地统计高斯过程与神经网络的融合方法综述

李锦韬

2201213292 lijintao@stu.pku.edu.cn

摘要

随着近年来深度学习技术的快速发展，许多研究尝试将神经网络模型引入地统计研究，然而地学领域对不确定度的衡量需求使得这一过程并不容易。本文在阅读大量有影响力的文献基础上，面向联合考察点参考数据属性值和空间分布的地统计任务特性，对高斯过程本身以及高斯过程和神经网络融合实现方法展开研究。本文详细梳理了高斯过程的基本原理和优缺点，调研目前已有的融合技术路线，并归纳整理为三种类型，对每种类型针对性分析。

关键词：地统计；高斯过程；神经网络；神经过程

1. 引言

地统计学（Geostatistics）是一门研究空间数据的统计学科，既考虑到样本值的大小，又重视样本空间位置及样本间的距离，弥补了经典统计学忽略空间方位的缺陷 [1, 2]。地统计学的研究对象通常为地质、地理、环境、气象等领域内的点参考数据（Point Referenced Data, PRD），主要任务是对点参考数据开展空间预测，其不仅关注插值的结果，还重视所插入值的不确定性 [3]，如图1所示。

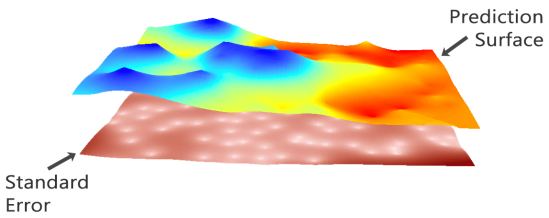


图 1. 地统计：兼顾插值预测与不确定估计 [4]

地统计学是高斯过程（Gaussian Processes, GP）最早应用的领域之一，其在 1960 年代便开始使用克里金（Kriging）方法，即高斯过程在回归任务上的实现 [5]。在实际应用中，如图2所示，地统计工作流使用基于高斯过程回归的各类克里格方法 [6, 7]，开展对点参考数据的建模，以生成未采样位置的预测，并提供对应不确定性的度量值，最终辅助决策与更进深入的探索研究。

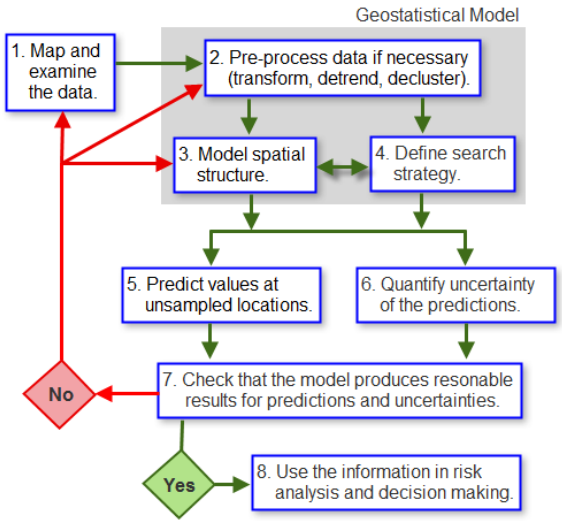


图 2. 地统计工作流 [8]

随着 2012 年以来的人工智能新浪潮崛起，深度学习技术的高速发展，以神经网络模型（Neural Network, NN）为代表的相关方法在计算机视觉、自然语言处理等领域取得了巨大的成功 [9]，其可以通过大量小批次的数据迭代训练，使用梯度下降进行参数调优，以高精度逼近有标签数据集，自适应特征提取效果好、可拓展性强 [10]。但是神经网络通常以黑箱形式存在，缺乏对不确定性的量化能力和可

解释性，并且作为一种参数量很大的模型，在普遍具有异质性的地学数据上迁移和泛化效果不佳，这两方面使得神经网络模型在地统计领域的应用受到限制 [11]。

高斯过程和神经网络各自具有优势，倘若能将两者结合起来，可为地统计领域带来新方法的机遇和新思路的拓展。

2. 高斯过程

高斯过程的相关研究最早可以追溯到统计学中的正态随机过程概念，是指由定义在时间或空间上的一组随机变量组成的随机过程，其中任意有限个随机变量的联合分布满足多元高斯分布 [12]。

在机器学习语境下，高斯过程实际上代表着基于正态随机过程、核方法和贝叶斯推断发展起来的机器学习方法。与传统监督学习中的参数化方法不同，高斯过程不局限于以模型参数选择为中心的建模思路，而是直接在函数空间上假设隐函数（Latent Function）满足先验的高斯分布，通过已知数据对隐函数的后验分布进行推断 [13, 14]，高斯过程建模的对象不是某个单一函数，而是函数的分布。

高斯过程根据预测目标类型可分为高斯过程回归和高斯过程分类两类。高斯过程是一种依赖于核（Kernel）的方法，高斯过程分类的实现效果没有支持向量机（SVM）理想，因而高斯过程回归的使用更加广泛 [15]。文献中的高斯过程也往往就是指代高斯过程回归，且地统计任务为回归问题。

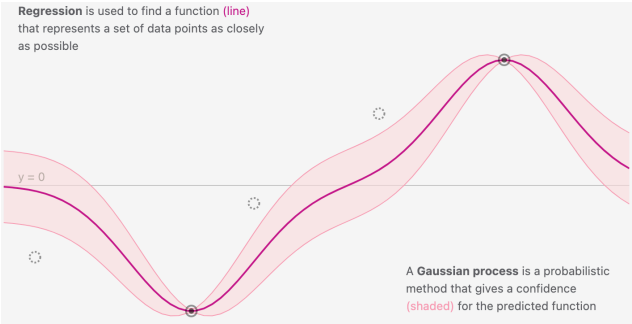


图 3. 高斯过程与回归方法不同：传统回归方法只能给出确定的预测值（图上深紫色线条），而高斯过程是一个概率模型，推断的是相应的分布（图上浅紫色阴影区），在给出预测均值同时估计预测值的不确定度 [16]。

2.1. 高斯过程基本原理

从函数空间（Function Space）的视角出发，高斯过程对函数分布建模，并直接在函数空间上进行贝叶斯推理。下面给出高斯过程（回归）的基本原理和关键步骤。

回归任务定义. 假设有训练集 $\mathcal{D} = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^n = (X, \boldsymbol{y})$，其中 $\boldsymbol{x}_i \in \mathcal{X}$ 是输入向量，构成输入矩阵 $X = [\boldsymbol{x_1}, \boldsymbol{x_2}, \cdots, \boldsymbol{x_n}]^\top$，$y_i \in \mathbb{R}$ 是对应的标签数值，$\boldsymbol{y} = [y_1, y_2, \cdots, y_n]^\top$。回归任务是根据训练集学习输入与输出之间的映射关系 $f(\boldsymbol{x})$，使得对于任意的新测试数据 $\boldsymbol{x}_* \in \mathcal{X}$，其回归预测值 $f(\boldsymbol{x}_*)$ 与真实值 y 的误差尽可能小。

高斯过程模型. 正态随机过程是任意有限个数量的具有联合高斯分布的随机变量的集合，其性质完全由均值函数和协方差函数决定：

$$\begin{cases} m(\boldsymbol{x}) = \mathbb{E}[f(\boldsymbol{x})] \\ k(\boldsymbol{x}, \boldsymbol{x}') = \mathbb{E}[(f(\boldsymbol{x}) - m(\boldsymbol{x}))(f(\boldsymbol{x}') - m(\boldsymbol{x}'))] \end{cases} \tag{1}$$

其中，$\boldsymbol{x}, \boldsymbol{x}' \in \mathcal{X}$ 为任意两个不同的输入向量，$m(\boldsymbol{x})$ 是均值函数，$k(\boldsymbol{x}, \boldsymbol{x}')$ 是协方差函数。$f(\boldsymbol{x})$ 是具体形式不确定或不固定的隐函数，用来表达与 $\boldsymbol{x}, \boldsymbol{x'}$ 对应的随机变量 $f(\boldsymbol{x}), f(\boldsymbol{x'})$，即所谓随机过程的截口随机变量 [12]。根据高斯过程定义，任意有限个隐函数 $f(\boldsymbol{x})$ 满足联合高斯分布 $f(\boldsymbol{x}) \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$。

高斯过程预测. 高斯过程预测的核心是通过训练数据集 $\mathcal{D}$ 对未知隐函数簇 $\boldsymbol{f}$ 的后验分布进行推断，之后根据训练样本与测试样本之间的相似性计算出测试样本的后验隐函数分布，这种相似性在地统计中被称为空间自相关性。即相近的事物更相似，正如地理学第一定律 [17]所概括的。

为使理论推导符号更加简洁，可假设数据均值为零（或进行预处理中心化），则隐函数簇 $\boldsymbol{f}$ 的先验分布为 $\boldsymbol{f} \sim \mathcal{N}(\boldsymbol{0}, K(X, X))$，其中 $K(X, X)$ 是协方差矩阵，$K(X, X)_{ij} = k(\boldsymbol{x_i}, \boldsymbol{x_j})$。根据任务定义，训练样本点构成的输入矩阵 X 对应的隐函数为 $\boldsymbol{f} = [f(\boldsymbol{x_1}), f(\boldsymbol{x_2}), \cdots, f(\boldsymbol{x_n})]^\top$，测试样

本点构成的输入矩阵 X_* 对应的隐函数为 $\boldsymbol{f_*} = [f(\boldsymbol{x_{*1}}), f(\boldsymbol{x_{*2}}), \cdots, f(\boldsymbol{x_{*m}})]^\top$。由多维高斯分布性质，$\boldsymbol{f}$ 和 $\boldsymbol{f_*}$ 的联合先验分布为：

$$\begin{bmatrix} \boldsymbol{f} \\ \boldsymbol{f_*} \end{bmatrix} \sim \mathcal{N}\left(\boldsymbol{0}, \begin{bmatrix} K(X,X) & K(X,X_*) \\ K(X_*,X) & K(X_*,X_*) \end{bmatrix}\right) \tag{2}$$

在实际的回归问题中，数据标签观测值 y 常常受到噪声污染，通常假设其也服从高斯分布，那么 $y = f(\boldsymbol{x}) + \epsilon$，$\epsilon \sim \mathcal{N}(0, \sigma_n^2)$，$\sigma_n^2$ 是噪声方差。由高斯性质，观测值向量 $\boldsymbol{y}$ 的先验分布为：

$$\boldsymbol{y} \sim \mathcal{N}(\boldsymbol{0}, K(X,X) + \sigma_n^2 \boldsymbol{I}) \tag{3}$$

其中，协方差矩阵 $K(X,X)_{ij} = k(\boldsymbol{x}_i, \boldsymbol{x}_j)$，$\boldsymbol{I}$ 为单位矩阵。继续应用高斯分布的性质，$\boldsymbol{f}$ 和 $\boldsymbol{f_*}$ 联合先验分布为：

$$\begin{bmatrix} \boldsymbol{f} \\ \boldsymbol{f}_* \end{bmatrix} \sim \mathcal{N}\left(\boldsymbol{0}, \begin{bmatrix} K(XX) + \sigma_n^2 \boldsymbol{I} & K(X,X_*) \\ K(X_*,\boldsymbol{X}) & K(X_*,X_*) \end{bmatrix}\right) \tag{4}$$

最后根据高斯联合分布下的条件分布计算公式（贝叶斯理论），测试样本隐函数 $\boldsymbol{f}_*$ 的后验分布为：

$$p(\boldsymbol{f}_*|X_*, X, \boldsymbol{y}) = \mathcal{N}(\bar{\boldsymbol{f}}_*, \boldsymbol{\Sigma}_*) \tag{5}$$

其中预测均值：

$$\bar{\boldsymbol{f}}_* = K(X_*,X)[K(X,X) + \sigma_n^2 \boldsymbol{I}]^{-1}\boldsymbol{y} \tag{6}$$

$\bar{\boldsymbol{f}}_*$ 可以被视为训练样本观测值向量 $\boldsymbol{y}$ 的线性组合，但其权重与测试输入 X_* 以及协方差结构有关。

预测方差：

$$\begin{aligned} \boldsymbol{\Sigma}_* = K(X_*,X_*) - \\ K(X_*,X)[K(X,X) + \sigma_n^2 \boldsymbol{I}]^{-1}K(X,X_*) \end{aligned} \tag{7}$$

$\boldsymbol{\Sigma}_*$ 与 $\boldsymbol{y}$ 无关，只依赖于输入 X 和 X_*，意味着 $\boldsymbol{f}_*$ 的方差仅和其与已观测点之间的相对位置有关，而和已观测点的函数值无关 [18]。

2.2. 高斯过程优缺点分析

高斯过程的数学定义优美，参数具有一定可解释性，是基于贝叶斯理论的核方法机器学习模型，而缺点和优点也是一体两面的。

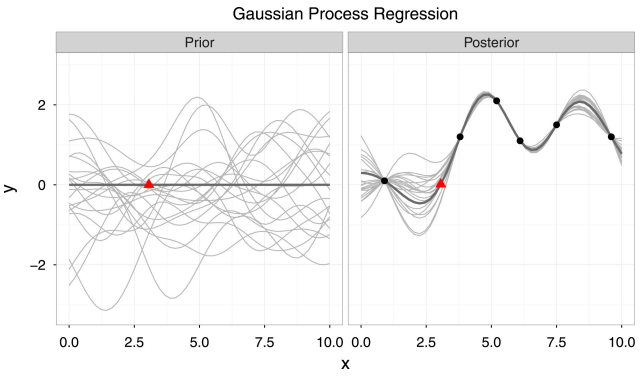


图 4. 一维高斯过程的先验和后验对比：左图是高斯过程先验，图上的若干灰色的曲线代表着可能的隐函数（事实上隐函数有无穷多个），深色线表示先验均值为 0。红色三角形为待测试点。右图为高斯过程后验，可见后验隐函数的分布发生了变化，相对拟合到训练数据（黑色点）上，且后验均值（深色线）通过所有训练数据点。[5]

高斯过程优点. 高斯过程相比于其他机器学习方法，尤其是和参数化建模方法相比，具有可适应于小样本量、非参数推断灵活以及输出具有概率意义等优点 [12, 19, 14, 20, 13, 21]。

- 对不确定性与预测结果同步输出，高斯过程是对隐函数分布建模的概率模型；
- 具有天然的贝叶斯正则化（或称遵循结构化风险最小化原则），高斯过程通过最大化边缘似然的方式自动确定合理的模型复杂度；
- 适应于小样本高度非线性问题，高斯过程不像神经网络模型对数据量依赖严重。

高斯过程缺点. 高斯过程与当今火热的深度学习相比，对大规模数据集的挖掘效率较差 [22, 23, 24, 25]。

- 高斯过程在某种意义上是理想化的，若数据本身不满足高斯分布假设，非高斯过程会更适合；
- 高斯过程在推断/预测时计算量很大，具有平方或立方级别的计算复杂度；
- 不同核函数的选择对结果的影响比较大，而核函数的选择依赖于先验知识。

在大数据量地统计中的应用难点. 高斯过程与神经网络的训练和预测不同，高斯过程是非参数模型，其参数数量是不固定的。根据公式(6)和公式(7)所示，

协方差矩阵内元素数量/参数数量与训练数据量相关。推断/预测时涉及大矩阵的求逆计算，不经过优化的普通高斯过程计算复杂度为 $\mathcal{O}(N^3)$，使其在地学大数据领域内的应用造成了困难。近年来许多研究聚焦在降低高斯过程计算复杂度上，开发出共轭梯度、降秩、矩阵稀疏计算、分区或局部计算以及近似算法等技术来加速 [26, 27, 22, 28, 29]，使计算复杂度逼近平方级别，但鲜有在工业领域应用的案例。

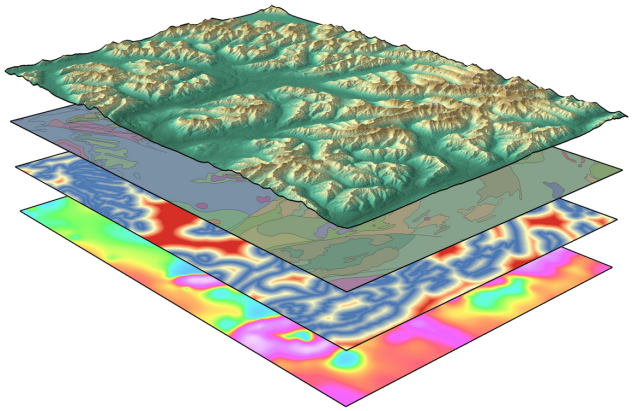


图 5. 大数据时代下地统计数据规模迅速扩大 [30]

3. 高斯过程与神经网络的结合

深度学习框架下的神经网络可视为一个直接训练的函数非线性逼近器，而高斯过程提供了一个学习非线性函数的分布的概率框架，双方优势互补。当数据量有限时，高斯过程因本身具备概率性质且可以描述不确定性而往往成为首选，但当面对大规模海量数据时，训练神经网络比高斯过程推断更容易且更具扩展性 [39]。因而，高斯过程和神经网络的结合是非常自然的想法。

本文对这一领域内的相关文献进行了梳理，参考 [40]中的归纳方式，将融合研究大致分为三种类型：（1）在高斯过程视角下加深对神经网络的理解，探讨神经网络的可解释性；（2）将神经网络与高斯过程组合，使用神经网络来学习核函数或协方差矩阵，也相当于以神经网络的预测优势来弥补高斯过程预测推断的高计算复杂度；（3）从元学习的理念出发，不再是用高斯过程的矩阵结构，直接使用神经网络来模拟高斯过程，此领域目前称为神经过程。更清晰的整理结果可见表1。

3.1. 高斯过程视角下的神经网络

该领域的开山之作源自 Neal 等在 1994 年对无限宽神经网络的先验分布的研究，其首次在理论上直接推导出单隐层无限宽神经网络等效于高斯过程 [31]。更进一步的，Williams 等在 1997 年计算出了单隐层神经网络的解析高斯过程核，并给出了使用高斯过程先验进行回归的精确贝叶斯推断方法 [41]。Hazan 等则在 2015 年进一步讨论了无限宽深度神经网络的等效核构建问题，但只限于两个非线性隐藏层 [42]。

最终，Lee 等在 2017 年直接论证分析了深度的无线宽神经网络等效于高斯过程，在 Neal 推导出了无限宽单隐层神经网络与高斯过程之间的等价性 [31]基础上，证明了“无限宽深度神经网络”与“高斯过程”之间的精确等价关系，并进一步开发了一个基于随机梯度下降算法（SGD）的高效计算管道，来计算高斯过程的协方差函数 [32]，作者将此过程称为神经网络高斯过程（Neural Network as Gaussian Process, NNGP）。Matthews 等在 2018 年也得到了和 Lee 等相似的结论，不同于前者使用的是 SGD 训练神经网络与高斯过程比较，Matthews 等使用的是基于马尔可夫链蒙特卡罗采样（MCMC）有限贝叶斯神经网络与高斯过程进行比较 [43]。

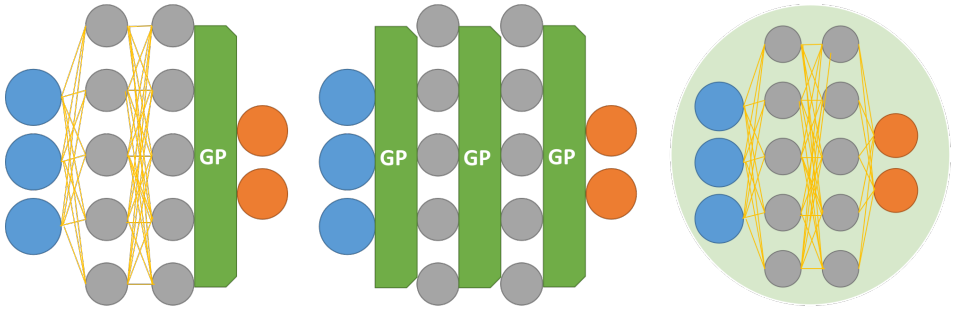
另外，Jacot 等在 2018 年剖析了神经网络训练期间的动态特性，并认为其训练动力学可以被视为一种神经正切核机制（Neural Tangent Kernel）[44]。Domingos 等在 2020 年提出了比神经正切核更进一步的路径核（Path Kernel）概念，认为所有通过梯度下降学得的模型，都可以被视为一种核机器 [33]。

3.2. 神经网络与高斯过程的组合

最早的尝试来源于 2013 年 Damianou 等对深度高斯过程（Deep Gaussian Process, DGP）的研究，提出使用多个等效于高斯过程的神经网络层堆叠形成一种新型的深度信念网络，其每个单层模型等效于标准高斯过程或含隐变量的高斯过程模型，创新地采用近似变分边缘化实现模型的推断 [34]。

表 1. 高斯过程与神经网络的结合方式

分类	主要观点	第一篇文献出现时间	代表文献
Explain NN with GP	(1) 无限宽深度神经网络可以等效于高斯过程 (2) 梯度下降训练得到的神经网络可视为核机器	1994	[31, 32, 33]
NN + GP (GP Kernel Identified by NN)	从神经网络的层结构或高斯过程的核函数切入: - 使用高斯过程作为神经网络中的一层 - 或视为使用神经网络构造高斯过程协方差矩阵	2013	[34, 35, 36]
NN is GP	从元学习的视角切入: - 使用神经网络实现类似高斯过程效果	2018	[37, 38]



(a) DKL[35] (b) DGP[45] (c) NNGP[32]

图 6. 高斯过程和深度学习组合的三种主要范例：(a) 在深度核学习 (DKL) 中，神经网的最后一层被高斯过程代替；(b) 深度高斯过程 (DGP) 更进一步，每一层都有一个高斯过程；(c) 在神经网络高斯过程 (NNGP) 中，高斯过程作为贝叶斯神经网络 (BNN) 序列的极限出现 [46]。

Wilson 等在 2015 年提出具有可扩展性的深度核学习模型 (Deep Kernel Learning, DKL)，其认为高斯过程与神经网络之间最大的不同在于基函数，神经网络只有有限的基函数 (参数数量固定)，而高斯过程通常使用无限多个固定的基函数 (例如谱分解后的特征函数)，因此提出了一种“前馈神经网络 (模拟非线性的特征映射函数) + 无线宽神经网络 (模拟高斯过程的无限个基函数)”构成的深度核学习神经网络，并给出了训练和推断的算法。该模型中的封闭形式的深度核可以直接替代标准核，核学习将高斯过程的边缘似然作为目标函数，n 个训练点的推断和学习成本为 $\mathcal{O}(n)$，每个测试点的预测成本为 $\mathcal{O}(1)$[35]。

Tibo 等在 2022 年提出归纳高斯过程网络 (Inducing Gaussian Process Networks, IGPN)，其在研究了 DKL 方法后，认为 DKL 模型在原始空间中选择稀疏归纳点不利于捕获特征空间中的交互，应当学习特征空间中的归纳点和深度核，将输入转换到特征空间，并在特征空间中设置归纳点，作为输入送入高斯过程，其优势在于能够同时学习特征空间中的归纳点和核参数 [36]。

3.3. 高斯过程的神经网络实现

深度神经网络擅长函数逼近，但通常针对每个新函数从头开始训练，而贝叶斯方法 (如高斯过程) 利用先验知识在测试时快速推断新的函数形状，但高斯过程的计算成本很高，而且很难设计出合适的先验。不同于第3.2节中的神经网络与高斯过程的组合模式，另一种想法是直接使用神经网络模拟高斯过程，某种程度上放弃了与高斯过程相关的数学保证，但好处是训练方便且能拓展到大型数据集上。

该领域来源自 DeepMind 团队在 2018 年的开创性研究，其提出了条件神经过程模型 (Conditional Neural Process, CNP)，意为其在给定一系列观察数据时能够定义函数的条件分布。该模型采用元学习的思想实现了深度学习灵活性和概率模型不确定性

的结合，模型可见图7，实现了端到端（end-to-end）的训练。但问题在于无法为相同的背景点生成不同的函数样本，缺少考虑目标之间的相关性（协方差），即缺少不确定性建模能力。[38]。

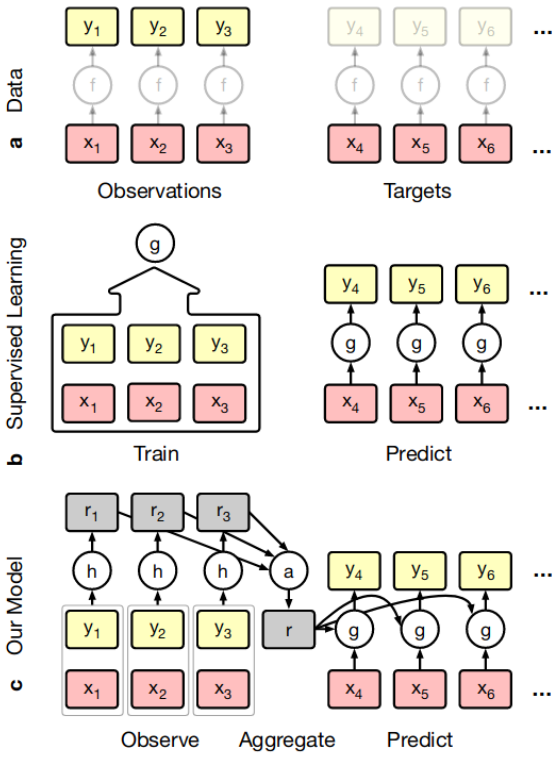


图 7. CNP 条件神经过程的说明 [38]：(a) f 代表从输入数据到标签的映射（可能是固定的、或是某个随机函数的一次实现）。(b) 传统监督学习模型：为每个新任务重新随机初始化参数化的函数 g，通过最小化近似函数 g 与 f 之间的损失，花费算力在大量数据上完成 g 对 f 的拟合。研究者可能拥有的有关 f 的先验信息，但一般是通过 g 的架构、损失函数或训练细节进行指定的。(c) CNP 模型：对观测的依赖性由神经网络进行参数化，该神经网络不受输入的排列顺序影响，之后生成每个观测值的编码（Embedding），然后将这些编码通过对称聚合器 a 聚合成 r，最终将其嵌入作为 g 的条件。

同年，DeepMind 为了提升不确定性建模能力，在 CNP 基础上增加了一个类似于 VAE 瓶颈的隐变量 z，其每一个随机样本都对应于随机过程的一个具体实现，见图8，这样就可以通过多个样本在解码器网络中的前向传递，生成目标处的预测分布，模型被命名为神经过程（Neural Process, NP）。此方法的问题在于单个预测输出虽然包含了不确定性（即测试点处的边缘分布），但不同点处的输出之间相互独立，无法对输出的相关性建模，这从某种程度上来说，失去了随机过程的优势 [37, 47]。

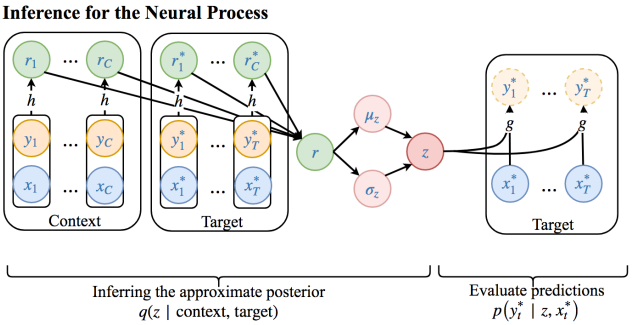


图 8. NP 神经过程 [47]

DeepMind 所提出的 CNP 和 NP 模型构成了神经过程家族（Neural Process Family, NPF）的雏形，后续有许多研究者将神经过程模型与当前主流热门的深度学习模型结合，逐渐使成员扩充 [48, 49]。

Bruinsma 等在 2021 年提出高斯神经过程，采用函数 KL 散度作为训练的代价函数，同时为解决输出相关性建模问题，引入了一个用于学习核函数的神经网络，并将其与神经过程网络的结合体称为高斯神经过程。[50] Markou 等在 2021 年提出高效的高斯神经过程回归，认为 Bruinsma 的高斯神经过程方法采用的 CNN 神经网络 [51]（本文作者称为为 FullConvGP）会限制输入的维度），因此提出了对原始高斯神经过程方法的改进，并将新模型称为卷积高斯神经过程（ConvGP）[52]。

DeepMind 团队在之前工作的基础上，于 2019 年提出的注意力神经过程（Attentive Neural Process），为了实现对输出相关性建模，首次在神经过程中引入注意力机制 [53]。Dutordoir 等在 2022 年提出神经扩散过程，首次将扩散模型引入神经过程 [54]。Nguyen 等在 2022 年提出 Transformer 神经过程，将自注意力机制实施的更进一步，直接使用自然语言处理领域中的强力架构 Transformer 来和神经过程搭配 [55]。

Bruinsma 等在 2023 年提出自回归条件神经过程，进一步为了提升相关性预测能力，但自回归条件神经过程并不对模型或训练过程进行任何修改，而

是像 MCDropout、神经自回归密度估计器 (NADE) 等一样，改变了 CNP 在测试阶段的部署方式，使用概率链式法则来自回归地定义联合预测分布，而不是对每个目标点独立进行预测。[56]

总的来说，神经过程方法融合高斯过程和神经网络的思想，具有相当好的拓展性，形成一系列神经过程家族。神经过程一定程度上实现了高斯过程所定义的函数上的分布，能够快速适应新的观测结果，并可以估计其预测中的不确定性；神经过程同时能够像神经网络一样，在训练和评估过程中计算效率很高，根据数据调整自身的先验，且在结构上具有较好的拓展性，能和最新的神经网络框架结合。神经过程更加适应当前的深度学习训练模式，但也损失了原有高斯过程中的部分数学性质。

4. 分析讨论

地理人工智能. 在地学领域内，部分学者将现在出现的各种地学人工智能方法概括称为 GeoAI (Geospatial Artificial Intelligence) [57]，并希望深度使用机器学习技术自动地挖掘地学领域内的知识，建立通用的处理框架，进而加速从数据到见解的过程。然而，由于地学数据来源极其广泛，数据类型和模态丰富多样，包括地理要素、关系型数据表格、栅格影像、时间序列、带有地理语义的文本等等 [58]，广义上而言任何带有地理位置意义的数据都是地学所关心的 [59]，意味着 GeoAI 的研究范围极其广泛，所涉及的领域极为多样。

对人类而言，人脑具备面向多尺度、多模式、多任务的地理空间认知能力，能够自然地把握空间异质性、空间依赖性、空间交互性和空间聚集性 [60]。但是对于计算机而言，这些能力要么都是需要人工设计的，要么得想方设法从数据或知识中学习得到，这也正是 GeoAI 研究的难点所在。短时间内，地学领域内很难独立地诞生统一的人工智能框架，地学相关学科必须和其他学科一道共同研究。

数据集与评价指标. 现阶段内，GeoAI 通常表现为面向特定任务目标或数据类型的应用型机器学习模型，例如遥感影像处理常常选择使用计算机视觉领域内的模型 [61]，其结果评价可采用与人工解译结果相对比的方式完成。但是，在地统计领域内的数据挖掘研究往往是案例式的（Case-by-Case），这是由点参考数据本身就是以环境指标类数据为主导致的，甚至在很多情况下就没有参考真值能够进行比对，例如 PM2.5 数据往往只有测站数据，缺少全局整个地理空间中的测量值，这意味着缺乏标准数据集和相应的评价指标，所使用和训练的模型的可解释性和可迁移性也相对较弱，部分仍然停留在工具化的使用层面 [62]。

本文所调研的文献中所使用的数据集和评价指标也是各不相同的，一方面是由于高斯过程和神经网络的结合本身是一个相当新和小众的研究领域，另一方面地学领域缺乏通用的标准数据集和评价指标，其构建仍然是一个有待于研究的问题。

融合研究的必要性. 在处理真实地理空间中的点参考数据时，经常会发现测量值受到不确定性和误差的影响，传统方法是使用高斯过程定义一个核函数来拟合数据，并为预测结果增加不确定性。地学研究者通过比较不同核函数在数据集上的效果，借由恰当地结合核函数或是为其选择参数来实现先验专家知识的嵌入 [63]。然而，在如今人类正以前所未有的速度获取着以往无法观测的数据，很多情况下我们往往无法拥有对应地统计领域的专家，或是获取先验知识的成本非常高昂。这使得从数据中自动地学习已然成为数据挖掘中的共识，

参考文献

[1] J.-P. Chiles and P. Delfiner, Geostatistics: modeling spatial uncertainty. John Wiley & Sons, 2012, vol. 713. 1

[2] M. Lu, “Spatial statistics 30: Origin of geostatistics origin.” [Online]. Available: https://blog.csdn.net/allenlu2008/article/details/103029221 1

[3] G. Matheron, “Principles of geostatistics,” Economic geology, vol. 58, no. 8, pp. 1246–1266, 1963. 1

[4] GISGeography. Kriging interpolation - the prediction is strong in this one. [Online]. Available: https://gisgeography.com/kriging-interpolation-prediction/ 1

[5] E. Schulz, M. Speekenbrink, and A. Krause, "A tutorial on gaussian process regression: Modelling, exploring, and exploiting functions," Journal of Mathematical Psychology, vol. 85, pp. 1–16, 2018. 1, 3

[6] M. A. Oliver, R. Webster et al., "Basic steps in geostatistics: the variogram and kriging," Springer, Tech. Rep., 2015. 1

[7] M. Oliver and R. Webster, "A tutorial guide to geostatistics: Computing and modelling variograms and kriging," Catena, vol. 113, pp. 56–69, 2014. 1

[8] ESRI, "Arcgis pro documentation: The geostatistical workflow." [Online]. Available: https://pro.arcgis.com/en/pro-app/latest/help/analysis/geostatistical-analyst/the-geostatistical-workflow.htm 1

[9] G. Hinton, Y. LeCun, and Y. Bengio, "Deep learning," Nature, vol. 521, no. 7553, pp. 436–444, 2015. 1

[10] Y. Bengio, I. Goodfellow, and A. Courville, Deep learning. MIT press Cambridge, MA, USA, 2017, vol. 1. 1

[11] J. S. Dramsch, "70 years of machine learning in geoscience in review," Advances in geophysics, vol. 61, pp. 1–55, 2020. 2

[12] G. Su, Machine Learning of Gaussian process and its Engineering Application. Science Press, Beijing, China, 2020. 2, 3, 10

[13] C. K. Williams and C. E. Rasmussen, Gaussian processes for machine learning. MIT press Cambridge, MA, 2006, vol. 2, no. 3. 2, 3, 10

[14] M. Seeger, "Gaussian processes for machine learning," International journal of neural systems, vol. 14, no. 02, pp. 69–106, 2004. 2, 3

[15] Z. he, "Overview of gaussian process regression," Control and Decision, vol. 28, no. 1121-1129+1137, 2013. 2

[16] J. Görtler, R. Kehlbeck, and O. Deussen, "A visual exploration of gaussian processes," Distill, vol. 4, no. 4, p. e17, 2019. 2, 11

[17] H. J. Miller, "Tobler's first law and spatial analysis," Annals of the association of American geographers, vol. 94, no. 2, pp. 284–289, 2004. 2

[18] K. P. Murphy, Machine learning: a probabilistic perspective. MIT press, 2012. 3

[19] D. J. MacKay et al., "Introduction to gaussian processes," NATO ASI series F computer and systems sciences, vol. 168, pp. 133–166, 1998. 3

[20] C. Williams and C. Rasmussen, "Gaussian processes for regression," Advances in neural information processing systems, vol. 8, 1995. 3

[21] G. Wang. Gaussian processes visualization and code implementation. [Online]. Available: https://borgwang.github.io/ml/2019/07/28/gaussian-processes.html 3, 11

[22] J. Hensman, N. Fusi, and N. D. Lawrence, "Gaussian processes for big data," arXiv preprint arXiv:1309.6835, 2013. 3, 4

[23] M. A. Alvarez and N. D. Lawrence, "Computationally efficient convolved multiple output gaussian processes," The Journal of Machine Learning Research, vol. 12, pp. 1459–1500, 2011. 3

[24] A. Davies, "Effective implementation of gaussian process regression for machine learning," Ph.D. dissertation, University of Cambridge, 2015. 3

[25] K. Cutajar, "Broadening the scope of gaussian processes for large-scale learning," Ph.D. dissertation, Sorbonne Université, 2019. [Online]. Available: https://theses.hal.science/tel-02968227 3

[26] N. Lawrence, M. Seeger, and R. Herbrich, "Fast sparse gaussian process methods: The informative vector machine," Advances in neural information processing systems, vol. 15, 2002. 4

[27] M. Titsias, "Variational learning of inducing variables in sparse gaussian processes," in Artificial intelligence and statistics. PMLR, 2009, pp. 567–574. 4

[28] E. Snelson and Z. Ghahramani, "Sparse gaussian processes using pseudo-inputs," Advances in neural information processing systems, vol. 18, 2005. 4

[29] M. Katzfuss and J. Guinness, "A general framework for vecchia approximations of gaussian processes," 2021. 4

[30] APEX Geoscience. Geostatistical services. [Online]. Available: https://www.apexgeoscience.com/service/geostatistical-services/ 4

[31] R. M. Neal and R. M. Neal, "Priors for infinite networks," Bayesian learning for neural networks, pp. 29–53, 1996. 4, 5

[32] J. Lee, Y. Bahri, R. Novak, S. S. Schoenholz, J. Pennington, and J. Sohl-Dickstein, "Deep neural networks as gaussian processes," arXiv preprint arXiv:1711.00165, 2017. 4, 5

[33] P. Domingos, "Every model learned by gradient descent is approximately a kernel machine," arXiv preprint arXiv:2012.00152, 2020. 4, 5

[34] A. Damianou and N. D. Lawrence, "Deep gaussian processes," in Artificial intelligence and statistics. PMLR, 2013, pp. 207–215. 4, 5

[35] A. G. Wilson, Z. Hu, R. Salakhutdinov, and E. P. Xing, "Deep kernel learning," in Artificial intelligence and statistics. PMLR, 2016, pp. 370–378. 5

[36] A. Tibo and T. D. Nielsen, "Inducing gaussian process networks," arXiv preprint arXiv:2204.09889, 2022. 5

[37] M. Garnelo, J. Schwarz, D. Rosenbaum, F. Viola, D. J. Rezende, S. Eslami, and Y. W. Teh, "Neural processes," arXiv preprint arXiv:1807.01622, 2018. 5, 6

[38] M. Garnelo, D. Rosenbaum, C. Maddison, T. Ramalho, D. Saxton, M. Shanahan, Y. W. Teh, D. Rezende, and S. A. Eslami, "Conditional neural processes," in International conference on machine learning. PMLR, 2018, pp. 1704–1713. 5, 6

[39] K. Märtens. Neural processes as distributions over functions. [Online]. Available: https://kasparmartens.rbind.io/post/np/ 4

[40] G. Pu. Neural network and gaussian process. [Online]. Available: https://xishansnow.github.io/posts/52ce42db.html 4

[41] C. Williams, "Computing with infinite networks," Advances in neural information processing systems, vol. 9, 1996. 4

[42] T. Hazan and T. Jaakkola, "Steps toward deep kernel methods from infinite neural networks," arXiv preprint arXiv:1508.05133, 2015. 4

[43] A. G. d. G. Matthews, M. Rowland, J. Hron, R. E. Turner, and Z. Ghahramani, "Gaussian process behaviour in wide deep neural networks," arXiv preprint arXiv:1804.11271, 2018. 4

[44] A. Jacot, F. Gabriel, and C. Hongler, "Neural tangent kernel: Convergence and generalization in neural networks," Advances in neural information processing systems, vol. 31, 2018. 4

[45] M. Havasi, J. M. Hernández-Lobato, and J. J. Murillo-Fuentes, "Inference in deep gaussian processes using stochastic gradient hamiltonian monte carlo," Advances in neural information processing systems, vol. 31, 2018. 5

[46] G. Pu. Review on gauss processes. [Online]. Available: https://xishansnow.github.io/posts/b5b2c876.html 5

[47] K. Märtens. Neural processes as distributions over functions. [Online]. Available: https://kasparmartens.rbind.io/post/np/ 6

[48] Y. Dubois, J. Gordon, and A. Y. Foong, "Neural process family," http://yanndubs.github.io/Neural-Process-Family/, September 2020. 6

[49] S. Jha, D. Gong, X. Wang, R. E. Turner, and L. Yao, "The neural process family: Survey, applications and perspectives," 2022. 6

[50] W. P. Bruinsma, J. Requeima, A. Y. Foong, J. Gordon, and R. E. Turner, "The gaussian neural process," arXiv preprint arXiv:2101.03606, 2021. 6

[51] J. Gordon, W. P. Bruinsma, A. Y. Foong, J. Requeima, Y. Dubois, and R. E. Turner, "Convolutional conditional neural processes," arXiv preprint arXiv:1910.13556, 2019. 6

[52] S. Markou, J. Requeima, W. Bruinsma, and R. Turner, "Efficient gaussian neural processes for regression," arXiv preprint arXiv:2108.09676, 2021. 6

[53] H. Kim, A. Mnih, J. Schwarz, M. Garnelo, A. Eslami, D. Rosenbaum, O. Vinyals, and Y. W. Teh, "Attentive neural processes," arXiv preprint arXiv:1901.05761, 2019. 6

[54] V. Dutordoir, A. Saul, Z. Ghahramani, and F. Simpson, "Neural diffusion processes," arXiv preprint arXiv:2206.03992, 2022. 6

[55] T. Nguyen and A. Grover, "Transformer neural processes: Uncertainty-aware meta learning via sequence modeling," arXiv preprint arXiv:2207.04179, 2022. 6

[56] W. P. Bruinsma, S. Markou, J. Requiema, A. Y. Foong, T. R. Andersson, A. Vaughan, A. Buonomo, J. S. Hosking, and R. E. Turner, "Autoregressive conditional neural processes," arXiv preprint arXiv:2303.14468, 2023. 7

[57] K. Janowicz, S. Gao, G. McKenzie, Y. Hu, and B. Bhaduri, "Geoai: spatially explicit artificial intelligence techniques for geographic knowledge discovery and beyond," International Journal of Geographical Information Science, vol. 34, no. 4, p. 625–636, Apr 2020. 7

[58] M. Graham and T. Shelton, "Geography and the future of big data, big data and the future of geography," Dialogues in Human geography, vol. 3, no. 3, pp. 255–261, 2013. 7

[59] J.-G. Lee and M. Kang, "Geospatial big data: challenges and opportunities," Big Data Research, vol. 2, no. 2, pp. 74–81, 2015. 7

[60] Y. Liu, H. Guo, H. Li, W. Dong, and T. Fei, "A note on geoai from the perspective of geographical laws," Acta Geodaetica et Cartographica Sinica, vol. 51, no. 6, p. 1062–1069, 2022. 7

[61] L. Zhang and L. Zhang, "Artificial intelligence for remote sensing data analysis: A review of challenges and opportunities," IEEE Geoscience and Remote Sensing Magazine, vol. 10, no. 2, pp. 270–294, 2022. 7

[62] S. Gao, "A review of recent researches and reflections on geospatial artificial intelligence," Geomatics and Information Science of Wuhan University, vol. 45, no. 12, p. 1865–1874, 2020. 7

[63] A. Pretorius, H. Kamper, and S. Kroon, "On the expected behaviour of noise regularised deep neural networks as gaussian processes," Pattern Recognition Letters, vol. 138, pp. 75–81, 2020. 7

[64] J. Shawe-Taylor, N. Cristianini et al., Kernel methods for pattern analysis. Cambridge university press, 2004. 10

[65] D. Duvenaud, "Automatic model construction with gaussian processes," Ph.D. dissertation, University of Cambridge, 2014. 10

[66] N. Cressie and H.-C. Huang, "Classes of nonseparable, spatio-temporal stationary covariance functions," Journal of the American Statistical association, vol. 94, no. 448, pp. 1330–1339, 1999. 10

[67] D. L. Zimmerman and M. B. Zimmerman, "A comparison of spatial semivariogram estimators and corresponding ordinary kriging predictors," Technometrics, vol. 33, no. 1, pp. 77–91, 1991. 10

[68] R. Frigola, F. Lindsten, T. B. Schön, and C. E. Rasmussen, "Bayesian inference and learning in gaussian process state-space models with particle mcmc," Advances in neural information processing systems, vol. 26, 2013. 10

[69] D. Duvenaud. Kernel cookbook: Advice on covariance functions. [Online]. Available: https://www.cs.toronto.edu/~duvenaud/cookbook/ 10

[70] B. Minasny and A. B. McBratney, "Spatial prediction of soil properties using eblup with the matérn covariance function," Geoderma, vol. 140, no. 4, pp. 324–336, 2007. 11

[71] J. N. Hendriks, C. Jidling, A. Wills, and T. B. Schön, "Evaluating the squared-exponential covariance function in gaussian processes with integral observations," arXiv preprint arXiv:1812.07319, 2018. 11

[72] S. Kamperis. An introduction to gaussian processes regression analysis. [Online]. Available: https://ekamperi.github.io/mathematics/2021/03/30/gaussian-process-regression 11

A. 附录：高斯过程训练

高斯过程中的协方差函数需要根据对数据的先验认识来选择，其反映了数据间的空间依赖关系（相关性），是高斯过程建模的核心。协方差函数需要满足对称半正定条件，与 Mercer 核函数条件一致 [64]，文献中常常直接将协方差函数和核函数等同。核函数的构造与超参数优化是地统计与高斯过程领域早期的主要的内容 [13, 12, 65, 66]。

高斯过程训练主要是确定核函数中的超参数，如径向基函数中的长度尺度参数 l 和方差参数 σ_f^2。最优化超参数的方法有很多，例如：

- 矩量法（在地统计领域克里金插值流程中称为半变异函数法 Variogram[67]）：计算所有样本点之间的距离和相关性，然后拟合一条与之最符合的曲线。
- 最大似然法 [13]：通过最大化训练数据的似然函数来确定超参数 $\boldsymbol{\theta}^* = \arg\max_\theta \log p(\boldsymbol{y}|X,\theta)$。
- 贝叶斯推断 [68]：通过最大化后验概率 $p(\theta|\boldsymbol{y},X)$ 来确定超参数 $\boldsymbol{\theta}^* = \arg\max_\theta p(\theta|\boldsymbol{y},X)$。

B. 附录：常用的核函数

优秀的核函数可以最大程度上挖掘训练样本和测试样本的相似特征，使得相似输入具有相似输出 [69]，且核函数间可组合。

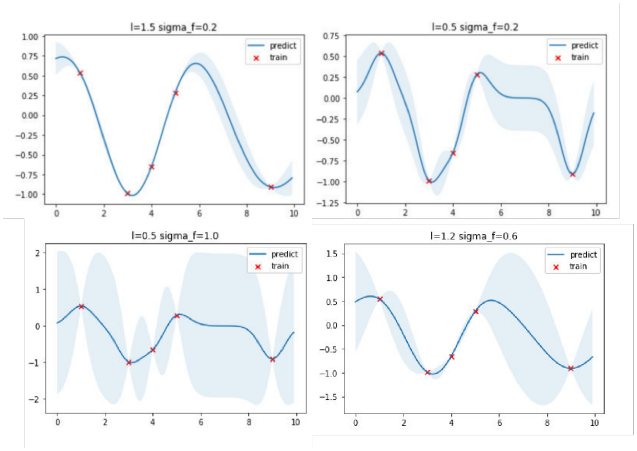
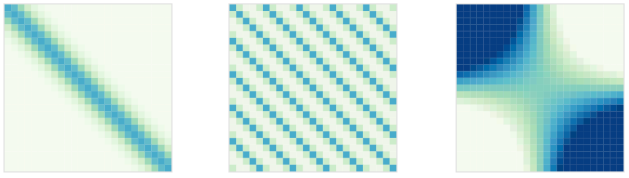


图 9. 不同的 REF 核函数超参数对推断的影响 [21]

Matérn 协方差函数或称马特恩协方差函数 [70]，形式为：

$$k(\boldsymbol{x},\boldsymbol{x}') = \sigma_f^2 \frac{2^{1-\nu}}{\Gamma(\nu)} \left(\frac{\sqrt{2\nu}}{l}||\boldsymbol{x}-\boldsymbol{x}'||\right)^{\nu} K_\nu \left(\frac{\sqrt{2\nu}}{l}||\boldsymbol{x}-\boldsymbol{x}'||\right) \tag{8}$$

其中 l 为长度尺度参数，σ_f^2 为方差参数，ν 为平滑度参数，$||\boldsymbol{x}-\boldsymbol{x}'||$ 为欧式距离，K_ν 为第二类修正 Bessel 函数。



(a)SE 核函数　(b)PER 核函数　(c)LIN 核函数

图 10. 不同核函数构成的协方差矩阵可视化对比 [16]

平方指数核函数（SE）或称径向基函数（RBF）[71]，是最常用的核函数之一，其形式为：

$$k(\boldsymbol{x},\boldsymbol{x}') = \sigma_f^2 \exp\left(-\frac{1}{2l^2}||\boldsymbol{x}-\boldsymbol{x}'||^2\right) \tag{9}$$

其中 l 为长度尺度参数，σ_f^2 为方差参数，$||\boldsymbol{x}-\boldsymbol{x}'||^2$ 为欧式距离。

周期核函数（PER）[71]，描述了每个观测不仅与相似的观测相关，也与更远的观测呈现周期性的相关性，其形式为：

$$k(\boldsymbol{x},\boldsymbol{x}') = \sigma_f^2 \exp\left(-\frac{2\sin^2(\pi||\boldsymbol{x}-\boldsymbol{x}'||/p)}{l^2}\right) \tag{10}$$

其中 l 为长度尺度参数，σ_f^2 为方差参数，p 为周期参数，$||\boldsymbol{x}-\boldsymbol{x}'||$ 为欧式距离。

线性核函数（LIN）[72]，形式为：

$$k(\boldsymbol{x},\boldsymbol{x}') = \sigma_b^2 + \sigma_f^2 \boldsymbol{x}^T \boldsymbol{x}' \tag{11}$$

其中 σ_f^2 为方差参数，σ_b^2 为偏置参数，$\boldsymbol{x}^T\boldsymbol{x}'$ 为内积。