fintech-kMC：用于机器学习系统设计与测试的金融平台代理模拟

Isaac Tamblyn, [1,2,3] **Tengkai Yu,** [1,4] **Ian Benlolo** [1,3]1 Cash App、Block，多伦多，安大略省 M5A 1J7，加拿大[2] Vector人工智能研究所，多伦多，安大略省 M5G 1M1，加拿大[3] 渥太华大学物理系，渥太华，安大略省 K1N 6N5，加拿大[4] 维多利亚大学计算机科学系，维多利亚，不列颠哥伦比亚省 V8P 5C2，加拿大
itamblyn@squareup.com

arXiv:2301.01807v1 [cs.LG] 4 Jan 2023

摘要

我们介绍模拟工具 fintech-kMC，该工具旨在生成用于机器学习模型开发与测试的合成数据。fintech-kMC 是一种基于智能体的模型，由动力学蒙特卡罗（又称连续时间蒙特卡罗）引擎驱动，用于模拟客户使用在线数字金融平台的行为。该工具能够以可解释、可复现且逼真的方式生成合成数据，可用于验证和测试将在面向真实客户的金融应用中使用的人工智能/机器学习模型及管道。

发表于*AAAI-23*桥接项目：金融服务人工智能，华盛顿特区，*2023*年*2*月*7*日至*8*日

引言

模拟数据有助于机器学习（ML）工作，以可重复、可验证的方式开发和测试算法。我们注意到，尽管历史上人们曾因仿真到现实的挑战而对模拟数据在机器学习应用中的效用有所争论（H¨ofer 等人，2020），但无论模拟数据是否有助于训练生产模型，它们在模型设计、单元测试和验证流程的初始阶段都极具价值。由于可以绝对确定真实标签，并对数据生成条件进行精细控制，因此能够消除嘈杂的真实世界数据所带来的初始复杂性和可变性，从而营造干净的开发环境，以验证机器学习管道和模型的功能与逻辑。当所研究的系统具有严重的类别不平衡、罕见事件和显著的标签噪声时，模拟尤其有价值。所有这些棘手的建模问题在金融科技机器学习中都很常见。

本文的组织结构如下。我们简要介绍合成数据集生成的可能方法，然后概述动力学蒙特卡罗（kMC）算法。接下来，我们说明 fintech-kMC 中定义智能体的属性，并讨论决定其行为的速率常数的处理方式。我们还提供仿真输出示例，以及一些利用这些输出的简单监督式机器学习测试。我们讨论该方法的局限性，最后作出总结。我们的主要贡献如下：

- 介绍了一种由动力学蒙特卡罗驱动的全新基于智能体的模拟工具
- 详细介绍了该工具，包括其设计、建模能力和输出格式
- 以一个预测模型为典型示例，展示了如何训练该模型来检测“不良行为者”；该模型使用我们的合成数据，是一种机器学习模型

合成数据生成协议有多种类型，包括简单的启发式方法（例如，如果某个输入特征取值为 X，则将输出设为标签 Y）、确定性模拟（Papageorgiou 和 Paskov，1999年）或随机模拟（Wang，2012年），以及利用真实系统的样本数据流进行训练的生成模型（Goodfellow 等，2014年；Albergo、Kanwar 和 Shanahan，2019年）。我们认为，生成模型虽然或许最能模仿真实世界数据集，但也会带来一系列全新的问题，因为它们本身依赖于高质量的数据管道、训练基础设施、监控等。随机模拟在逼真程度上足以发现机器学习工作流中的实现错误，同时又足够简单，不会引入这些错误，因此是理想的折中方案。

动力学蒙特卡罗（kMC）

kMC 长期以来一直用于物理科学领域，用于描述各种系统的微观时间演化，例如催化表面上的分子（Stamatakis 和 Vlachos，2012年）、自组装与生长（Ghosh 等，2021年）以及辐射损伤（Voter，2007）。该方法也已成功应用于社会科学领域，用于模拟大型在线社交图的结构和信息流动，例如 Twitter（Ryczko 等，2017年）。

与其他一些基于蒙特卡罗的模拟技术（最著名的是马尔可夫链蒙特卡罗，MCMC）不同，kMC 是一种无拒绝算法。每个模拟步骤都会发生一个事件。此外，我们注意到，模拟步长 Δt 并非固定不变。假设某个系统中只有一种事件类型，其发生速率为特征速率 r，那么模拟时间中每一步之间的间隔大约为

版权所有 © 2023，美国人工智能促进协会（www.aaai.org）。保留所有权利。

算法 1：动力学蒙特卡洛算法

参数：智能体数量、最大时间等

输出：日志文件

1：**当** 模拟时间 < 最大时间**时**

2: 更新速率()

3: 速率：= $[r_1, r_2, ..., r_n]$ {事件 n 的速率数组}

4: 累积数组 =cumsum(rates/sum(rates))

5: u_1 := rand() {获取随机数 $u_1 \in (0, 1]$}

6: 事件：= 二分搜索(累积数组, u_1)

7: 执行事件(event)

8: u_2 := $rand()$ {获取随机数 $u_2 \in (0, 1]$, 向上-日期模拟时间}

9: 模拟时间：= 模拟时间 $-ln(u_2)/R$

10: 将事件信息写入日志文件

11: **结束循环**

$\Delta t = r^{-1}$。在存在多种可能事件（及其速率）的模拟中，模拟会根据系统的总有效速率（例如 $R^{-1} = r_1^{-1} + r_2^{-1} + ...$）向前推进。在实践中，这意味着可能发生的事件总数越多，连续事件之间的时间间隔就越短。现实世界中的系统正是如此。考虑一个处于早期阶段、恰有 1 位客户的金融平台。如果这位客户每周存款一次，那么一个月后，交易日志中将大约有 4 条记录，间隔约为一周。如果客户数量增加 100 倍（且他们存款的时间随机分布），那么同样一个月后，我们预计会有 400 条记录，更新之间的平均时间间隔将缩短 100 倍。动力学蒙特卡洛模拟能够模拟这种行为，并允许事件以不规则的时间间隔发生，且 Δt 的值会随着模拟推进而变化。

我们注意到，在上面的示例中，我们假设客户存款的速率彼此不相关，且发生时间是随机的。一般来说，这当然无法保证为真，尤其是与客户工资支票、租金、贷款还款等相关的付款，它们往往会在每月或每年的特定时间发生。有关我们如何处理这一问题的进一步讨论，请参见第**动态速率**节。

免责声明：在以下讨论中，出于教学目的，我们举例说明了可能在金融科技平台上发生的操作或规则。这些纯属假设，并非旨在描述真实金融科技平台所建议采用的一套政策。此类平台受大量政策和法规约束（尤其是在身份要求、客户年龄限制等方面）。虽然 fintech-kMC 可以根据需要进行调整，以纳入这些具体要求，但在本讨论中，我们不处理这种复杂性或地区特定限制。

智能体

在 fintech-kMC 中，客户被表示为智能体。目前，我们的模型中包含 2 种不同类型的智能体：

- 个人客户：这是典型模拟中最常见的代理类型，代表个人客户。在这种类型中，存在若干不同的原型（参见第**代理类型**节）。
- 企业：代表组织或企业，这些智能体只能执行受限的一组操作（例如，它们不会支付房租、偿还贷款或购买比特币）。有关智能体操作的进一步讨论，请参见**“可用操作”**一节。目前，我们这里只有一种原型。

个人或企业账户也可能是“不良行为者”（例如从事信用卡欺诈、消费者诈骗等）。这些智能体采取的行动主要与正标签相关。这些智能体与正常、健康客户之间细微的行为差异，正是机器学习算法能够进行预测的原因。

静态、动态和触发速率

fintech-kMC 支持同时使用静态速率和动态速率。新客户加入平台的速率就是静态速率的一个例子。在我们通常考虑的时间尺度内，可以假设该速率保持不变。相反，动态速率可能会在模拟过程中发生变化。有些动态速率可以连续变化，而另一些则会根据预先设定的一组规则，或在达到特定模拟状态时被激活（例如，某个智能体成功完成身份验证）。

动态速率 动态速率计划可以基于时间设定（例如，模拟中的绝对世界时钟时间，或相对时间，如客户成为平台用户的天数）。定期事件（如工资支票存入）也可以通过这种方式纳入。对于工资支票、租金等已安排的事件，我们会在对应付款日期（例如每月1日）将所有受影响智能体的速率设为一个非常高的值。这意味着该事件被选中的概率会显著高于其他任何过程。事件一发生，我们就将该速率重置为0。通过这种方法，可以自然地将这类“保证事件”纳入kMC方法。

管理动态速率的其他规则可以基于智能体的当前状态。例如，只要其账户余额高于某个阈值且已通过身份验证流程，智能体就可能以每 ≈ 1/周一次的频率购买比特币（BTC）。类似地，智能体可以通过点对点（p2p转账）付款进行每月一次的贷款还款，直到余额为*0*。允许灵活、动态地调整决定智能体行为的速率常数，有助于以更真实的方式对人类行为进行建模。

可执行操作 智能体通过在模拟中采取的行动相互交互，并随时间不断演化。此类行动有很多种。

- 存入现金：资金转入账户。可选的转账金额范围取决于该智能体是否已完成身份验证。
- 取出现金：资金从账户中扣除。可选的转账金额范围取决于代理是否已完成身份验证。

• 点对点转账：资金从账户中扣除，并通过点对点支付转给另一位代理。

• 身份验证：代理成功通过身份验证后，即可“解锁”其他操作（具体来说是比特币买入），并改变允许存入和取出的最高现金金额。客户会持续尝试此操作，直至成功；不良行为者成功的概率较低。

• 比特币买入：代理通过身份验证操作成功验证身份后，即可购买比特币。此操作会减少其现金余额并增加其比特币余额。

• 支付房租：代理若需支付房租，将每隔 30 天支付一次。

• 存入工资支票：代理收到工资支票后，每隔 14 天会将其存入账户。

• 偿还贷款：贷款还款按 7 天为一个周期进行（每次偿还原始贷款金额的固定比例），直至贷款余额归零。

请注意，上述任何涉及资金流动的操作都只有在余额充足时才能进行（账户余额不能为负）。

重要的是，为了体现真实客户在金融科技平台上观察到的客户行为多样性，每个代理所采用的各项速率并不相同，而是从某个统计分布中随机抽取。我们还采用了代理类型这一概念。

代理类型 虽然人与人各不相同，但在建模和商业分析中，采用客户类型这一概念通常很有用。通过了解平台上客户的行为和动机，我们可以将他们划分为行动和（可能的）动机相似的群体。在 fintech-kMC 中，我们运用这一理念定义不同的代理类型，从而产生各异的行为。

例如，加密货币爱好者这一代理类型，其比特币购买速率将从正态分布 $N(\mu,\ \sigma)$ 中抽取，其中 $\mu = 2$ 和 $\sigma = 1$（单位为天$^{-1}$）；而加密货币怀疑者的该速率则设为零。μ 和 σ 的具体数值是可调整的输入参数，建模者可利用它们尽可能准确地模拟其希望模拟的特定金融科技平台的细节。同样，挥金如土的客户，其平均存入现金金额可能显著高于同类客户。定义代理类型会决定速率的分布以及代理在模拟过程中使用的典型资金流动数值范围。这使得

用于在数据集中创建具有实质意义的多样化智能体。属于同一原型的智能体，其行为不会完全相同，而是从相同的统计分布中抽取其偏好和倾向。

输出

fintech-kMC 以结构化文件的形式提供仿真输出，这种格式对于从事数据科学和机器学习领域的实践者来说应该十分熟悉。这些 CSV文件由所有事件的带时间戳的记录组成，并包含生成智能体、所执行的操作以及与之相关的任何值等信息。例如，当某个智能体完成身份验证时，报告的信息如下：时间、发起令牌、操作、值。例如，记录可能如下所示：2022-09-03 04:50:05.00, C 83dhpqzz, 身份验证, True。点对点货币兑换还包含一个与接收令牌对应的附加字段。例如：2022-11-14 04:50:05.00, C b6589f, 点对点转出, 2000, C 83dhpqzz。

这里的发起令牌是代理唯一数值 ID 的哈希值。通常，现实系统会使用这类令牌，在不泄露任何个人身份信息 (PII) 数据的情况下，唯一标识特定客户。我们采用这种格式，尽可能贴近现实系统原始日志文件中的数据形式。同样需要注意的是，最终数据框中存在必须处理的空值（这在真实数据集中很常见），因为不同事件可能具有不同数量的属性。我们有意以这种格式提供输出数据，其中操作与代理交错排列，以便验证和测试机器学习流程中使用的预处理步骤。

测试机器学习模型

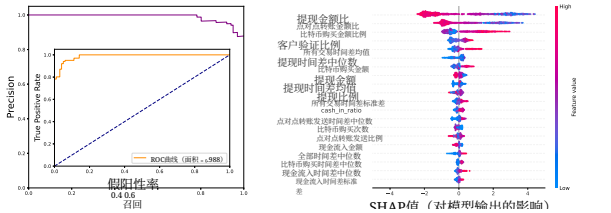


图1：a) 我们的简单示例二元分类器模型的精确率-召回率曲线和受试者工作特征曲线（ROC，插图）；b) SHAP（Lundberg 和 Lee，2017年）分析。模型训练数据和验证数据由 fintech-kMC 工具生成。用于训练模型的完整特征列表见在线补充信息。

为了展示 fintech-kMC 与机器学习工具结合的实用性，我们在此展示一项简化实验的结果，用于检测并区分不良行为者与普通客户。在这项特定模拟中，我们将不良行为者的身份验证成功率设得较低（50%，而普通客户为 75% ），并设定其点对点资金转账行为略有不同（平均转账金额为 5 ± 3 vs 8 ± 3，且只有余额高于 15 ± 3 vs 30 ± 3时才愿意进行此类转账）。这里，$\pm$ 表示用于抽样并确定具体转账金额的正态分布的标准差。

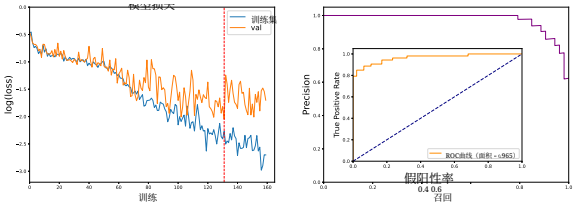


图 2：a）训练集和验证集的长短期记忆网络损失曲线；b）模型的精确率-召回率曲线和 ROC 曲线（插图）（在红色竖线标示的训练轮次进行评估）。超参数列于在线补充信息中。

我们创建了一个由 1000 个智能体组成的群体，其中 50% 是不良行为者，其余则是普通的正常客户。我们的模拟涵盖了 ≈ 8 天的时间段；根据所用的各种速率常数，这相当于每个智能体约执行 100 次操作。我们约定，正标签代表有害行为（例如诈骗、账户接管或其他我们希望识别并预防的活动）。为了训练模型，我们构建了一系列特征，详见在线补充信息。我们采用 80%-10%-10% 的训练-验证-测试集划分，并在数据上训练了一个 XGBoost 二元分类器（Chen 和 Guestrin, 2016），见图 1；此外，还训练了一个长短期记忆网络（Hochreiter 和 Schmidhuber，1997），见图 2（完整的模型训练超参数详见在线补充信息）。

方法的局限性

- 引入新类型的操作需要进行更多软件实现。我们无法直接从真实系统的数据流中学习。
- 控制文件中的速率常数概率分布由用户手动设置。若要使其适用于特定数据集，必须在单独的循环中对其进行优化。理论上，这可以通过自动微分实现，但目前尚未包含此类功能。
- 由于使用该工具进行模拟的智能体数量通常较少（例如 1k），目前尚未能很好地描述大规模网络效应。原则上，可以使用基于代理的动力学蒙特卡洛来模拟更大的“世界”（例如 1-10M），但为求简便，我们尚未实现实现此类性能所需的常见优化或并行化技术。
- 当前实现采用 Python 编写。这有利于原型开发和添加新特征，但也导致工具的性能未能达到应有水平。
- 在当前实现中，模拟日内的任何时刻都可以发生操作；而在真实世界数据中，客户往往会表现出日趋势。不过，这并非方法论上的根本限制。
- 目前，我们在客户层级而非单笔交易层级分配正标签。我们正在努力解除这一限制。
- 不应将该模型生成的数据视为生产模型的数据来源。

结论

fintech-kMC 是一种基于智能体的模型，可模拟在线数字金融平台客户的行为。该工具实现了此类平台上的许多典型操作，例如点对点资金转账、身份验证和加密货币购买。借助动力学蒙特卡罗，事件会以符合现实的时间尺度发生，并可能存在有意义的先后依赖关系。该工具生成的数据可用于以可控且可重复的方式测试和验证机器学习工作流。

参考文献

Albergo, M. S.; Kanwar, G.; and Shanahan, P. E. 2019年. 用于格点场论中马尔可夫链蒙特卡罗的基于流的生成模型。*Phys.Rev.D*，100：034515。

Chen, T.; and Guestrin, C. 2016年。XGBoost：可扩展的树提升系统。载于第*22*届 *ACM SIGKDD* 知识发现与数据挖掘国际会议论文集，KDD ’16，785–794。美国纽约州纽约市：计算机协会。ISBN 9781450342322。

Ghosh, P.; Gupta, N.; Dhankhar, M.; and Ranganathan, M. 2021年。Si(001)上锗岛自组织的动力学蒙特卡罗模拟。《物理化学化学物理》，23：19022–19031。

Goodfellow, I. J.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; and Bengio, Y. 2014年。生成对抗网络。。

Hochreiter,S.; and Schmidhuber,J. 1997. 长短期记忆网络. 《神经计算》，9(8): 1735–1780.

H¨ofer, S.; Bekris, K.; Handa, A.; Gamboa, J. C.; Golemo, F.; Mozifian, M.; Atkeson, C.; Fox, D.; Goldberg, K.; Leonard, J.; Liu, C. K.; Peters, J.; Song, S.; Welinder, P.; and White, M. 2020. 关于机器人学仿真到现实迁移的观点：R:SS 2020研讨会综述。.

Lundberg, S.; and Lee, S.-I. 2017年. 解释模型预测的统一方法。.

Papageorgiou, A.; and Paskov, S. H. 1999年. 面向风险管理的确定性模拟. 《投资组合管理杂志》,25(5): 122–127.

Ryczko, K.; Domurad, A.; Buhagiar, N.; 和 Tamblyn, I. 2017年。Hashkat：在线社交网络的大规模模拟。《社会网络分析与挖掘》，7(1): 4.

Stamatakis, M.; 和 Vlachos, D. G. 2012年。通过动力学蒙特卡罗模拟揭示催化反应的复杂性：现状与前沿。*ACS*催化，2(12)：2648–2663。Voter, A. F. 2007。动力学蒙特卡罗方法简介。载于 Sickafus, K. E.; Kotomin, E. A.; 和 Uberu- aga, B. P. 编，《固体中的辐射效应》，1–23。多德雷赫特：施普林格荷兰。ISBN 978-1-4020-5295-8。Wang, H. 2012。《应用于金融的蒙特卡罗模拟》。CRC出版社。

补充信息

我们的长短期记忆网络和XGBoost模型的架构细节见表 S1和表 S2。我们还提供了用于训练基于特征的机器学习模型的手工设计特征列表（表 S3）。

基分 _	0.5
提升器	梯度提升树
按层列采样 _	1
按节点列采样	1
按树列采样 _	1
伽马参数	0
学习率 _	0.1
最大增量步长	0
最大深度	3
最小子节点权重	1
缺失	None
估计器数量	100
线程数	1
目标函数	二分类逻辑回归
L1 正则化系数	0
L2 正则化系数	1
正类权重缩放系数	1
seed	0
子采样	1
详细程度	1
树构建方法	hist

表1：用于训练我们的 XGBoost（Chen 和 Guestrin, 2016）模型的参数

层	Type	输出单元	丢弃率	返回序列
0	双向长短期记忆网络	64	0.2	True
1	双向长短期记忆网络	64	0.2	True
2	双向长短期记忆网络	64	0.2	错误
3	全连接层	1		

表2：Architex 构（层定义），我们的长短期记忆网络（Hochreiter 和 Schmidhuber，1997）神经网络模 el

事件总数
现金存入次数
客户验证次数
现金取出次数
点对点转账发送次数
比特币购买次数
现金存入比例
客户验证比例
现金取出比例
点对点转账发送比例
比特币购买比例
现金存入金额
客户验证金额
现金取出金额
点对点转账金额
比特币购买金额
现金存入金额比率
客户验证金额比率
现金取出金额比率
点对点转账金额比率
比特币购买金额比率
所有时间差均值
所有时间差中位数
所有时间差标准差
现金存入时间差均值
现金存入时间差中位数
现金存入时间差标准差
客户验证时间差均值
客户验证时间差中位数
客户验证时间差标准差
现金取出时间差均值
现金取出时间差中位数
现金取出时间差标准差
点对点转出时间差均值
点对点转出时间差中位数
点对点转出时间差标准差
比特币购买时间差均值
时间差中位数 比特币买入
时间差标准差 比特币买入

表 3：我们创建并用作 XGBoost 模型输入的手工设计特征