面向大规模数据的高斯过程机器学习模型

研究进展

贺建军，许小可

大连民族大学

引言

高斯过程模型是源于贝叶斯神经网络而逐渐发展起来的一种机器学习算法建模工具。20 世纪 90 年代中期，Neal[1] 在对贝叶斯神经网络的研究过程中发现，具有一个隐藏层的神经网络当隐藏节点的个数趋于无穷时，该神经网络等价于一个高斯过程。此发现激发了人们直接利用高斯过程处理监督学习问题的兴趣，Rasmussen[2] 在他的博士论文里详细研究了面向回归问题的高斯过程模型；文献 [3] 系统地研究了基于高斯过程模型的回归和分类算法。随后，许多学者从不同方面展开了研究。经过20年的发展，高斯过程模型目前已经发展成为了一种具有完整理论体系的建模工具，并被用于回归 [2]、分类 [3]、多任务学习 [4]、强化学习 [5]、半监督学习 [6]、排序学习 [7]、多示例多标记学习 [8]、多视角学习 [9] 和偏标记学习 [10] 等各种机器学习框架中。

令 D 表示观察数据、H 表示假设空间，机器学习的目的就是利用数据 D 在假设空间 H 中寻找能对新样本的类别标记进行精确预测的假设 $f \in H$。高斯过程模型的基本思想是赋予 f 一个高斯过程先验 $f(x) \sim \mathrm{GP}(0, k(x,x'))$，然后利用贝叶斯公式 $p(f \mid D) = \dfrac{p(D \mid f)p(f)}{\int p(D \mid f)p(f)\mathrm{d}f}$ 推理得到 f 的后验概率分布 $p(f \mid D)$，从而得到最佳假设。其中，$k(x,x')$ 表示协方差函数；$p(D|f)$ 是似然，表示 f 成立的情况下观察到数据 D 的概率，它实质上反映的就是经验风险。以回归和二分类问题为例，令 $\boldsymbol{X}=\{\boldsymbol{x}_i\}_{i=1}^n$ 和 $\boldsymbol{Y}=[y_1,y_2,\cdots,y_n]^{\mathrm{T}}$ 分别表示训练样本的特征向量和类别标记构成的集合，根据高斯过程模型中先验概率的定义，潜变量函数（假设）f 在 $\boldsymbol{X}$ 上的函数值 $\boldsymbol{F}_X=[f_1, f_2,\cdots, f_n]^{\mathrm{T}}$ 的先验概率是一个多元高斯分布

$p(\boldsymbol{F}_X|\boldsymbol{X})=N(\boldsymbol{F}_X|0,K_X)$，其中 $\boldsymbol{K}_X$ 为 $\boldsymbol{X}$ 中样本间的协方差矩阵 $(\boldsymbol{K}_X(i,j)=k(\boldsymbol{x}_i,\boldsymbol{x}_j))$；然后利用贝叶斯公式，$\boldsymbol{F}_X$ 的后验概率分布可以表示为

$$p(\boldsymbol{F}_X|\boldsymbol{X},\boldsymbol{Y})=\frac{p(\boldsymbol{F}_X|\boldsymbol{X})\,p(\boldsymbol{Y}|\boldsymbol{F}_X)}{\int p(\boldsymbol{F}_X|\boldsymbol{X})p(\boldsymbol{Y}|\boldsymbol{F}_X)\mathrm{d}\boldsymbol{F}_X}$$

最后根据 $p(\boldsymbol{F}_X|\boldsymbol{X},\boldsymbol{Y})$ 可以计算得到 f 在待预测样本 $\boldsymbol{x}_*$ 上的函数值的后验概率

$$p(f_*|\boldsymbol{X},\boldsymbol{Y})=\int p(f_*|\boldsymbol{F}_X)p(\boldsymbol{F}_X|\boldsymbol{X},\boldsymbol{Y})\mathrm{d}\boldsymbol{F}_X$$

从而可以得到 $\boldsymbol{x}_*$ 的类别标记。对于回归问题，定义的似然函数通常是一个高斯型函数，如 $p(\boldsymbol{Y}|\boldsymbol{F}_X)=N(\boldsymbol{Y}|\boldsymbol{F}_X,\sigma^2\boldsymbol{I})$。因此 $p(\boldsymbol{F}_X|\boldsymbol{X},\boldsymbol{Y})$ 仍然是一个高斯分布，从而可以得到 $p(f_*|\boldsymbol{X},\boldsymbol{Y})$ 的分析表达式

$$p(f_*|\boldsymbol{X},\boldsymbol{Y})=N\ (f_*|\boldsymbol{K}_{X*}^{\mathrm{T}}(\boldsymbol{K}_X+\sigma^2\boldsymbol{I})^{-1}\boldsymbol{Y},\boldsymbol{K}_*-\boldsymbol{K}_{X*}^{\mathrm{T}}(\boldsymbol{K}_X+\sigma^2\boldsymbol{I})^{-1}\boldsymbol{K}_{X*})$$

其中 $\boldsymbol{K}_{X*}$ 表示 $\boldsymbol{X}$ 中样本和 $\boldsymbol{x}_*$ 之间的协方差矩阵，$\boldsymbol{K}_*=k(\boldsymbol{x}_*,\boldsymbol{x}_*)$。对于分类问题，似然函数通常是一个非高斯型函数，如常用的似然函数 $p(\boldsymbol{Y}|\boldsymbol{F}_X)=\prod_{i=1}^{n}\frac{1}{1+\mathrm{e}^{-y_if_i}}$ 是一个 S 型函数。因此 $p(\boldsymbol{F}_X|\boldsymbol{X},\boldsymbol{Y})$ 不是一个高斯分布，需要先计算 $p(\boldsymbol{F}_X|\boldsymbol{X},\boldsymbol{Y})$ 的一个近似表达式 $q(\boldsymbol{F}_X)$（通常是一个高斯分布）；然后利用 $q(\boldsymbol{F}_X)$ 来近似地计算 $p(f_*|\boldsymbol{X},\boldsymbol{Y})\approx\int p(f_*|\boldsymbol{F}_X)q(\boldsymbol{F}_X)\mathrm{d}\boldsymbol{F}_X$，以上就是传统高斯过程模型解决回归或分类问题的基本过程。由于需要计算并存储协方差矩阵 $\boldsymbol{K}_X$ 及其逆矩阵，而且当似然函数是非高斯函数时还需要计算后验概率分布 $p(\boldsymbol{F}_X|\boldsymbol{X},\boldsymbol{Y})$ 的近似表达式，传统高斯过程模型的存储和计算复杂度通常分别是 $O(n^2)$ 和 $O(n^3)$，因此只能处理训练样本规模比较小的问题。为了降低模型的复杂度，使其可以处理大规模数据，近年来，人们先后提出了诱导变量、协方差函数分解、协方差矩阵逼近和贝叶斯委员会机器等几种解决方法，本文将对这些方法进行简要介绍。

诱导变量方法

诱导变量方法的基本思想是从样本集合 $\boldsymbol{X}$ 中，选择一个样本数目较少的子集 $\boldsymbol{U}\subset\boldsymbol{X}$，构造一组诱导变量来辅助计算 $\boldsymbol{F}_X$ 的概率分布，从而降低模型的复杂度。由于回归问题的主要计算量来自 $\boldsymbol{F}_X$ 的先验概率 $p(\boldsymbol{F}_X|\boldsymbol{X})$，因此早期的诱导变量方法主要侧重如何降低 $p(\boldsymbol{F}_X|\boldsymbol{X})$ 的计算复杂度。利用乘法法则可以将 $p(\boldsymbol{F}_X|\boldsymbol{X})$ 分解为 $p(\boldsymbol{F}_X|\boldsymbol{F}_U,\boldsymbol{X})$ 和 $p(\boldsymbol{F}_U|\boldsymbol{X})$ 的乘积，即

$$p(\boldsymbol{F}_X|\boldsymbol{X})=p(\boldsymbol{F}_U|\boldsymbol{X})p(\boldsymbol{F}_X|\boldsymbol{F}_U,\boldsymbol{X}),$$

其中 $\boldsymbol{F}_U$ 表示 f 在 U 上的函数值。根据条件概率公式，$p(\boldsymbol{F}_X|\boldsymbol{F}_U,\boldsymbol{X})$ 的表达式可以分析得到，即

$$p(\boldsymbol{F}_X|\boldsymbol{F}_U,\boldsymbol{X})=\boldsymbol{N}(\boldsymbol{F}_X|\boldsymbol{K}_{XU}\boldsymbol{K}_U^{-1}\boldsymbol{F}_U,\boldsymbol{\Sigma}_X)$$

其中 $\boldsymbol{K}_{XU}$ 和 $\boldsymbol{K}_U$ 是对应的协方差矩阵、$\boldsymbol{\Sigma}_X=\boldsymbol{K}_X-\boldsymbol{K}_{XU}\boldsymbol{K}_U^{-1}\boldsymbol{K}_{XU}^{\mathrm{T}}$。由于 $p(\boldsymbol{F}_X|\boldsymbol{F}_U,\boldsymbol{X})$ 的计算复杂度仍然较高，因此人们进一步提出在“已知 $\boldsymbol{F}_U$ 的条件下，$\boldsymbol{F}_X$ 中的变量之间具有一定的独立性”这个假设条件下来近似地表示 $p(\boldsymbol{F}_X|\boldsymbol{F}_U,\boldsymbol{X})$，比较有代表性的有 DTC[11]、FITC[12] 和 PITC[13] 三种。DTC 方法假设 $\boldsymbol{F}_X$ 中的变量之间的协方差为 0，因此在该类方法中

$$p(\boldsymbol{F}_X|\boldsymbol{F}_U,\boldsymbol{X})\approx\boldsymbol{N}(\boldsymbol{F}_X|\boldsymbol{K}_{XU}\boldsymbol{K}_U^{-1}\boldsymbol{F}_U,0)$$

FITC 方法假设 $\boldsymbol{F}_X$ 中的各个变量之间是相互独立的，因此

$$p(\boldsymbol{F}_X|\boldsymbol{F}_U,\boldsymbol{X})\approx\boldsymbol{N}(\boldsymbol{F}_X|\boldsymbol{K}_{XU}\boldsymbol{K}_U^{-1}\boldsymbol{F}_U,\mathrm{diag}(\boldsymbol{\Sigma}_X))$$

PITC 方法假设 $\boldsymbol{F}_X$ 的各个变量之间是部分独立的，因此条件变量 $\boldsymbol{F}_X|\boldsymbol{F}_U$ 之间的协方差矩阵是一个分块对角矩阵，$p(\boldsymbol{F}_X|\boldsymbol{F}_U,\boldsymbol{X})$ 可

以表示为 $N(\boldsymbol{F}_X|\boldsymbol{K}_{XU}\boldsymbol{K}_U^{-1}\boldsymbol{F}_U,\text{blockdiag}(\boldsymbol{\Sigma}_X))$。

可以看出，利用以上几种方法近似计算先验概率 $p(\boldsymbol{F}_X|\boldsymbol{X})$ 的复杂度是 $O(nm^2)$，其中 m 表示诱导样本的个数。由于当 m 的取值通常达到数百时（远远小于 n），算法就可以取得较高的精度，因此采用以上描述的诱导变量方法可以有效降低模型的计算复杂度。但是该类方法也存在很多问题，例如利用边缘似然函数学习参数的效果将较差，对于非高斯型似然函数，$\boldsymbol{F}_X$ 的后验概率分布的计算仍然很困难，不利于处理回归问题以外的机器学习问题。针对这些问题，近年来，人们开始关注直接逼近后验概率分布 $p(\boldsymbol{F}_X|\boldsymbol{X},\boldsymbol{Y})$ 的诱导变量方法的研究。与传统高斯过程模型中利用贝叶斯公式直接计算 $p(\boldsymbol{F}_X|\boldsymbol{X},\boldsymbol{Y})$ 的方式不同，该类方法的基本思想是先计算联合后验概率分布 $p(\boldsymbol{F}_X,\boldsymbol{F}_U|\boldsymbol{X},\boldsymbol{Y})$ 的一个高斯逼近 $q(\boldsymbol{F}_X,\boldsymbol{F}_U)$（或者后验概率分布 $p(\boldsymbol{F}_U|\boldsymbol{X},\boldsymbol{Y})$ 的高斯逼近 $q(\boldsymbol{F}_U)=N(\boldsymbol{F}_U|\boldsymbol{\mu},\boldsymbol{\Sigma})$）；然后对 $\boldsymbol{F}_U$ 积分来计算 $p(\boldsymbol{F}_X|\boldsymbol{X},\boldsymbol{Y})$，即

$$p(\boldsymbol{F}_X|\boldsymbol{X},\boldsymbol{Y})=\int p(\boldsymbol{F}_X,\boldsymbol{F}_U|\boldsymbol{X},\boldsymbol{Y})\mathrm{d}\boldsymbol{F}_U \approx \int q(\boldsymbol{F}_X,\boldsymbol{F}_U)\mathrm{d}\boldsymbol{F}_U$$

或者

$$p(\boldsymbol{F}_X|\boldsymbol{X},\boldsymbol{Y})=\int p(\boldsymbol{F}_X|\boldsymbol{F}_U,\boldsymbol{X})p(\boldsymbol{F}_U|\boldsymbol{X},\boldsymbol{Y})\mathrm{d}\boldsymbol{F}_U \approx \int p(\boldsymbol{F}_X|\boldsymbol{F}_U,\boldsymbol{X})q(\boldsymbol{F}_U)\mathrm{d}\boldsymbol{F}_U$$

这类方法的代表性成果是 2009 年 Titsias[14] 面向回归问题提出的 VAR 算法。该算法先将联合概率分布 $p(\boldsymbol{F}_X,\boldsymbol{F}_U|\boldsymbol{X},\boldsymbol{Y})$ 表示为

$$p(\boldsymbol{F}_X,\boldsymbol{F}_U|\boldsymbol{X},\boldsymbol{Y})=\frac{p(\boldsymbol{F}_X|\boldsymbol{F}_U,\boldsymbol{X})p(\boldsymbol{F}_U|\boldsymbol{X})p(\boldsymbol{Y}|\boldsymbol{F}_X)}{p(\boldsymbol{Y}|\boldsymbol{X})}$$

然后通过最小化 KL 散度 $\mathrm{KL}(q(\boldsymbol{F}_X,\boldsymbol{F}_U)\|p(\boldsymbol{F}_X,\boldsymbol{F}_U|\boldsymbol{X},\boldsymbol{Y}))$ 来计算 $q(\boldsymbol{F}_X,\boldsymbol{F}_U)$，并且假设 $q(\boldsymbol{F}_X,\boldsymbol{F}_U)=p(\boldsymbol{F}_X|\boldsymbol{F}_U)q(\boldsymbol{F}_U)$，因此只需通过 $\min_{\boldsymbol{\mu},\boldsymbol{\Sigma}}\mathrm{KL}(q(\boldsymbol{F}_X,\boldsymbol{F}_U)\|p(\boldsymbol{F}_X,\boldsymbol{F}_U|\boldsymbol{X},\boldsymbol{Y}))$ 计算得到 $\boldsymbol{\mu},\boldsymbol{\Sigma}$ 即可。VAR 算法不仅计算复杂度是 $O(nm^2)$，而且提供了一个可以优化诱导点位置的似然下界，因此受到了人们广泛的关注。由于 $\min_{\boldsymbol{\mu},\boldsymbol{\Sigma}}\mathrm{KL}(q(\boldsymbol{F}_X,\boldsymbol{F}_U)\|p(\boldsymbol{F}_X,\boldsymbol{F}_U|\boldsymbol{X},\boldsymbol{Y}))$ 等价于

$$\max_{\boldsymbol{\mu},\boldsymbol{\Sigma}}\iint q(\boldsymbol{F}_X,\boldsymbol{F}_U)\log\frac{p(\boldsymbol{F}_X,\boldsymbol{F}_U,\boldsymbol{Y}|\boldsymbol{X})}{q(\boldsymbol{F}_X,\boldsymbol{F}_U)}\mathrm{d}\boldsymbol{F}_X\mathrm{d}\boldsymbol{F}_U$$

而 $\iint q(\boldsymbol{F}_X,\boldsymbol{F}_U)\log\frac{p(\boldsymbol{F}_X,\boldsymbol{F}_U,\boldsymbol{Y}|\boldsymbol{X})}{q(\boldsymbol{F}_X,\boldsymbol{F}_U)}\mathrm{d}\boldsymbol{F}_X\mathrm{d}\boldsymbol{F}_U$ 就是人们熟知的对数似然下界（evidence lower bound, ELBO），即

$$\begin{aligned}&\log p(\boldsymbol{Y}|\boldsymbol{X})\\&=\log\iint p(\boldsymbol{F}_X,\boldsymbol{F}_U,\boldsymbol{Y}|\boldsymbol{X})\,\mathrm{d}\boldsymbol{F}_X\mathrm{d}\boldsymbol{F}_U\\&\geqslant\iint q(\boldsymbol{F}_X,\boldsymbol{F}_U)\log\frac{p(\boldsymbol{F}_X,\boldsymbol{F}_U,\boldsymbol{Y}|\boldsymbol{X})}{q(\boldsymbol{F}_X,\boldsymbol{F}_U)}\mathrm{d}\boldsymbol{F}_X\mathrm{d}\boldsymbol{F}_U\end{aligned}$$

因此也可以直接通过最大化对数似然下界来计算 $q(\boldsymbol{F}_U)$。考虑到 VAR 算法不易采用随机梯度法来计算 $q(\boldsymbol{F}_U)$，Hensman 等[15] 提出了新的对数似然下界

$$\begin{aligned}&\log p(\boldsymbol{Y}|\boldsymbol{X})\\&\geqslant\iint q(\boldsymbol{F}_U)p(\boldsymbol{F}_X|\boldsymbol{F}_U,\boldsymbol{X})\log p(\boldsymbol{Y}|\boldsymbol{F}_X)\mathrm{d}\boldsymbol{F}_X\mathrm{d}\boldsymbol{F}_U\\&-\mathrm{KL}(q(\boldsymbol{F}_U)\|p(\boldsymbol{F}_U|\boldsymbol{X}))\end{aligned}$$

并且将其推广到了分类问题中[16]。以上几种基于变分推理方法来计算 $q(\boldsymbol{F}_U)$ 的策略，虽然在回归或者二分类问题中可以取得较好的效果，但是不易处理似然函数更为复杂的问题，为此本课题组提出了一种基于拉普拉斯方法来计算 $q(\boldsymbol{F}_U)$ 的策略[17]。首先将 $p(\boldsymbol{F}_U|\boldsymbol{X},\boldsymbol{Y})$ 表示为

$$p(\boldsymbol{F}_U|\boldsymbol{X},\boldsymbol{Y})=\frac{p(\boldsymbol{Y}|\boldsymbol{F}_U)p(\boldsymbol{F}_U|\boldsymbol{X})}{\int p(\boldsymbol{Y}|\boldsymbol{F}_U)p(\boldsymbol{F}_U|\boldsymbol{X})\mathrm{d}\boldsymbol{F}_U}$$

其中 $p(\boldsymbol{Y}|\boldsymbol{F}_U)=\int p(\boldsymbol{Y}|\boldsymbol{F}_X)p(\boldsymbol{F}_X|\boldsymbol{F}_U,\boldsymbol{X})\mathrm{d}\boldsymbol{F}_X$；然后利用拉普拉斯方法来计算高斯逼近 $q(\boldsymbol{F}_U)$，即

$$\boldsymbol{\mu}=\underset{\boldsymbol{F}_U}{\operatorname{argmax}}\ \log\ p(\boldsymbol{F}_U\mid\boldsymbol{X},\boldsymbol{Y})$$
$$\boldsymbol{\Sigma}=-(\nabla\nabla\log\ p(\boldsymbol{F}_U\mid\boldsymbol{X},\boldsymbol{Y})\big|_{\boldsymbol{F}_U=\boldsymbol{\mu}})^{-1}$$

该方法不仅在二分类问题上可以取得较高的精度，而且可以处理具有复杂似然函数的问题，目前已经用于处理多示例学习问题和偏标记学习问题。此外，Hernández-Lobato 等[18]在表达式

$$p(\boldsymbol{F}_U|\boldsymbol{X},\boldsymbol{Y})=\frac{p(\boldsymbol{Y}|\boldsymbol{F}_U)p(\boldsymbol{F}_U|\boldsymbol{X})}{\int p(\boldsymbol{Y}|\boldsymbol{F}_U)p(\boldsymbol{F}_U|\boldsymbol{X})\mathrm{d}\boldsymbol{F}_U}$$

的基础上，并且在 FITC 条件先验假设 $p(\boldsymbol{F}_X\mid\boldsymbol{F}_U,\boldsymbol{X})\approx\boldsymbol{N}(\boldsymbol{F}_X|\boldsymbol{K}_{XU}\boldsymbol{K}_U^{-1}\boldsymbol{F}_U,\mathrm{diag}(\boldsymbol{\Sigma}_X))$ 下，利用 EP 方法得到了 $p(\boldsymbol{F}_U\mid\boldsymbol{X},\boldsymbol{Y})$ 的一个新的逼近 $q(\boldsymbol{F}_U)=\prod_{i=1}^{n}\tilde{\phi}_i(\boldsymbol{F}_U)p(\boldsymbol{F}_U\mid\boldsymbol{X})/\boldsymbol{Z}_q$，并在文献 [19] 中将其推广到了多分类问题，2017 年，Bui 等[20]将 EP 方法和变分方法统一在同一个框架下，建立了一种基于 Power EP 方法的逼近算法。

协方差函数分解方法

该类方法的策略是先构造一组显式特征映射 $\boldsymbol{\phi}(\boldsymbol{x})$ 来近似地逼近协方差函数，即 $k(\boldsymbol{x},\boldsymbol{y})\approx\boldsymbol{\phi}(\boldsymbol{x})^{\mathrm{T}}\boldsymbol{\phi}(\boldsymbol{y})$；换句话而言，就是利用由这组显式特征映射确定的低维显式特征空间，近似表示由协方差函数确定的高维（无限维）隐式特征空间；然后在 $\boldsymbol{\phi}(\boldsymbol{x})$ 确定的显式特征空间中构造一个线性高斯过程模型来建立学习算法。由于线性高斯模型的计算开销比较小，因此可以较好地扩展到大规模数据问题。Bochner 定理[21]表明，一个连续函数正定的充要条件是该函数可以表示为一个有限非负 Borel 测度的傅里叶变换。对于平稳高斯过程的协方差函数 $k(\boldsymbol{x},\boldsymbol{y})=k(\boldsymbol{x}-\boldsymbol{y})$，根据 Bochner 定理可知，只要对 $k(\boldsymbol{x},\boldsymbol{y})$ 进行合适的尺度缩放 $\tilde{k}(\boldsymbol{x},\boldsymbol{y})=\frac{1}{\sigma^2}k(\boldsymbol{x},\boldsymbol{y})$，就可存在一个概率分布 $s(\boldsymbol{\omega})$使得它们之间存在如下的关系：

$$\tilde{k}(\boldsymbol{x},\boldsymbol{y})=\int_{R^d}s(\boldsymbol{\omega})\mathrm{e}^{i\boldsymbol{\omega}^{\mathrm{T}}(\boldsymbol{x}-\boldsymbol{y})}\mathrm{d}\boldsymbol{\omega}=E_{\omega}(\zeta_{\omega}(\boldsymbol{x})\zeta_{\omega}(\boldsymbol{y})^{*})$$

其中 $\zeta_{\omega}(\boldsymbol{x})=\mathrm{e}^{i\boldsymbol{\omega}^{\mathrm{T}}\boldsymbol{x}}$。因此，如果根据概率分布 $s(\boldsymbol{\omega})$ 对 $\boldsymbol{\omega}$ 进行采样，可以利用 $\zeta_{\omega}(\boldsymbol{x})\zeta_{\omega}(\boldsymbol{y})^{*}$ 得到 $\tilde{k}(\boldsymbol{x},\boldsymbol{y})$ 的一个无偏估计，即 $\tilde{k}(\boldsymbol{x},\boldsymbol{y})\approx\sum_{m=1}^{M}\zeta_{\omega_m}(\boldsymbol{x})\zeta_{\omega_m}(\boldsymbol{y})^{*}$，注意到 $s(\boldsymbol{\omega})$ 和 $\tilde{k}(\boldsymbol{x},\boldsymbol{y})$ 都是实值函数，因此可以抛弃 $\zeta_{\omega}(\boldsymbol{x})$ 的虚部，用 $z_{\omega}(\boldsymbol{x})=[\cos(\boldsymbol{\omega}^{\mathrm{T}}\boldsymbol{x})\ \ \sin(\boldsymbol{\omega}^{\mathrm{T}}\boldsymbol{x})]^{\mathrm{T}}$ 来表示 $\tilde{k}(\boldsymbol{x},\boldsymbol{y})$，即 $\tilde{k}(\boldsymbol{x},\boldsymbol{y})=E_{\omega}(z_{\omega}(\boldsymbol{x})^{\mathrm{T}}z_{\omega}(\boldsymbol{x}))$。根据以上性质，文献 [22] 利用蒙特卡洛采样方法，建立了一组随机傅里叶特征映射

$$\boldsymbol{\phi}(\boldsymbol{x})=\frac{1}{\sqrt{M}}[\cos(\boldsymbol{\omega}_1^{\mathrm{T}}\boldsymbol{x})\ \sin(\boldsymbol{\omega}_1^{\mathrm{T}}\boldsymbol{x})\cdots\ \cos(\boldsymbol{\omega}_M^{\mathrm{T}}\boldsymbol{x})\ \sin(\boldsymbol{\omega}_M^{\mathrm{T}}\boldsymbol{x})]^{\mathrm{T}}$$

利用该特征映射可以将潜变量函数的高斯先验 $f(\boldsymbol{x})\sim\mathrm{GP}(0,k(\boldsymbol{x},\boldsymbol{x}'))$ 近似地表示为 $f(\boldsymbol{x})\sim\mathrm{GP}(0,\sigma^2\boldsymbol{\phi}(\boldsymbol{x})^{\mathrm{T}}\boldsymbol{\phi}(\boldsymbol{x}'))$，这将等价于一个线性高斯过程模型 $f(\boldsymbol{x})=\boldsymbol{w}^{\mathrm{T}}\boldsymbol{\phi}(\boldsymbol{x})$，$\boldsymbol{w}\sim N(0,\sigma^2\boldsymbol{I})$，从而极大地降低了原始模型的复杂度。考虑到随机采样方法需要的采样点数 M 通常较大，文献 [21] 将 $\boldsymbol{\omega}_m$ 作为模型参数建立了一种利用边缘似然对其进行优化选择的算法；文献 [23] 建立了一种 Fastfood 方法来通过随机矩阵近似生成 $\boldsymbol{\omega}_m$；文献 [24] 在 Fastfood 方法的基础上建立了一种核学习算法。文献 [25] 将协方差函数看作一个伪微分算子，利用希尔伯特空间逼近方法建立了如下形式的特征映射：

$$k(\boldsymbol{x},\boldsymbol{y})\approx\sum_{j=1}^{M}s(\sqrt{\lambda_j})\phi_j(\boldsymbol{x})\phi_j(\boldsymbol{y})=\boldsymbol{\phi}(\boldsymbol{x})^{\mathrm{T}}S\boldsymbol{\phi}(\boldsymbol{y}),$$
$$S=\mathrm{diag}(s(\sqrt{\lambda_1}),s(\sqrt{\lambda_2}),\cdots,s(\sqrt{\lambda_M}))$$

从而将原始高斯模型转换成了具有如下先验的线性高斯过程模型：

$$f(\boldsymbol{x})=\boldsymbol{w}^{\mathrm{T}}\boldsymbol{\phi}(\boldsymbol{x}),\ \boldsymbol{w}\sim N(0,\boldsymbol{S})$$

最近，Hensman 等 [26] 将变分方法与傅里叶特征映射进行结合，建立了一种变分傅里叶特征映射。

协方差矩阵逼近方法

该类方法的策略是直接利用低维（或者稀疏）矩阵近似地表示协方差矩阵 $\boldsymbol{K}_X$，从而降低高斯过程模型的复杂度。在有的实际应用问题中，训练样本的特征向量采自于一个笛卡尔网格，对于这样的问题，文献 [27] 发现，当采用张量积核作为高斯过程的协方差函数时，协方差矩阵可以写成 d 个 $n^{1/d}$ 阶方阵 $\{\boldsymbol{K}_i|1,2,\cdots,d\}$ 的克罗内克积的形式，即 $\boldsymbol{K}_X=\bigotimes_{i=1}^{d}\boldsymbol{K}_i$，其中 d 表示特征向量的维数。由于克罗内克积具有 $\boldsymbol{K}_X^{-1}=\bigotimes_{i=1}^{d}\boldsymbol{K}_i^{-1}$、$(\boldsymbol{K}_X+\sigma\boldsymbol{I})^{-1}=Q(\boldsymbol{\Lambda}+\sigma\boldsymbol{I})^{-1}\boldsymbol{Q}^{\mathrm{T}}$、$\boldsymbol{Q}=\bigotimes_{i=1}^{d}\boldsymbol{Q}_i$、$\boldsymbol{\Lambda}=\bigotimes_{i=1}^{d}\boldsymbol{\Lambda}_i$ 的性质，其中 $\boldsymbol{K}_X=\boldsymbol{Q}\boldsymbol{\Lambda}\boldsymbol{Q}^{\mathrm{T}}$ 和 $\boldsymbol{K}_i=\boldsymbol{Q}_i\boldsymbol{\Lambda}_i\boldsymbol{Q}_i^{\mathrm{T}}$ 分别是 $\boldsymbol{K}_X$ 和 $\boldsymbol{K}_i$ 的特征分解，所以根据以上性质，高斯回归模型的存储和计算复杂度可以分别降低为 $O(dn^{2/d})$ 和 $O(dn^{(d+1)/d})$。随后，文献 [28] 将以上克罗内克积表示方法推广到了高斯过程分类模型中，Wilson 等 [29] 放松了对训练数据必须采样于一个笛卡尔网格的要求，使算法可以处理只有一部分数据采样于笛卡尔网格的问题。考虑到训练数据采样于笛卡尔网格是一个比较严苛的条件，文献 [30] 借助于诱导变分方法的思想提出了一种矩阵插值的方法，先假设 $\boldsymbol{K}_X$ 可以近似地表示为 $\boldsymbol{K}_X\approx\boldsymbol{K}_{XU}\boldsymbol{K}_U^{-1}\boldsymbol{K}_{UX}$；然后将 $\boldsymbol{K}_{XU}$ 表示为一个稀疏的插值权重矩阵 $\boldsymbol{W}$ 和 $\boldsymbol{K}_U$ 的乘积形式 $\boldsymbol{K}_{XU}\approx\boldsymbol{W}\boldsymbol{K}_U$，利用该矩阵插值方法，高斯过程模型的存储和计算复杂度可以降低到 $O(n)$。

贝叶斯委员会机器

贝叶斯委员会机器 [31-32] 的基本思想是把训练集分割成不同的小数据集，在每个小数据集上训练一个学习器，然后利用贝叶斯公式把这些子学习器组合得到最终的学习器。令 $\{D_i|i=1,2,\cdots,l\}$ 表示对训练集 D 的一个分割，则利用贝叶斯委员会机器得到的 f_* 的后验概率分布的近似表达式为

$$p(f_*\mid D,x_*)\approx Z\frac{\prod_{i=1}^{l}p(f_*|D_i,x_*)}{\prod_{i=2}^{l}p(f_*|x_*)}$$

其中 Z 是常数。可以看出，贝叶斯委员会机器本质上是一种集成学习算法，它也可以用于高斯过程模型以外的其他各种机器学习模型。

结束语

本文对构建面向大规模数据的高斯过程模型的几种主要方法进行了简要介绍，这些方法各有优缺点。贝叶斯委员会机器虽然比较简单、直观，但是由于这种将大数据集拆分为小数据集的策略丧失了不同小数据集样本之间的关联关系，通常并不能取得好的结果，更适合于与其他方法配合使用。协方差函数分解和协方差矩阵逼近方法，虽然在有的应用问题上可以取得较好的结果，但是目前这两类方法都对协方差函数的类型有特定要求，不适合处理一般性的问题。诱导变量方法由于对协方差函数没有特定要求，而且可以同时解决诱导变量的选择和后验概率分布的逼近问题，相比较而言比其他方法更具一般性，因此也受到了人们更多的关注，目前已经建立了可以有效解决回归和分类问题的多

种诱导变量算法。由于高斯过程模型已经被用于排序学习、多视角学习等各种机器学习框架中，因此如何将诱导变量方法推广到这些机器学习框架是下一步需要解决的问题，此外建立面向流数据、类不平衡数据的诱导变量方法也是值得研究的方向。

基金项目：国家自然科学基金项目(61503058, 61374170)；辽宁省自然科学基金项目(201602190)；大连市青年科技之星项目(2016RQ072)。

参考文献

[1] Neal R M. Bayesian learning for neural networks [J]. Lecture Notes in Statistics. 1996, 118.
[2] Rasmussen C E. Evaluation of Gaussian processes and other methods for non-linear regression [D]. (PhD Thesis), Toronto: University of Toronto, 1996.
[3] Gibbs MN. Bayesian Gaussian processes for regression and classification [D]. (PhD Thesis),Cambridge: University of Cambridge, 1997.
[4] Bonilla E V, Chai K M, Williams C. Multi-task Gaussian process prediction[C]. Advances in neural information processing systems, 2008.
[5] Engel Y, Mannor S, Meir R. Reinforcement learning with Gaussian processes[C]. Proceedings of the 22nd international conference on Machine learning, 2005:201–208.
[6] Lawrence N, Jordan M. Semi-supervised learning via Gaussian processes[C]. Advances in Neural Information Processing Systems, 2005.
[7] Guiver J, Snelson E. Learning to rank with softrank and gaussian processes [C]. Proceedings of the 31st Annual International ACM SIGIR Conference On Research and Development in Information Retrieval, 2008.
[8] He J, Gu H, Wang Z. Bayesian multi-instance multi-label learning using Gaussian process prior[J]. Machine learning, 2012, 88(1-2): 273-295.
[9] Liu Q, Sun S. Multi-view Regularized Gaussian Processes[C]. Pacific-Asia Conference on Knowledge Discovery and Data Mining, 2017: 655-667.
[10] Zhou Y, He J, Gu H. Partial Label Learning via Gaussian Processes[J]. IEEE Transactions on Cybernetics, online, 2017.
[11] Seeger M, Williams C, Lawrence N. Fast forward selection to speed up sparse Gaussian process regression[C]. Ninth International Workshop on Artificial Intelligence and Statistics, 2003.
[12] Snelson E, Ghahramani Z. Sparse Gaussian processes using pseudo-inputs[C]. Advances in neural information processing systems, 2006: 1257-1264.
[13] Quiñonero-Candela J, Rasmussen C E. A unifying view of sparse approximate Gaussian process regression[J]. Journal of Machine Learning Research, 2005, 6: 1939-1959.
[14] Titsias M K. Variational learning of inducing variables in sparse Gaussian processes[C]. International Conference on Artificial Intelligence and Statistics, 2009: 567-574.
[15] Hensman J, Fusi N, Lawrence N D. Gaussian processes for Big data[C] Proceedings of the Twenty-Ninth Conference on Uncertainty in Artificial Intelligence, 2013: 282-290.
[16] Hensman J, Matthews A G, Ghahramani Z. Scalable variational Gaussian process classification[J]. Journal of Machine Learning Research, 2015, 38: 351-360.
[17] 马彪 , 贺建军 , 李厚杰 . 基于拉普拉斯方法的大规模高斯过程分类算法 [J]. 控制与决策 , 2017, 32(7): 1319-1324.
[18] Hernández-Lobato D, Hernández-Lobato J M. Scalable gaussian process classification via expectation propagation[C]. Proceedings of the 19th International Conference on Artificial Intelligence and Statistics, 2016: 168-176.

[19] Villacampa-Calvo C, Hernández-Lobato D. Scalable Multi-Class Gaussian Process Classification using Expectation Propagation[C]. International Conference on Machine Learning, 2017: 3550-3559.
[20] Bui T D, Yan J, Turner R E. A Unifying Framework for Sparse Gaussian Process Approximation using Power Expectation Propagation[J]. arXiv preprint arXiv:1605.07066, 2016.
[21] Lázaro-Gredilla M, Quiñonero-Candela J, Rasmussen C E, et al. Sparse Spectrum Gaussian Process Regression[J]. Journal of Machine Learning Research, 2010, 11: 1865-1881.
[22] Rahimi A, Recht B. Random features for large-scale kernel machines[C]. Advances in neural information processing systems, 2008: 1177-1184.
[23] Le Q, Sarlós T, Smola A. Fastfood-computing hilbert space expansions in loglinear time[C]. Proceedings of the 30th International Conference on Machine Learning , 2013: 244-252.
[24] Yang Z, Wilson A, Smola A, et al. A la carte-learning fast kernels[C]. International Conference on Artificial Intelligence and Statistics, 2015: 1098-1106.
[25] Solin A, Särkkä S. Hilbert space methods for reduced-rank Gaussian process regression[C]. Gaussian Process Approximations Workshop, 2015.
[26] Hensman J, Durrande N, Solin A. Variational Fourier features for Gaussian processes[J]. arXiv preprint arXiv:1611.06740, 2016.
[27] Saatçi Y. Scalable inference for structured Gaussian process models[D]. University of Cambridge, 2012.
[28] Flaxman S, Wilson A, Neill D, et al. Fast Kronecker inference in Gaussian processes with non-Gaussian likelihoods[C]. International Conference on Machine Learning, 2015: 607-616.
[29] Wilson A, Gilboa E, Cunningham J P, et al. Fast kernel learning for multidimensional pattern extrapolation[C]. Advances in Neural Information Processing Systems, 2014: 3626-3634.
[30] Wilson A, Nickisch H. Kernel interpolation for scalable structured Gaussian processes (KISS-GP)[C]. International Conference on Machine Learning, 2015: 1775-1784.
[31] Tresp V. A Bayesian committee machine[J]. Neural computation, 2000, 12(11): 2719-2741.
[32] Deisenroth M P, Ng J W. Distributed gaussian processes[C]. International Conference on Machine Learning, 2015: 1481-1490.

贺建军

博士，大连民族大学信息与通信工程学院副教授，硕士生导师。主要研究方向为机器学习与数据挖掘。

许小可

博士，大连民族大学信息与通信工程学院教授，硕士生导师。主要研究方向为社交网络大数据分析与机器学习。