利用遗传编程进行随机微分方程的符号发现

Sigur de Vries[1*]、Sander W. Keemink[1], 和 Marcel A. J. van Gerven[1]

[1]机器学习与神经计算系，唐德斯脑、认知与行为研究所，拉德堡德大学，荷兰奈梅亨[*]
sigur.devries@donders.ru.nl

arXiv:2603.09597v1 [cs.NE] 10 Mar 2026

摘要

自动化科学发现旨在通过机器学习提升科学理解。该领域的一种核心方法是符号回归，它利用遗传编程或稀疏回归，学习可解释的数学表达式以解释观测数据。传统上，符号回归主要用于识别常微分方程。普遍观点认为，噪声只会增加恢复确定性动力学的难度。然而，在随机微分方程中显式学习噪声分量的符号函数，可以增强建模能力、增加知识获取，并实现生成式采样。我们提出一种基于遗传编程的随机微分方程符号发现方法，通过最大似然估计联合优化漂移函数和扩散函数。结果表明，该方法能够准确恢复控制方程、高效扩展至高维系统、稳健处理稀疏采样问题，并推广至随机偏微分方程。本研究将符号回归拓展至随机动力系统的可解释发现，为在充满噪声且不断变化的世界中实现科学自动化作出贡献。

1 引言

自动化科学发现（ASD）旨在为科学研究开发并部署机器学习 [1,2]。换言之，ASD 专注于通过机器学习，从观测数据中获取有关未知系统或过程的知识。通常，ASD 包括以数据驱动的方式生成并验证假设，与人类研究相比，能大幅加快这一过程。ASD 已在材料科学 [3], 、药物发现 [4], 、化学 [5]等诸多领域取得成功。ASD 的一个重要方向是系统辨识，即建立数学模型，从数据中学习系统的内在机制。这些数学模型通常完全是确定性的，因此在存在噪声的情况下会受到限制；噪声可能是系统动态的一部分（偶然不确定性），也可能源于观测不完整（认知不确定性）。在数学模型中刻画噪声正受到越来越多的关注，因为这有助于生成式建模和不确定性量化。

系统辨识的一种方法是符号回归，它通过发现符号结构来解释变量之间的关系 [6, 7]。与神经网络等传统黑盒优化方法相比，符号回归能够生成可解释的模型，这些模型不仅能拟合数据，还能提供有关系统内在机制的信息。符号回归主要通过遗传编程（GP） [8, 9, 10]或稀疏回归 [11], 实现，不过一些不那么传统的算法也开始展现出有竞争力的性能 [12, 13, 14, 15]。遗传编程以进化原理为灵感，对以解析树表示的计算机程序的结构进行优化 [8], 。这使得人们能够直接学习数学方程的参数和结构，从而减少人为偏见并增强可解释性。

符号回归的大多数应用都专注于从数据中发现支配定律或动力系统 [10,7, 9, 11, 16]。在动力系统领域，学习的主要是常微分方程（ODE），尽管近年来偏微分方程（PDE）也日益受到关注 [17, 18, 19]。另一类动力系统是随机微分方程（SDE）（图1a）。一个随机微分方程由

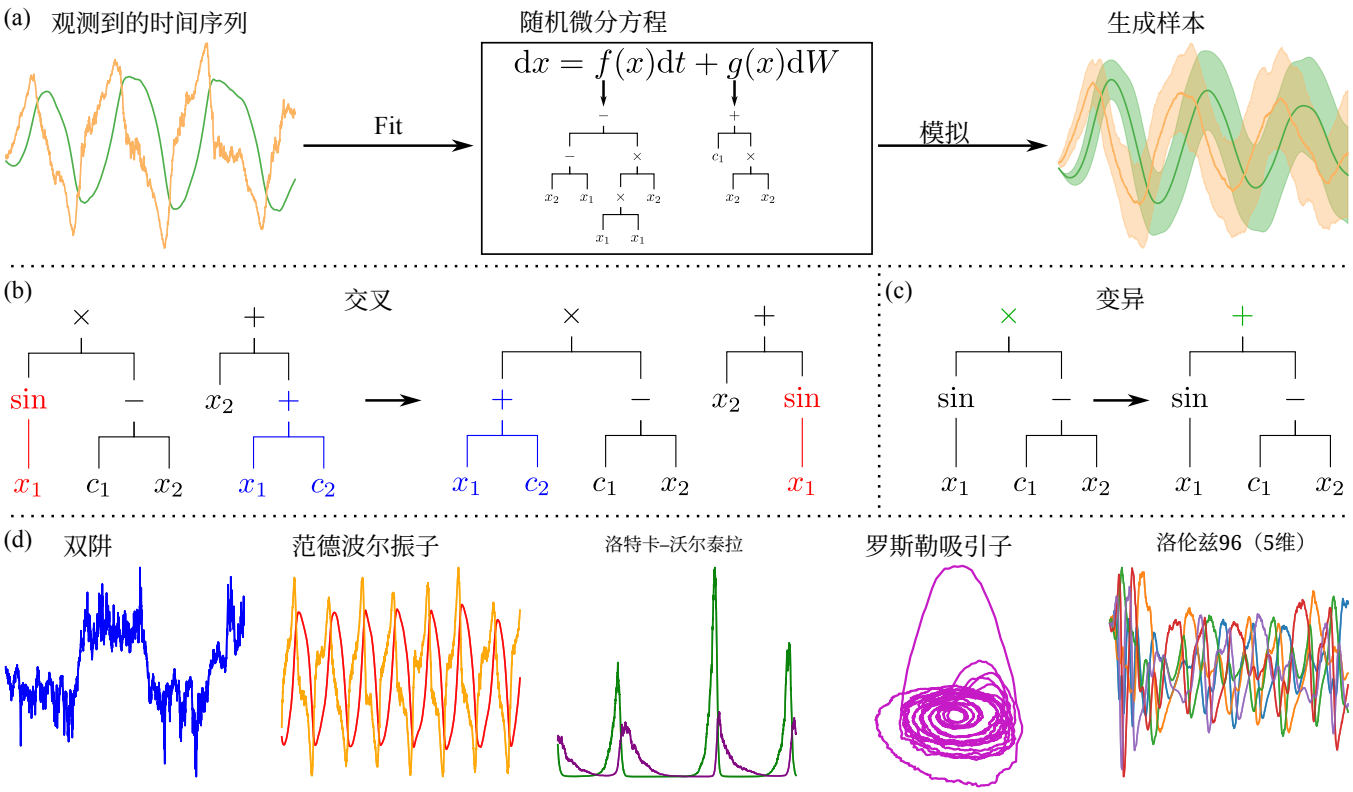


图1：**随机微分方程遗传编程概述**。（a）观测随机时间序列，并用符号表达式拟合随机微分方程。由此得到的随机微分方程具有可解释性，并可用于生成式建模。漂移 $f(x)$和扩散 $g(x)$的符号方程通过遗传编程进行优化。（b）在遗传编程中，通过交叉来调整树，即交换子树；图中展示了红色和蓝色子树的交换。（c）除了交叉，变异也会改变单棵树的结构。可以对多个方面进行变异，此例展示了绿色运算符的变化。（d）所提出的方法可应用于的系统示例。每个子图中的曲线展示了状态遵循随机系统动力学时的行为。

确定性函数和额外的噪声过程构成，这些噪声过程会随机扰动系统动力学。在现实应用中使用随机微分方程可以实现生成式建模，并获得更强的近似能力 [20, 21, 22]。与常微分方程和偏微分方程相比，随机微分方程的符号结构学习仍鲜有探索，且尚未通过遗传编程进行研究。在识别确定性控制方程时，系统中的噪声通常被视为一项额外挑战，用于衡量方法的鲁棒性 [11, 12], ，但很少有人学习支配噪声的项。

目前，符号回归在随机微分方程中的应用仅采用稀疏回归实现，展示了对含高斯噪声 [23, 24, 25], L´évy 跳跃 [26, 27], 以及随机偏微分方程（ SPDEs ） [28] 的系统正确结构的识别。通过捕捉噪声项，既能提高对动力学确定性部分的学习准确率，也能为动力学的随机分量发现一种可解释函数；与拟合随机微分方程的黑盒方法相比，这一做法更具优势 [1, 21, 29]。将稀疏回归应用于随机微分方程时，克拉默斯-莫亚尔展开 [30] 采用分箱方法，分别估计动力学确定性分量和随机（扩散）分量的系数。这会带来若干局限，例如漂移和扩散无法同时优化，展开和回归过程中都可能出现误差，而且该方法容易受到维数灾难的影响 [31]。

或者，也可以扩展遗传编程，使其学习随机微分方程的结构。已有研究表明，对于结构固定的（非线性）随机微分方程，可以通过直接优化最大似然估计（MLE） [32, 33, 34, 35]来拟合参数。因此，MLE 也可作为遗传编程中的合适目标函数，用于学习随机微分方程的符号表达式，并通过多树表示联合演化漂移和扩散，同时摆脱对数据分箱和系数估计的依赖。

本文旨在将遗传编程应用于随机微分方程的符号发现任务（图 1b+c），并展示这一方法在一系列系统辨识问题中的应用（图 1d）。结果表明，该方法能够有效学习随机分量，与仅演化常微分方程相比，还能提升漂移恢复效果。在低维问题上，该方法的表现可与基于克拉默斯–莫亚尔展开的稀疏回归相媲美，同时在扩展至高维系统时展现出更好的可扩展性。此外，结合积分能够从稀疏采样数据中更好地恢复方程。最后，我们表明，该方法还可以利用最大似然估计识别随机偏微分方程的结构。该框架发现的方程不仅揭示了底层系统的知识，还能生成生成样本。总之，所提出的方法为随机系统辨识提供了一种可扩展且有效的替代方案，这是科学发现中的一个重要方向。

2 相关工作

自从符号回归被提出作为遗传编程 [8], 的一种可能应用以来，它在系统辨识领域 [6, 36]受到了广泛关注。多年来，人们开发了许多符号回归工具，这些工具或基于遗传编程 [9, 10, 37], 、稀疏回归 [9] ，或基于物理信息方法 [7], ，并提出了若干基准 [7, 12, 38]。符号回归是一种广受欢迎的方法，可用于发现描述物理定律和自然定律的符号方程，包括基因调控网络 [39], 、能量消耗 [40] 或化学工程 [41]。

常微分方程构成了一类重要的系统，符号回归已广泛应用于这类系统 [9, 11, 12, 16,42, 43]。通过观测到的时间序列数据，可以推断出控制常微分方程的符号表达式，从而得到对底层动态过程具有可解释性的模型。传统上，候选方程通过对常微分方程系统进行数值积分，并将所得轨迹与数据进行比较来评估，这一过程的计算成本可能很高。有限差分法无需对常微分方程系统进行数值积分，而是将评估转化为回归任务，针对每个变量分别将系统状态映射到其导数。这大幅降低了计算成本，也简化了问题。SINDy及其扩展方法基于有限差分法 [11], ，该方法也被用于遗传编程中的适应度评估 [9, 37]。

自SINDy证明了利用稀疏回归对动力系统进行符号发现的可能性以来，它已被扩展到多种应用，包括偏微分方程 [17, 18, 19], 部分观测系统 [44,45], 随机微分方程 [23, 24, 25, 26, 27, 46, 47, 48, 49] 以及随机偏微分方程 [28]。当前所有用于随机微分方程符号回归的方法都将有限差分法替换为克拉默斯-莫亚尔展开 [30], ，从而在动力学的每个时间步估计漂移与扩散的系数。传统方法在计算系数之前对数据进行分箱；然而，使用原始时间序列会在低采样频率下引入偏差—方差权衡，并增加计算成本 [49]。在进行克拉默斯-莫亚尔展开之后，漂移与扩散的辨识可被视为相互独立的回归问题，对此SINDy是一个强有力的候选方法。尽管稀疏回归与克拉默斯-莫亚尔展开的结合行之有效，但它仍有若干局限。漂移与扩散系数的准确估计需要高频采样以及每个分箱内有足够的数据，这使得该方法极易受维数灾难影响 [31]。其次，该组合作为一种两阶段方法，意味着辨识过程中的多个阶段都可能出现不精确。稀疏回归的一个普遍局限是过度依赖函数库中选定的项，当所需基函数缺失时会导致错误的恢复。

随机微分方程的符号回归是一个相对鲜有人涉足的领域，但近年来，人们已开发出黑盒方法，利用离散观测拟合随机微分方程模型。例如，神经随机微分方程或潜变量随机微分方程会将参数化的漂移函数和扩散函数拟合到观测数据，并通过变分推断生成新的样本轨迹 [20, 21, 29]。另一方面，物理信息神经网络也展现出学习随机过程的能力 [50, 51, 52]。神经随机微分方程和物理信息神经网络都会产生可解释性较弱的黑盒模型，而遗传编程可以为这类问题演化出符号表达式。当随机微分方程的结构固定时，可以通过贝叶斯推断 [53, 54] 或最小化最大似然估计 [32, 34, 35], 来进行参数估计，甚至还有无需参数的方法 [33]。然而，这些方法依赖于预定义的模型结构，因此会引入偏差；与通过符号回归搜索最佳方程相比，它们的建模能力也会降低。此外，当观测间隔显著增大时，可以通过在观测点之间对提出的解进行多步积分，更准确地优化最大似然估计 [34, 55], ，从而提升稀疏采样数据上的性能。

遗传编程（GP）尚未被直接用于学习随机微分方程的结构，更不用说随机偏微分方程了。不过，参考文献 [27] 使用遗传编程构建了用于稀疏回归的基函数库。通过遗传编程进行符号回归，使该方法不那么依赖

给定的函数库，而稀疏回归通常更依赖它。分别学习漂移和扩散并不能保证模型一致或有效，而使用遗传编程优化最大似然估计，则能够针对每个变量对二者进行联合评估，从而得到更协调的解。此外，将最大似然估计作为目标函数，可摆脱对结合分箱的克拉默斯-莫亚尔展开的依赖，使该方法更易扩展到更高维的问题。

3 方法

3.1 随机微分方程

随机微分方程通过刻画动力系统中的随机性来扩展常微分方程。本文考虑随机微分方程的一般时不变形式，定义为

$$\mathrm{d}\mathbf{x}(t) = f(\mathbf{x}(t))\mathrm{d}t + G(\mathbf{x}(t))\mathrm{d}\mathbf{W}. \tag{1}$$

此处，$\mathbf{x}(t) \in \mathbb{R}^N$支配 N维状态，$f(\mathbf{x}(t)) \in \mathbb{R}^N$与 $G(\mathbf{x}(t)) \in \mathbb{R}^{N\times N}$分别表示漂移和扩散，且$\mathbf{W} \in \mathbb{R}^N$ 表示标准维纳过程。当 $G(\mathbf{x}(t))$依赖于$\mathbf{x}$时，噪声称为乘性，而常数 $G(\mathbf{x}(t))$称为加性。我们假设系统中每个变量均由独立噪声驱动，因此可将动力学显式分解为

$$f(\mathbf{x}(t)) = \begin{bmatrix} f_1(\mathbf{x}(t)) \\ \vdots \\ f_N(\mathbf{x}(t)) \end{bmatrix}, \quad G(\mathbf{x}(t)) = \mathrm{diag}(g_1(\mathbf{x}(t)), \ldots, g_N(\mathbf{x}(t))), \tag{2}$$

其中，$f_i(\mathbf{x}(t))$与 $g_i(\mathbf{x}(t))$对于 $i = 1, \ldots\ldots, N$均输出标量。

将此框架扩展到随机偏微分方程（随机偏微分方程），即将有限维状态$\mathbf{x}(t)$替换为空间依赖场 $u(t, \mathbf{x})$，其定义于空间域$\Omega \subset \mathbb{R}^d$上。具体而言，该随机偏微分方程可写为

$$\mathrm{d}u(t, \mathbf{x}) = f(u(t, \mathbf{x}))\mathrm{d}t + g(u(t, \mathbf{x}))\mathrm{d}W(t, \mathbf{x}), \quad \mathbf{x} \in \Omega. \tag{3}$$

此处，f 和 g 的定义方式与式 (1) 相同，而$\mathbf{x}$ 为空间坐标。为了对 SPDE 进行数值模拟，将$\mathbf{x}$ 离散为等间距点，并且必须定义初始条件 $u(0,\mathbf{x})$ 和边界条件 $u(t, 0)$。

在实际应用中，时间序列来自动力学未知的系统观测，但为了衡量所学模型的准确率，我们将关注具有已知真实值的模拟系统。需要注意的是，我们假设系统具有完全可观测性，因此数据的所有维度均可用，且不添加观测噪声。为生成一组时间序列，在给定初始条件 $\mathbf{x}(0)$ 和维纳过程 $\mathbf{W}$ 的情况下，对随机微分方程在时间窗口 $[t_0, T]$ 上进行积分。随机微分方程的积分使用 Diffrax [56], ，采用针对 Stratonovich 随机微分方程的随机龙格-库塔方法 [57] ，时间步长为 $\Delta t = 0.001$。随后，在离散时间 $t_k = t_0 + k\tau$ 处从积分结果中采样观测，其中 $k = 0, \ldots, K$ 且 $K = T/\tau$，从而得到 $\{\mathbf{x}(t_k)\}_{k=0}^{K}$。为收集一批时间序列，采样不同的初始条件和维纳过程，并用于系统积分。

3.2 遗传编程

遗传编程（GP）是一种进化算法，最初用于优化计算机程序的结构 [8] ，此后已成为一种强大的符号回归方法。遗传编程通过受生物进化启发的机制，采用基于种群的无梯度优化算法，同时演化数学表达式的结构和参数，以拟合数据。从随机初始化的种群开始，其中包含 p个最大深度为 m的个体，遗传编程会在 g 代中迭代执行评估、选择和繁殖。针对漂移和扩散演化而调整的遗传编程算法见算法 1。本文使用了 Kozax 库 [58], ，这是一个基于 JAX 构建的快速、灵活的遗传编程框架。

个体被表示为由函数节点和叶节点的预定义库构建的解析树。（图1a）展示了一棵示例树，其对应的符号方程为$\sin(x_1)\times(3-x_2)$。候选解通过适应度函数 $\mathcal{F}$ 进行评估，该函数赋予一个适应度值。在评估种群之后，个体

算法 1 遗传编程算法

输入 世代数 g、种群规模 p、适应度函数 $\mathcal{F}$

种群 $P \longleftarrow \{\}$
对于 i 在 p**中执行**
    采样漂移 $\hat{f}$ 和扩散 $\hat{g}$
    追加($\hat{f}, \hat{g}$) 到 P
结束循环
对 j 中的 g **执行**
    计算 F 在 $\mathcal{F}$上的 p 适应度
    子代 $O \longleftarrow \{\}$
    使用 NSGA-II 确定 R 的等级 P
    将 P 的帕累托前沿追加到 O
    当 大小(O) $< p$**时**
        从 P 中根据给定的 R进行锦标赛选择，随机选择亲本 p_1， p_2
        从 {交叉、变异}中随机选择进行繁殖
        子代 $c \longleftarrow$繁殖（ $\hat{f}$， $\hat{g}$) 来自 p_1和 p_2
        追加 c到 O
    结束 while 循环
    $P \longleftarrow O$
结束 for 循环
返回 P中的适应度最高的个体

被选中进行繁殖，优先选择适应度较高者。在Kozax中，解按照NSGA-II算法 [59] 进行选择，以适应度和复杂度为目标，复杂度定义为节点数量。被选中的个体在繁殖过程中通过交叉和变异发生结构变化。交叉在个体之间交换随机选择的子树（图1b），而变异则修改算子或叶节点、插入或删除节点（图1c）。在繁殖过程中，树中的节点数量受超参数 s约束。

为了同时演化漂移项和扩散项，个体由多棵树组成 [60], ，计算得到的适应度对应于这些树的组合。在繁殖过程中，交叉只能应用于两个个体中位于相同位置的一对树。此外，还加入了整树交叉操作，以促进优良树之间的交换 [61]。为了在扩大的解空间中更高效地遍历解空间，使用梯度下降优化树中的常数 [62]。为限制计算成本，此优化仅应用于 p^* 个个体的子集，优先选择适应度最高的解，进行 g^* 次迭代，步长为 η。为避免过早收敛并保持多样性，种群被划分为多个子种群 [63]，每个子种群具有不同的交叉和变异概率 [64]。每隔几代，个体便会迁移，以便在其他子种群中引入多样且有前景的结构 [63]。遗传编程算法实现中使用的超参数列于附录B的表4中。

3.3 问题定义

本文的核心任务是从观测到的含噪时间序列数据中识别控制方程（图1a）。我们提出一种利用遗传编程演化随机微分方程符号方程的方法。我们通过一组实验研究了该方法的有效性，并将其与演化常微分方程以及对克拉默斯-莫亚尔展开进行稀疏回归进行了比较。

3.3.1 实验

为证明本方法的有效性与鲁棒性，我们考察了若干系统与设定。首先，在加性噪声、线性乘性噪声和非线性乘性噪声下对一维双势阱问题进行辨识，以检验在扩散函数日益复杂时能否准确恢复。随后，我们研究了范德波尔振子和 R¨ossler 吸引子，二者均扩展了乘性噪声。为检验向高维问题的可扩展性，对具有五个、十个和二十个变量的 Lorenz96 模型进行了符号回归。在不同采样率下研究了 Lotka-Volterra 模型，以考察数据有限时的性能。图 1d 显示

这些环境的示例轨迹。最后，研究了 Fisher-KPP 与二维热方程，以检验向随机偏微分方程的泛化能力。关于各环境的更多细节，请参阅附录 A。在所有实验中，遗传编程的节点集由 $+$、$\times$ 以及环境中的变量数量组成。对于随机偏微分方程，梯度与拉普拉斯算子也被加入节点集。

3.3.2 适应度函数

ˆ 遗传编程共同演化变量 x_i 的分量漂移 $f_i(\mathbf{x}(t))$ 和扩散 ˆ$g_i(\mathbf{x}(t))$ 函数的符号表达式。为了评估候选解的适应度，需要根据数据确定最大似然估计。我们的方法聚焦于正态分布噪声，因此最大似然估计可定义为高斯分布的负对数似然。最大似然估计针对每次状态转移进行计算，表示在时间 t 下，给定均值 $\mu_i(t)$ 和方差 $\sigma_i^2(t)$ 时，$x_i(t)$ 的概率，时间范围为 $i = 1, ..., N$。适应度通过对成对条件转移概率求和来计算，与无模拟有限差分法 [11] 类似。由于假设具有完全可观测性，因此可以针对目标系统中的每个变量分别优化漂移和扩散。这样可以缩小搜索空间，因为需要同时演化的树更少。我们将在结果部分将这种方法称为 GP-SDE。具体而言，一对函数的适应度函数定义为

$$F(\hat{f}_i, \hat{g}_i) \coloneqq \log p(\{x_i(t_k)\}_{k=0}^K \mid \{\mu_i(t_k)\}_{k=0}^K, \{\sigma_i^2(t_k)\}_{k=0}^K) = \sum_{k=1}^K \left[\frac{1}{2} \log\left(2\pi\sigma_i(t_k)^2\right) + \frac{(x_i(t_k) - \mu_i(t_k))^2}{2\sigma_i(t_k)^2} \right] \tag{4}$$

其中 $\mu_i(t_k) = x_i(t_{k-1}) + \tau$ ˆ$f_i(\mathbf{x}(t_{k-1}))$ 和 $\sigma(t_k) = \sqrt{\tau}$ ˆ$g_i(\mathbf{x}(t_{k-1}))$.

当观测数据的采样率过低时，直接最小化转移对的最大似然估计可能会导致模型不准确。另一种方法是同时演化多个变量的方程 [42]。因此，可以在观测值之间以更小的时间步长对方程组进行数值积分 [34, ˆ55]。随后，通过 $\boldsymbol{\mu}(t_k) = \mathbf{y}(L)$ 计算下一个状态的均值，其中 $\mathbf{y}(l) = \mathbf{y}(l-1) + \frac{\tau}{L} f(\mathbf{y}(l-1))$ 且 $\mathbf{y}(0) = \mathbf{x}(t_{k-1})$，适用于 $l = 0, ..., L$，其中 L 设为 5。类似地，方差变为 $\boldsymbol{\sigma}(t_k) = \frac{\sqrt{\tau}}{L} \sum_{l=0}^{L-1}$ ˆ$g(\mathbf{y}(l))$。在结果部分余下内容中，将 GP-SDE 扩展到多个积分步（MS）称为 GP-SDE-MS。

3.3.3 基线方法

常微分方程对确定性动力系统进行符号回归已取得显著成果，甚至能够从含噪观测数据中恢复正确的常微分方程 [12, 65]。然而，当动力学具有随机性时，这类方法的性能可能会下降。在这种情况下，如果也能通过符号回归恢复随机分量，将会很有帮助。为了展示我们方法的优势，我们ˆ将其与传统的遗传编程方法进行比较；该方法仅从数据中学习每个变量的漂移项 f_i，我们称之为 GP-ODE。GP-ODE 要最小化的适应度函数为

$$F(\hat{f}_i) \coloneqq \mathrm{MSE}(\{x_i(t_k)\}_{k=0}^K \mid \{\mu_i(t_k)\}_{k=0}^K) = \sum_{k=1}^K (x_i(t_k) - \mu_i(t_k))^2, \tag{5}$$

其中 $\mu_i(t_k) = x_i(t_{k-}$ ˆ$_1) + \tau f_i(\mathbf{x}(t_{k-1}))$。GP-ODE 也可以扩展为对多个步长上的漂移进行积分 s 观测之间 ，这称为 GP-ODE-MS。

此外，我们提出的方法还将与基于克拉默斯-莫亚尔展开和稀疏回归的 SDE 符号恢复成熟方法进行比较，我们将其称为 KM-SR。下文将简要介绍克拉默斯-莫亚尔展开和稀疏回归，更详细的说明请参见参考文献 [23]。

克拉默斯-莫亚尔展开 克拉默斯-莫亚尔展开是时间序列数据的泰勒展开，可给出变量 x_i 的漂移$D_{i,1}$ 和扩散$D_{i,2}$ 系数的显式估计。这些系数可以用

$$D_{i,1}(\{x_i(t_k)\}_{k=0}^K) = \mathbb{E}_{k\in[0,K]}\left[\frac{1}{\tau}(x_i(t_k) - x_i(t_{k-1}))\right], \quad D_{i,2}(\{x_i(t_k)\}_{k=0}^K) = \sqrt{\mathbb{E}_{k\in[0,K]}\left[\frac{1}{\tau}(x_i(t_k) - x_i(t_{k-1}))^2\right]}. \tag{6}$$

为提高估计的计算效率和准确率，漂移系数和扩散系数在分箱内取平均。分箱的确定方式是将观测数据中最低值与最高值之间的范围均匀划分为选定数量的分箱 b。对于多维系统，这将形成一个网格，其中每个维度被划分为 b 个分箱。仅当某个分箱内的数据点数量大于指定阈值 β时，才纳入该分箱的系数。

稀疏回归 稀疏回归将回归中的 L1 正则化与函数库相结合，从数据中学习符号方程，并成为符号回归中 GP 的有力替代方法 [11]。函数库 $\boldsymbol{\Theta}$ 通过计算观测变量在指定次数 d以内的所有组合多项式构建而成。将函数库与向量$\mathbf{c}$相乘，其中向量$\mathbf{c}$通过采用 L1 正则化的最小二乘回归拟合观测数据获得。最优系数向量$\mathbf{c}^*$ 由下式确定：

$$\mathbf{c}^* = \arg\min_{\mathbf{c}} \left[\|\mathbf{y} - \mathbf{c}\boldsymbol{\Theta}\|_2 + \alpha \|\mathbf{c}\|_1 \right]. \tag{7}$$

其中， α 决定 L1 正则化的强度，alpha 越大，得到的解越稀疏。得到 $\mathbf{c}^*$后，将低于阈值 λ 的系数设为零，并从函数库中移除相应函数，以得到 $\boldsymbol{\Theta}'$ [46]。使用 $\boldsymbol{\Theta}'$再次运行方程（7），重复此过程，直到$\mathbf{c}^*$ 中没有系数小于 λ。

克拉默斯-莫亚尔展开返回观测时间序列中一个变量的漂移和扩散系数，随后分别对漂移和扩散应用稀疏回归，从而得到两个方程。对于具有多个维度的系统，克拉默斯-莫亚尔展开针对每个变量独立进行。KM-SR 的两个阶段都对所选超参数值较为敏感，并且需要针对不同的问题设置进行调优。为提高 KM-SR 的泛化能力，在每个实验中，分别针对每个变量的漂移和扩散独立执行超参数搜索。可调节的超参数包括 b、 β、 α 和 λ。不同实验中各超参数的取值集合见表3。

比较 三种方法 KM-SR、GP-ODE 和 GP-SDE 分别使用不同的指标进行优化，因此直接比较并非易事。为了公平地评估这些方法，我们生成了三个数据集：优化集、验证集和测试集。优化集用于演化或拟合模型，使其与数据相匹配。将真实系统应用于验证集和测试集，从而得到漂移和扩散的目标值。这样便可以计算漂移的估计输出与目标输出之间的均方误差（MSE），其计算方式为

$$\mathrm{MSE}(\{\mathbf{x}(t_k)\}_{k=0}^K) = \frac{1}{NK} \sum_{i=1}^{N} \sum_{k=0}^{K} \left[f_i(\boldsymbol{x}(t_k)) - \hat{f}_i(\boldsymbol{x}(t_k)) \right]^2, \tag{8}$$

ˆ 其中 $f_i(\mathbf{x}(t))$ 是真实漂移函数， $f_i(\mathbf{x}(t))$ 是通过任意方法发现的漂移函数。类似地，扩散的均方误差使用 $g_i(\mathbf{x}(t))$ 和 ˆ$g_i(\mathbf{x}(t))$ 计算。KM-SR 在超参数搜索过程中会生成不同的解，并返回验证集上均方误差最低的解。在 GP-ODE 和 GP-SDE 中，则选取最终帕累托前沿中验证集均方误差最低的解。最后，针对每种方法的最佳解计算测试数据上的均方误差，以便进行比较。

3.3.4 硬件配置

基于 GP 的方法在一台配备 Intel Xeon Platinum 8360Y（2.40 GHz）36 核处理器和单个 NVIDIA A100 GPU（40 GiB HBM2内存）的计算机上运行，而 KM-SR 则在一台配备 AMD Genoa 9654（2.40 GHz）192 核处理器的计算机上运行。

4 结果

本文提出一种基于遗传编程的方法，用于识别随机微分方程（GP-SDE）。我们将 GP-SDE 与符号回归中的标准方法进行比较，后者仅用遗传编程演化漂移函数（GP-ODE）。此外，我们还将其与基于 Kramers-Moyal 展开和稀疏回归的既有基线（KM-SR）进行比较。这些方法在多种环境中进行评估，考察在不同噪声项、维度和采样率下对系统的恢复能力。代码和数据见 https://github.com/sdevries0/GP-SDE。

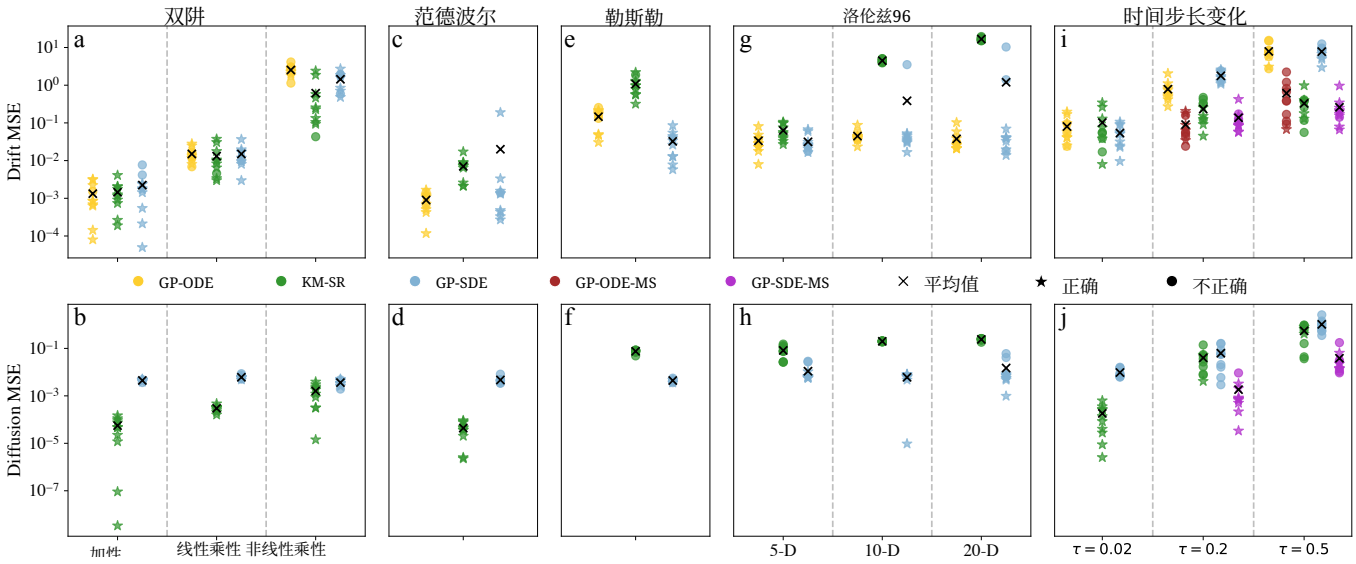


图2：**GP-SDE 能够准确恢复随机动力系统的控制方程**。展示了在测试数据上，漂移和扩散的真实函数与学习函数之间的均方误差（MSE），共采用十个不同的随机种子。每种方法十次运行的平均值均以黑色十字标出。此外，对于每次运行，星号表示所有变量的方程结构均被正确识别，而圆圈表示至少有一个方程的结构不正确。这些方法包括先进行克拉默斯-莫亚尔展开、再进行稀疏回归（KM-SR），常微分方程的遗传编程（GP-ODE）——该方法仅学习漂移函数，以及随机微分方程的遗传编程（GP-SDE）。（a、b）带有加性、线性乘性和非线性乘性扩散项的双阱问题结果。（c、d）范德波尔振子结果。（e、f）R¨ossler 吸引子结果。（g、h）含有5、10和20个变量的 Lorenz96 模型结果。（i、j）采样率为0.02、0.2和0.5的洛特卡-沃尔泰拉模型结果。GP-ODE 和 GP-SDE 均通过多步积分进行扩展，分别标记为 GP-ODE-MS 和 GP-SDE-MS。

GP-SDE 在低维问题上可与 KM-SR 媲美

为评估 GP-SDE 能否从数据中准确恢复漂移和扩散，我们首先研究具有不同扩散项的一维双势阱问题。由 GP-ODE、KM-SR 和 GP-SDE 得到的漂移与扩散的均方误差见图 2a+b。对于加性（标量）扩散，GP-ODE 始终以较低的均方误差恢复出正确的漂移函数。KM-SR 在漂移准确率上与之相当，同时还能识别扩散。GP-SDE 表现具有竞争力，但在不同运行之间的一致性较差。

将双阱问题中的扩散改为乘性扩散后，由于系统动力学变得更加不规则，任务也更加复杂。在线性乘性噪声下，GP-ODE 的均方误差有所上升，表明其对过程噪声较为敏感。KM-SR 仍恢复出了漂移和扩散的正确结构，但均方误差更高。GP-SDE 的表现与 KM-SR 相近，但扩散项的均方误差略高。在非线性乘性噪声下，各方法的均方误差都大幅增加。GP-ODE 的表现始终不如其他方法，这表明若不对复杂的扩散项建模，就无法准确恢复漂移。KM-SR 的表现通常优于 GP-ODE，但在某些随机种子下性能有所下降。同样，就均方误差而言，KM-SR 略优于 GP-SDE，但两种方法往往都能演化出正确的方程。

双阱问题上的实验表明，在一维设定中，GP-SDE 是一种可与 KM-SR 相竞争的方法，尽管 KM-SR 更为精确，并且演化扩散项相较于 GP-ODE 具有优势。为进一步探究各方法的优势与局限，我们考虑具有随机多维动力学的基准环境。第一个基准环境是随机范德波尔振子（图 2c+d），其中扩散项仅对一个状态变量加入乘性噪声，从而要求各方法区分随机变量与确定性变量。漂移由 GP-ODE 识别得最为准确，而 KM-SR 正确识别了漂移与扩散二者的结构。GP-SDE 取得了与基线方法相当的漂移均方误差，尽管有一次运行得到了不正确的方程。扩散的结构未能恢复

在任何一次 GP-SDE 运行中均未得到恢复，但其均方误差仍相对接近 KM-SR。

R¨ossler 吸引子进一步增加了任务的难度，因为它是一个具有混沌行为的三维系统（图 2e+f）。尽管如此，GP-ODE 在部分运行中仍能恢复正确的方程。更高的维度和混沌轨迹降低了 KM-SR 分箱的准确率，导致漂移和扩散的均方误差均有所增加，尽管它通常能够找到正确的结构。相比之下，基于 GP 的方法不依赖分箱，因此 GP-SDE 在两个组成部分的均方误差上都优于 KM-SR。此外，GP-SDE 还能从学习扩散项中获益：与 GP-ODE 相比，GP-SDE 的平均漂移均方误差更低，恢复结果也更加稳定。

GP-SDE 在高维问题中仍然稳健

R¨ossler 吸引子表明，维度增加会因分箱不准确以及函数库不断扩大而降低 KM-SR 的稳健性。为进一步探究这一局限性，我们在具有五个、十个和二十个变量的随机 Lorenz96 模型上评估了这些方法，均方误差见图 2g+h。对于这三种情况，GP-ODE 均恢复了正确的漂移函数。当变量数为五时，KM-SR 仍表现良好，取得了具有竞争力的均方误差；但变量数增至十和二十时，其性能显著下降，无法准确识别漂移和扩散。这种性能退化可归因于分箱过程的可扩展性较差。变量数为十时，每个维度使用四个分箱，总数便已超过一百万，而函数库的规模为 66；变量数增至二十时，函数库规模还会进一步扩大。GP-SDE 的扩展性优于 KM-SR，在所有设置下都取得了更低的均方误差，并且在五维情况下，其漂移准确率甚至与 GP-ODE 相当。尽管对于十个和二十个变量，GP-SDE 未能始终恢复精确的方程，但在大多数运行中都能演化出准确的解，表现明显优于 KM-SR。

通过积分，GP-SDE 能够从稀疏数据中识别方程

在此前的实验中，轨迹采样密集，因此方程恢复相对容易。然而，实际应用中的数据可能采样稀疏或不规则。为评估各方法在稀疏数据条件下的稳健性，我们在随机洛特卡-沃尔泰拉模型上比较了它们在不同采样率下的表现（图 2i+j）。在原始采样时间 0.02 下，KM-SR、GP-ODE 和 GP-SDE 的表现不相上下；其中，KM-SR 对扩散的估计更准确，而 GP-SDE 的漂移恢复效果更好。将采样时间增至 0.2 后，KM-SR 的均方误差略有上升，而 GP-ODE 和 GP-SDE 的表现则明显变差，尤其是在漂移项上。由于遗传编程能够针对多个变量同时演化漂移和扩散，因此可以在观测数据点之间对函数进行数值积分，从而获得更细粒度的预测。加入这一功能后，GP-ODE-MS 和 GP-SDE-MS 的表现均优于各自的标准版本，且两种方法都胜过 KM-SR。GP-SDE-MS 在识别正确的方程结构方面也最为成功。当采样时间为 0.5 时，多步积分的优势更加明显。在这种情况下，GP-SDE-MS 优于 GP-ODE-MS，进一步表明对随机性进行建模有助于恢复确定性函数。

学习得到的方程生成了信息丰富的生成样本

图2e和f表明，在均方误差方面，GP-SDE在R¨ossler吸引子上的表现优于KM-SR和GP-ODE。在大多数运行中，这三种方法都能恢复漂移的正确结构，但KM-SR和GP-SDE均未能识别扩散结构。为检视学习得到的系统，表1列出了每种方法表现最佳的一次运行所得到的方程，以及真实方程。GP-ODE恢复了 f_x 和 f_y的正确方程及参数，而恢复得到的 f_z 在常数项上略有偏差。其主要局限在于没有学习到任何函数来刻画随机性。KM-SR虽然确实识别出了一个扩散函数，但拟合得到的方程与真实方程的相似度较低，尽管它识别出了漂移的正确方程结构。KM-SR未能在 g_x 中纳入与状态的乘法项，并在 g_z中加入了一个额外的常数。GP-SDE则同时学习漂移和扩散，并实现了极高精度的恢复。漂移中的常数接近真实值，且对于 f_z，其拟合效果优于GP-ODE。扩散项接近真实函数，但常数略小。与KM-SR类似，GP-SDE也在 g_z中学习到了一个额外的常数，不过其值更接近于零，同时对状态系数的拟合也更好。

为进一步分析所发现方程的质量，我们对系统进行模拟，并将结果与真实轨迹进行比较。首先，在给定固定初始条件的情况下，我们根据真实值生成三条轨迹。图3a 中的黑色曲线显示真实时间序列，而黄色曲线则展示了通过 GP-ODE 学到的系统的确定性轨迹。由于 GP-ODE 返回的是一个确定性系统，因此给定一个初始

表1：使用常微分方程遗传编程（GP-ODE）、克拉默斯-莫亚尔展开和稀疏回归（KM-SR），以及随机微分方程遗传编程（GP-SDE）发现的 R¨ossler 吸引子的真实系统对应方程和最佳解。对于每个变量 x、y 和 z，均展示了漂移和扩散的方程。GP-ODE 只对漂移函数进行演化，因此不展示扩散方程。

方法	$f_x(x,\ y,\ z)$	$g_x(x,\ y,\ z)$	$f_y(x,\ y,\ z)$	$g_y(x,\ y,\ z)$	$f_z(x,\ y,\ z)$	$g_z(x,\ y,\ z)$
True	$-x-y$	$0.100x$	$x+0.200y$	$0.100y$	$xz-5.700z+0.200$	$0.100z$
GP-ODE	$-0.999x-0.999y$	–	$x+0.196y$	–	$xz-5.790z+0.282$	–
KM-SR	$-0.950x-1.486y$	0.306	$1.003x+0.198y$	$0.146y$	$0.868xz-4.849z+0.157$	$0.064z+0.055$
GP-SDE	$-x-0.981y$	$0.083x$	$x+0.191y$	$0.083y$	$xz-5.670z+0.188$	$0.091z-0.008$

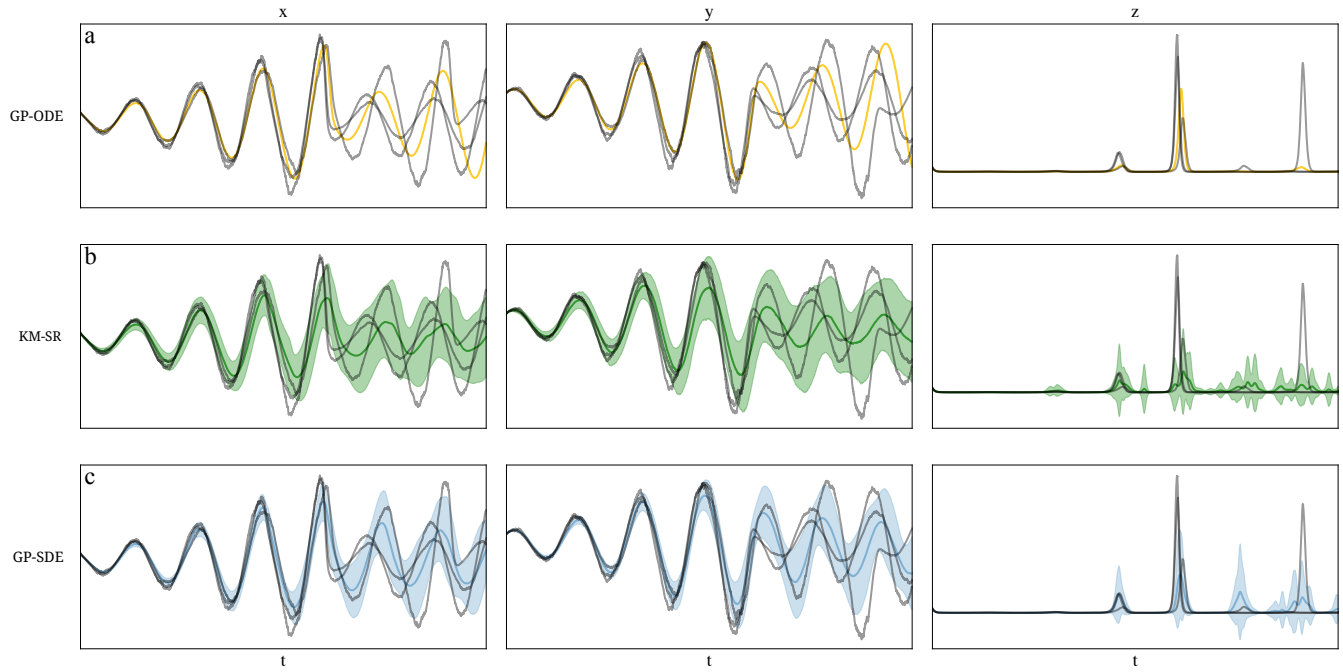


图3：**对发现的模型进行仿真：R¨ossler吸引子**。(a) 给定固定的初始条件，使用不同的维纳过程（黑色）模拟 R¨ossler 吸引子的三条轨迹。遗传编程（GP-ODE）演化得到的最佳常微分方程从相同的初始条件出发进行积分（橙色）。(b) 显示真实系统的相同三条轨迹，以及基于克拉默斯-莫亚尔展开和稀疏回归的方法（KM-SR）所发现的最佳模型生成的50条轨迹的均值（绿色）和标准差（阴影区域）。这些轨迹的初始条件固定，但各自使用不同的维纳过程。(c) 与(b)采用相同方法，但使用遗传编程发现的最佳随机微分方程（GP-SDE）。各方法对应的方程列于表1。

条件时，只能得到一条轨迹。这条确定性轨迹在 x 和 y中呈现出与真实轨迹相似的模式，但经过一段时间后便偏离真实样本。 z 的轨迹在正确的时间点出现了较大的向上尖峰，但未能捕捉到较小的尖峰。由于确定性系统只能生成一个样本，因此无法展示可能样本路径的演化过程。

由于 KM-SR 还会返回扩散项，生成多条轨迹便能覆盖随机微分方程的可能路径。50 个样本的均值和标准差以绿色显示在图 3b 中。 x 和 y 的采样路径在轨迹初期就已明显扩散。最终，这些样本覆盖了状态空间的很大一部分，却无法与真实时间序列紧密吻合，尤其是在真实轨迹的极值处。

表2：展示了与真实随机偏微分方程相对应的系统，以及使用 GP-SDE 演化得到的解。

方法	Fisher-KPP方程 $\mathrm{d}u(t,x)$	二维热传递 $\mathrm{d}u(t,\ x,\ y)$
真实值	$(u_{xx}+u(1-u))\,\mathrm{d}t+0.100u\mathrm{d}W$	0。$100(u_{xx}+u_{yy})\mathrm{d}t+(u_x+u_y)\mathrm{d}W$
GP-SDE	$(u_{xx}+u(0.929-0.936u))\,\mathrm{d}t+(0.094u-0.011)\mathrm{d}W$	0。$098(u_{xx}+u_{yy})\mathrm{d}t+(u_x+u_y)\mathrm{d}W$

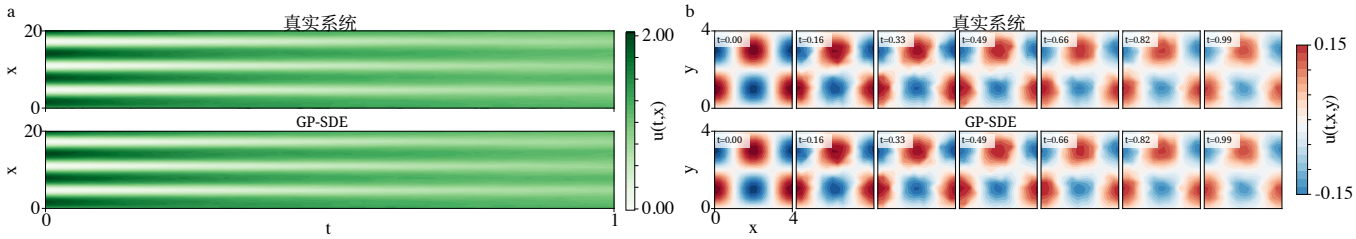


图4：**GP-SDE 能够还原随机偏微分方程**。(a) 给定真实值方程以及表2中所示由 GP-SDE 识别出的系统，展示 Fisher-KPP 方程随 x 和时间的演化。(b) 展示二维热传导随 x、 y 和时间演化的快照。同样，图中给出了由 GP-SDE 得到的真实系统和学习系统。

与真实值样本相比，变量 z 的样本中峰值出现得更频繁，但峰值高度不准确，出现的时间点也不正确。

图3c 中以蓝色展示了根据 GP-SDE 的解生成样本的均值和标准差。这些样本拟合得更好，对于变量 x 和 y，其均值和标准差更准确，覆盖了极值，且呈现出更多振荡。变量 z 的尖峰与真实轨迹大体重合，但出现的时间和高度也有不准确之处。这些模拟表明，在所提出的设置下可以识别出有效系统，同时也展示了求得随机解的优势，以及我们的方法相较于 KM-SR 的有效性。

GP-SDE 扩展至随机偏微分方程

为进一步检验所提出方法的泛化能力，我们使用 GP-SDE 学习随机偏微分方程的控制方程。首先，研究了带有乘性噪声的一维Fisher-KPP方程。GP-SDE 恢复了漂移项的正确结构，但常数略有偏差，并在随机分量中加入了一个较小的常数（表2）。Fisher-KPP方程的演化模拟显示，真实系统与拟合系统高度相似（图4a）。为检验该方法在随机偏微分方程领域的可扩展性，我们还将 GP-SDE 应用于二维热传导系统。在该系统中，GP-SDE 演化出了完整方程的正确结构，且常数与真实值非常接近（表2）。正如预期，真实系统与演化所得系统的模拟呈现出相似的模式。这些实验表明，我们的方法能够正确识别随机偏微分方程的结构，同时仍以最大似然估计为目标。

GP-SDE 的运行时间可高效扩展

实验表明，使用遗传编程拟合随机微分方程在稀疏数据约束下具有更好的可扩展性和稳健性。然而，对于实际应用而言，这些方法的计算时间也是一个重要属性。图5给出了GP-ODE、KM-SR和GP-SDE在不同维度的Lorenz96 模型上的平均计算时间。KM-SR的运行时间在四个和十六个分箱下进行评估，同时仍对其余参数执行超参数搜索。GP-ODE和GP-SDE以500的种群规模、50代进行评估。在低维度下，KM-SR明显快于基于遗传编程的方法，更多分箱仅略微增加了运行时间。然而，在十六个分箱时，当维度为五时运行时间急剧增长，而在维度为十时，该KM-SR配置在计算上不可行。即使只有四个分箱，KM-SR在十维问题上也变得极其缓慢，同时还产生了不准确

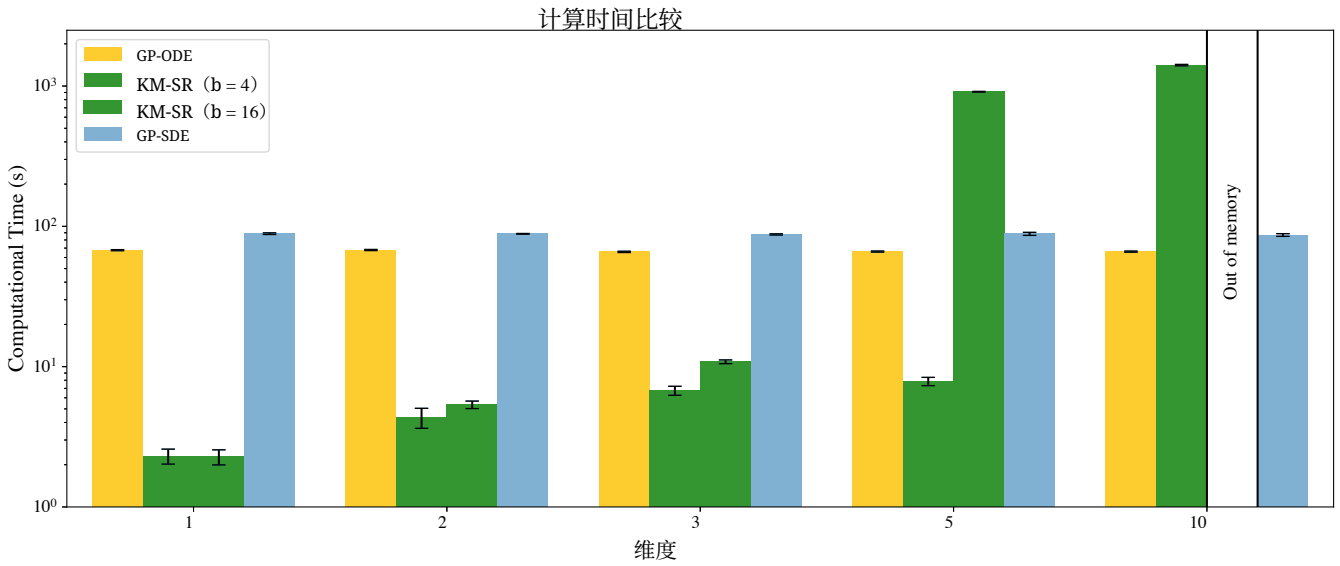


图 5：**各方法的运行时间**。随着维度增加，GP-ODE、KM-SR 和 GP-SDE 在 Lorenz96 模型上的计算时间。每次实验先运行一个随机种子以编译算法，随后对其余十个随机种子的运行时间取平均。KM-SR 分别使用四个和十六个分箱进行评估（b）。

结果（图2g+h）。尽管在低维问题上GP-ODE和GP-SDE比KM-SR更慢，但运行时间几乎不随问题规模增加。对于所有问题，GP-SDE都略慢于GP-ODE，因为需要演化更多的树，但相对差异不会随变量增多而扩大。这些结果表明，遗传编程是一种更具可持续性的方法，因为它在计算上和效果上都能很好地扩展到更高维度。

5 讨论

本文介绍了一种用于数据驱动发现随机微分方程符号方程的方法。我们通过优化最大似然估计目标函数，利用遗传编程学习了一对描述漂移和扩散的方程。大多数符号回归方法仅专注于学习漂移函数，而识别扩散不仅能增强建模能力，还能提供有关底层系统的信息。目前，随机微分方程符号回归唯一的替代方法是将克拉默斯-莫亚尔展开与稀疏回归结合起来 [23]。与该方法相比，我们提出的方法具有诸多优势，例如无需采用两步法或分箱方法，对超参数不那么敏感，并且能够联合优化函数，从而提高所得系统与观测数据的一致性。

我们在多种实验中测试了该方法，考察了基准系统、可扩展性以及对低采样率的鲁棒性。将遗传编程扩展到学习扩散项后，我们发现两个函数都能被准确识别，同时确定性分量的质量并未降低。此外，对于更复杂的任务，与仅学习漂移相比，同时演化两个函数甚至有助于学习漂移函数。遗传编程与克拉默斯-莫亚尔展开的稀疏回归相比表现不逊，且在维度较高的环境中表现更优。对于变量较多的情况，克拉默斯-莫亚尔展开中的分箱方法效率很低，尤其是在数据值范围也较大的情况下。即使维度增加到十或二十，我们的方法仍能恢复真实方程，同时保持显著更高的计算效率。此外，我们的方法还可扩展为同时学习所有变量的方程，从而能够在观测数据点之间进行数值积分。通过这一扩展，仍能从稀疏采样数据中准确恢复方程。最后，该框架还展现出从随机偏微分方程中学习结构的泛化能力。

学习动力系统，尤其是随机微分方程时，一个根本挑战是可辨识性和非唯一性 [66, 67]。不同的系统可能产生统计上无法区分的动力学行为，因此在稀疏数据约束或含噪动力学条件下，难以识别真实方程。在随机微分方程推断中，这一挑战

因漂移和扩散的明确分离而更加突出。随机效应可能部分被漂移吸收，或者确定性结构通过扩散得到解释。尽管我们的结果表明，通常能够恢复正确的函数结构，但有时也会观察到，使用（部分）错误的符号形式仍能获得准确的模型拟合。因此，将此方法应用于真实世界系统时，重要的是要认识到，良好的预测拟合并不一定意味着找到了正确的控制方程。

我们的方法依赖于一些较强的假设，这些假设可能会限制其在实际应用中的适用性。该方法假设数据具有完全可观测性，但观测结果可能含有噪声，也可能存在未观测到的变量。要处理这类情况，需要使用变分推断来学习潜变量随机微分方程 [20, 21, 68]。SINDy 已扩展为能够从部分观测数据中学习潜在符号动力学，但其函数空间仍局限于确定性系统 [44,45]。将符号回归扩展到潜变量随机微分方程面临多项挑战，包括联合演化整个系统、推断初始条件以及整合完整轨迹。此外，还必须调整适应度函数，以便根据观测数据评估推断出的潜在轨迹。可辨识性问题也因此更加突出，因为将观测噪声与过程随机性区分开来本身就很困难。

另一个可能妨碍实际应用的假设是，该方法仅限于识别噪声呈正态分布且可与漂移分离的系统。文献 [26] 表明，KM-SR 也可用于识别控制 L´évy 跳跃的项。本文提出的适应度函数应推广至学习其他类型噪声对应的方程，理想情况下，甚至能够选择最适合解释观测数据的噪声类型。还可以进一步推广该方法，取消漂移与随机性可分离这一假设，方法是将随机噪声作为状态方程的输入。

总而言之，本文证明遗传编程是实现随机微分方程符号发现的可行方法，并为将其应用于要求在高维度或稀疏数据条件下具备鲁棒性的实际问题开辟了道路。我们的贡献有助于理解呈现噪声数据的动力学，既能改善拟合效果，也能提升对未知系统中确定性分量和随机分量的可解释性。未来的工作将整合所提出的改进，使该框架成为自动化科学发现领域中完整且通用的方法体系。

6 致谢

本出版物是项目 ROBUST：可信人工智能可持续增长系统的一部分，项目编号为 KICH3.L TP.20.006。该项目由荷兰研究理事会（NWO）、ASMPT 和荷兰经济事务与气候政策部（EZK）通过 LTP KIC 2020-2023 计划提供部分资助。本文所有内容均代表作者的观点，不一定得到其各自雇主和/或资助方的认同或认可。

参考文献

[1] Christopher Rackauckas 等人。“用于科学机器学习的通用微分方程”。载于：*arXiv* 预印本 *arXiv:2001.04385*（2020）。[2] Hanchen Wang 等人。“人工智能时代的科学发现”。载于：《自然》620.7972（2023），第47–60页。[3] Dane Morgan 与 Ryan Jacobs。“机器学习在材料科学中的机遇与挑战”。载于：《材料研究年度评论》50.1（2020），第71–103页。[4] Gisbert Schneider。“药物发现自动化”。载于：《自然综述：药物发现》17.2（2018），第97–113页。[5] Connor W Coley、Natalie S Eyke 与 Klavs F J ensen。“化学科学中的自主发现（第一部分）：进展”。载于：《应用化学国际版》59.51（2020），第22858–22893页。[6] Nour Makke 与 Sanjay Chawla。“基于符号回归的可解释科学发现：综述”。载于：《人工智能评论》57.1（2024），第2页。[7] Silviu-Marian Udrescu 与 Max Tegmark。“AI Feynman：一种受物理学启发的符号回归方法”。载于：《科学进展》6.16（2020），eaay2631。[8] John R Koza。“遗传编程：借助自然选择为计算机编程的一种手段”。载于：*Statisticsand Computing* 4.2（1994），第87–112页。

[9] Josh Bongard 与 Hod Lipson。“非线性动力系统的自动逆向工程”。载于：美国国家科学院院刊 104.24（2007），第9943–9948页。[10] Michael Schmidt 与 Hod Lipson。“从实验数据中提炼自由形式的自然定律”。载于：《科学》324.5923（2009），第81–85页。[11] Steven L Brunton、Joshua L Proctor 与 J Nathan Kutz。“通过非线性动力系统的稀疏辨识从数据中发现控制方程”。载于：美国国家科学院院刊 113.15（2016），第3932–3937页。[12] St´ephane d’Ascoli 等人。“ODEformer：利用 Transformer 对动力系统进行符号回归”。载于：arXiv 预印本 arXiv:2310.05573（2023）。[13] Samuel Kim 等人。“将基于神经网络的符号回归融入深度学习以促进科学发现”。载于：IEEE 神经网络与学习系统汇刊 32.9（2020），第4166–4177页。[14] Luca Biggio 等人。“可扩展的神经符号回归”。载于：国际机器学习会议。PMLR。2021，第936–945页。[15] Ying Jin 等人。“贝叶斯符号回归”。载于：arXiv 预印本 arXiv:1910.08892（2019）。[16] Markus Quade 等人。“基于符号回归的动力系统预测”。载于：Physical Review E 94.1（2016），第012214页。[17] Daniel A Messenger 与 David M Bortz。“用于偏微分方程的弱形式 SINDy”。载于：Journal of Computational Physics 443（2021），第110525页。[18] Michael Zhang 等人。“用于发现参数方程的深度学习与符号回归”。载于：IEEE 神经网络与学习系统汇刊（2023）。[19] Patrick AK Reinbold 与 Roman O Grigoriev。“含潜变量的偏微分方程模型的数据驱动发现”。载于：Physical Review E 100.2（2019），第022219页。[20] Ahmed ElGazzar 与 Marcel van Gerven。“通过潜在随机微分方程对神经动力学进行生成建模”。载于：arXiv 预印本 arXiv:2412.12112（2024）。[21] Belinda Tzen 与 Maxim Raginsky。“神经随机微分方程：扩散极限下的深度潜在高斯模型”。载于：arXiv 预印本 arXiv:1905.09883（2019）。[22] Yang Song 等人。“通过随机微分方程实现基于分数的生成建模”。载于：arXiv 预印本 arXiv:2011.13456（2020）。[23] Lorenzo Boninsegna、Feliks N¨uske 与 Cecilia Clementi。“随机动力学方程的稀疏学习”。载于：《化学物理学杂志》148.24（2018）。[24] Yasen Wang 等人。“随机微分方程的数据驱动发现”。载于：《工程》17（2022），第244–252页。[25] Yalda Rajabzadeh、Amir Hossein Rezaie 与 Hamidreza Amindavar。“用于重构随机微分方程模型的稳健非参数框架”。载于：《物理学A：统计力学及其应用》450（2016），第294–304页。[26] Yang Li 与 Jinqiao Duan。“发现含非高斯 L´evy 噪声的随机动力系统的数据驱动方法”。载于：《物理学D：非线性现象》417（2021），第132830页。[27] Yang Li、Shengyuan Xu 与 Jinqiao Duan。“基于非局部 Kramers–Moyal 公式发现非高斯随机动力系统的进化方法”。载于：《非线性科学与数值模拟通讯》145（2025），第108751页。[28] Yogesh Chandrakant Mathpati 等人。“利用变分贝叶斯推断从有限数据中发现随机偏微分方程”。载于：Computer Methods in Applied Mechanics and Engineering 418（2024），第116512页。[29] Junteng Jia 与 Austin R Benson。“神经跳跃随机微分方程”。载于：《神经信息处理系统进展》32（2019）。[30] Hannes Risken。“福克-普朗克方程”。载于：《福克-普朗克方程：求解方法与应用》。施普林格，1996，第63–95页。

[31] Christoph Honisch 和 Rudolf Friedrich。《低采样率下克拉默斯-莫亚尔系数的估计》。载于：Physical Review E—统计、非线性与软物质物理学 83.6（2011），第 066701 页。[32] Jan Nygaard Nielsen、Henrik Madsen 和 Peter C Young。《随机微分方程中的参数估计：综述》。载于：Annual Reviews in Control 24（2000），第 83–94 页。[33] Yacine Ait-Sahalia。《离散采样扩散过程的最大似然估计：一种闭式近似方法》。载于：Econometrica 70.1（2002），第 223–262 页。[34] Asger Roer Pedersen。《基于离散观测的随机微分方程最大似然估计新方法》。载于：Scandinavian Journal of Statistics（1995），第 55–71 页。[35] Niels Rode Kristensen、Henrik Madsen 和 Sten Bay Jørgensen。《随机灰箱模型中的参数估计》。载于：Automatica 40.2（2004），第 225–237 页。[36] Dimitrios Angelis、Filippos Sofos 和 Theodoros E Karakasidis。《物理科学中的人工智能：符号回归的趋势与展望》。载于：Archives of Computational Methods in Engineering（2023），第 1 页。[37] Miles Cranmer。《借助 PySR 和 SymbolicRegression.jl 为科学实现可解释机器学习》。载于：arXiv 预印本 arXiv:2305.01582（2023）。[38] William La Cava 等。《当代符号回归方法及其相对性能》。载于：《神经信息处理系统进展》2021.DB1（2021），第 1 页。[39] Erina Sakamoto 和 Hitoshi Iba。《利用遗传编程推断基因调控网络的微分方程组》。载于：2001 年进化计算大会论文集。第 1 卷。IEEE。2001，第 720–726 页。[40] Mauro Castelli 等。《住宅建筑能源性能预测：一种遗传编程方法》。载于：Energy and Buildings 102（2015），第 67–74 页。[41] Ben McKay、Mark Willis 和 Geoffrey Barton。《利用遗传编程对化学过程系统进行稳态建模》。载于：Computers & Chemical Engineering 21.9（1997），第 981–996 页。[42] Hongqing Cao 等。《利用遗传编程对常微分方程组进行演化建模》。载于：Genetic Programming and Evolvable Machines 1.4（2000），第 309–337 页。[43] Hitoshi Iba。《利用遗传编程推断微分方程模型》。载于：Information Sciences 178.23（2008），第 4453–4468 页。[44] Joseph Bakarji 等。《利用深度延迟自编码器从部分测量中发现控制方程》。载于：Proceedings of the Royal Society A 479.2276（2023），第 20230422 页。[45] Paolo Conti 等。《VENI、VINDy、VICI：一种具备不确定性量化功能的生成式降阶建模框架》。载于：Neural Networks（2026），第 108543 页。[46] Jared L Callaham 等。《利用朗之万回归进行非线性随机建模》。载于：Proceedings of the Royal Society A 477.2250（2021），第 20210092 页。[47] Yunfei Huang 等。《从数据中稀疏推断随机微分方程并进行主动学习》。载于：Scientific Reports 12.1（2022），第 21691 页。[48] Mozes Jacobs 等。《Hypersindy：非线性随机控制方程的深度生成建模》。载于：arXiv 预印本 arXiv:2310.04832（2023）。[49] Arshed Nabeel 等。《从生态时间序列数据中发现随机动力学方程》。载于：The American Naturalist 205.4（2025），E100–E117。[50] Liu Yang、Dongkun Zhang 和 George Em Karniadakis。《用于随机微分方程的物理信息生成对抗网络》。载于：SIAM Journal on Scientific Computing 42.1（2020），A292–A317。[51] Weiheng Zhong 和 Hadi Meidani。《PI-VAE：用于随机微分方程的物理信息变分自编码器》。载于：Computer Methods in Applied Mechanics and Engineering 403（2023），第 115664 页。[52] Hyomin Shin 和 Minseok Choi。《用于随机微分方程不确定性量化的物理信息变分推断》。载于：Journal of Computational Physics 487（2023），第 112183 页。

[53] Simo S¨arkk¨a 等。《通过非线性高斯滤波和自适应马尔可夫链蒙特卡罗对随机微分方程参数进行后验推断》。载于：*Statistics and Computing* 25.2（2015），第 427–437 页。[54] Philipp Batz、Andreas Ruttor 和 Manfred Opper。《随机微分方程的近似贝叶斯学习》。载于：*Physical Review E* 98.2（2018），第 022109 页。[55] Khalid B Nowman。《利率期限结构单因子连续时间模型的高斯估计》。载于：*The Journal of Finance* 52.4（1997），第 1695–1706 页。[56] Patrick Kidger。《关于神经微分方程》。载于：*arXiv* 预印本 *arXiv:2202.02435*（2022）。[57] James M Foster、Goncalo Dos Reis 和 Calum Strange。《满足交换性条件的随机微分方程高阶分裂方法》。载于：*SIAM Journal on Numerical Analysis* 62.1（2024），第 500–532 页。[58] Sigur de Vries、Sander W Keemink 和 Marcel AJ van Gerven。《Kozax：基于 JAX 的灵活且可扩展遗传编程》。载于：遗传与进化计算会议伴会论文集。2025，第 603–606 页。[59] Kalyanmoy Deb 等。《快速精英多目标遗传算法：NSGA-II》。载于：*IEEE Transactionson Evolutionary Computation* 6.2（2002），第 182–197 页。[60] William B Langdon。《遗传编程与数据结构：遗传编程 + 数据结构 =自动编程！施普林格科学与商业媒体，1998。[61] Durga Prasad Muni、Nikhil R Pal 和 Jyotirmoy Das。《使用遗传编程设计分类器的新方法》。载于：*IEEE Transactions on Evolutionary Computation* 8.2（2004），第 183–196 页。[62] Alexander Topchy、William F Punch 等。《基于数值叶节点值局部梯度搜索的快速遗传编程》。载于：遗传与进化计算会议论文集（*GECCO-2001*）。第 155162 卷。Morgan Kaufmann，旧金山，加利福尼亚州。2001。[63] Francisco Fernandez、Marco Tomassini 和 Leonardo Vanneschi。《多种群遗传编程的实证研究》。载于：*Genetic Programming and Evolvable Machines* 4（2003），第 21–51 页。[64] Francisco Herrera 和 Manuel Lozano。《渐进式分布式实数编码遗传算法》。载于：*IEEE Transactions on Evolutionary Computation* 4.1（2000），第 43–63 页。[65] Patrick AK Reinbold 等。《通过物理约束符号回归，从含噪、不完整的高维实验数据中进行稳健学习》。载于：*Nature Communications* 12.1（2021），第 3219 页。[66] Toshio Yamada 和 Shinzo Watanabe。《关于随机微分方程解的唯一性》。载于：*Journal of Mathematics of Kyoto University* 11.1（1971），第 155–167 页。[67] Alexander P Browning 等。《系统生物学中随机微分方程模型的可辨识性分析》。载于：*Journal of the Royal Society Interface* 17.173（2020）。[68] Ali Hasan 等。《识别潜在随机微分方程》。载于：*IEEE Transactions on Signal Processing* 70（2021），第 89–104 页。

附录

A 环境详情

双阱问题 随机双阱描述了一维粒子的位置，其

$$\mathrm{d}x = (x - x^3)\mathrm{d}t + g(x)\mathrm{d}w. \tag{9}$$

在确定性情形下，双阱在 -1 和 1 处有两个吸引不动点，但加入额外噪声后，粒子会在这两个不动点之间移动。x 的初始条件从 $\mathcal{N}(0,\ 0.1)$ 中采样，轨迹将模拟至 $T = 50$。双阱通过三种不同的扩散项来表征：加性噪声 $g(x) = 0.5$、线性乘性噪声 $g(x) = 0.5x$ 以及非线性乘性噪声 $g(x) = 0.5(1 + x^2)$。数据集包含八条轨迹。

范德波尔振子 范德波尔振子是一种围绕零点振荡的非线性二维系统。随机动力学定义为

$$\mathrm{d}\mathbf{x}(t) = \mathrm{d}\begin{bmatrix} v \\ w \end{bmatrix} = \begin{bmatrix} w \\ \mu w(1 - v^2) - v \end{bmatrix}\mathrm{d}t + \sigma \begin{bmatrix} 0 & 0 \\ 0 & 1 + \frac{1}{2}v^2 \end{bmatrix}\mathrm{d}W \tag{10}$$

其中 $\mu = 1.0$ 和 $\sigma = 0.2$。两个维度的初始条件均从 $\mathcal{N}(0,\ 1)$ 中采样。时间序列积分至 $T = 50$，并收集了八条轨迹。

R¨ossler 吸引子 R¨ossler 吸引子是一个三维非线性系统，表现出混沌行为。其随机动力学由以下方程给出：

$$\mathrm{d}\mathbf{x}(t) = \mathrm{d}\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} -(y - z) \\ x + ay \\ b + z(x + c) \end{bmatrix}\mathrm{d}t + \sigma \begin{bmatrix} x & 0 & 0 \\ 0 & y & 0 \\ 0 & 0 & z \end{bmatrix}\mathrm{d}W, \tag{11}$$

其中 $a = b = 0.2$、$c = 5.7$ 和 $\sigma = 0.1$。$\mathbf{x}(0)$ 从 $\mathcal{N}(0, I)$ 中采样，$T = 50$，批量大小设为八。

Lorenz96 模型 Lorenz96 模型是一个 N 维系统，其方程会根据变量数量系统性地变化。变量 x_i 的一般更新形式加入扩散项后，得到：

$$\mathrm{d}x_i = ((x_{i+1} - x_{i-2})x_{i-1} - x_i + F)\mathrm{d}t + \sigma x_i \mathrm{d}W_i \tag{12}$$

其中 $F = 4$、$\sigma = 0.2$ 和 $x_0 = x_N$。模型分别使用五个和十个变量进行模拟，每个变量的初始值从 $\mathcal{N}(0, 0.1)$ 中采样，并对模型进行积分，直至 $T = 25$，生成八条轨迹。由于每个变量的方程都具有相同的一般结构，实验中仅对 x_1 的方程应用这些方法进行学习。

洛特卡-沃尔泰拉模型 洛特卡-沃尔泰拉模型描述了猎物和捕食者种群之间的相互作用。其动力学定义为

$$\mathrm{d}\mathbf{x}(t) = \mathrm{d}\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \alpha x - \beta xy \\ \delta xy - \gamma y \end{bmatrix}\mathrm{d}t + \sigma \begin{bmatrix} x & 0 \\ 0 & y \end{bmatrix}\mathrm{d}W. \tag{13}$$

在我们的实验中，我们使用 $\alpha = 1.1$, $\beta = 0.4$, $\delta = 0.1$, $\gamma = 0.4$, $\sigma = 0.2$ 和 $T = 50$。初始条件从 $\mathcal{U}(5,10)$ 中采样，数据集由八条轨迹组成。

Fisher-KPP方程 Fisher-KPP方程是一种用于模拟种群增长的一维偏微分方程。加入乘性噪声后的随机动力学定义为

$$\mathrm{d}u(t, x) = (Du_{xx} + u(1 - u))\,\mathrm{d}t + \sigma u \mathrm{d}W, \tag{14}$$

其中我们使用 $D = 1$, $\sigma = 0.1$ 并采用 $u_{xx} = \frac{\partial^2 u}{\partial x^2}$。此外，$x \in [0, 20]$ 表示坐标空间，$t \in [0, 1]$ 是时间跨度，而 $u(0,\ x) = 1 + \sin(x)$ 是初始条件。边界条件定义为$u(t,\ 0) = u(t,\ 20)$。使用有限差分法，将坐标空间离散为 64 个点，时间采样率为 0.02。总共模拟了四条轨迹。

热传递 我们考虑存在乘性噪声且无反应的二维热传导。噪声分量由相对于坐标空间的梯度进行缩放。因此，动力学遵循

$$\mathrm{d}u(t,x,y) = D(u_{xx} + u_{yy})\mathrm{d}t + (\alpha u_x + \beta u_y)\mathrm{d}W. \tag{15}$$

此处， $D = 0.1$、 $\alpha = 1$和 $\beta = 1$。坐标空间由 $x \in [0, 4]$、 $y \in [0, 4]$定义，两个维度均被离散化为16个点。此外， $t \in [0, 1]$ 的采样率为0.02，初始分布根据 $u(0, x, y) = 0.2 \sin\left(\frac{\pi}{2}x\right) \cos\left(\frac{\pi}{2}y\right)$生成。边界条件为 $u(t, 0, y) = u(t, 4, y)$ 以及 $u(t, x, 0) = u(t, x, 4)$。 同样，¯为该数据集模拟了四条轨迹。

B 实验细节

超 参数在每次实验中由这些方法所使用的取值见表3和表4 .

表3：克拉默斯-莫亚尔展开与稀疏回归算法的超参数集。在每次运行中，均进行超参数优化，以最大化恢复的准确率。超参数的定义见第3.3.3节。

环境	b	β	α	λ	d
双阱问题	[5, 10, 25, 50, 100]	[1, 10, 20]	[0.001、0.01、0.1]	[0.05，0.1，0.2]	3
范德波尔振子	[5, 10, 25, 50, 100]	[1, 10, 20]	[0.001，0.01，0.1]	[0.05、0.1、0.2]	3
R¨ossler 吸引子	[5, 10, 25]	[1, 10, 20]	[0.001、0.01、0.1]	[0.05、0.1、0.2]	2
五维 Lorenz96 模型	[5, 10, 15]	[1, 10, 20]	[0.001、0.01、0.1]	[0.05，0.1，0.2]	2
十维 Lorenz96 模型	[4]	[1, 10, 20]	[0.001, 0.01, 0.1]	[0.05，0.1，0.2]	2
二十维 Lorenz96 模型	[2]	[1, 10, 20]	[0.001，0.01，0.1]	[0.05、0.1、0.2]	2
洛特卡-沃尔泰拉模型	[5, 10, 25, 50, 100]	[1, 10, 20]	[0.001、0.01、0.1]	[0.05、0.1、0.2]	3

表4：用于识别常微分方程和随机微分方程的遗传编程算法超参数。这些超参数在第3.2节中定义。

环境	p	g	m	s	p^*	g^*	η
双阱问题	500	50	5	15	100	15	0.1
范德波尔振子	500	50	5	15	100	15	0.1
R¨ossler 吸引子	500	50	5	15	100	15	0.1
五维 Lorenz96 模型	1000	100	5	20	200	15	0.1
十维 Lorenz96 模型	2000	200	5	20	200	15	0.1
二十维 Lorenz96 模型	2000	200	5	20	200	15	0.1
洛特卡-沃尔泰拉模型	500	50	5	15	100	15	0.1
随机偏微分方程	1000	50	5	15	200	15	0.1
多步遗传编程							
洛特卡-沃尔泰拉模型（$\tau = 0.2$）	2000	100	5	15	500	50	0.1
洛特卡-沃尔泰拉模型（$\tau = 0.5$）	3000	100	5	15	500	50	0.1