{t+1}(x, x_1), k_{t+1}(x, x_2), \ldots, k_{t+1}(x, x_N)\big) \\ =& \big(k_t(x, x_1) - k_t(x, \mathbf{x}^\intercal)[\mathbf{K}_t + \gamma_t\mathbf{I}_N]^{-1}k_t(\mathbf{x}, x_1), \ldots, \\ & k_t(x, x_N) - k_t(x, \mathbf{x}^\intercal)[\mathbf{K}_t + \gamma_t\mathbf{I}_N]^{-1}k_t(\mathbf{x}, x_N)\big) \\ =& k_t(x, \mathbf{x}^\intercal) - \big(k_t(x, \mathbf{x}^\intercal)[\mathbf{K}_t + \gamma_t\mathbf{I}_N]^{-1}k_t(\mathbf{x}, x_1), \ldots, \\ & k_t(x, \mathbf{x}^\intercal)[\mathbf{K}_t + \gamma_t\mathbf{I}_N]^{-1}k_t(\mathbf{x}, x_N)\big) \\ =& k_t(x, \mathbf{x}^\intercal) - k_t(x, \mathbf{x}^\intercal)[\mathbf{K}_t + \gamma_t\mathbf{I}_N]^{-1}\mathbf{K}_t \\ =& k_t(x, \mathbf{x}^\intercal)\left(\mathbf{I}_N - [\mathbf{K}_t + \gamma_t\mathbf{I}_N]^{-1}\mathbf{K}_t\right)\end{aligned}$$

对递推关系进行迭代 $k_t(x,\ \mathbf{x})$ 并使用上面的（26）可得到 $t = 1,\ \ldots,$

$$\begin{aligned}k_t(x, \mathbf{x}^\intercal) &= k_0(x, \mathbf{x}^\intercal)\prod_{j=0}^{t-1}\mathbf{O}\underset{i}{\mathrm{diag}}\left(\frac{\lambda_i^{(0)}\gamma_{j-1}^- + 1}{\lambda_i^{(0)}\gamma_j^- + 1}\right)\mathbf{O}^\intercal \\ &= k_0(x, \mathbf{x}^\intercal)\mathbf{O}\underset{i}{\mathrm{diag}}\left(\prod_{j=0}^{t-1}\frac{\lambda_i^{(0)}\gamma_{j-1}^- + 1}{\lambda_i^{(0)}\gamma_j^- + 1}\right)\mathbf{O}^\intercal \\ &= k_0(x, \mathbf{x}^\intercal)\mathbf{O}\underset{i}{\mathrm{diag}}\left(\frac{1}{\lambda_i^{(0)}\gamma_{t-1}^- + 1}\right)\mathbf{O}^\intercal\,.\end{aligned} \tag{27}$$

接下来，我们证明（10）。首先，我们注意到（6）给出了关于 $m_{t+1}(\mathbf{x})$ 的递推关系：

$$\begin{aligned} m_{t+1}(\mathbf{x}) &= m_t(\mathbf{x}) + k_t(\mathbf{x},\mathbf{x}^\intercal)[k_t(\mathbf{x},\mathbf{x}^\intercal) + \gamma_t \mathbf{I}_N]^{-1}(\mathbf{y} - m_t(\mathbf{x})) \\ &= m_t(\mathbf{x}) + \mathbf{K}_t[\mathbf{K}_t + \gamma_t \mathbf{I}_N]^{-1}(\mathbf{y} - m_t(\mathbf{x})) \\ &= (\mathbf{I}_t - \mathbf{K}_t[\mathbf{K}_t + \gamma_t \mathbf{I}_N]^{-1}) m_t(\mathbf{x}) + \mathbf{K}_t[\mathbf{K}_t + \gamma_t \mathbf{I}_N]^{-1}\mathbf{y} \end{aligned} \tag{28}$$

由归纳法可得

$$m_t(\mathbf{x}) = \mathbf{O}\underset{i}{\mathrm{diag}}\left(\frac{\gamma_{t-1}^- \lambda_i^{(0)}}{\gamma_{t-1}^- \lambda_i^{(0)} + 1}\right)\mathbf{O}^\intercal \mathbf{y} \tag{29}$$

当 $t = 0$时，该结论显然成立。假设（29）成立，并结合（28）以及（24）和（26），可得

$$\begin{aligned} & m_{t+1}(\mathbf{x}) \\ &= \mathbf{O}\underset{i}{\mathrm{diag}}\left(\frac{\lambda_i^{(0)}\gamma_{t-1}^- + 1}{\lambda_i^{(0)}\gamma_t^- + 1}\right)\mathbf{O}^\intercal \mathbf{O}\underset{i}{\mathrm{diag}}\left(\frac{\gamma_{t-1}^- \lambda_i^{(0)}}{\gamma_{t-1}^- \lambda_i^{(0)} + 1}\right)\mathbf{O}^\intercal \mathbf{y} + \mathbf{O}\underset{i}{\mathrm{diag}}\left(\frac{\lambda_i^{(0)}}{\gamma_t(\lambda_i^{(0)}\gamma_t^- + 1)}\right)\mathbf{O}^\intercal \mathbf{y} \\ &= \mathbf{O}\underset{i}{\mathrm{diag}}\left(\frac{\gamma_{t-1}^- \lambda_i^{(0)}}{\lambda_i^{(0)}\gamma_t^- + 1}\right)\mathbf{O}^\intercal \mathbf{y} + \mathbf{O}\underset{i}{\mathrm{diag}}\left(\frac{\lambda_i^{(0)}}{\gamma_t(\lambda_i^{(0)}\gamma_t^- + 1)}\right)\mathbf{O}^\intercal \mathbf{y} \\ &= \mathbf{O}\underset{i}{\mathrm{diag}}\left(\frac{\gamma_{t-1}^- \lambda_i^{(0)}}{\lambda_i^{(0)}\gamma_t^- + 1} + \frac{\lambda_i^{(0)}}{\gamma_t(\lambda_i^{(0)}\gamma_t^- + 1)}\right)\mathbf{O}^\intercal \mathbf{y} \\ &= \mathbf{O}\underset{i}{\mathrm{diag}}\left(\frac{\gamma_t\gamma_{t-1}^- \lambda_i^{(0)} + \lambda_i^{(0)}}{\gamma_t(\lambda_i^{(0)}\gamma_t^- + 1)}\right)\mathbf{O}^\intercal \mathbf{y} \\ &= \mathbf{O}\underset{i}{\mathrm{diag}}\left(\frac{\lambda_i^{(0)}(\gamma_{t-1}^- + 1/\gamma_t)}{\lambda_i^{(0)}\gamma_t^- + 1}\right)\mathbf{O}^\intercal \mathbf{y} \\ &= \mathbf{O}\underset{i}{\mathrm{diag}}\left(\frac{\lambda_i^{(0)}\gamma_t^-}{\lambda_i^{(0)}\gamma_t^- + 1}\right)\mathbf{O}^\intercal \mathbf{y} \end{aligned}$$

按要求。d □

定理3.3。递推关系（6）和（7）的解为

$$\begin{aligned} m_t(x) &= k_0(x,\mathbf{x}^\intercal)(\mathbf{K} + \mathbf{I}_N/\gamma_{t-1}^-)^{-1}\mathbf{y} \\ k_t(x,y) &= k_0(x,y) - k_0(x,\mathbf{x}^\intercal)(\mathbf{K} + \mathbf{I}_N/\gamma_{t-1}^-)^{-1}k_0(\mathbf{x},y) \end{aligned} \tag{11}$$

对于 $t = 1,\ 2,\ \ldots\ldots$。

证明。我们首先考虑 k_t。结合（7）、（23）、（24）和（27），可得

$$\begin{aligned} k_{t+1}(x,y) =& k_t(x,y) - k_0(x,\mathbf{x}^\intercal)\mathbf{O}\underset{i}{\mathrm{diag}}\left(\frac{1}{\gamma_t(\lambda_i^{(0)}\gamma_t^- + 1)(\lambda_i^{(0)}\gamma_{t-1}^- + 1)}\right)\mathbf{O}^\intercal k_0(\mathbf{x},y) \\ =& k_t(x,y) - k_0(x,\mathbf{x}^\intercal)\mathbf{O}\underset{i}{\mathrm{diag}}\left(\frac{1}{\lambda_i^{(0)}}\left(\frac{1}{\lambda_i^{(0)}\gamma_{t-1}^- + 1} - \frac{1}{\lambda_i^{(0)}\gamma_t^- + 1}\right)\right)\mathbf{O}^\intercal k_0(\mathbf{x},y)\,. \end{aligned} \tag{30}$$

对上述方程进行迭代可得 i

$$\begin{aligned} k_t(x,y) &= k_0(x,y) - \sum_{j=0}^{n-1} k_0(x,\mathbf{x}^\intercal)\mathbf{O}\underset{i}{\mathrm{diag}}\left(\frac{1}{\lambda_i^{(0)}}\left(\frac{1}{\lambda_i^{(0)}\gamma_{j-1}^- + 1} - \frac{1}{\lambda_i^{(0)}\gamma_j^- + 1}\right)\right)\mathbf{O}^\intercal k_0(\mathbf{x},y) \\ &= k_0(x,y) - k_0(x,\mathbf{x}^\intercal)\mathbf{O}\underset{i}{\mathrm{diag}}\left(\frac{1}{\lambda_i^{(0)}}\sum_{j=0}^{n-1}\left(\frac{1}{\lambda_i^{(0)}\gamma_{j-1}^- + 1} - \frac{1}{\lambda_i^{(0)}\gamma_j^- + 1}\right)\right)\mathbf{O}^\intercal k_0(\mathbf{x},y) \\ &= k_0(x,y) - k_0(x,\mathbf{x}^\intercal)\mathbf{O}\underset{i}{\mathrm{diag}}\left(\frac{1}{\lambda_i^{(0)}}\left(\frac{1}{\lambda_i^{(0)}\gamma_{-1}^- + 1} - \frac{1}{\lambda_i^{(0)}\gamma_{t-1}^- + 1}\right)\right)\mathbf{O}^\intercal k_0(\mathbf{x},y) \end{aligned}$$

并且由于 $\gamma_{-1}^- = 0$，我们可知。

$$\begin{aligned} k_t(x,y) &= k_0(x,y) - k_0(x,\mathbf{x}^\intercal)\mathbf{O}\operatorname*{diag}_i\left(\frac{1}{\lambda_i^{(0)}}\left(1-\frac{1}{\lambda_i^{(0)}\gamma_{t-1}^- + 1}\right)\right)\mathbf{O}^\intercal k_0(\mathbf{x},y) \\ &= k_0(x,y) - k_0(x,\mathbf{x}^\intercal)\mathbf{O}\operatorname*{diag}_i\left(\frac{1}{\lambda_i^{(0)} + 1/\gamma_{t-1}^-}\right)\mathbf{O}^\intercal k_0(\mathbf{x},y) \\ &= k_0(x,y) - k_0(x,\mathbf{x}^\intercal)(\mathbf{K} + \mathbf{I}_N/\gamma_{t-1}^-)^{-1}k_0(\mathbf{x},y) \end{aligned}$$

如有需要。

接下来，我们考虑（11）。由（6）可得

$$m_{t+1}(x) = m_t(x) + k_t(x,\mathbf{x}^\intercal)[\mathbf{K}_t + \gamma_t\mathbf{I}_N]^{-1}(\mathbf{y} - m_t(\mathbf{x}))$$

利用（9）、（10）和（24），我们可以将其改写为

$$\begin{aligned} &m_{t+1}(x) = \\ &m_t(x) + k_0(x,\mathbf{x}^\intercal)\mathbf{O}\operatorname*{diag}_i\left(\frac{1}{\gamma_t(\lambda_i^{(0)}\gamma_t^- + 1)}\right)\mathbf{O}^\intercal(\mathbf{y} - \mathbf{O}\operatorname*{diag}_i\left(\frac{\lambda_i^{(0)}\gamma_{t-1}^-}{\lambda_i^{(0)}\gamma_{t-1}^- + 1}\right)\mathbf{O}^\intercal\mathbf{y}) \\ &m_t(x) + k_0(x,\mathbf{x}^\intercal)\mathbf{O}\operatorname*{diag}_i\left(\frac{1}{\gamma_t(\lambda_i^{(0)}\gamma_t^- + 1)} - \frac{1}{\gamma_t(\lambda_i^{(0)}\gamma_t^- + 1)}\frac{\lambda_i^{(0)}\gamma_{t-1}^-}{(\lambda_i^{(0)}\gamma_{t-1}^- + 1)}\right)\mathbf{O}^\intercal\mathbf{y} \\ &m_t(x) + k_0(x,\mathbf{x}^\intercal)\mathbf{O}\operatorname*{diag}_i\left(\frac{1}{\gamma_t(\lambda_i^{(0)}\gamma_t^- + 1)(\lambda_i^{(0)}\gamma_{t-1}^- + 1)}\right)\mathbf{O}^\intercal\mathbf{y} \end{aligned} \tag{31}$$

注意，出现在（31）中的对角矩阵的元素与出现在（30）中的元素相同。使用与上述相同的计算，我们得到

$$\begin{aligned} m_{t+1}(x) &= m_0(x) + k_0(x,\mathbf{x}^\intercal)\mathbf{O}\operatorname*{diag}_i\left(\frac{1}{\lambda_i^{(0)} + 1/\gamma_{t-1}^-}\right)\mathbf{O}^\intercal\mathbf{y} \\ &= k_0(x,\mathbf{x}^\intercal)\mathbf{O}\operatorname*{diag}_i\left(\frac{1}{\lambda_i^{(0)} + 1/\gamma_{t-1}^-}\right)\mathbf{O}^\intercal\mathbf{y}\,, \end{aligned}$$

由此（11）可得。 □

推论3.4。 如果我们令

$$\mathcal{D}_t = \{(x_{ij}, y_{ij}) \mid i = 1,\ldots,N\,, \quad j = 1,\ldots,t\}$$

其中 $x_{ij} = x_i$ 对所有 i 和 j，则 $\mathbf{f}$ 的后验分布为 $\mathcal{N}(\boldsymbol{\mu}_t, \boldsymbol{\Sigma}_t)$ 其中

$$\boldsymbol{\mu}_t = \begin{pmatrix} \mathbf{K}(\mathbf{K} + \gamma/t\mathbf{I}_N)^{-1}\mathbf{y} \\ \vdots \\ \mathbf{K}(\mathbf{K} + \gamma/t\mathbf{I}_N)^{-1}\mathbf{y} \end{pmatrix} \tag{12}$$

$$\boldsymbol{\Sigma}_t = \mathbf{1}\mathbf{1}^\intercal \otimes (\mathbf{K} - \mathbf{K}(\mathbf{K} + \gamma/t\mathbf{I}_N)^{-1}\mathbf{K}) \tag{13}$$

其中 $\otimes$ 是克罗内克积。

证明。 为简单起见，我们仅证明 $t = 2$ 的情况，因为一般证明类似。关键观察是先验协方差矩阵可以写成

$$\begin{Bmatrix} \mathbf{K} & \mathbf{K} \\ \mathbf{K} & \mathbf{K} \end{Bmatrix} + \gamma\mathbf{I}_{2N} = \begin{Bmatrix} \mathbf{K} + \gamma\mathbf{I}_N & \mathbf{K} \\ \mathbf{K} & \mathbf{K} + \gamma\mathbf{I}_N \end{Bmatrix}$$

利用分块矩阵求逆公式，我们得到

$$\begin{Bmatrix} \mathbf{K} + \gamma\mathbf{I}_N & \mathbf{K} \\ \mathbf{K} & \mathbf{K} + \gamma\mathbf{I}_N \end{Bmatrix}^{-1} = \begin{Bmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} \\ \mathbf{A}_{21} & \mathbf{A}_{22} \end{Bmatrix}$$

where

$$\mathbf{A}_{11} = \mathbf{A}_{22} = (\mathbf{K} + \gamma\mathbf{I}_N - \mathbf{K}(\mathbf{K} + \gamma\mathbf{I}_N)^{-1}\mathbf{K})^{-1}$$
$$\mathbf{A}_{12} = \mathbf{A}_{21} = -(\mathbf{K} + \gamma\mathbf{I}_N - \mathbf{K}(\mathbf{K} + \gamma\mathbf{I}_N)^{-1}\mathbf{K})^{-1}\mathbf{K}(\mathbf{K} + \gamma\mathbf{I}_N)^{-1}$$

并且这些可以进一步简化得到

$$\left(\begin{Bmatrix}\mathbf{K} & \mathbf{K}\\ \mathbf{K} & \mathbf{K}\end{Bmatrix} + \mathbf{I}_{2N}\right)^{-1} = \begin{Bmatrix}-(2\mathbf{K}\gamma + \gamma^2\mathbf{I}_N)^{-1}(\mathbf{K} + \gamma\mathbf{I}_N) & (2\mathbf{K}\gamma + \gamma^2\mathbf{I}_N)^{-1}\mathbf{K}\\ -(2\mathbf{K}\gamma + \gamma^2\mathbf{I}_N)^{-1}\mathbf{K} & (2\mathbf{K}\gamma + \gamma^2\mathbf{I}_N)^{-1}(\mathbf{K} + \gamma\mathbf{I}_N)\end{Bmatrix},$$

因此我们得到

$$\begin{aligned}\boldsymbol{\mu}_2 &= \begin{Bmatrix}\mathbf{K} & \mathbf{K}\\ \mathbf{K} & \mathbf{K}\end{Bmatrix}\begin{Bmatrix}(2\mathbf{K}\gamma + \gamma^2\mathbf{I}_N)^{-1}(\mathbf{K} + \gamma\mathbf{I}_N) & -(2\mathbf{K}\gamma + \gamma^2\mathbf{I}_N)^{-1}\mathbf{K}\\ -(2\mathbf{K}\gamma + \gamma^2\mathbf{I}_N)^{-1}\mathbf{K} & (2\mathbf{K}\gamma + \gamma^2\mathbf{I}_N)^{-1}(\mathbf{K} + \gamma\mathbf{I}_N)\end{Bmatrix}\begin{pmatrix}\mathbf{y}\\ \mathbf{y}\end{pmatrix}\\ &= \begin{Bmatrix}\mathbf{K} & \mathbf{K}\\ \mathbf{K} & \mathbf{K}\end{Bmatrix}\begin{pmatrix}(2\mathbf{K}\gamma + \gamma^2\mathbf{I}_N)^{-1}\gamma\mathbf{y}\\ (2\mathbf{K}\gamma + \gamma^2\mathbf{I}_N)^{-1}\gamma\mathbf{y}\end{pmatrix}\\ &= \begin{pmatrix}2\mathbf{K}(2\mathbf{K}\gamma + \gamma^2\mathbf{I}_N)^{-1}\gamma\mathbf{y}\\ 2\mathbf{K}(2\mathbf{K}\gamma + \gamma^2\mathbf{I}_N)^{-1}\gamma\mathbf{y}\end{pmatrix} = \begin{pmatrix}\mathbf{K}(\mathbf{K} + (\gamma/2)\mathbf{I}_N)^{-1}\mathbf{y}\\ \mathbf{K}(\mathbf{K} + (\gamma/2)\mathbf{I}_N)^{-1}\mathbf{y}\end{pmatrix}.\end{aligned}$$

类似地，协方差矩阵为

$$\boldsymbol{\Sigma}_2 = \begin{Bmatrix}\mathbf{K} & \mathbf{K}\\ \mathbf{K} & \mathbf{K}\end{Bmatrix} - \begin{Bmatrix}\mathbf{K} & \mathbf{K}\\ \mathbf{K} & \mathbf{K}\end{Bmatrix}\begin{Bmatrix}(2\mathbf{K}\gamma + \gamma^2\mathbf{I}_N)^{-1}(\mathbf{K} + \gamma\mathbf{I}_N) & -(2\mathbf{K}\gamma + \gamma^2\mathbf{I}_N)^{-1}\mathbf{K}\\ -(2\mathbf{K}\gamma + \gamma^2\mathbf{I}_N)^{-1}\mathbf{K} & (2\mathbf{K}\gamma + \gamma^2\mathbf{I}_N)^{-1}(\mathbf{K} + \gamma\mathbf{I}_N)\end{Bmatrix}\begin{Bmatrix}\mathbf{K} & \mathbf{K}\\ \mathbf{K} & \mathbf{K}\end{Bmatrix}$$

可以简化得到

$$\boldsymbol{\Sigma}_2 = \begin{Bmatrix}\mathbf{K} - \mathbf{K}(\mathbf{K} + (\gamma/2)\mathbf{I}_N)^{-1}\mathbf{K} & \mathbf{K} - \mathbf{K}(\mathbf{K} + (\gamma/2)\mathbf{I}_N)^{-1}\mathbf{K}\\ \mathbf{K} - \mathbf{K}(\mathbf{K} + (\gamma/2)\mathbf{I}_N)^{-1}\mathbf{K} & \mathbf{K} - \mathbf{K}(\mathbf{K} + (\gamma/2)\mathbf{I}_N)^{-1}\mathbf{K}\end{Bmatrix},$$

如预期。 □

B 第 节的证明 4 （高斯过程分类的自蒸馏）

命题 4.1。 设 $C(\sigma(a))$ 按(20) 定义，且 $\lambda = \sigma(a)$，则有

$$C(\sigma(a)) = \begin{cases}2 & \textit{if } a = 0\\ a\coth\left(\frac{a}{2}\right) & \textit{otherwise},\end{cases}$$
$$\frac{d}{da}\log(C(\sigma(a))) = \begin{cases}0 & \textit{if } a = 0\\ \frac{1}{a} - \frac{1}{\sinh(a)} & \textit{otherwise},\end{cases}$$
$$\frac{d^2}{d^2a}\log(C(\sigma(a))) = \begin{cases}\frac{1}{6} & \textit{if } a = 0\\ -\frac{1}{a^2} + \frac{\coth(a)}{\sinh(a)} & \textit{otherwise},\end{cases}$$

其中 双曲余切函数 和 双曲正弦函数 分别是双曲余切函数和正弦函数。

证明。 首先我们注意到，对于 $a \neq 0$，可以证明 that

$$C(\sigma(a)) = \frac{2\tanh^{-1}(1 - 2\sigma(a))}{1 - 2\sigma(a)} = a\coth\left(\frac{a}{2}\right), \tag{32}$$

这源于 $\tanh(x) = \frac{e^{2x}-1}{e^{2x}+1}$ 这一事实 ——

$$1 - 2\sigma(a) = 1 - \frac{2e^a}{1 + e^a} = \frac{1 + e^a}{1 + e^a} - \frac{2e^a}{1 + e^a} = \frac{1 - e^a}{1 + e^a} = \tanh\left(-\frac{a}{2}\right)$$

因此可直接推出

$$2\tanh^{-1}(1 - 2\sigma(a)) = 2\tanh^{-1}\left(\tanh\left(-\frac{a}{2}\right)\right) = \frac{-2a}{2} = -a$$

最后通过双曲余切函数$(x) = \frac{1}{\tanh(x)}$以及$\tanh(-x) = -\tanh(x)$的

$$\frac{1}{1-2\sigma(a)} = \frac{1}{-\tanh\left(\frac{a}{2}\right)} = -\coth\left(\frac{a}{2}\right),$$

结合后得到（32）中的结论。现在，根据双曲函数的性质[1] 通过直接计算可得

$$\begin{aligned}
\frac{d}{da}\log(C(\sigma(a))) &= \frac{d}{da}\log(a) + \frac{d}{da}\log\coth\left(\frac{a}{2}\right) && \text{for } a \neq 0 \\
&= \frac{1}{a} + \frac{1}{\coth\left(\frac{a}{2}\right)}\frac{d}{da}\coth\left(\frac{a}{2}\right) && \text{for } a \neq 0 \\
&= \frac{1}{a} + \frac{\sinh\left(\frac{a}{2}\right)}{\cosh\left(\frac{a}{2}\right)}\left(-\frac{1}{2\sinh^2\left(\frac{a}{2}\right)}\right) && \text{for } a \neq 0 \\
&= \frac{1}{a} - \frac{1}{2\cosh\left(\frac{a}{2}\right)\sinh\left(\frac{a}{2}\right)} && \text{for } a \neq 0 \\
&= \frac{1}{a} - \frac{1}{\sinh(a)} \qquad \left(= \frac{1}{a} - \operatorname{csch}(a)\right) && \text{对于 } a \neq 0.
\end{aligned}$$

此外，根据双曲函数的进一步性质[2] 可直接得到

$$\begin{aligned}
\frac{d^2}{d^2a}\log(C(\sigma(a))) &= -\frac{1}{a^2} - \frac{d}{da}\operatorname{csch}(x) && \text{for } a \neq 0 \\
&= -\frac{1}{a^2} + \operatorname{csch}(a)\coth(a) && \text{for } a \neq 0. \\
&= -\frac{1}{a^2} + \frac{1}{\sinh(a)}\frac{1}{\tanh(a)} && \text{for } a \neq 0.
\end{aligned}$$

还需证明关于 $a = 0$的两个导数。首先令 $f(a) = \log(C(\sigma(a))) = \log a$双曲余切函数 $\left(\frac{a}{2}\right)$ for$a \neq 0$，而在其他地方为 log(2)，则有 $f(-x) = \log\left(-x\text{双曲余切函数}\left(\frac{-x}{2}\right)\right) = \log(x\text{双曲余切函数}\left(\frac{x}{2}\right)) = f(x)$，并且不失一般性地，我们只需考虑右极限。注意到

$$\begin{aligned}
\lim_{h\downarrow 0}\frac{f(h)-f(0)}{h} &= \lim_{h\downarrow 0}\frac{d}{dx}f(h) \\
&= \lim_{h\downarrow 0}\left(\frac{1}{h} - \frac{1}{\sinh(h)}\right) \\
&= \lim_{h\downarrow 0}\left(\frac{\sinh(h)-h}{h\sinh(h)}\right),
\end{aligned}$$

并且对 $0/0$ 极限应用洛必达法则，我们得到

$$\lim_{h\downarrow 0}\frac{f(h)-f(0)}{h} = \lim_{h\downarrow 0}\left(\frac{\cosh(h)-1}{\sinh(h)+x\cosh(h)}\right) = 0.$$

通过类似论证可得，由于 $\sinh(-x) = -\sinh(x)$，因此 $\frac{d}{dx}f(-x) = -\frac{d}{dx}f(x)$，并且我们可以不失一般性地只考虑右极限并应用洛必达法则，从而得到

$$\lim_{h\downarrow 0}\frac{\frac{d}{dx}f(h) - \frac{d}{dx}f(0)}{h} = \lim_{h\downarrow 0}\frac{d^2}{d^2x}f(h) = \lim_{h\downarrow 0}\left(-\frac{1}{a^2} + \frac{1}{\sinh(a)}\frac{1}{\tanh(a)}\right) = \frac{1}{6}$$

至此证毕。 □

命题*4.2*。（*A*）使用 $\mathcal{D}_t$ 和高斯先验$\mathcal{GP}(0,k)$进行一次分布中心蒸馏，所得后验分布与使用高斯先验 $\mathcal{GP}(0,tk)$进行一次分布中心蒸馏所得的后验分布相同。（*B*）使用 $\mathcal{D}$ 进行 t 次分布中心蒸馏，并以初始高斯先验 $\mathcal{GP}(0,k)$ 开始，其所得后验分布近似等于使用高斯先验 $\mathcal{GP}(0,tk)$进行一次分布中心蒸馏所得的后验分布。

[1] In particular that $\sinh(2x) = 2\sinh(x)\cosh(x)$ and that $\frac{d}{da}\coth(x) = -\frac{1}{\sinh^2(x)}$ for $x \neq 0$.

[2] Here, in particular $\frac{d}{da}\operatorname{csch}(x) = -\operatorname{csch}(x)\coth(x)$, $\coth(x) = \frac{1}{\tanh(x)}$ and $\operatorname{csch}(x) = \frac{1}{\sinh(x)}$.

证明。（**A**）
首先，我们注意到，复制数据只意味着对应于给定潜变量时数据条件分布的项会重复 t 次。使用第A.1节中的记号，这意味着对数后验将满足

$$\psi(\mathbf{f}) \stackrel{\text{def}}{=} \log p(\mathbf{f} \mid \mathbf{y}) \underset{\mathbf{f}}{\propto} \log p(\mathbf{f}) + t \log p(\mathbf{y} \mid \mathbf{f})$$

其梯度为

$$\nabla\psi(\mathbf{f}) = -\mathbf{K}^{-1}\mathbf{f} + t(\mathbf{y} - \boldsymbol{\sigma})$$

ˆ 并令其等于零，我们可以看到，最大化器$\mathbf{f}$ 将满足

$$\hat{\mathbf{f}} = (t\mathbf{K})(\mathbf{y} - \boldsymbol{\sigma})\,.$$

将其与式（17）所隐含的单步蒸馏对应方程进行比较，可以看出这两种方法给出相同的结果。

接下来，我们希望证明（**B**）。为简单起见，我们将 $t = 2$ 视为自变量；对于一般的 t，证明过程类似。由（**A**）可知，只需证明两步蒸馏的结果近似等同于基于 $\mathcal{D}_2$ 的单步蒸馏结果，即可得到所需结论。对于后者，由（**A**）的证明可知，后验分布正比于

$$\left(\prod_{n=1}^{N} \sigma(f_n)^{t_n}(1 - \sigma(f_n))^{1-t_n}\right)^2 \exp\left(-\frac{1}{2}\mathbf{f}^\intercal\mathbf{K}^{-1}\mathbf{f}\right)\,. \tag{33}$$

定义

$$\varphi(\mathbf{f}) \stackrel{\text{def}}{=} \left(\prod_{n=1}^{N} \sigma(f_n)^{t_n}(1 - \sigma(f_n))^{1-t_n}\right) \exp\left(-\frac{1}{2}\mathbf{f}^\intercal\mathbf{K}^{-1}\mathbf{f}\right)$$

我们可以将（33）重写为

$$\left(\prod_{n=1}^{N} \sigma(f_n)^{t_n}(1 - \sigma(f_n))^{1-t_n}\right) \varphi(\mathbf{f})\,. \tag{34}$$

现在我们只需注意到 2 步蒸馏对应于将 $\varphi(\mathbf{f})$ 在 (34) 中替换为来自拉普拉斯分布的适当高斯分布，因此该近似源于 $\varphi(\mathbf{f})$ 本身并不与某个高斯分布严格成比例。

□