arXiv:2602.04270v1 [cs.LG] 4 Feb 2026

跨类别标签多重整合组件识别（MILCCI）

Noga Mudrik [1] **Yuxi Chen** [2] **Gal Mishne** [3] **Adam S. Charles** [1]

摘要

许多领域通过重复测量（“试次”）收集大规模时序数据，每个试次都标注有一组涵盖多个类别的元数据变量。例如，神经科学研究中的一个试次可能关联类别（a）的一个取值：任务难度，以及类别（b）的一个取值：动物选择。时间序列分析面临的一项关键挑战，是理解这些标签如何编码在多试次观测中，并区分不同类别中各个标签条目的独特作用。本文提出一种新颖的数据驱动方法 MILCCI，它能够：i）识别数据中潜在的可解释成分；ii）捕捉跨试次变异；iii）整合标签信息，以理解数据对各类别的表征。MILCCI 扩展了一种稀疏逐试次分解方法，该方法利用各类别内部的标签相似性，在成分组成中进行细微的、由标签驱动的跨试次调整，并区分各类别的贡献。MILCCI 还学习每个成分对应的时间轨迹，该轨迹随时间在每个试次内演变，并可在不同试次间灵活变化。我们通过合成示例和真实世界示例展示 MILCCI 的性能，其中包括投票模式、在线页面浏览趋势以及神经元记录。

1. 引言

理解高维、随时间演变的系统（例如大脑）的一种关键方法，是分析来自多个重复观测（以下简称“试次”）的时间序列数据。每个试次通常都带有一组实验元数据变量。此类元数据往往涉及多个类别；例如，神经元记录中的每个试次都可以标注类别 (a) 任务难度和类别 (b) 动物选择中的一个属性；天气测量中的每个试次可以是一整天的温度时间序列，并标注 (a) 城市、(b) 湿度水平和 (c) 降水。因此，我们将一个试次的标签称为其各类别取值组成的元组，例如“（任务简单，选择正确）”或“（纽约州，湿度 90%，降雪 1 英寸）”。值得注意的是，不同试次的标签可能相似，也可能不同。当试次之间某个类别发生变化（例如任务难度：简单与困难）时，相应的标签条目 会发生变化，而对应其他类别的标签条目 可能保持不变，也可能同时变化。

面对这类多试次、多标签（“多路”）数据，要理解观测值如何随标签空间变化，会受到数据高维性和试次间变异的影响。一个实用的方法是在低维潜在空间中分析数据（Ma &Zhu，2013），此时标签相关结构更易于解读。在这个空间中，活动可以用一组数量较少的成分来描述——这些单元能够捕捉跨试次和标签变化的主要来源。用于分析多路数据的现有降维方法通常会将一个大型张量分解为这样的成分（Harsh-man 等人，1970年），但它们通常忽略试次标签，并要求数据结构满足特定约束（例如，试次长度相同）。另一种方法是分别对每个试次进行分解，这样可以适应不同的试次结构，但会损失有关跨试次关系的信息。

因此，需要开发新型灵活且可解释的方法，以：(1) 发掘高维多路数据中的潜在结构；(2) 揭示该结构如何捕获标签信息；以及 (3) 分离每个类别的影响。这反过来又要求利用标签所捕获的试次间关系，并理解这些关系如何支配观测值。

在本文中，我们提出了 MILCCI，这是一种新颖的方法，用于揭示多路时间序列数据的潜在结构，并从结构和时间两个层面解析多类别标签如何嵌入其中。我们的贡献

[1]生物医学工程、Kavli NDI、数据科学数学研究所、成像科学中心、约翰斯·霍普金斯大学，巴尔的摩，马里兰州 [2]约翰斯·霍普金斯大学神经科学系，巴尔的摩，马里兰州 [3]加州大学圣迭戈分校 Halıcıo˘glu 数据科学研究所，圣迭戈，加利福尼亚州。联系人：Noga Mu-drik <nmudrik1@jhu.edu>、Gal Mishne <gmishne@ucsd.edu>、Adam S. Charles <adamsc@jhu.edu>。

预印本。*2026*年*2*月*5*日。

包括：

• 我们提出了 MILCCI，这是一种灵活的模型，能够发现多路数据潜在的、可解释的稀疏成分，并揭示这些成分如何捕捉多样的标签类别。

• 我们识别出能够捕捉试次间标签驱动变异的成分，并追踪这些成分的激活在单个试次内如何演变，从而涵盖试次间变异的全部范围。

• 我们在合成数据上验证了 MILCCI，结果表明，与其他方法相比，它能更准确地恢复真实成分。

•我们展示了 MILCCI 在真实数据中发掘可解释的有意义模式的能力，包括发现与已知事件相符的美国各州投票趋势、反映语言和设备的在线活动模式，以及支持多区域记录中决策的神经元集群。

2. 相关工作

分析多路数据的简单方法是对每个试次分别进行降维，或将多个试次堆叠成一个矩阵后，对所有试次联合降维。这可以通过线性矩阵分解来实现，例如主成分分析、独立成分分析（Hyvarinen 等人，2001）、非负矩阵分解（Lee 与 Seung，1999），也可以采用提升可解释性的稀疏分解方法（例如稀疏主成分分析（Zou 等人，2006）），或采用非线性嵌入方法（例如 t-SNE（Maaten 与 Hinton，2008））。然而，逐试次分析会忽略跨试次关系，而用单一映射分析所有试次，则会忽略内部结构的试次间变异。

解混主成分分析（dPCA）（Kobak 等人，2016）能够将与任务相关的神经变异分离为低维成分，但无法处理缺失数据、不同的试次时长以及变化的试次采样率，从而妨碍了异质试次之间的对齐。 Mudrik 等人 (2024) 最近提出了一种统一的跨试次模型，可识别多阵列数据中编码标签信息的构建模块；然而，该方法仅处理标签变化的单一维度，因此无法区分多个类别在试次间共同变化或分别变化所产生的影响。TDR 及其扩展（Mante 等人，2013；Aoi& Pil-low，2018年）通过对固定矩阵进行逐试次标量重加权来捕获多类别标签，但假设跨试次变异仅源于对固定时间信号的线性重加权，且并非专为捕获共享同一标签的试次间变异而设计。

张量分解（TF），例如 PARAFAC（Harshman 等人，1970年）和 HOSVD（Lathauwer 等人，2000），通过将试次作为多维数组中的额外数据维度来处理，从而超越单个试次。然而，现有的 TF 方法，包括那些结合了高斯过程（Till- inghast 等人，2020；Xu 等人，2012；Zhe 等人，2016）或动态信息（Wang 与 Zhe，2022）的方法，并非为区分标签驱动的变异性与其他变异性来源而设计，且往往会产生难以解释的成分。SliceTCA（Pellegrino 等人，2024）扩展了张量分解，在同一数据集中同时分离神经、试次和时间协变性类别，使成分能够捕捉不同类型神经变异性中的结构。然而，SliceTCA 未纳入显式标签信息，并强制要求数据具有张量结构。Chen等人（2015）基于元数据信息实现了跨试次表征的灵活性，但其成分正交性假设使模型无法捕捉相关或部分重叠的模式，限制了模型的表达能力。

与上述方法不同，MILCCI（1）能够识别跨会话变化的多试次、多标签数据的结构；（2）能够捕捉试次间变异，包括同一标签重复测量时的变异；（3）能够分离数据中各类别的编码方式。

表*1.* 主要记号。

记号	描述
m	试次编号
(k)$\in C$	类别
$L^{(m)} = (L^{(m)}{}_{(a)} L^{(m)}{}_{(b)} \ldots)$	试次 m 的标签
$\boldsymbol{Y}^{(m)} \in \mathbb{R}^{N \times T^{(m)}}$	试次 m 的观测值
$\boldsymbol{\Phi}^{(m)} \in \mathbb{R}^{P \times T^{(m)}}$	试次 m 的轨迹
$\mathcal{G}^{(\mathrm{k})}$	类别 (k) 的轨迹索引
$\mathcal{A}^{(\mathrm{k})} \in \mathbb{R}^{N \times p^{(\mathrm{k})} \times \lvert(\mathrm{k})\rvert}$	类别 (k) 的成分
$\boldsymbol{A}^{(L^{(m)})} \in \mathbb{R}^{N \times P}$	试次 m 的成分

3. 我们的成分识别方法

问题表述：

设 $\boldsymbol{Y} = \{\boldsymbol{Y}^{(m)}\}_{m=1}^{M}$ 为一组 M 时间序列（试次），其中每个 $\boldsymbol{Y}_{(m)} \in \mathbb{R}^{N \times T^{(m)}}$ 表示从 N 个通道在 $T^{(m)}$ 个时间点上的测量值（图 1、**I**）。我们假设各试次中的 N 通道身份保持不变，而试次时长 $T^{(m)}$可以有所不同。每个试次 $\boldsymbol{Y}_{(m)}$ 均在属于以下类别的试次相关元数据变量条件下进行观测： $C{:}:=$ {a、b.....、f}（例如，（a）任务难度、（b）动物的选择等），其中 $|C|$表示类别数量。我们将试次 m 的标签定义为包含其 $|C|$ 个元数据变量的元组 $L^{(m)}$，其中每个标签条目 $L_i^{(m)}$ 都是该试次中第 i 个类别的取值（例如， $L^{(m)} =$ （任务难度：容易，选择：正确））。值得注意的是，不同试次可能具有相同的标签、部分重叠的标签，或完全不同的标签。我们的目标是理解这些标签如何通过其底层成分和轨迹编码在多路观测值中。

一种简约的建模策略是，将每个试次 m 中的观测值 $\boldsymbol{Y}^{(m)}$ 建模为由一小组核心成分 $\boldsymbol{A}^{(m)} \in \mathbb{R}^{N \times P}$ 线性生成，并由相应的时序活动 $\boldsymbol{\Phi}^{(m)} \in \mathbb{R}^{P \times T^{(m)}}$加以表征 P，使得

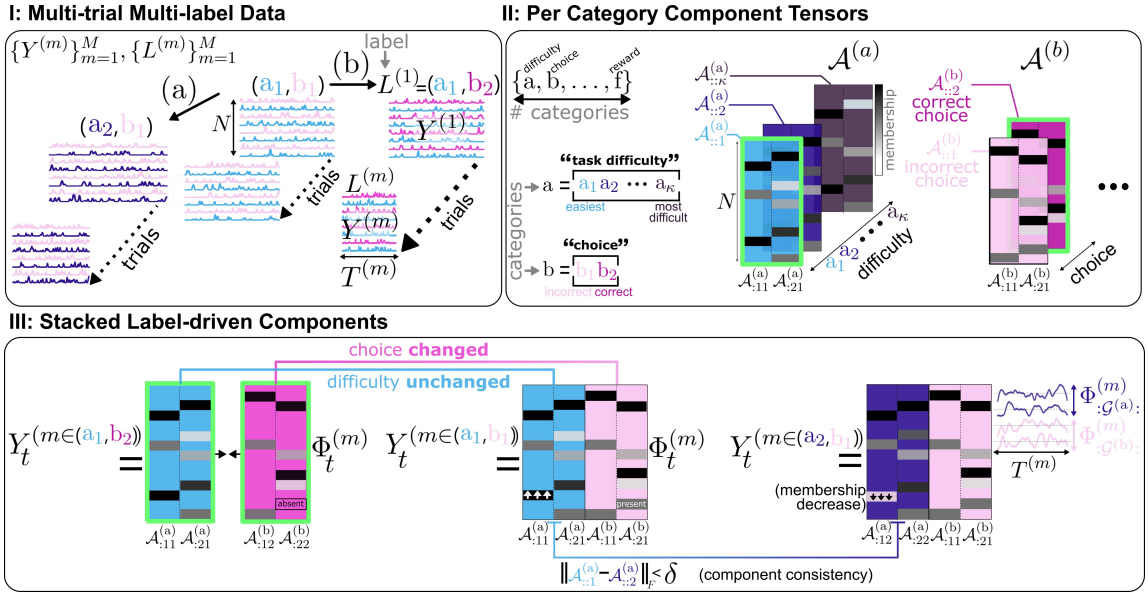

图 *1.* **示意图。I：** 跨 M 个时长不一（$\{T^{(m)}\}_{m=1}^{M}$）的试次的时间序列（例如脑记录）。每个试次 m 都关联一个标签 $L^{(m)}$，该标签由一组跨越不同类别的实验变量组成（例如 $L^{(m)} =$（简单任务，选择正确））。**II：** 每个类别 (k) 的成分由张量 $\mathcal{A}^{(\mathrm{k})}$ 表示，其中第 i 个变体（$\mathcal{A}^{(\mathrm{k})}{}_{::i}$）对应于该类别的第 i 个选项（例如，若类别 (b) 的第 2 个选项为选择正确，则 $\mathcal{A}^{(\mathrm{b})}{}_{::2}$ 表示选择正确的成分）。**III：** 每个试次 m 都通过稀疏分解建模：根据该试次的标签，从每个类别的张量中选取一个变体（层）（绿色边框，**II**），然后将所有选定的变体水平拼接（绿色边框，**III**），从而构成该试次的载荷矩阵。重要的是：1）标签相同的试次使用相同的载荷；2）当相应类别的取值发生变化时，成分可以略微 调整其组成以保持一致性（例如，任务难度为 1 和 2 时的同一成分：$\|\mathcal{A}^{(\mathrm{a})}{}_{::1} - \mathcal{A}^{(\mathrm{a})}{}_{::2}\|_F < \epsilon$）；3）成分时间轨迹（$\{\mathbf{\Phi}^{(m)}\}_{m=1}^{M}$）可以在不同试次间灵活变化。

满足 $\boldsymbol{Y}_{(m)} = \boldsymbol{A}^{(m)}\mathbf{\Phi}^{(m)} + \epsilon$，其中 ϵ 表示例如独立同分布的高斯噪声。尤其是，传统方法要么使用逐试次矩阵来反映可在不同试次间自由变化的成分，要么通过一个在试次间共享的矩阵（对堆叠后的试次进行矩阵/张量分解）进行建模，使得所有试次中的成分完全相同。因此，这些方法无法捕捉在不同试次间发生标签变化时会细微地调整其组成的成分。

例如，考虑一个脑网络（即一个神经元“成分”）：它在困难任务中会招募额外的神经元，但在简单任务中不会。如果将对固定成分进行线性缩放的方法独立应用于不同任务难度下的各个试次，就无法识别出该网络在这些条件下是相同的。另一种做法是将所有试次堆叠后再应用这类方法，但这样会强制简单任务和困难任务具有相同的成分结构，无法如实表示额外招募的神经元，也会扭曲与这些成分相关的轨迹。

因此，我们需要能够识别多路数据中一致成分的方法，了解这些成分如何根据标签变化进行调整，并捕捉它们在单个试次内及不同试次之间的时间轨迹演化。

3.1. 我们的模型：CP 分解方法

成分： 我们首先假设，为了捕捉类别特定信息，$\boldsymbol{A}$ 中的每个成分都对应一个类别 (k)。因此，$\boldsymbol{A}$ 由一组 $|C|$ 类别特异性成分矩阵组成，记为$\{\boldsymbol{A}^{(\mathrm{k})}\}_{(\mathrm{k})\in C}$；其中，每个 $\boldsymbol{A}^{(\mathrm{k})} \in \mathbb{R}^{N\times p^{(\mathrm{k})}}$ 包含与类别 (k) 相关联的 $p^{(\mathrm{k})}$ 个成分 $\in C$，使得$\sum_{(\mathrm{k})\in C} p^{(\mathrm{k})} = P$。每个条目 $A^{(\mathrm{k})}_{nj}$ 表示通道 n属于类别 (k) 的第 j个成分的程度（例如，神经元 n 参与该神经元集群的程度）；而 $\boldsymbol{A}^{(\mathrm{k})}_{nj} = 0$ 表示不属于该成分（图 1，II）。我们假设成分归属具有稀疏性，即每个通道 n 仅属于少数几个成分。因此，我们对每个条目设置拉普拉斯先验：$A^{(\mathrm{k})}_{nj} \sim \mathrm{Laplace}(0, \frac{1}{\gamma_1})$，其中 γ_1 是稀疏性缩放参数（附录 C）。每个 P成分在每个试次 (m) 中都有一条时变轨迹，这些轨迹共同由每试次轨迹矩阵 $\mathbf{\Phi}^{(m)} \in \mathbb{R}^{P\times T}$的各行表示。随后，每个试次中的观测值建模为 $\boldsymbol{Y}_{(m)} \approx \sum_{(\mathrm{k})\in C} \boldsymbol{A}^{(\mathrm{k})}\mathbf{\Phi}^{(m)}_{\mathcal{G}^{(\mathrm{k})}:}$，其中 $\mathcal{G}^{(\mathrm{k})}$ 是 $\boldsymbol{A}^{(\mathrm{k})}$的轨迹对应的行索引（$\mathrm{length}(\mathcal{G}^{(\mathrm{k})}) = p^{(\mathrm{k})}$）。

扩展至标签依赖组件： 我们的完整模型超越了固定组件结构；相反，我们假设成分不仅是类别特定的，每个类别的成分还可以表现出组合

位置变体，即通过在标签变化下微妙地调整其成员关系。因此，每个类别 (k) 的成分表示不再局限于单个矩阵，而是扩展为三维张量 $\mathcal{A}^{(\mathrm{k})} \in \mathbb{R}^{N \times p^{(\mathrm{k})} \times |(\mathrm{k})|}$，其中 $|(\mathrm{k})|$ 是该类别的唯一标签选项数量（例如，$|(\mathrm{k})| = 2$ 对应二元选择，而 $|(\mathrm{k})| = \kappa$ 个选项对应任务难度，图 1，**II**）。第 i个类别 (k) 的成分变体是第i个沿成分张量第三模态的切片 $\mathcal{A}^{(\mathrm{k})}$，并记为 $\mathcal{A}^{(\mathrm{k})}{}_{::i}$（图 1，**II**）。

因此，对于任何带有标签 $L^{(m)}$ 的试次 m，类别 (k) 都会贡献成分变体 $\mathcal{A}^{(\mathrm{k})}{}_{::\arg(L^{(m)}_{(\mathrm{k})})}$，其中 $\arg(L^{(m)}_{(\mathrm{k})}) \in \mathbb{Z}_{\geq 0}$ 将标签值映射到该类别中的对应变体索引（例如，arg(选择：正确) = 2，因为“正确”是选择类别中的第 2 个选项）。

虽然每个成分的变体不必完全相同，但我们假设它们在结构上相似，且相似程度与其标签值的相似程度成正比。因此，对于类别 (k) 中不同的类别选项 $(\mathrm{k})_{i'} \neq (\mathrm{k})_i$，我们假设 $\|\mathcal{A}^{(\mathrm{k})}{}_{::i'} - \mathcal{A}^{(\mathrm{k})}{}_{::i}\|_F^2 < \delta((\mathrm{k})_i, (\mathrm{k})_{i'})$，其中类别依赖距离 δ 较小。值得注意的是，这一约束促进了不同变体之间成分的对齐，迫使 $\{\mathcal{A}^{(\mathrm{k})}\}$ 张量捕捉不同标签组合之间有意义的协同作用，而不是对单个试次过拟合，最终揭示标签之间共享但可调整的模式。

因此，对于每个试次 m，将分解模型扩展为以下特定标签的表述：

$$\boldsymbol{Y}^{(m)} = \sum_{(\mathrm{k}) \in C} \mathcal{A}^{(\mathrm{k})}_{::\arg(L^{(m)}_{(\mathrm{k})})} \boldsymbol{\Phi}^{(m)}_{\mathcal{G}^{(\mathrm{k})}:} + \boldsymbol{\epsilon},\ \epsilon_{ij} \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \sigma^2). \quad (1)$$

3.2. 模型拟合过程：

直接从 $\{\boldsymbol{Y}^{(m)}\}$ 学习成分组成及其轨迹，会受到所有标签在各类别之间的效应混合的阻碍。我们通过三阶段过程来解决这一问题：

• 阶段 *1*：预计算标签相似度图。为同时适应类别标签和有序标签，MILCCI 会为每个类别 (k) 预先计算标签相似度图 $\boldsymbol{\lambda}^{(\mathrm{k})} \in \mathbb{R}^{|(\mathrm{k})| \times |(\mathrm{k})|}$（详见附录 A.2）。这些图可以整合到阶段 *3*中，以确保组成调整能够反映有序标签的标签间距离。

• 阶段 *2*：初始化。我们按照附录 A.1 初始化成分和轨迹。

• 阶段 *3*：迭代优化。按照下文所述更新 $\{\mathcal{A}^{(\mathrm{k})}\}_{(\mathrm{k}) \in \mathbf{C}}, \{\boldsymbol{\Phi}^{(m)}\}_{m=1}^{M}$，直至收敛。

推断 $\{\mathcal{A}^{(\mathrm{k})}\}_{(\mathrm{k}) \in \mathbf{C}}$：对于每个唯一类别 (k) 的选项，$(\mathrm{k})_i$（例如，选择：正确），我们推断第 i个变体 $\mathcal{A}^{(\mathrm{k})}{}_{::i}$，所依据的是在 $(\mathrm{k})_i$（$\widetilde{M} = \{m \in \{1, \ldots, M\} \mid L^{(m)}{}_{(\mathrm{k})} = (\mathrm{k})_i\}$）下观察到的所有试次，而不考虑这些试次在其他类别中的值（例如，所有选择：正确的试次，而不考虑试次难度）。

由于每个试次 m 都是通过对多个类别的成分进行分解来建模的（公式 1），为了学习该变体（$\mathcal{A}^{(\mathrm{k})}{}_{::i}$）的结构，我们需要将其贡献与其他类别成分的贡献区分开来。因此，对于每个 $m \in M$，我们首先计算残差矩阵 $\widetilde{\boldsymbol{Y}}_{(m,\mathrm{k})}$，该矩阵表示观测值 $\boldsymbol{Y}_{(m)}$ 与基于除 $\mathcal{A}^{(\mathrm{k})}{}_{::i}$ 之外的所有成分得到的部分重构之间的差异。具体而言：$\widetilde{\boldsymbol{Y}}_{(m,\mathrm{k})} := \boldsymbol{Y}_{(m)} - \sum_{(\mathrm{k}') \neq (\mathrm{k})} \mathcal{A}^{(\mathrm{k}')}{}_{::\arg(L^{(m)}_{(\mathrm{k}')})} \boldsymbol{\Phi}^{(m)}_{\mathcal{G}^{(\mathrm{k}')}:}$。随后，我们通过 LASSO（Tibshirani，1996）推断 $\mathcal{A}^{(\mathrm{k})}{}_{::i}$：

$$\widehat{\mathcal{A}}^{(\mathrm{k})}_{::i} = \arg\min_{\mathcal{A}^{(\mathrm{k})}_{::i}} \|\widetilde{\boldsymbol{Y}}^{(m,\mathrm{k})} - \mathcal{A}^{(\mathrm{k})}_{::i} \boldsymbol{\Phi}^{(m)}_{\mathcal{G}^{(\mathrm{k})}:}\|_F^2 + \gamma_1 \|\mathcal{A}^{(\mathrm{k})}_{::i}\|_{1,1} + \gamma_2 \sum_{i' \neq i} \lambda^{(\mathrm{k})}_{i',i} \|\mathcal{A}^{(\mathrm{k})}_{::i'} - \mathcal{A}^{(\mathrm{k})}_{::i}\|_F^2 \quad (2)$$

其中$\lambda^{(\mathrm{k})}_{i',i}$用于提升同一类别（k）的不同变体之间的相似性（附录A.2），而 γ_1，γ_2 是控制稀疏性和变体间相似性的超参数。值得注意的是，该模型还支持对成分施加非负约束（$\mathcal{A}^{(\mathrm{k})}_{nji} \geq 0 \, \forall n,\ j,\ i$），适用于预期取值为正的应用场景。

总体而言，公式2通过 $\boldsymbol{\lambda}^{(\mathrm{k})}$（第 3 项），在数据保真度（第 1 项）、成分稀疏性（第 2 项），以及对应成分之间与其标签相似度成正比的一致性之间取得平衡。随后，将每个成分归一化至固定总和，以避免与 $\boldsymbol{\Phi}^{(m)}$之间的尺度歧义。

更新 $\{\boldsymbol{\Phi}^{(\mathrm{m})}\}_{m=1}^{M}$：由于各试次的轨迹独立于标签变化（即属于无监督数据），因此可以针对每个试次 m，使用该试次中实现分量的标签驱动载荷矩阵进行学习。该载荷矩阵记为 $\boldsymbol{A}^{(L^{(m)})} \in \mathbb{R}^{N \times P}$，是一个辅助变量，通过根据试次的标签 $L^{(m)}$，从每个类别张量中选取一个切片 $\{\mathcal{A}^{(k)}\}_{(k) \in C}$ 并将这些切片水平拼接而构建（图1，**III**）。在每个试次中$m = 1 \ldots M$，随后可以按如下方式更新 $\boldsymbol{\Phi}^{(m)}$：

$$\widehat{\boldsymbol{\Phi}}^{(m)} = \arg\min_{\boldsymbol{\Phi}^{(m)}} \underbrace{\|\boldsymbol{Y}^{(\mathrm{m})} - \boldsymbol{A}^{(L^{(m)})} \boldsymbol{\Phi}^{(m)}\|_F^2}_{\text{data fidelity}} + \gamma_3 \underbrace{\sum_{t=1}^{T^{(m)}} \|\boldsymbol{\Phi}^{(m)}_{:,t} - \boldsymbol{\Phi}^{(m)}_{:,t-1}\|_2^2}_{\text{temporal smoothness}} + \gamma_4 \underbrace{\|(\mathbf{C} \odot (\mathbf{1} - \mathbf{I}_P)) \odot \boldsymbol{D}\|_{1,1}}_{\text{within-trial trace decorrelation}}, \quad (3)$$

其中 γ_3，γ_4 为超参数，$\odot$ 为逐元素乘法，$\mathbf{C} := \mathrm{Gram}(\boldsymbol{\Phi}^{(m)})$，且 $\boldsymbol{D} \in \mathbb{R}^{p \times p}$用于归一化（$D_{j,\,j'} := \|\boldsymbol{\Phi}^{(m)}{}_{:\,j}\|_2^{-1} \|\boldsymbol{\Phi}^{(m)}{}_{:\,j'}\|_2^{-1}$）。式3 总体上促进数据保真度（第 1 项），鼓励平滑性（第 2 项），并惩罚各轨迹之间的相关性，这些轨迹来自

不同成分（第 3 项）。参见算法、记号及图示，见算法 1、表1和图1。

4. 实验

我们在合成数据上验证 MILCCI，并在四个真实世界数据集上证明其有效性。

MILCCI 从合成数据中恢复真实成分：我们生成了由 $P = 4$ 个具有时变轨迹的稀疏成分产生的合成数据（$T = 500$ 个时间点；$M = 250$ 个试次）。我们定义了两个类别：(a)“任务难度”（5 个选项）和 (b)“选择”（2 个选项），使得 $p^{(a)} = p^{(b)} = 2$ 个集成随 (a) 和 (b) 的变化而调整（图 6B）。每条轨迹均生成为高斯过程，其参数在不同成分和试次之间变化：有些反映任务难度，有些反映选择，有些则在每个试次中变化（图 2、B，图 6、A，附录 D）。我们在该数据上运行 MILCCI，并与以下方法进行比较：(1) Tucker（Tucker，1966年），(2) PARAFAC（Harshman 等人，1970年），(3) 非负 PARAFAC（Shashua 与 Hazan，2005），(4) 奇异值分解，(5) SiBBlInGS（Mudrik 等，2024），以及 (6) sliceTCA（Pellegrino 等人，2024）；所有方法均使用相同的 $P = 4$ 个成分（图 2、F、G，第 H 节）。由于张量分解对成分排列具有不变性，我们使用线性和指派将每种方法的成分与真实值对齐。

MILCCI 恢复了真实的成分（图 2，C）和轨迹（图 2，D），两者与真实值均高度相关（图 2，E）。与其他方法（图 2，F、G）相比，MILCCI 与真实值的相似度最高。虽然 SiBBlInGS 的相关性不相上下，但它生成的轨迹较为模糊（例如，图 2，F），且不具备 MILCCI 的可解释性，无法区分各类别的贡献。

MILCCI 揭示按政党和公职划分的州级投票模式：接下来，我们在以下投票数据上测试 MILCCI（Data& Lab，2017a；b；c），数据包含不同政党（民主党、共和党、自由意志党；表 3）和公职（总统职位、参议院、众议院）的投票数；每个 $\boldsymbol{Y}_{(m)} \in \mathbb{R}^{51\times 23}$ 表示所有州多年来的投票数。我们首先对数据进行了预处理，包括处理因各州选举日程不同而产生的缺失值（图 8，附录E）。

我们使用以下方法应用 MILCCI：$P = 8$ 成分，类别包括 (a) 政党和 (b) 公职（$p^{(a)} = p^{(b)} = 4$ 各自）。MILCCI 可发现捕捉各州特定投票模式的成分，这些模式因公职、政党和时间而异。例如，成分 $\mathcal{A}^{(\text{party})}{}_{:1:}$（图 3，B.1）突出显示蒙大拿州 (MT) 和宾夕法尼亚州 (PA) 在“其他”类别中的成员人数增加，这主要分别由独立党（图 16，右）和宪法党（图 16，左）推动。这与蒙大拿州历史上对个人主义的重视相符（Kitayama 等人，2010），而宾夕法尼亚州则是宪法党总部所在地。同一成分还识别出俄勒冈州自由意志党成员人数增加，这与放宽小党派选票获取权的2001年法律相符（俄勒冈州政党手册），也反映出俄勒冈州自由意志党是最早成立州分支机构的政党之一。值得注意的是，其轨迹（$\boldsymbol{\Phi}_{:\mathcal{G}_1^{(\text{party})}}$，图3，B.1，左上）显示民主党—共和党激活相互重叠并逐渐分化 ∼2004，民主党活动上升而共和党活动下降，这反映了该时期全国政治变迁所推动的长期党派重组（例如伊拉克战争、2003年）。

在成分 $\mathcal{A}^{(\text{party})}{}_{:3:}$ (图3，B.1，右侧），MILCCI 将阿拉斯加州、俄克拉何马州、阿拉巴马州、亚利桑那州、密西西比州和蒙大拿州归为一组。这种分组与这些州在立法上的相似性相符，例如严格的选民身份证明法（全国州议会会议，2025），这表明 MILCCI 能够直接从观测值中有效提取潜在趋势。从时间维度来看，其轨迹 $\boldsymbol{\Phi}_{:\mathcal{G}_3^{(\text{party})}}$（图 3，B，左下）显示民主党与共和党的激活趋势相反：共和党激活上升，而自由意志党的活动在 2016 年前后开始显现。这一趋势反映出共和党选票增加、民主党选票减少，以及这些州的部分民主党支持者可能转而支持自由党。这些以及 MILCCI 识别出的其他模式（附录 E.2）展现了该方法揭示州—政党—公职依赖模式的能力。我们进一步验证了（附录 E.3），MILCCI 发现的投票组件捕捉到了真实结构，因为各州对重构均作出了有意义的贡献（图 9，A-C），且该模型的表现显著优于置换零模型（$p<0.001$）。更多投票方面的见解见附录 E.2。

值得注意的是，其他 TF 方法得到的成分（图 10、13）要么是稠密的（PARAFAC），要么包含负值（SVD、PARAFAC、Tucker），而 SiBBlInGS 则未能将政党与公职区分开来，这些问题都会妨碍可解释性（图 12）。此外，由于张量结构的限制，PARAFAC 和 Tucker 无法捕捉组合调整，也不能灵活地改变其轨迹，以捕捉试次间的时间变异性。

MILCCI 跨设备和语言发现维基百科页面簇：接下来，我们提取了维基百科页面（Wikipages）页面浏览量计数（Meta，2022）（10月20日至24日，$T = 1482$ 天）的 $N = 32$ 个来自不同领域的随机维基百科页面。我们在 3 种条件下跟踪浏览量：（a）代理：用户/爬虫（即网络爬虫）；（b）平台：桌面端/网页/应用程序；以及（c）语言：en/ar/es/fr/he/hi/zh。每个 $\boldsymbol{Y}^{(m)} \in \mathbb{R}^{N\times T}$ 都包含所有32个维基百科页面在不同时间条件下的每日页面浏览量，例如（英语、桌面端、用户）。因此，我们旨在识别能够捕捉具有相似兴趣模式的维基百科页面的成分，并了解这些模式如何随不同条件而变化（数据与预处理的详细信息见附录F.1）。我们在数据上运行 MILCCI，每个类别设置 $p^{(k)} = 4$个成分，并识别出

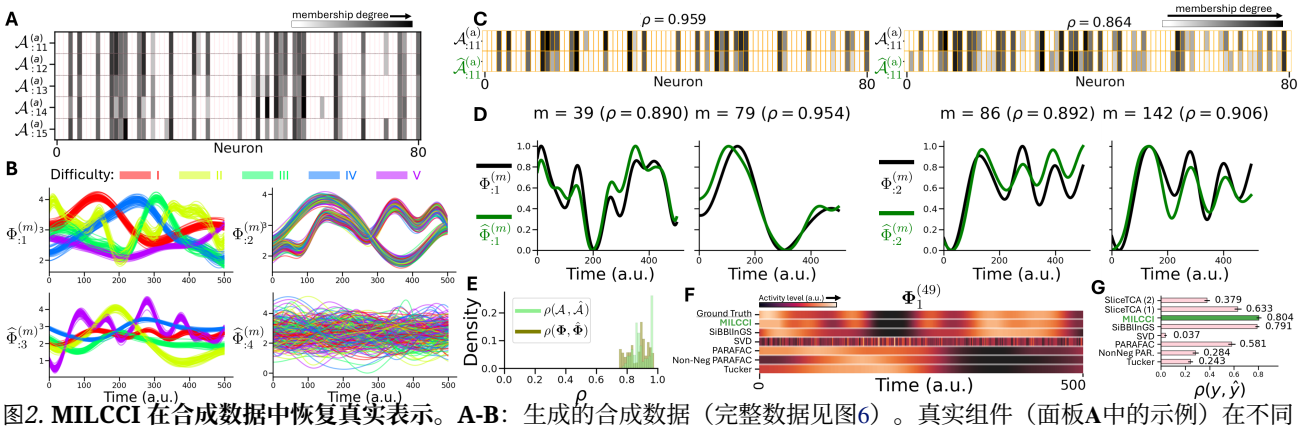

图*2.* **MILCCI 在合成数据中恢复真实表示。A-B**：生成的合成数据（完整数据见图6）。真实组件（面板**A**中的示例）在不同标签间略有差异，但在相同标签的试次（行）中保持不变。真实轨迹随试次变化（**B**，按难度着色）。**C-D**：随机试次中已识别组件与真实组件（**C**）及时间轨迹（**D**）的比较。**E**：已识别组件和轨迹与其真实对应项之间相关性的直方图。**F-G**：基于轨迹（随机试次，4 个组件维度相同）和重建性能（**F**）将 MILCCI 与其他方法进行比较（**G**，基线方法详情见第H节）。

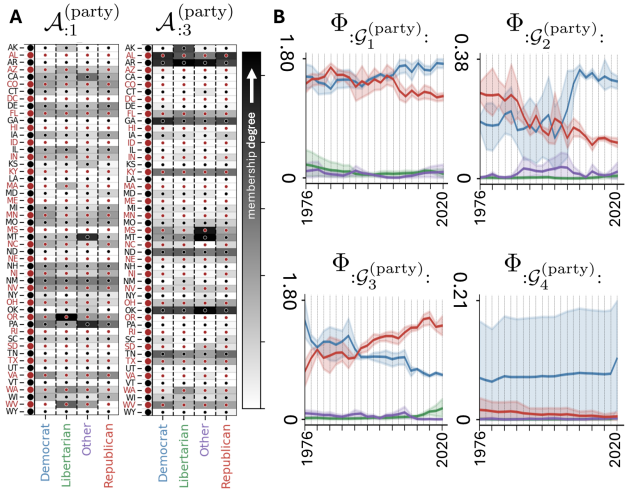

图*3.* **投票结果：**已识别的示例组件（**A**）和轨迹（**B**，均值和80% 置信区间）。完整内容见图8。这些组件将相关的维基百科页面聚为一类，并随类别而变化（图4A），某些页面（例如无监督学习）会出现在多个组件中，这突显了 MILCCI 捕捉多义术语的能力。

具体而言，我们识别出例如 $\mathcal{A}^{(\text{agent})}_{:1:}$ 将与学习理论（心理学）相关的维基百科页面归为一组（图4，A，红色箭头，列表见附录F.2）；$\mathcal{A}^{(\text{agent})}_{:2:}$ 将社交媒体页面归为一组（紫色箭头）；以及成分 $\mathcal{A}^{(\text{platform})}_{:4:}$ 将计算机科学基础归为一组（蓝色箭头）。探究成分构成如何随用户↔爬虫以及桌面端↔网页↔应用程序而变化时，会发现一些有趣的模式。

对于 $\mathcal{A}^{(\text{agent})}_{:1:}$（心理学中的学习理论），MILCCI 发现代理之间存在细微差异（爬虫与用户，图 4，A，左）。例如，波波娃娃实验的维基百科页面（一项关于社会学习理论的心理学实验）出现在“爬虫”下，而不在“用户”下。这与其相比该簇中的其他心理学术语对普通人而言较不熟悉这一点相一致，而网络爬虫则通过实际的维基百科链接（将该页面与其他相关术语连接起来）与之建立关联。因此，其他维基百科页面，如古典条件作用和操作性条件作用（心理学的基础），在“用户”中的成员度更高。“无监督学习”和“嵌入”在该组件中也表现出较低的成员度，且“用户”中的成员度高于“爬虫”。有趣的是，这些实际的维基百科页面指向计算机科学术语（而非心理学），但由于它们在心理学学习中也具有相关含义，因此“用户”相较于“爬虫”更高的成员度符合人类行为：用户进入页面后发现该术语属于不同领域，便会离开，而网络爬虫则会遵循可预测的导航模式。这些发现凸显了 MILCCI 在同一组件中揭示人类与网络爬虫不同行为的能力，也强调了允许组合调整以捕捉细微组件适应的重要性。该组件的轨迹（$\mathbf{\Phi}_{:\mathcal{G}_1^{(\text{agent})}:}$，图 4，B，左上）体现了其波动，以及相较于其他语言在英语中的更高激活。这与专业术语在英语中更加详尽这一点相吻合，因为英语通常使用在其他语言中未被完整定义或使用的行话；同时，非英语母语者也可能更倾向于用英语阅读专业材料（Miquel-Rib´e &Laniado，2018年）。

社交媒体组件 $\mathcal{A}^{(\text{agent})}_{:2:}$ 显示出细微的结构调整：用户↔爬虫：「马克·扎克伯格」在「爬虫」中的占比更高，而「社交媒体」在「用户」中的占比更高，这可能反映出人们对人物的兴趣有所下降，而对「社交媒体」这类常见术语的兴趣相对更高。其轨迹（$\mathbf{\Phi}_{:\mathcal{G}_2^{(\text{agent})}:}$）在英语和非英语内容中都持续上升，直到2024年3月达到峰值（图4B，左下），这与社交媒体多年来日益普及的总体趋势相符，也可能与例如2024年3月佛罗里达州社交媒体禁令有关，该禁令在相应的「社交媒体」维基百科页面

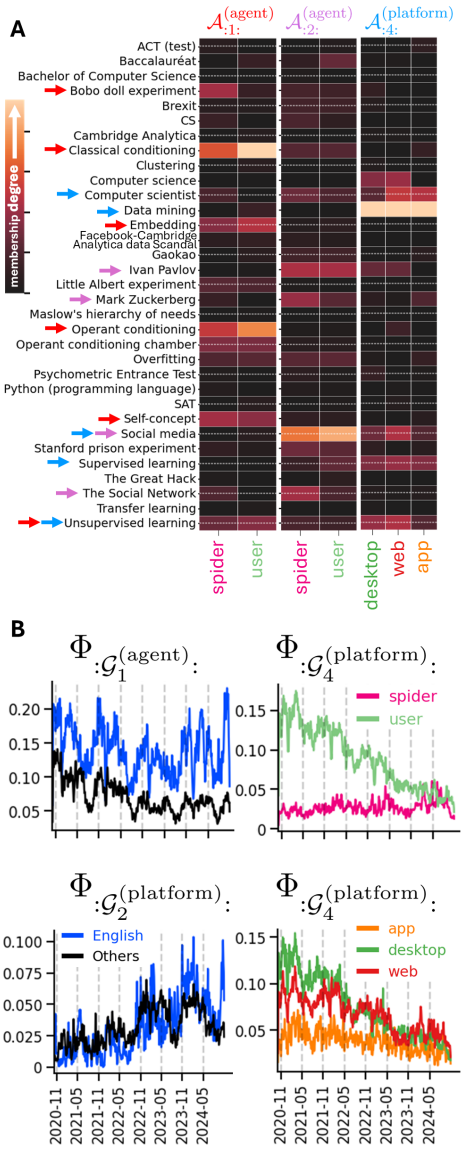

图4. 识别出的Wiki页面浏览示例成分（**A**）以及按不同类别平均的轨迹（**B**）。

由该成分捕获。 $\mathcal{A}^{(\text{platform})}{}_{:4}$（计算机科学基础术语）捕捉了平台间成员构成的调整，例如，计算机科学家（一个简短且不含数学内容和图的维基百科页面）在应用程序/网页端的成员占比高于桌面端。这与例如无监督学习形成对比：其页面内容更复杂（包含数学内容和图），且在应用程序中的成员占比低于其他平台。这或许符合用户的偏好：在应用程序上阅读「简单」术语，在桌面端阅读更复杂的术语；同时也展现了MILCCI揭示不同平台间可解释处理差异的能力。该组件的时间轨迹（$\mathbf{\Phi}_{:\mathcal{G}_4^{(\text{platform})}}$）显示英语内容中的活动更活跃（图18），访问主要来自桌面端和移动网页（图4，B，右下）。时间轨迹主要由用户而非网络爬虫驱动（图4，B，右上）；用户轨迹随时间下降，这与这些术语在计算机科学中属于基础（“旧”）概念相一致（相较于大语言模型等较新趋势）。这凸显了MILCCI在发现平台特定参与度和行为指纹方面的能力。一些更有趣的模式包括，例如，$\mathcal{A}^{(\text{platform})}{}_{:2}$ 它捕获了与剑桥分析相关的术语；其轨迹在2020年前后达到峰值，主要出现在英语中，此后下降，与该事件的时间线相符。参见图18、21、20、22、23 以进行比较并查看完整轨迹。

MILCCI发现多区域大脑数据背后的神经元集群： 随后，我们将MILCCI应用于小鼠在决策任务中多区域、单细胞分辨率记录所得的神经元活动模式（Laboratory 等人，2025；Angelaki 等人，2025），附录G）。在该实验中，小鼠通过向左或向右转动转轮，报告不同对比度视觉光栅的位置（图5，A）。我们使用一次随机实验时段的数据，提取可用的尖峰时间数据，并通过施加高斯卷积估计放电率。我们移除不活跃的神经元，并拆分为 $M = 1011$ 个试次，使得每个 $\boldsymbol{Y}^{(m)} \in \mathbb{R}^{N\times T}$（具有 $N = 270$ 和 $T = 137$）表示试次 m 中神经元随时间变化的放电率（例如，图31）。我们为每个类别定义了 $p^{(\text{k})} = 2$ 个成分（“神经元集群”）：（a）试次编号，（b）先验（预期）刺激侧，（c）所施加的刺激侧，以及（d）跨试次固定的成分。该设置（1）能够将可能与学习或注意相关的表征漂移（Rule 等人，2019；Driscoll 等人，2022）与任务变量区分开来，并且（2）证明MILCCI能够同时支持非调整成分与调整成分，并跨越类别型与有序型类别。

我们发现了对多种任务变量具有选择性的神经元集群（图5）。例如，$\mathbf{\Phi}^{(m)}{}_{:\mathcal{G}_1^{(\text{a})}}$ 对选择正确性进行了调谐，其活动在刺激呈现期间激增（图5，B 左），但仅在正确选择时出现。有趣的是，MILCCI 还发现了一个对错误决策敏感的集群，该集群在刺激呈现后立即被激活（$\mathbf{\Phi}_{:\mathcal{G}_2^{(\text{b})}:}$，图32）。值得注意的是，这些成分有助于揭示神经元集群如何整合刺激信息以支持正确决策，并将这些轨迹与所涉及的特定神经元联系起来（图34）。另一个例子是，$\mathbf{\Phi}^{(m)}{}_{:\mathcal{G}_1^{(\text{b})}:}$ 主要在具有随机先验的试次中活动（即，p（左）$= 0.5$），贯穿刺激呈现之前、期间和之后；而在先验偏向某一侧的试次中则基本不活动（图5，B 右）。这凸显了 MILCCI 在提示先验参与决策方面的有效性，以及其揭示相似条件轨迹之间相似性的能力。

此外，MILCCI 能够分离出具有

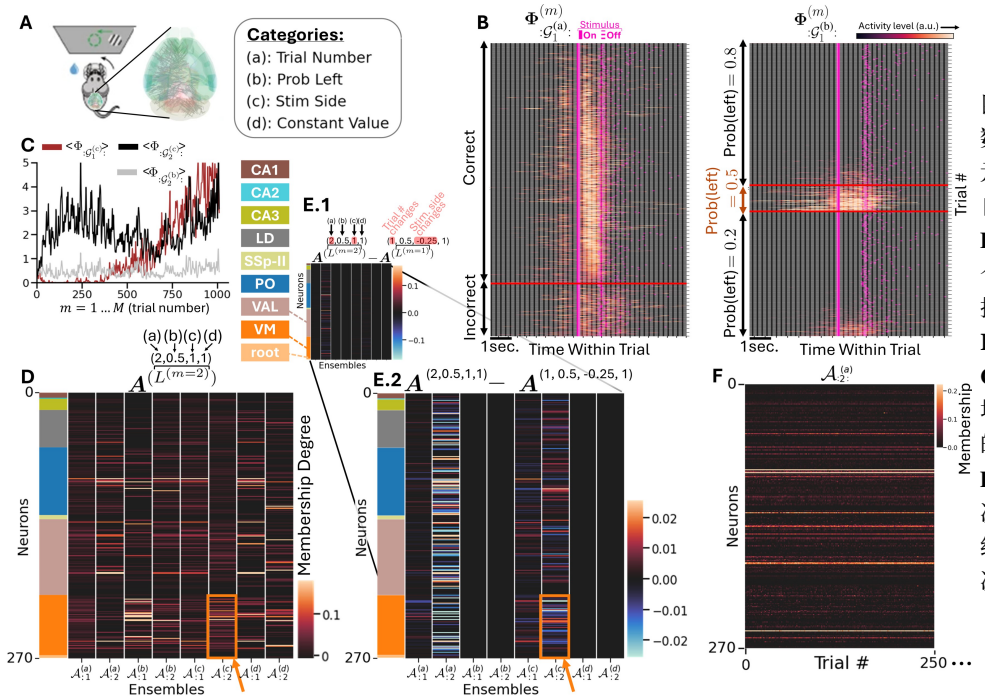

图5. **MILCCI 在真实世界脑数据中识别出有意义的神经元集群。A**：实验设置（引自（Angelaki 等人，2025））。**B**：组件轨迹（全部 1011 个试次）按试次正确性水平排序（左图），并按 Prob(左侧）（右图）排序。**C**：示例轨迹的试次内数值均值揭示了轨迹随试次变化的时间漂移程度各不相同。**D**：识别出的集群（示例试次）。**E**：不同试次间集群组成的差异。**F**：前250个试次中经试次调整的集群组成。

追踪时间漂移的特征（$\mathbf{\Phi}^{(m)}:\mathcal{G}_1^{(c)},:$，图5，C，棕色；33）揭示了神经元编码如何随试次演变，这可能是学习、注意或适应的结果。MILCCI 揭示了集群组成如何捕捉脑区间独特的局部和广泛结构（图5，D；左侧颜色标示神经元所属脑区）。例如，$\mathcal{A}^{(c)}:2:$ 包含许多丘脑腹内侧核（VM，图5D 中的橙色箭头）中的神经元，这表明它可能参与唤醒调节和运动协调。我们还可以通过集群组成识别有趣的模式（图5，D），以及这些模式在不同试次间如何变化（例如，刺激侧不同的两个示例试次之间；图5，E.1；图5，E.2 放大显示了变化）。这些结果揭示了集群组成调整的模式：第一个集群 $\mathcal{A}^{(a)}$（图5，E.2，左列）的调整很小，而第二个集群则在多个区域呈现分布式调整。在刺激调节型集群中（第5、6列），调整发生在 VM 周围（例如，$\mathcal{A}^{(c)}:2:$，图5），表明 VM 组成具有适应性。

MILCCI 识别唤醒和频率依赖性神经元集群：最后，在附录 J中，我们展示了 MIL- CCI 能够识别反映唤醒和刺激频率的成分，并且对超参数具有稳健性。

5. 讨论、局限性与未来方向

我们介绍了 MILCCI，这是一种用于分析多试次多标签时间序列的数据驱动方法。MILCCI 能够识别数据背后可解释的成分，捕捉跨试次关系并整合标签信息，同时考虑标签效应之外的试次间变异。MILCCI 允许成分随标签变化调整其构成，从而揭示在固定成分分解中可能被掩盖的相似性。与其他方法不同，MILCCI 在对跨试次变异建模、整合标签并区分标签驱动与非标签驱动的变异来源时，仍能保持可解释性。MILCCI 的另一项优势是能够在单个数据集中同时处理多种类型的标签，包括类别标签、非连续有序标签、连续标签以及随试次变化的标签（例如 IBL实验中的情况）。我们在合成数据上验证了 MILCCI，结果表明其性能优于其他方法；我们还在真实世界数据上展示了其有效性：（1）揭示与真实事件相吻合的投票模式；（2）恢复可解释的维基百科浏览成分，其成员资格随语言、平台和代理而变化；（3）揭示在不同试次和任务变量下不断调整的神经元集群。值得注意的是，MILCCI 的架构（低秩稀疏成分）自然地约束模型，使其学习能够在不同试次间灵活复用的模式。MILCCI 的另一项特点是其模块化设计，支持未来以其他特定应用的代价指标替代当前的均方误差指标（该指标反映正态分布假设）。值得注意的是，尽管本文为清晰起见聚焦于时间序列，MILCCI 也适用于时序数据以外的模态。

一个潜在的问题是成分与轨迹之间的尺度歧义。MILCCI 通过在每次迭代后对成分进行归一化来解决这一问题，同时允许轨迹在不同试次间灵活变化，以捕捉试次特异性振幅。尽管每个类别的组件维度是一个超参数，MILCCI 也能自然地处理这一点，因为稀疏性会使冗余成分趋于零。未来的研究方向包括扩展作为可解释性核心的线性假设，

适用于非线性结构（例如，通过核化）；针对大型数据集进行并行优化或批处理；或采用描述成分时间演化的动态先验（例如，附录。I，J）。

致谢

N.M. 获得了卡夫利神经数据发现研究所提供的研究员资助。Y.C. 获得了美国公共卫生服务局下属美国国家老龄研究所提供的 P01AG009973 项目资助。G.M. 获得了美国国家科学基金会编号为 CCF-2217058 的项目资助。A.S.C. 获得了美国国家科学基金会 CAREER 奖（编号 2340338）的部分资助。

影响声明

本研究推动了用于分析不同领域多视图数据的机器学习方法的发展。我们预计这一方法学贡献不会带来负面的社会影响。

参考文献

Angelaki, D., Benson, B., Benson, J., Birman, D., Bonacchi, N., Bougrova, K., Bruijns, S. A., Carandini, M., Catarino, J. A., 等。《复杂行为期间神经活动的全脑图谱》。《自然》，645(8079):177–191，2025年。

Aoi, M. 和 Pillow, J. W. 面向神经元群体数据的基于模型的目标维度约简。《神经信息处理系统进展》，31，2018年。

Chen, P.-H. C., Chen, J., Yeshurun, Y., Hasson, U., Haxby, J. 和 Ramadge, P. J. 一种降维的功能性磁共振成像共享响应模型。载于 Cortes, C., Lawrence, N., Lee, D., Sugiyama, M. 和 Garnett, R.（编），《神经信息处理系统进展》，第28卷。Curran Associates, Inc.，2015年。

Chen, Y., Mudrik, N., Johnsen, K.A., Alagapan, S., Charles, A. S., 和 Rozell, C. 用于稳健发现潜在神经动力学的概率分解线性动力系统。《神经信息处理系统进展》，37:104443–104470，2024。

数据，M. E. 和实验室，S. 美国总统 1976–2020，2017a。网址https://doi.org/10.7910/DVN/42MVDX。

数据，M.E. 和实验室，S. 美国众议院 1976–2022，2017b。网址https://doi.org/10.7910/DVN/IG0UN2。

数据，M. E. 和实验室，S. 美国参议院各州数据 1976–2020，2017c。网址https://doi.org/10.7910/DVN/PEJ5QU。

Driscoll, L. N.、Duncker, L. 和 Harvey, C. D. 表征漂移：持续学习的新兴理论与实验未来方向。《神经生物学当前观点》，76:102609，2022年。

Harshman, R. A. 等。PARAFAC 程序的基础：“解释性”多模态因子分析的模型与条件。《加州大学洛杉矶分校语音学工作论文》，16 (1):84，1970年。

Hyvarinen, A.、Karhunen, J. 和 Oja, E. 独立成分分析与盲源分离，2001年。Kitayama, S.、Conway III, L. G.、Pietromonaco, P. R.、Park, H. 和 Plaut, V. C. 美国各地区的独立性精神：文化变迁的生产-采纳模型。《美国心理学家》，65(6):559，2010年。Kobak, D.、Brendel, W.、Constantinidis, C.、Feierstein, C. E.、Kepecs, A.、Mainen, Z. F.、Qi, X.-L.、Romo, R.、Uchida, N. 和 Machens, C. K. 神经群体数据的解混主成分分析。《eLife》，5:e10989，2016年。

Laboratory, I.B., Benson, B., Benson, J., Birman, D.,Bonacchi, N., Carandini, M., Catarino, J., Chapuis,G.,Dayan, P.,DeWitt, E.,Engel,T.,Fabbri,M.,Faulkner,M., Fiete, I., Findling,C., Freitas-Silva, L.,Gerc.ek, B., Harris, K., Hofer, S., Hu, F., Hubert, F., Hunten- burg, J., Khanal, A., Langdon, C., Lau, P., Meijer, G., Miska, N., Noel, J.-P., Nylund, K., Pan-Vazquez, A., Pouget, A., Rossant,C., Roth, N., Schaeffer, R., Schart- ner, M., Shi, Y., Socha, K., Steinmetz, N., Svoboda, K., Urai, A., Wells, M., West, S., Whiteway, M., Win- ter, O., and Witten, I. Ibl - 大脑全域图谱 [已弃用](草稿版本)。https://dandiarchive.org/dandiset/000409/draft，2025年。数据集，DANDI档案库。

Lathauwer, L. D., Moor, B. D. 和 Vandewalle, J. 多线性奇异值分解。*SIAM*矩阵分析与应用期刊，21:1253–1278，2000。网址：https://api.semanticscholar.org/CorpusID:14344372。

Lee, D. D. 和 Seung, H. S. 通过非负矩阵分解学习物体的组成部分。《自然》，401(6755): 788–791，1999。

Ma, Y. 和 Zhu, L. 关于降维的综述。《国际统计评论》，81(1):134–150，2013。

Maaten, L. v.d. 和 Hinton, G. 使用 t-SNE 可视化数据。《机器学习研究杂志》，9(Nov): 2579–2605，2008。

Mairal, J., Bach, F., Ponce, J. 和 Sapiro, G. 用于稀疏编码的在线字典学习。载于第 *26* 届国际机器学习年度会议论文集，第 689–696 页，2009。

Mante,V., Sussillo, D., Shenoy,K.V.,and Newsome, W. T. 前额叶皮层中的循环动力学执行的上下文依赖型计算。《自然》, 503(7474):78–84, 2013.

Meta. 页面浏览量分析——Meta，关于维基媒体项目的讨论，2022年。URLhttps://meta.wikimedia.org/w/index.php?title=页面浏览量_分析&oldid=23610159。[在线；访问日期：10月31日-2024]。

Miquel-Rib´e, M. 和 Laniado, D. 维基百科文化差异：量化40 种语言版本之间的内容失衡。《物理学前沿》，6:54，2018年。

Mudrik, N., Mishne, G. 和 Charles, A. S. SiBBlInGS：利用图跨状态进行相似性驱动的构件推断。发表于第四十一届国际机器学习会议，2024。

Mudrik, N., Ly, R., Ruebel, O. 和 Charles, A. S. Creimbo：多视图脑观测中的跨区域集成相互作用。发表于第十三届国际学习表征会议，2025。

全国州议会会议。选民身份识别法。https://www.ncsl.org/elections-and-campaigns/voter-id，2025。更新于2025年6月2日。

Papadopoulos, L., Jo, S., Zumwalt, K., Wehr, M., McCormick, D. A. 和 Mazzucato, L. 亚稳态集成动力学的调节解释了听觉皮层在中度唤醒时的最优编码。*bioRxiv*，2024。Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M. 和 Duch- esnay, E. Scikit-learn：Python 机器学习。《机器学习研究杂志》，12:2825–2830, 2011。网址 http://jmlr.org/papers/v12/pedregosa11a.html。

Pellegrino, A., Stein, H. 和 Cayco-Gajic, N. A. 利用切片张量分量分析实现超越神经子空间的降维。《自然·神经科学》，27(6)：1199–1210，2024。

Ravasi, M. 和 Vasconcelos, I. PyLops——一个用于可扩展代数和优化的线性算子 Python 库。*SoftwareX*，11：100495，2020年。doi：https://doi.org/10.1016/j.softx.2020.100495。网址 https://www.sciencedirect.com/science/article/pii/S2352711020302449。

R¨ubel, O., Tritt, A., Ly, R., Dichter, B. K., Ghosh, S., Niu, L., Baker, P., Soltesz, I., Ng, L., Svoboda, K. 等。无边界神经数据生态系统：神经生理数据科学。《eLife》，11：e78362，2022。

Rule, M. E., O'Leary, T. 和 Harvey, C. D. 表征漂移的成因及其后果。《神经生物学当前观点》，58:141–147, 2019.

Shashua, A. 和 Hazan, T. 非负张量分解及其在统计学和计算机视觉中的应用。载于第*22*届国际机器学习会议论文集，pp. 792–799, 2005.

Tibshirani, R. 基于 LASSO 的回归收缩与变量选择。《英国皇家统计学会会刊*B*辑：统计方法学》，58(1):267–288，1996年。

Tillinghast, C., Fang, S., Zhang, K. 和 Zhe, S. 概率神经核张量分解。发表于*2020*年*IEEE*国际数据挖掘会议*(ICDM)*，第 531–540 页。IEEE，2020 年。

Tucker, L. R. 关于三模因子分析的一些数学注记。《心理测量学》，31(3):279–311，1966年。

Van Den Berg, E. 和 Friedlander, M. P. 探索基追踪解的帕累托前沿。《暹罗科学计算杂志》，31(2):890–912，2009年。

Wang, Z. 和 Zhe, S. 用于动态张量分解的非参数因子轨迹学习。发表于国际机器学习会议，第 23459–23469 页。机器学习研究会议论文集，2022年。

Xu, Z., Yan, F., 和 Qi, Y. 无限Tucker分解：用于多路数据分析的非参数贝叶斯模型。载于《第29届国际机器学习会议论文集》，第1675–1682页，2012年。

Zhe, S., Zhang, K., Wang, P., Lee, K.-c., Xu, Z., Qi, Y., 和 Ghahramani, Z. 分布式灵活非线性张量分解。《神经信息处理系统进展》，29，2016年。

Zou, H.、Hastie, T. 和 Tibshirani, R. 稀疏主成分分析。《计算与图形统计学杂志》，15(2):265–286，2006。

附录

目录

A 额外拟合细节： **14**

A.1 初始化：.14 A.2 关于状态相似性图计算的细节：.14 A.3 关于稀疏性与一致性超参数.14

B 运行详情 **15**

B.1 数据与代码可用性.15 B.2 版本.15

C 成分的弹性网络先验 15

D 补充信息——合成实验 15

E 补充信息——投票实验 **16**

E.1 投票数据预处理.16 E.2 补充发现——投票数据.18 E.3 投票数据的事后验证分析.20

F 补充信息–维基百科实验 **30**

F.1 维基百科页面浏览量数据预处理.30 F.2 对发现的澄清——维基百科数据.31

G 关于神经元集群实验的补充信息 39

H 关于基线计算与执行的信息 **39**

H.1 与奇异值分解、Tucker、PARAFAC的比较.39 H.2 与SliceTCA的比较.40 H.3 SiBBlInGS.40

I 通过动态先验对轨迹的替代推断 **41**

I.1 线性动态系统先验.41 I.2 修改后的目标函数与推断.41 I.3 初始化.41

J 第四个真实世界实验：MILCCI识别出随唤醒水平和刺激频率调整并依循动力学规则演化的神经元集群 42

J.1 数据与预处理 42

J.2 扩展版 MILCCI 的演示：采用非线性变换和轨迹演化上的动力学先验.42 J.3 超参数敏感性分析.43

K 伦理声明和LLM使用 **46**

A. 额外拟合细节：

A.1. 初始化：

我们使用字典学习（Mairal 等人，2009）初始化成分和轨迹，并对成分施加稀疏性；具体使用 Sklearn.decomposition(Pedregosa 等人，2011) DictionaryLearning。在模型中，通过 PyLops（Ravasi & Vasconcelos，2020年）SPGL1 求解器（Van Den Berg & Friedlander，2009）实现稀疏性。

A.2. 状态相似性图计算的详细信息：

MILCCI 能够解耦类别型、连续型和非连续有序类别。它还支持根据各类别对应标签之间的相似程度，使成分在标签变化的试次间进行调整。例如，如果我们假设神经元集群在学习过程中逐渐发生组成变化，就需要捕捉试次逐渐变化的有序顺序。另一个例子是，当某个类别表示一个连续变量（例如刺激的 x 位置）时，我们同样希望捕捉标签关系（即标签之间的距离）。

因此，MILCCI 为模型增添了一组标签驱动图，这些图在迭代优化过程开始之前预先计算，并在各次迭代中重复使用，以实现更平滑的跨组件正则化，同时保持标签相似性。对于每个类别 (k)，我们构建图 $\boldsymbol{\lambda}^{(\mathrm{k})} \in \mathbb{R}^{\#\mathrm{k}\times\#\mathrm{k}}$，其中 #k 表示类别 (k) 下观测到的不同选项数量（例如，如果类别“选择”可以为正确/错误，则 #k=2）。该图刻画了其可能取值之间的相似程度。

对于类别标签，我们为图使用一个常数值（例如，$\lambda_{i,\,i'}^{(\mathrm{k})} = 1 \;\; \forall i, \;\; i'$）。

对于有序标签：我们使用高斯核 $\lambda_{i,\,i'}^{(\mathrm{k})} = e^{\frac{\|\mathrm{k}_i - \mathrm{k}_{i'}\|_2^2}{2\sigma^2}} \;\; \forall i, \;\; i',$ 其中 k_i 和 $\mathrm{k}_{i'}$ 分别是类别 (k) 的第 i 个和第 i' 个选项（例如，任务难度为 1 与 5）。值得注意的是，MILCCI 支持集成多种图计算距离度量，因此，如果认为标签之间的相似性可以用不同方式来刻画，也可以轻松采用其他距离度量（非高斯距离度量）。

图计算完成后，我们建议按行将图归一化，使每行的绝对值和为 1，以确保不同标签受到相同程度的正则化：

$$\lambda_{i,:}^{(\mathrm{k})} \leftarrow \frac{\lambda_{i,:}^{(\mathrm{k})}}{\|\lambda_{i,:}^{(\mathrm{k})}\|_1} \;\; \forall i, i'.$$

i 中第 $\boldsymbol{\lambda}^{(\mathrm{k})}$ 行为零意味着，类别 (k) 的第 i 个选项不会经过正则化以与其他选项保持一致。如果有意创建完全随试次变化、能够在不同试次之间灵活变化的组件，则可以采用这种方式；这也是 MILCCI 提供的另一项功能。

A.3. 关于稀疏性与一致性超参数

与大多数机器学习模型一样，MILCCI 包含用于控制模型行为的超参数，不过其数量明显少于深度网络等复杂模型。MILCCI 中有两个关键超参数用于控制组件行为： γ_1（稀疏性）和 γ_2 （跨标签一致性）。

γ_1 控制 ℓ_1 组件隶属度上的正则化（公式2）。较高的 γ_1 值会产生更稀疏的组件，其中非零项更少，这对于可解释性至关重要，但可能会遗漏较弱的关系。较低的 γ_1值允许生成更稠密的组件，从而捕捉更多细微模式，但也可能包含噪声。 γ_2 通过对同一类别内的组件变体之间的距离进行 ℓ_2 正则化（公式 2），促进组件变体之间的相似性。较高的 γ_2 值会迫使变体保持几乎完全相同，从而失去捕捉标签驱动的调整的能力。较低的 γ_2 值允许更大的灵活性，但可能导致不同标签之间的组件差异过大。

我们建议通过考察 γ_1 的组件稀疏性和可解释性（例如，检查非零项的数量及其是否具有实际意义），并在训练后检验信息准则（例如，赤池信息准则、贝叶斯信息准则）来进行选择。对于 γ_2，我们建议分析同类别组件变体之间成对距离的分布，以确保它们保持相似，同时又允许有意义的调整。如果具备领域知识（例如，在神经科学中，我们可能会根据所记录的数据量估算

多少个神经元构成一个 出的组数），这还能进一步为超参数的选择提供指导 n.

B. 运行细节

B.1. 数据与代码可用性

本文使用的所有真实世界数据均可在线公开获取，来源已在相应章节中引用。模型实现与合成数据生成的代码将在发表后公开。

B.2. 版本

我们使用 Python 3.10.4（conda-forge）以及 matplotlib 3.8.2、scikit-learn 1.0.2、seaborn 0.11.2、NumPy 1.23.5、pandas 1.5.0、PyLops 1.18.2 和 SPGL1 0.0.2 训练了该模型。

C. 成分的弹性网络先验

在第 3.1 节中，我们规定成分矩阵服从拉普拉斯分布：

$$\boldsymbol{A}_{nj}^{(\mathrm{k})} \sim \text{Laplace}(0, \frac{1}{\gamma_1}) = \frac{\gamma_1}{2} \exp\left(-\gamma_1 |\boldsymbol{A}_{nj}^{(\mathrm{k})}|\right)$$

当我们随后将模型扩展到每个类别 k 的多个成分变体，并对它们之间的距离施加约束 ℓ_2（即，$\|\mathcal{A}_{n:i}^{(\mathrm{k})} - \mathcal{A}_{n:i'}^{(\mathrm{k})}\|_2 < \epsilon$）时，实质上采用了分层贝叶斯框架。因此，特定于变体的成分$\mathcal{A}_{n:i}^{(\mathrm{k})}$服从弹性网络先验：

$$p(\{\mathcal{A}_{n:i}^{(\mathrm{k})}\}) \propto \exp\left(-\gamma_1 \sum_{n,j} |\mathcal{A}_{njl}^{(\mathrm{k})}| - \gamma_2 \sum_{i' \neq i} \lambda_{i',i}^{(\mathrm{k})} \|\mathcal{A}_{::i'}^{(\mathrm{k})} - \mathcal{A}_{::i}^{(\mathrm{k})}\|_2^2\right)$$

第一项对应于继承自拉普拉斯先验的 ℓ_1 惩罚项，促进稀疏性。第二项对变体差异施加 ℓ_2 惩罚项，鼓励同一变体组内保持相似。这种组合产生了弹性网络正则化，它自然地源于分层结构：各变体通过共同的基础成分共享统计强度。

D. 补充信息—合成实验

我们生成了包含**80 个通道**、**4 个成分**（2 个类别，$\times$ 2 个成分按类别调整）、**每个试次 500 个时间点**和**250 个试次**的合成数据集。每个试次在每个轴上都有一个标签：类别 (a)（难度）的标签从集合 {I、II、III、VI、V}（5 个级别）中采样，类别 (b)（选择）的标签则从 {I、II}（2 个级别）中采样，由此最多可得到**10 种不同的标签组合**。参考试次的成分到神经元映射以 $[0.5, 1.0]$ 中的值初始化，随后根据标签变体进行更新；更新依据的是根据试次标签计算得出的试次相似性图，并将其阈值设为**第 60 百分位数**，以确保成分组成具有稀疏性。

每个标签对的时序活动均从采用 RBF 核的高斯过程先验中抽取，核按逐样本振幅进行缩放：振幅按样本在 $\approx [0.2, 1.533]$ 中抽取，核长度尺度按样本在$[0.05, 0.2]$ 中抽取，单位为归一化时间单位（**0–1**）；在每个试次有 500 个样本的情况下，这大致对应**25–100 个时间点**。核中包含 1×10^{-8} 的白噪声项。我们为每个标签抽取一个 GP样本，然后通过添加协方差按 σ^2 缩放的多元正态扰动，生成多个相似的试次轨迹，其中 $\sigma = 0.15$($\sigma^2 = 0.0225$)，因此，共享同一标签的试次会呈现相关动力学。

其中一个成分被指定为随机成分（随试次变化的成分）。对成分激活进行平移，使其变为非负值，并重新缩放，使其第 98 百分位数与成分映射的第 98 百分位数相匹配。观测数据的生成方式是

算法 1MILCCI 算法

输入： 观测试验数据 $\{\boldsymbol{Y}_{(m)}\}_{m=1}^{M}$，及其关联的多类别标签 $\{L^{(m)}\}_{m=1}^{M}$。
预先计算： 每个类别的标签相似度图 $\boldsymbol{\lambda}^{(\mathrm{k})}$（k）（附录 A.2）。
初始化： 稀疏成分 $\{\mathcal{A}^{(\mathrm{k})}\}_{\mathrm{k}\in\text{Categories}}$ 和轨迹 $\{\boldsymbol{\Phi}^{(m)}\}_{m=1}^{M}$（附录 A.1）
重复
  对于每个类别 (k)**执行**
    对于每个标签值 k_i **执行**
      计算标签为 k_i 的试次的残差
      求解 $\mathcal{A}^{(\mathrm{k})}{}_{::i}$，通过 LASSO ((2))实现跨标签一致性和稀疏性
      将每个分量归一化，使其总和为 1 ($\mathcal{A}^{(\mathrm{k})}{}_{:ji} \leftarrow \frac{\mathcal{A}^{(\mathrm{k})}{}_{:ji}}{\|\mathcal{A}^{(\mathrm{k})}{}_{:ji}\|_1}$)，以避免与 Φ 相关的尺度歧义
    结束循环
  结束循环
  对于每个带有标签 ℓ 的试次 m**执行**
    通过从每个 $\mathcal{A}^{(\mathrm{k})}$ 中选择一个变体，构建堆叠分量矩阵 $\boldsymbol{A}^{(\ell)}$
    更新轨迹 $\boldsymbol{\Phi}^{(m)}$ 以最小化数据保真度、平滑性和去相关（（3））
  结束循环
直到收敛

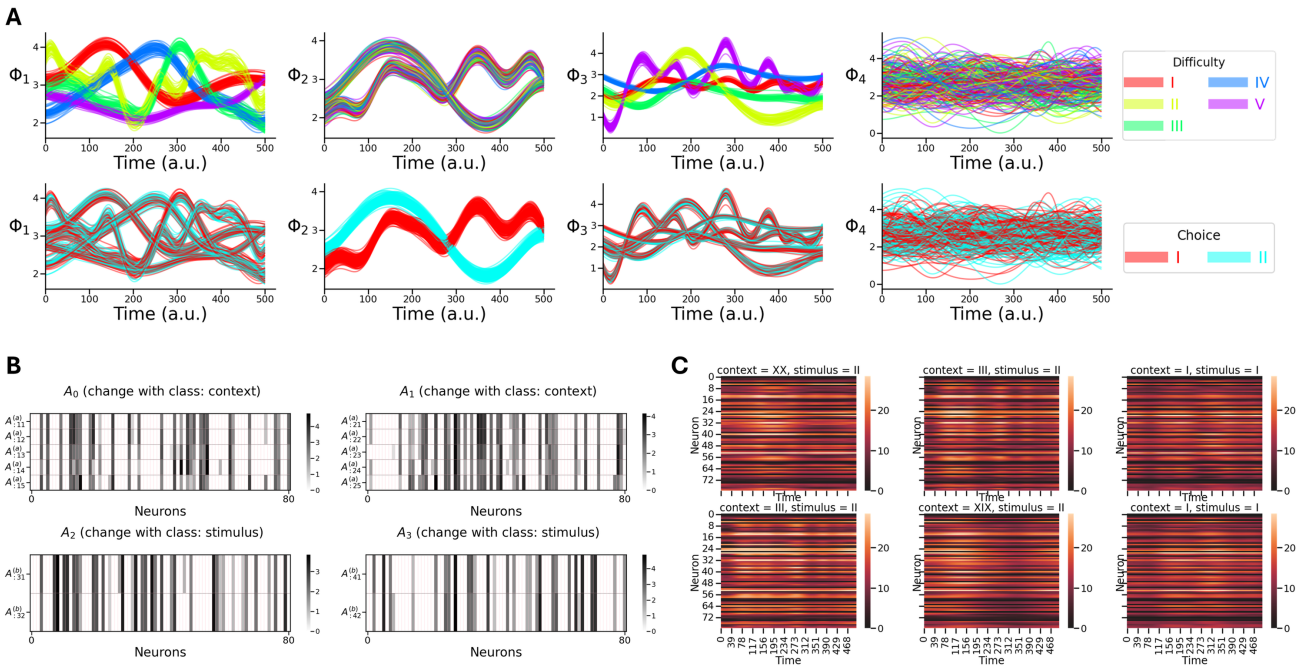

图*6*。**生成的合成数据**。**A：** 生成的轨迹，按难度（上）或选择（下）着色。**B：** 生成的成分。每个子图展示一个成分及其随各类别标签变化的情况（各行之间的变化）。换言之，每个子图对应于某个成分 $\mathcal{A}^{(\mathrm{k})}{}_{:\,j:\;\;j}$。**C：** 随机生成的合成试次示例 $\{\boldsymbol{Y}_{(m)}\}$。

将每个试次的分量到神经元的映射乘以该试次的时间激活，从而得到形状为 **(神经元 ×时间 × 试次) = (80 × 500 × 250)** 的数据。

E. 补充信息——投票实验

E.1. 投票数据预处理

数据来自（Data & Lab，2017a；b；c），其中包含51个州（包括华盛顿特区）的总统、参议院和众议院选举投票信息。总统和参议院选举数据集涵盖1976年至2020年

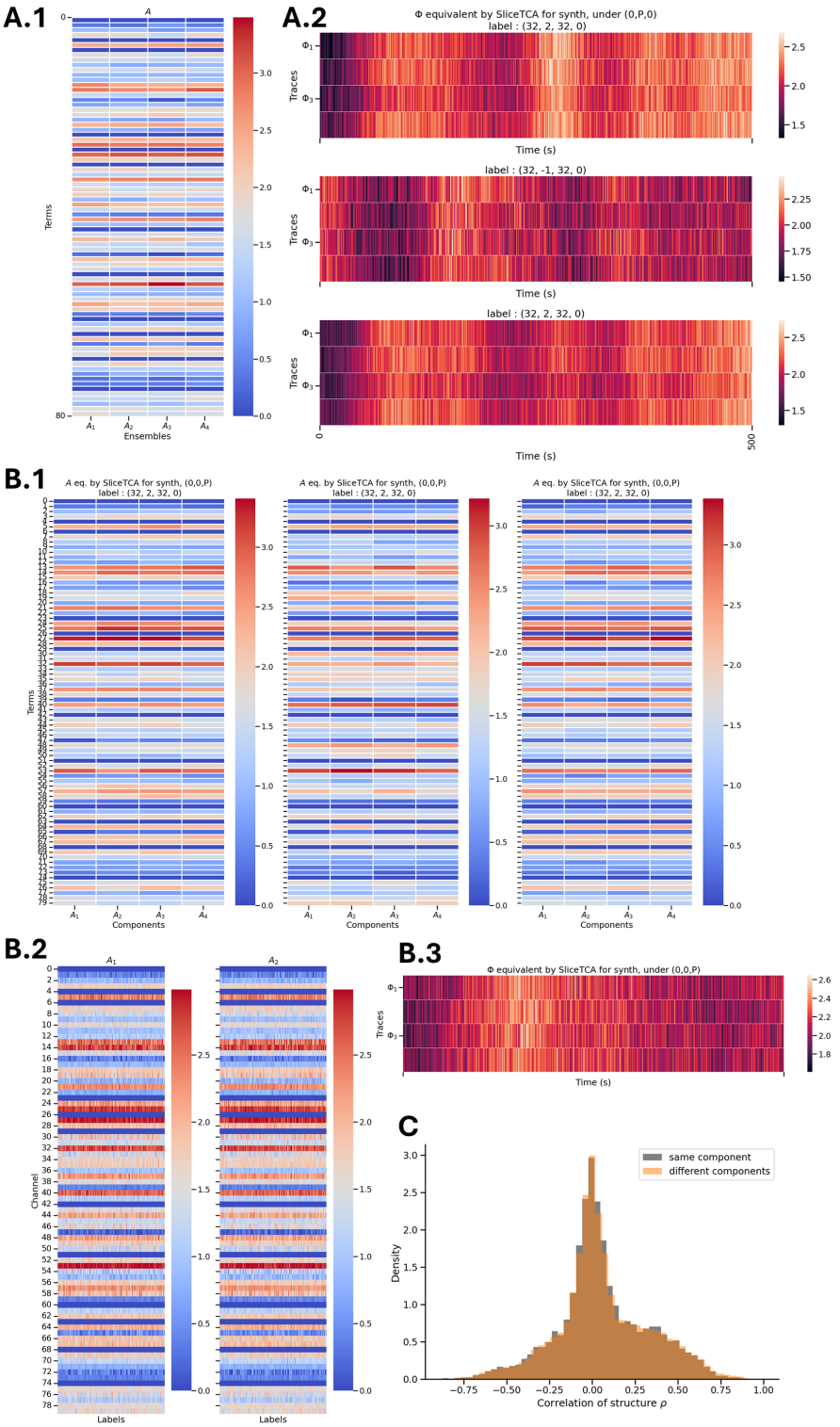

图7. **SliceTCA 在合成数据上识别出的组件和轨迹。A** 配置1（第 H.2 节）的结果，其中 **A.1** 表示组件，**A.2** 表示来自 sliceTCA ***A*** 矩阵的对应轨迹。 **B** sliceTCA 配置2的结果。 **B.1** 3 个示例试次的已识别组件；每个子图表示一个试次的所有组件。 **B.2** 展示已识别组件在不同试次间的变化。 **B.1 和 B.2** 共同表明，sliceTCA 识别出的组件彼此非常接近，其组件间变异与同一组件在不同试次间的变异相似。这说明不同试次中的组件未必会按身份进行匹配，如 **C** 面板所示，该面板展示了相同组件与不同组件之间的相关性分布。 **B.3** sliceTCA 在配置2下获得的时间轨迹，所有试次共享。

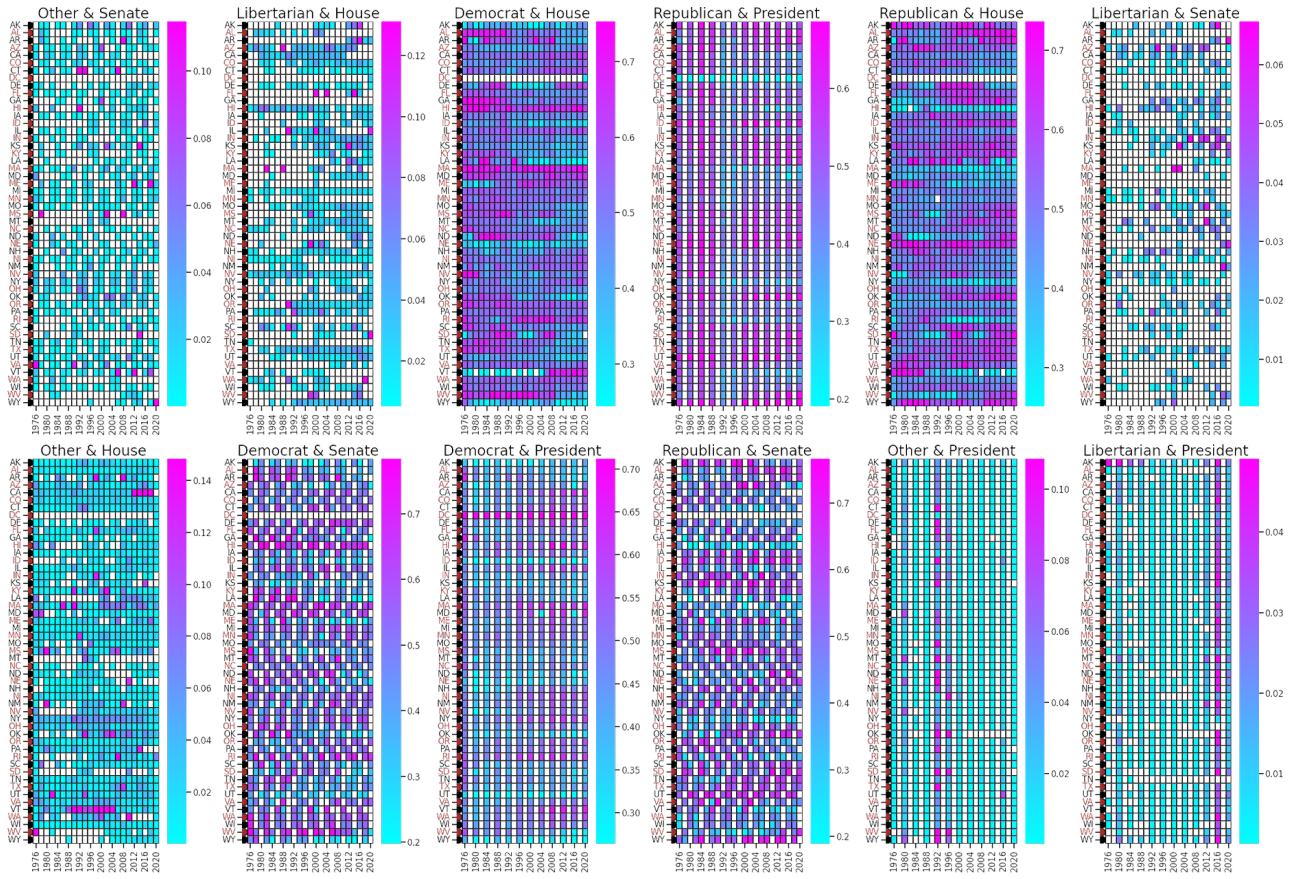

图 *8.* **投票数据**：($\{\boldsymbol{Y}_{(m)}\}_{m=1}^{M}$) 展示了多样的数据结构

选举；众议院选举则涵盖 1976 年至 2022 年。为确保涵盖所有职位类型，我们在分析中采用 1976 年至 2020 年的数据范围，因此众议院和参议院各有 23 个时间点，总统选举有 12 个时间点。对于每一年和每个州，我们统计各政党类别（民主党、自由意志党、共和党、其他）获得的选票总数，并分别除以该州当年选举的总投票数。我们排除了特别选举。我们设计模型来捕捉以下两个类别：1）政党（4 个选项）；2）公职（3 个选项）。我们对每个类别使用 $p^{(\mathrm{k})}=4$ 个成分运行 MILCCI，模型可根据类别进行结构调整，最终得到 $p=8$ 个成分。由于数据仅包含正值，我们对单元到组件的隶属关系和时间轨迹均施加了非负约束。

E.2. 其他发现——投票数据

还有一些 有趣的投票模式，除正文中讨论的内容之外，还可以在图 15 和 11 中找到 .

在成分 $\mathcal{A}^{(\text{party})}{}_{:4:}$ 中，华盛顿特区呈现为一个独立的簇。这很可能反映了其作为美国首都极为独特的政治特征：民主党占据压倒性优势，选民投票率也非常高，与其他州有显著差异。在 $\mathcal{A}^{(\text{office})}{}_{:2:}$（即总体上的第 6 个成分）中，与此前的成分不同，该成分的构成会因公职而异：西弗吉尼亚州（WV）在众议院和参议院投票中被纳入，但在总统职位投票中被排除。该成分反映了这些立法机构选举中的民主党主导地位。西弗吉尼亚州的这种区别或许与其历史投票模式相符：在地方和州级职位（众议院和参议院）选举中，该州更倾向于支持民主党人，而在总统选举中则逐渐倾向共和党人，这体现了选民行为因公职而异。查看所有按政党着色的轨迹（图 15 上方）时，民主党与共和党的差异是主导轴，始终推动着随时间变化最显著的区分。在一些轨迹中（$\boldsymbol{\Phi}_{:\mathcal{G}_1^{(\text{party})}}$、$\boldsymbol{\Phi}_{:\mathcal{G}_2^{(\text{party})}}$、$\boldsymbol{\Phi}_{:\mathcal{G}_3^{(\text{party})}}$），2000年前后出现了明显的分化，这表明两党从那时起开始更加鲜明地分立。这一趋势可能与2000年总统选举争议（布什对阵戈尔）的后续影响有关；该争议加剧了持久的党派分化，而这种分化可能还进一步

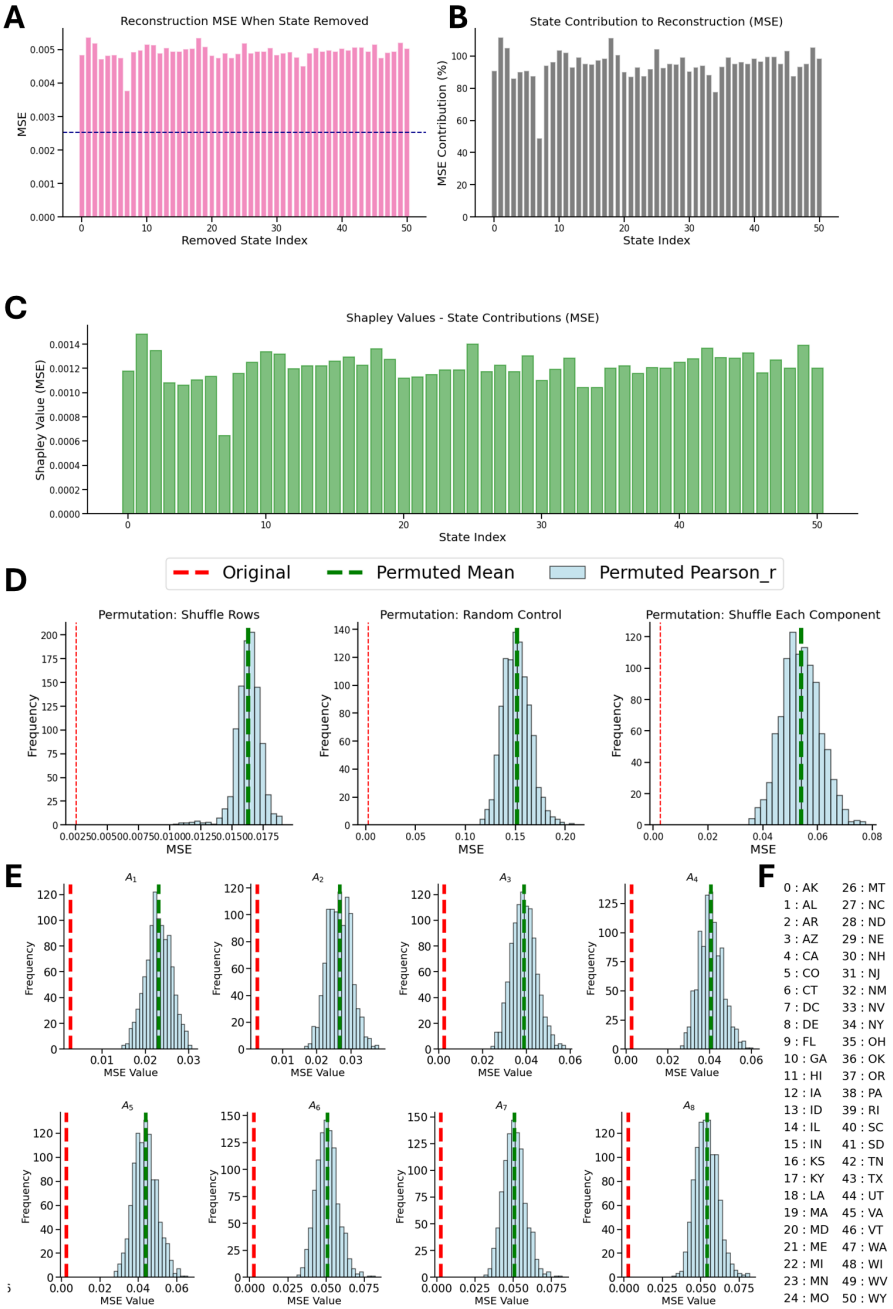

图*9*。 对MILCCI发现的投票组件进行事后验证（附录。 E.3）。**A:** 留一分析显示逐个移除每个州时的重建均方误差，基线均方误差（虚线）表示未省略任何数据时的模型性能。**B:** 各州对重构的单独贡献，即移除每个州后模型性能下降的程度（$100 * \frac{\text{均方误差}_{\text{有省略}} - \text{均方误差}_{\text{均方误差}}}{\text{均方误差}}$）。**C:** Shapley值衡量每个州对整体重构的贡献。**D:** 置换检验将原始重构误差（红线）与三个零假设进行比较：在各行之间打乱州分配（左）、用随机噪声替换数据（中）以及在每个组件维度内打乱各州（右）。所有检验均显示$p < 0.001$。**E:** 逐组件置换结果表明，所发现的组件分别具有稳健性。**F:** 状态缩写对照表。

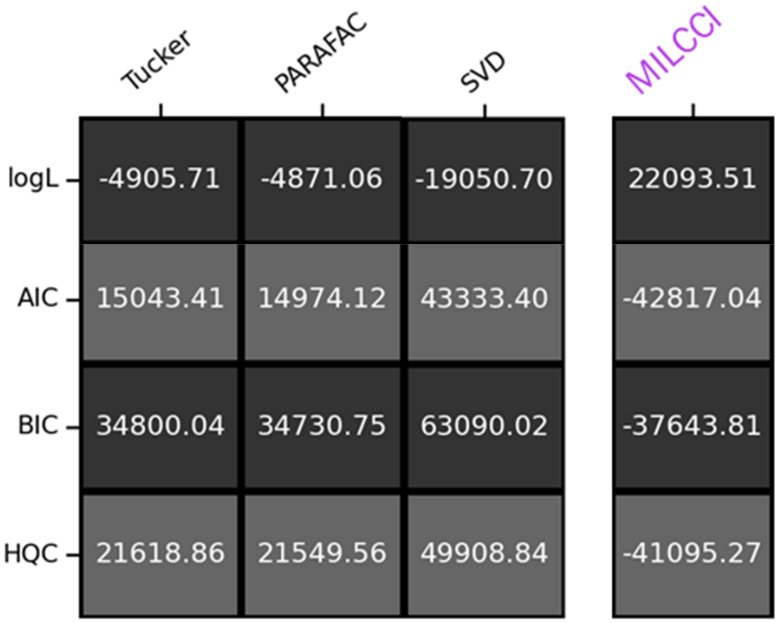

图 *10.* **投票实验。** 在重构和信息准则方面与基线方法进行比较。比较 Tucker、PARAFAC 和奇异值分解在重构和信息准则方面的表现，包括：1）根据已识别组件计算的观测值对数似然；2）平衡重构效果与模型复杂度的信息准则：AIC（赤池信息准则）、BIC（贝叶斯信息准则）和 HQC（汉南-奎因信息准则）。数值越低表示性能越好。

在 9·11事件之后，意识形态极化加剧了这一趋势。其他轨迹，例如 $\mathbf{\Phi}_{:\mathcal{G}_2^{(office)}:}$ 和 $\mathbf{\Phi}_{:\mathcal{G}_4^{(office)}}$，则显示出方向相反、随时间波动幅度更大的变化，这可能反映了政党之间更深层、由来已久的历史差异或结构性差异。轨迹 $\mathbf{\Phi}_{:\mathcal{G}_3^{(office)}}$ 似乎捕捉到了短期变化或噪声，包括政党投票行为逐年出现的峰值。相比之下，基于公职的投影（图 15 ）呈现的分离现象要少得多。各条轨迹大幅重叠，且置信区间较宽，这表明选举行为受职位类型的区分程度低于党派归属。

E.3. 投票数据的事后验证分析

为验证 MILCCI 发现的是真实的投票结构，而非虚假的相关性，我们进行了全面的统计分析，以考察各个州的贡献，并检验多个零假设（图 9）。

我们通过将某州的贡献归零，从成分矩阵中移除该州，并测量重构性能的下降幅度。对于 N 个州中的每一个，我们都使用经过修改、排除目标州的成分计算重建均方误差。图 9A 显示移除各州后的重建均方误差，并以虚线标出基线均方误差，表示纳入所有州时的性能。图 9B 展示了各州的贡献，计算方式为相对于基线性能的均方误差百分比变化。

我们使用博弈论 Shapley值计算各州的公平贡献，该方法会考虑州的所有可能组合。由于 N=52 个州带来的计算限制（需要 2^{52} 次计算），我们使用 500 个随机组合进行近似。对于每种组合，我们仅在成分矩阵中纳入该组合的成员，并将组合之外的州归零；随后计算每个州的边际贡献，即将该州加入组合前后重构质量的差异。图 9C 显示 Shapley值，用于量化各州对整体重构质量的单独贡献。

我们检验三个零 假设，通过将原始重建性能与随机化版本进行比较 s:

打乱行 （图 9D，左）：
我们随机将完整的州投票概况重新分配到不同的州位置。该检验用于判断地理州与其投票行为模式之间的特定对应关系是否有意义。

随机对照 （图9D，中）：
我们替换整个 com ponents矩阵，其随机值取自具有相同全局mea的正态分布 n

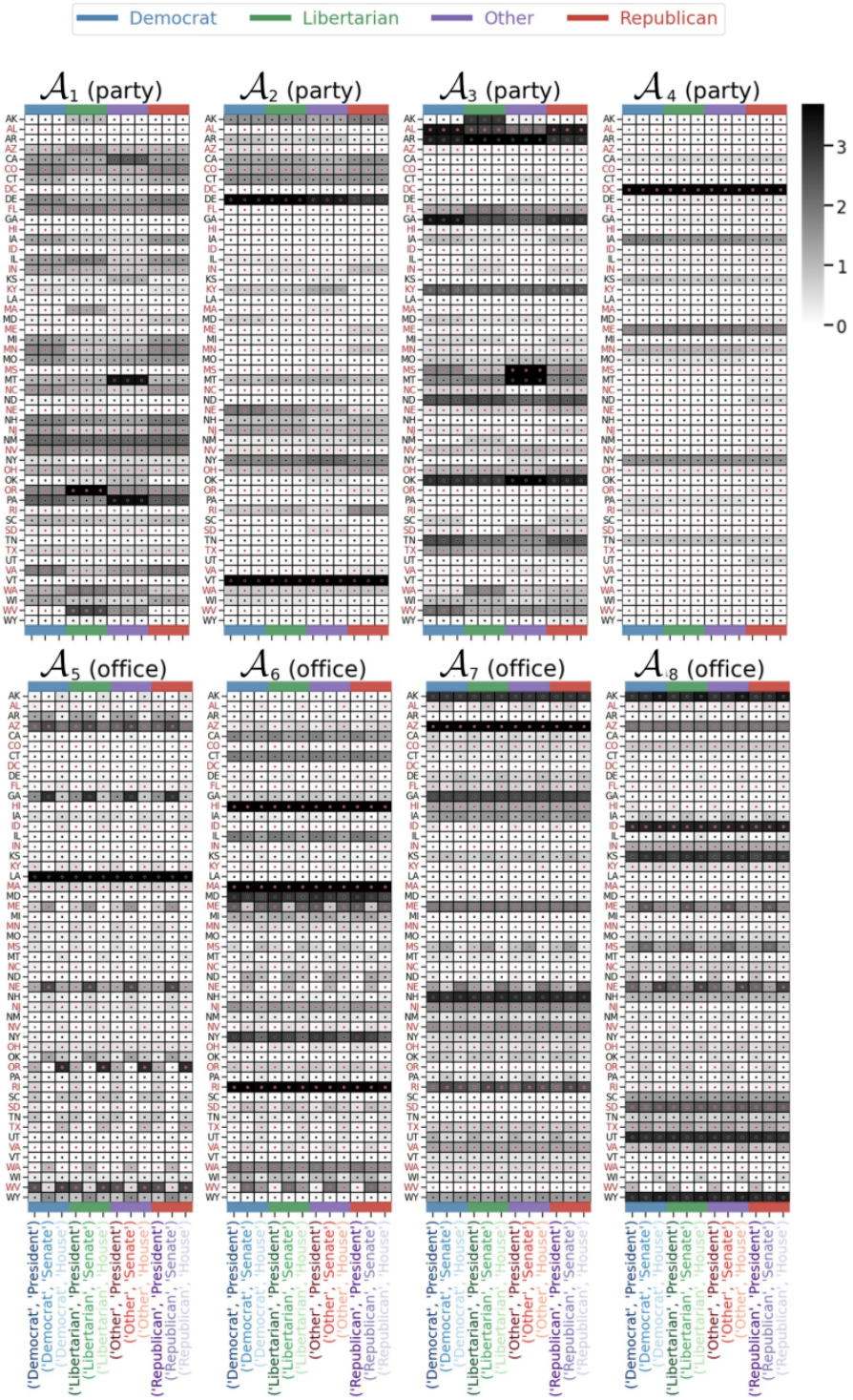

图 *11.* 投票实验的已识别集成。

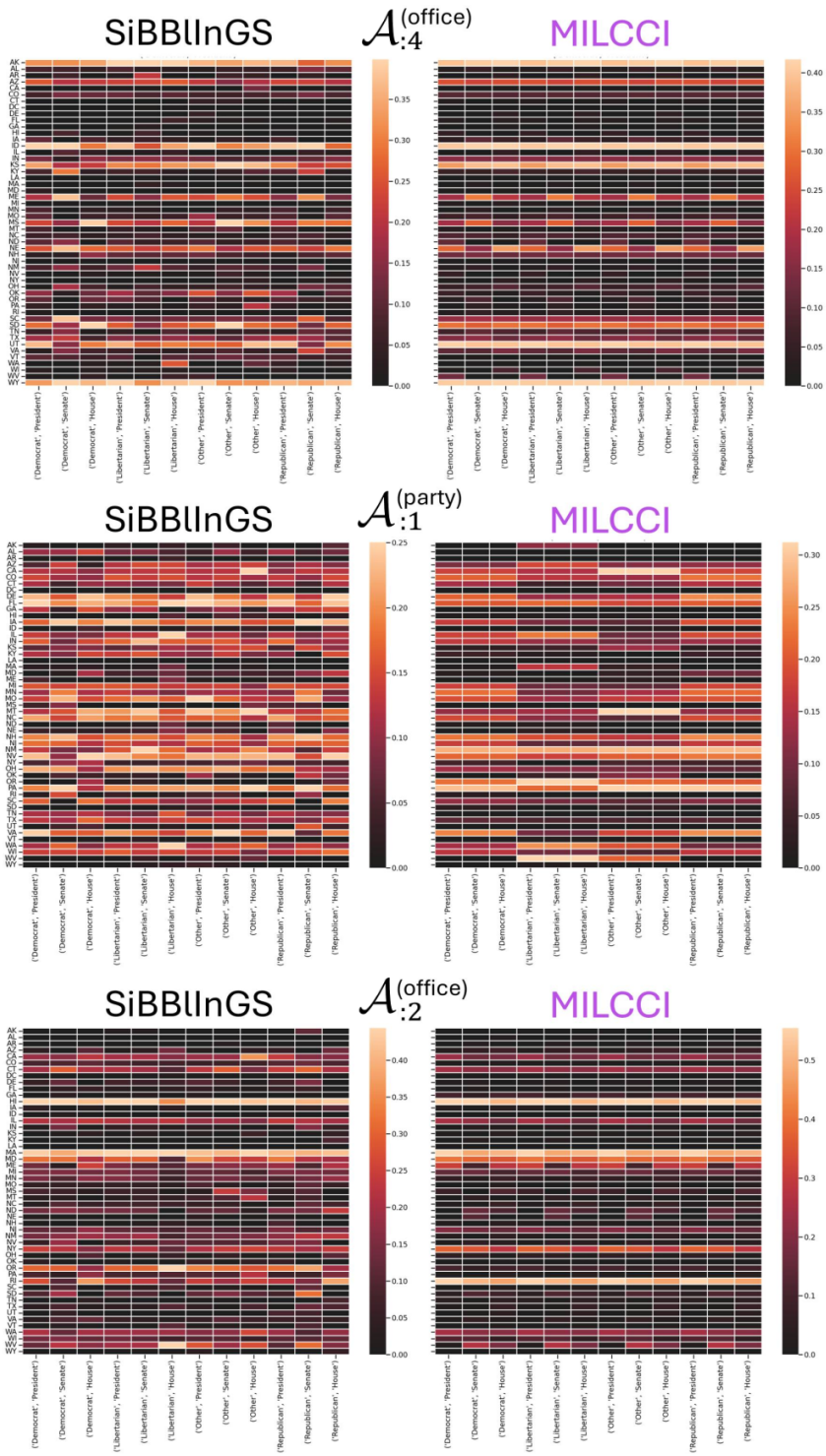

图*12*。 在相同参数和随机种子下，比较 MILCCI 与 SiBBlInGS 针对三个示例标签识别出的成分。为匹配 SiBBlInGS 的 x轴刻度标签，对 MILCCI 的成分进行了复制。SiBBlInGS 在每个标签下都呈现出难以解释的变化，即使某些部分是共享的；而 MILCCI 则将它们分离开来。

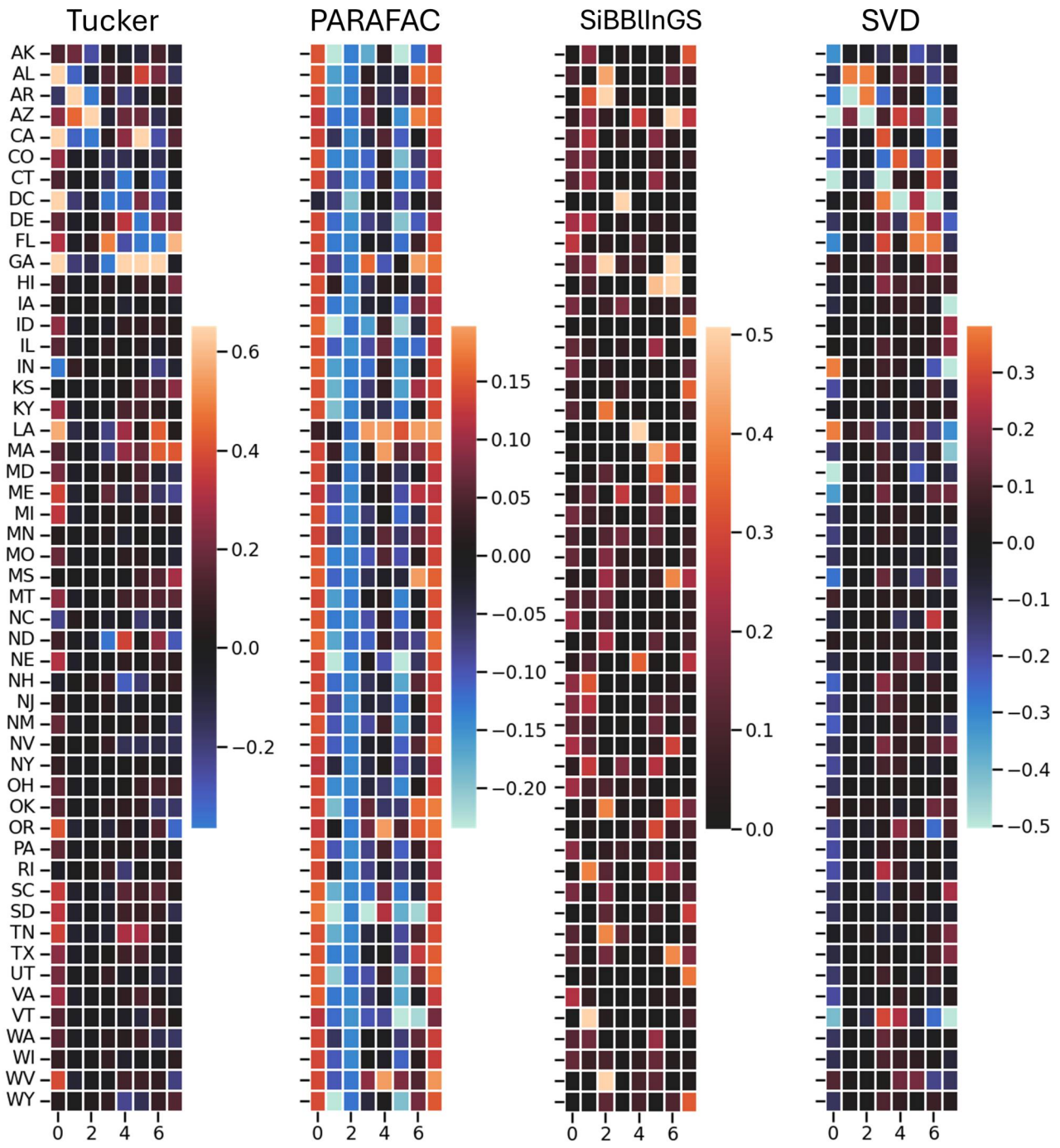

图*13.* **投票实验基线比较**。以下基线方法所识别的成分：1）Tucker分解（高阶奇异值分解），2）PARAFAC，3）SiBBlInGS（针对单个随机标签条目：（民主党，总统）），4）奇异值分解（对所有拼接的试次）。

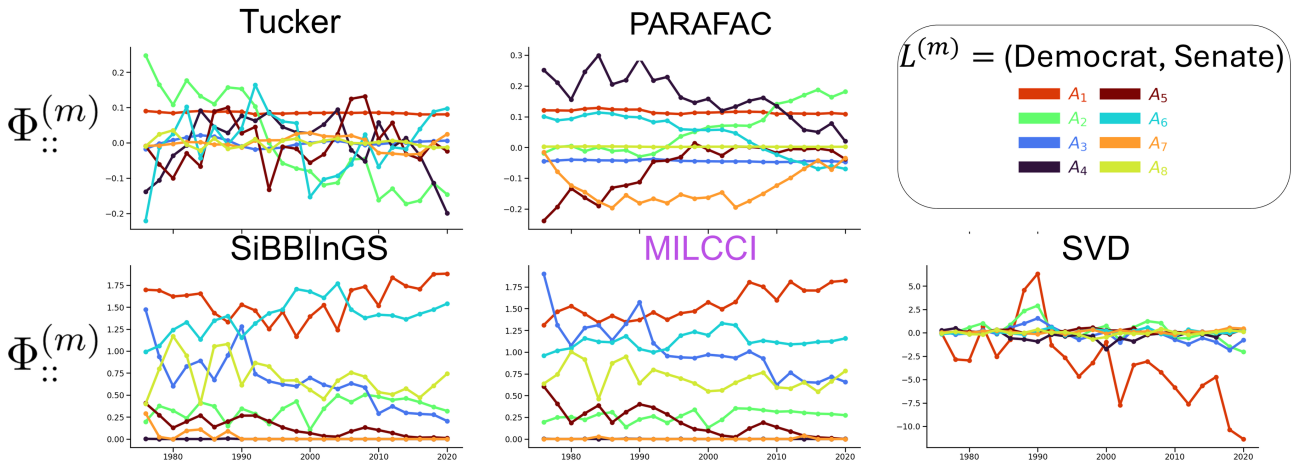

图 *14.* **投票实验。** MILCCI 识别的轨迹与其他基线方法在随机试次示例中的对比。

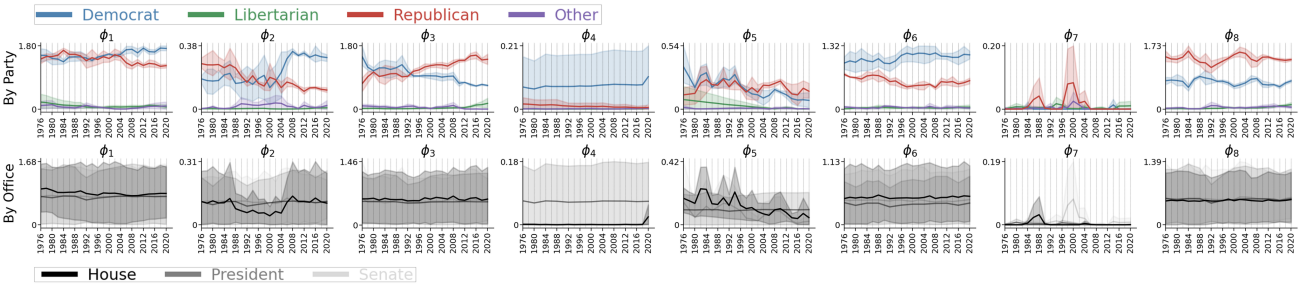

图 *15.* **投票轨迹。** 上：按政党着色。下：按公职着色。

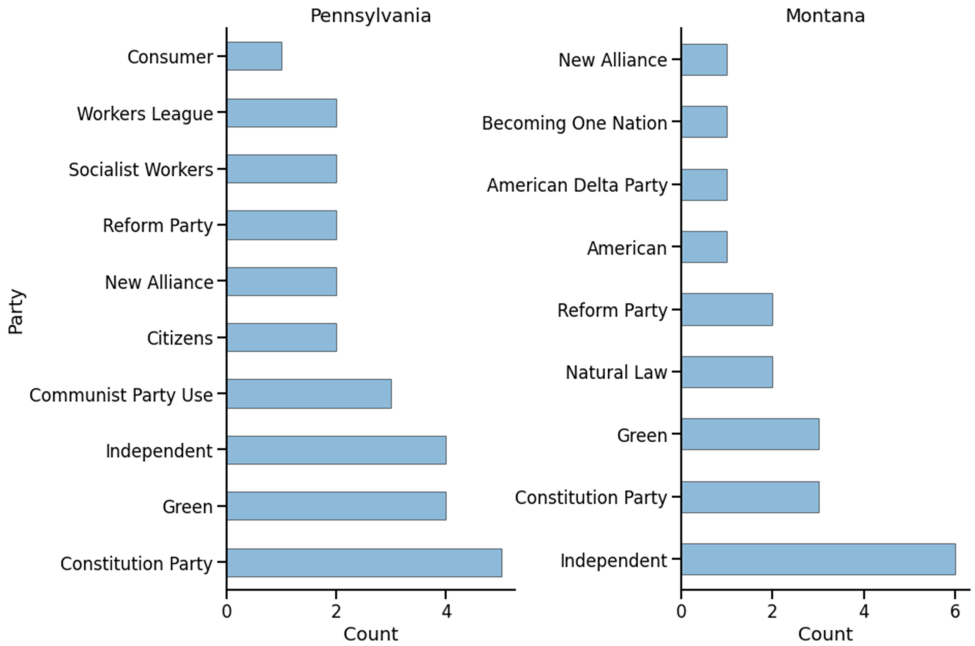

图 *16.* 蒙大拿州和宾夕法尼亚州“其他”政党的实例计数

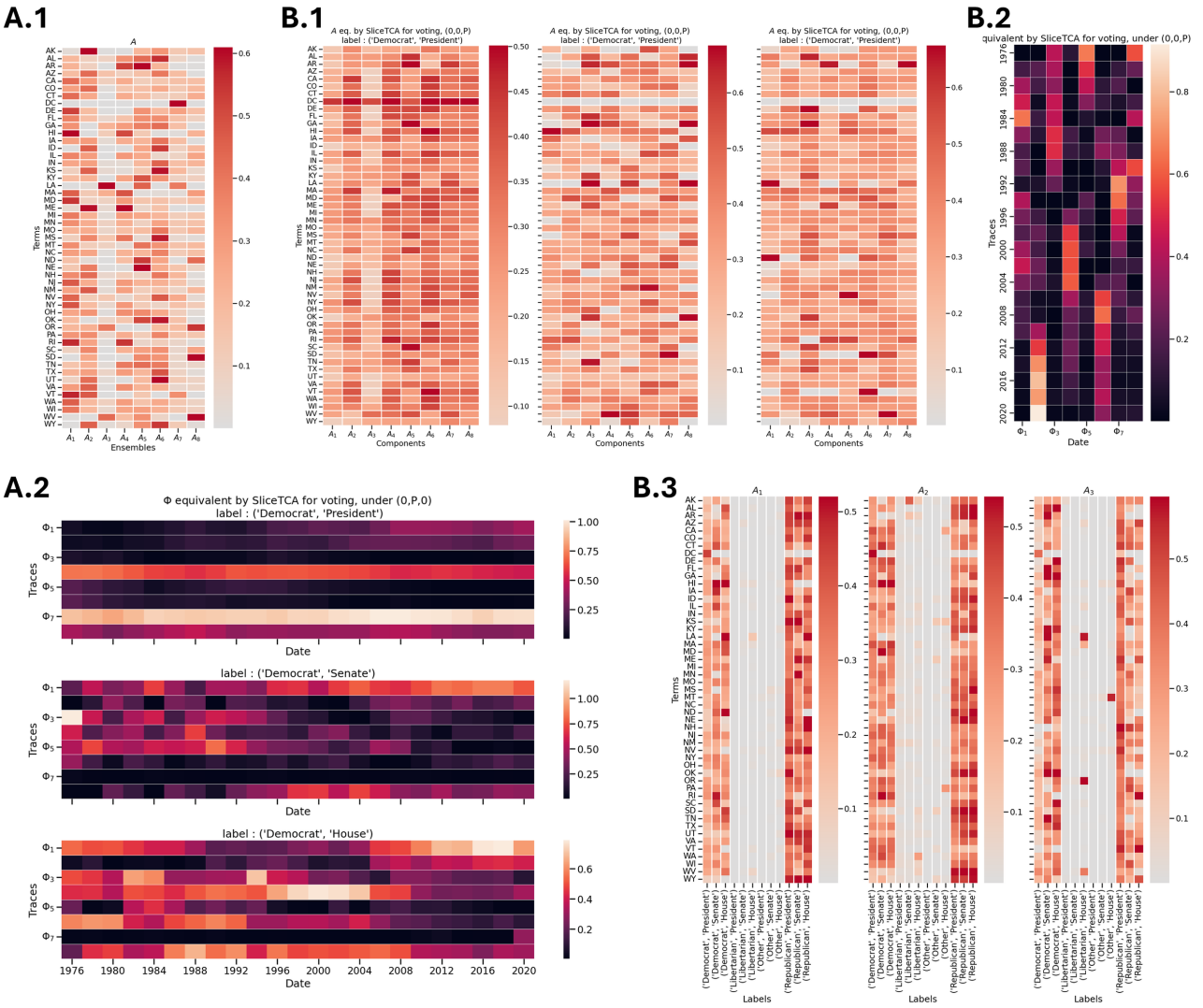

图*17*。 **投票实验：SliceTCA 对比**。**A.1** 固定成分情况中的成分（即来自 u）。**A.2** 固定成分情况中的轨迹（即来自 A）。**B.1** 变化成分情况中的成分（即 B）。每个子图代表一个试次，列表示该试次的不同成分。**B.2** 变化成分情况中的成分（即 v）。**B.3** 变化成分情况中的成分。每个子图代表一个成分，列表示各个试次。与不同成分之间相比，我们可以观察到，同一成分（B.3）在不同试次之间并不一致。

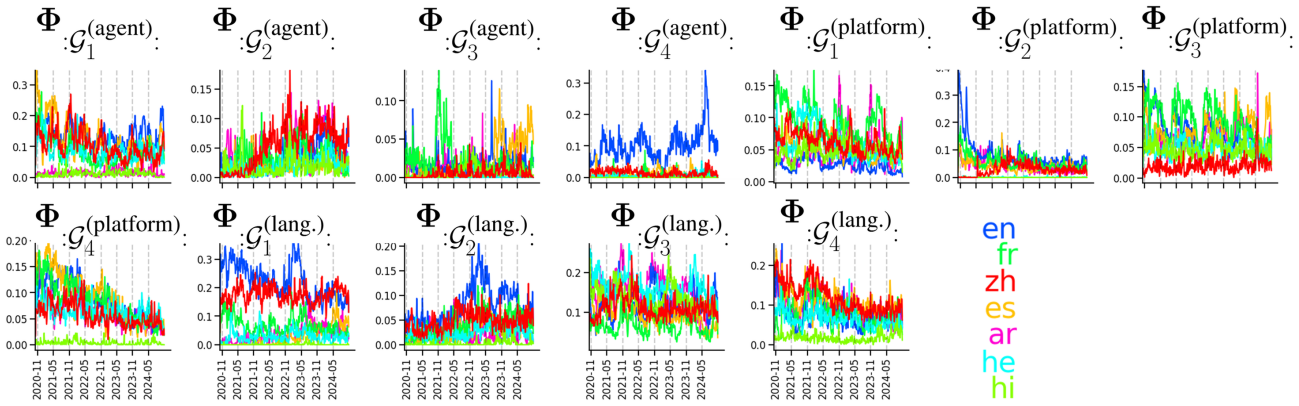

图*18.* **维基百科页面浏览量实验**。所有集成中按语言（阿拉伯语、英语、西班牙语、法语、希伯来语、印地语、中文）着色的轨迹。

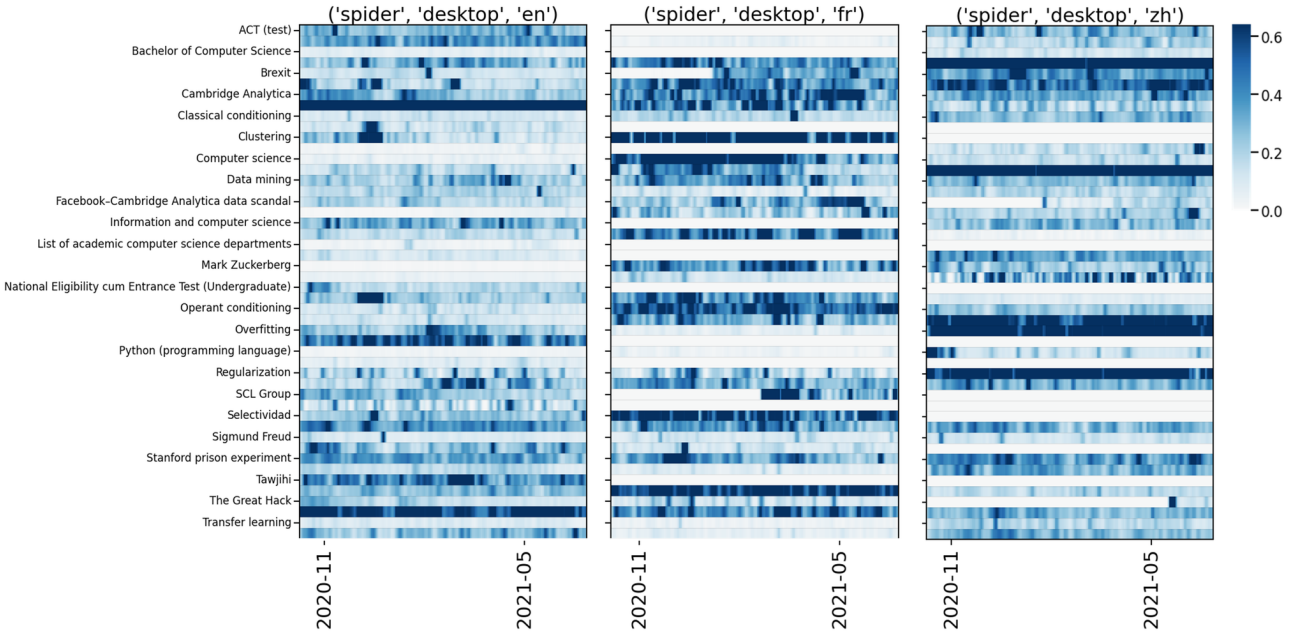

图*20.* 维基百科页面浏览量数据，示例试次

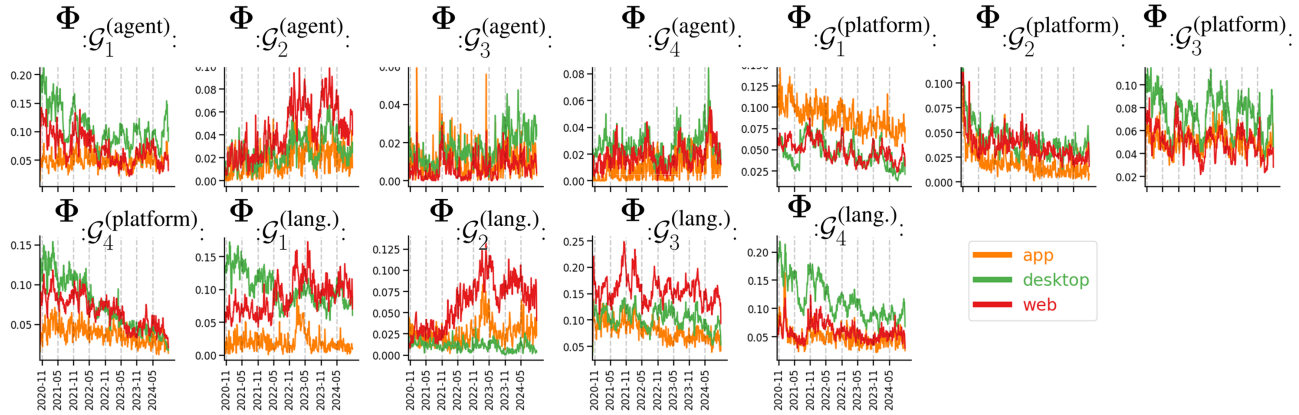

图*20.* **维基百科页面浏览量实验**。所有集成中按平台（桌面端、移动网页、移动应用）着色的轨迹

缩写	州	缩写	州
AL	阿拉巴马州	AK	阿拉斯加州
AZ	亚利桑那州	AR	阿肯色州
CA	加利福尼亚州	CO	科罗拉多州
CT	康涅狄格州	DE	特拉华州
DC	哥伦比亚特区 佛罗里达州		佛罗里达州
GA	佐治亚州	HI	夏威夷州
ID	爱达荷州	IL	伊利诺伊州
IN	印第安纳州	IA	IOWA
KS	堪萨斯州	KY	肯塔基州
LA	路易斯安那州	ME	缅因州
MD	马里兰州	MA	马萨诸塞州
MI	密歇根州	MN	明尼苏达州
MS	密西西比州	MO	密苏里州
MT	蒙大拿州	NE	内布拉斯加州
NV	内华达州	NH	新罕布什尔州
NJ	新泽西州	NM	新墨西哥州
NY	纽约州	NC	北卡罗来纳州
ND	北达科他州	OH	OHIO
OK	俄克拉何马州	OR	俄勒冈州
PA	宾夕法尼亚州	RI	罗得岛州
SC	南卡罗来纳州	SD	南达科他州
TN	田纳西州	TX	得克萨斯州
UT	UTAH	VT	佛蒙特州
VA	弗吉尼亚州	WA	华盛顿州
WV	西弗吉尼亚州	WI	威斯康星州
WY	怀俄明州		

表 2. 美国各州及其缩写列表

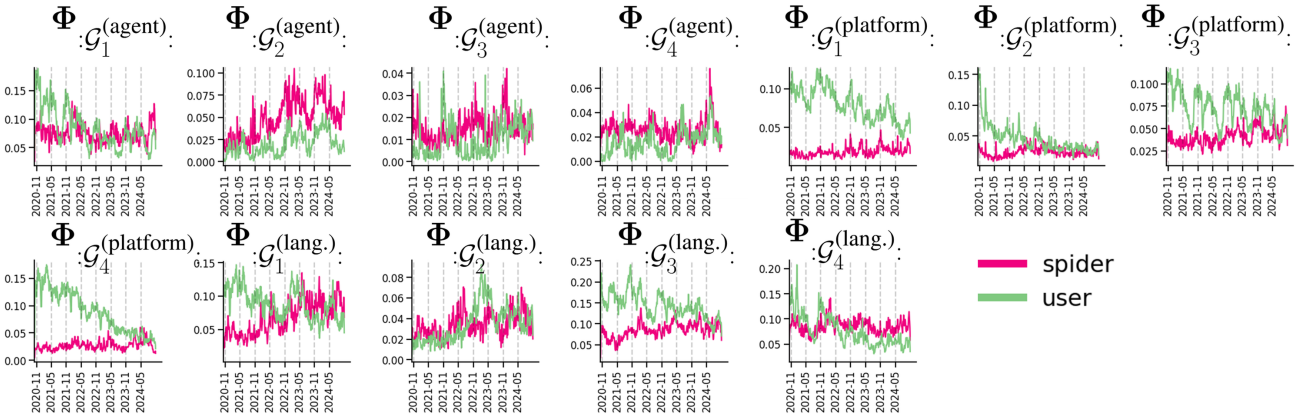

图21。 **维基百科页面浏览量实验**。按代理（爬虫与用户）着色，涵盖所有集成

并具有与原始数据相同的标准差。这用于检验纯统计噪声基线。

打乱每个成分（图 9D，右）：
对于每个组件维度，我们随机置换该组件内状态的分配，同时保留逐组件统计量。这用于检验每个状态内成分之间的具体协调是否重要。

每次置换检验运行 1000 次迭代，p 值计算为达到不低于原始结果性能的置换重构所占的比例。

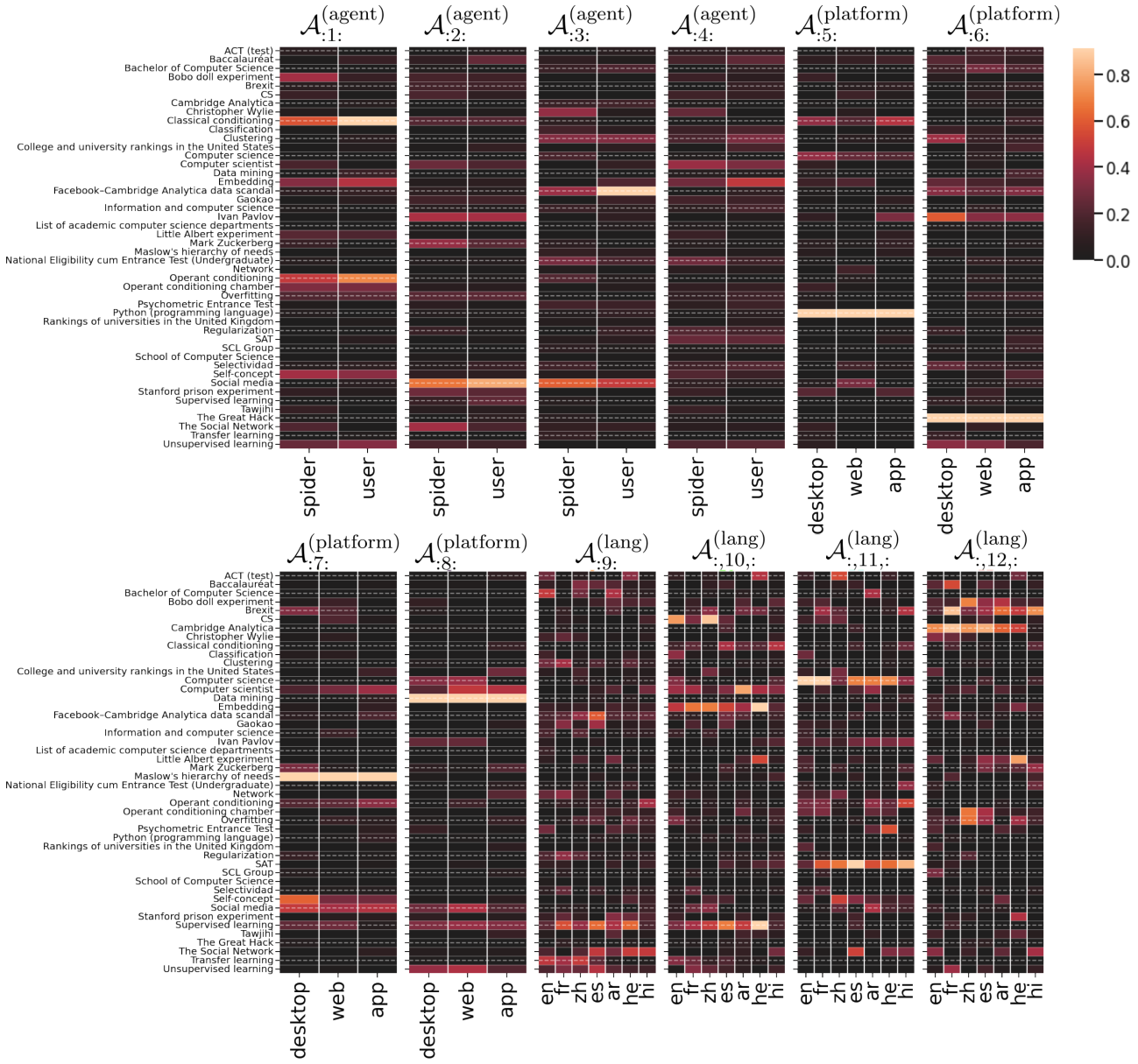

图22。 为维基百科页面浏览量实验识别的成分。

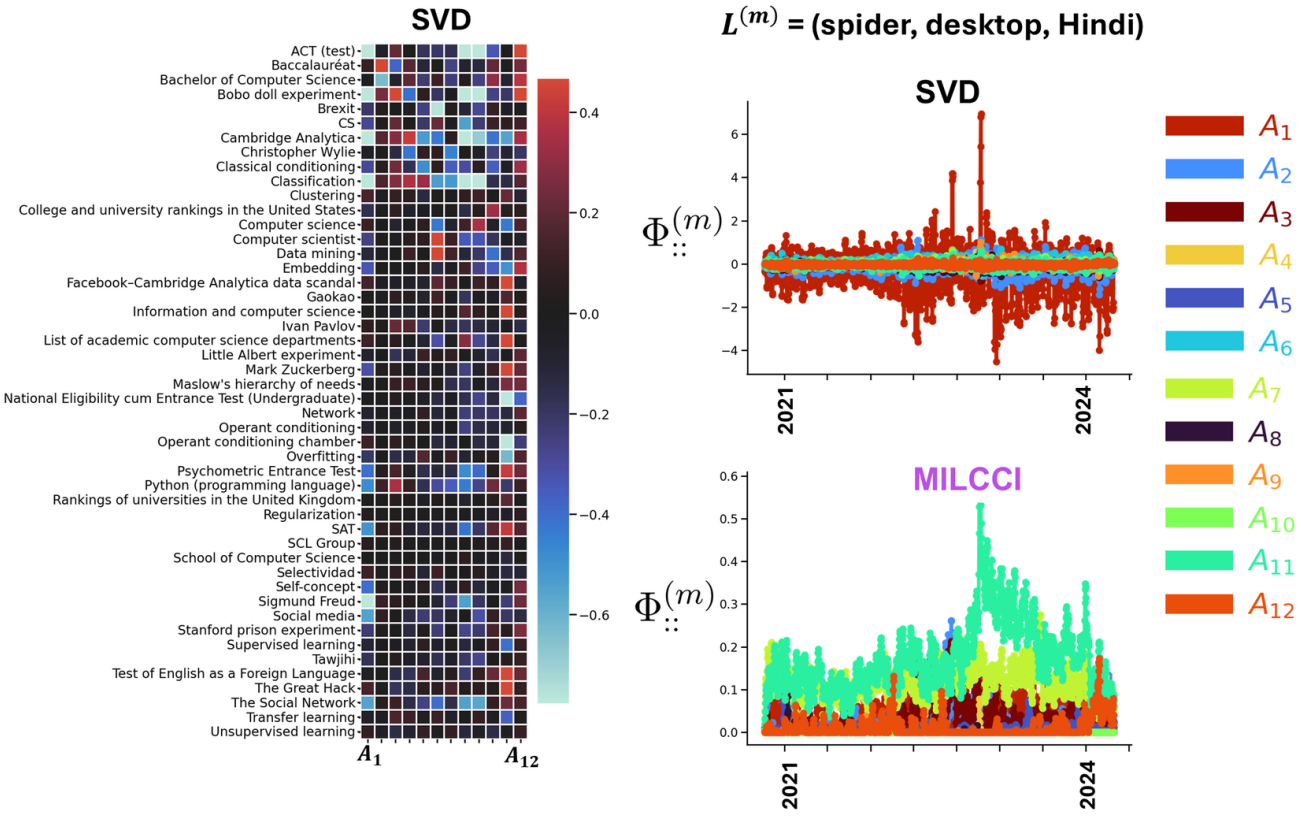

图*23.* **维基百科实验与奇异值分解的比较**。奇异值分解识别出的成分（左侧为组成，右侧为示例试次轨迹）。

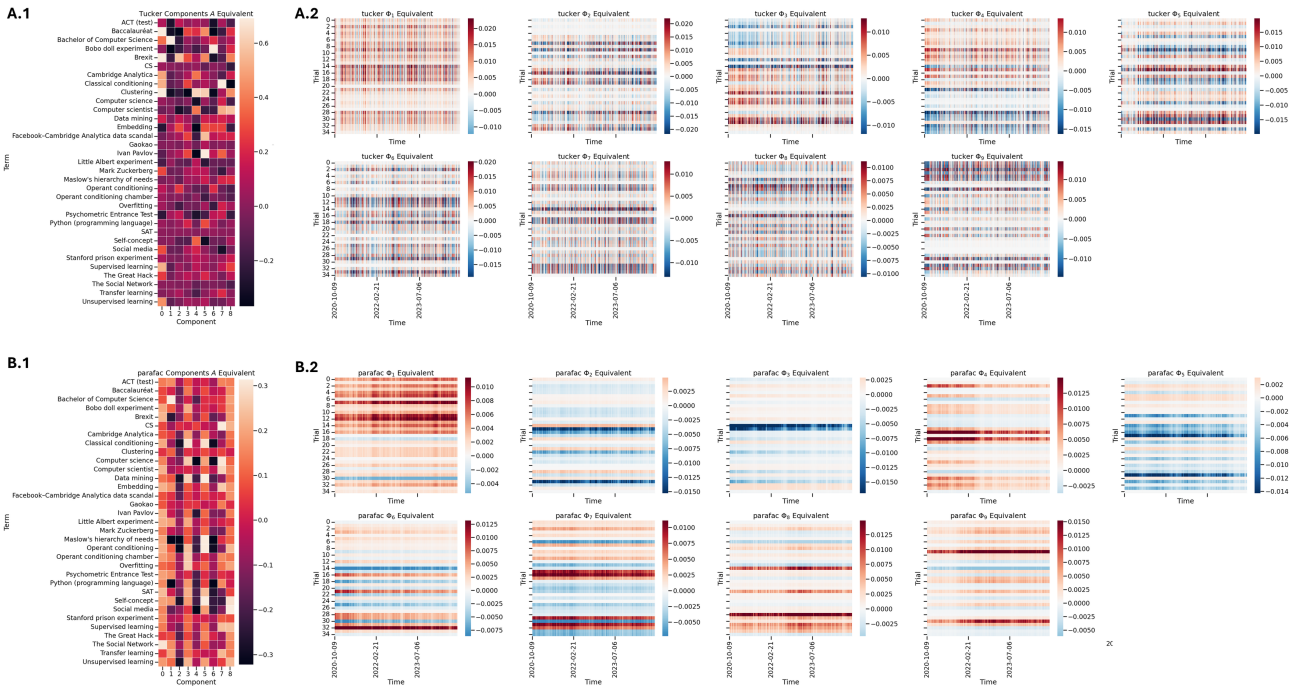

图*24.* 维基百科实验中由 Tucker（**A.1**,**A,2**）和 PARAFAC（**2.1**,**B.2**）识别出的成分，参见附录 H

表 3. 政党详细版本与简化版本。“其他”仅包括在所有年份和各州中出现次数至少为 10 次的政党。

民主党	共和党	其他	自由意志党
民主党；民主党-农民-劳工党；民主党-无党派联盟；民主-Npl；民主党（未在选票上标识）	共和党	禁酒党；独立党；美国党独立党；U.S. 劳动；社会党工人党；美国党；保守党；社会主义劳工党；独立党美国党；宪法党；社会党；自由联盟；政治家；公民；New 联盟；工人世界；工人联盟；独立；民粹主义者；请愿提名；基层；No 政党 隶属关系；绿党；自然法党；无党派；其他；工薪家庭党；联盟；无党派；宪法党；美国-ican 独立党政党；共产党政党 Use；和平与自由；纳税人-纳税人党；改革党；U.S. 纳税人党；社会主义与解放党；美国党特拉华州政党；美国团结政党；社会主义和解放党；成为 One Nation	自由意志党

组件特异性置换结果（图 9E）表明，所发现的模式在各个独立投票组件中都具有稳健性。我们存储了每个成分的中间重构结果，从而可以检验各成分对随机化的特异性稳健性。图 9F 提供了状态缩写对照表以供参考。

所有统计检验均得出 $p < 0.001$，这表明 MILCCI 发现的投票模式具有真实结构。

F. 补充信息——维基百科实验

F.1. 维基百科页面浏览量数据预处理

我们提取了 2020年10月9日 至 2024年10月29日的维基百科每日页面浏览量数据（$T = 1482$ 个时间点），涵盖了与大学、计算机科学、机器学习和心理学专业相关的 48 个不同页面（“词条”）（（Meta，2022））。值得注意的是，我们特意选择了既有推论变量又有协变量的主题。针对每个词条，我们分别收集了三个访问平台的数据：（1）桌面端、（2）移动网页和（3）移动应用程序。我们还区分了访问数据的代理：1）用户或 2）爬虫（由于应用程序和爬虫组合的稀疏性极高，爬虫数据仅从网页和桌面平台提取）+。

我们重点研究了代表世界不同地区的七种语言：英语 (en)、中文 (zh)、西班牙语 (es)、印地语 (hi)、阿拉伯语 (ar)、法语 (fr) 和希伯来语 (he)。为确保可比性，我们使用第99百分位数对每种语言在所有术语和时间点上的数据进行范围归一化，以减少离群值影响：$\boldsymbol{Y}_{:,:,l} \leftarrow (\boldsymbol{Y}_{:,:,l} - \min(\boldsymbol{Y}_{:,:,l}))/\mathrm{perc}(\boldsymbol{Y}_{:,:,l}, 99)$，其中$Y_{:,:,l}$是语言 l的完整数据集。

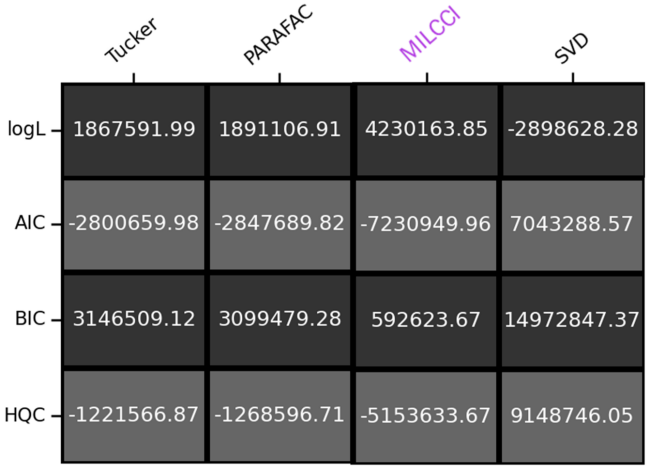

图*25.* 维基百科实验的聚类信息准则：MILCCI 与基线方法对比。值得注意的是，由于奇异值分解的不稳定性，PARAFAC 和 Tucker 无法收敛到 12 个成分。值得注意的是，PARAFAC 和 Tucker 遇到了不稳定问题，因此此处展示的 Tucker 和 PARAFAC 结果分别是秩为 9 并添加了 $\sigma = 0.1$ （Tucker）和 $\sigma = 0.2$ （PARAFAC）噪声的结果。

随后，使用相同的第99百分位数方法，对每个术语在所有语言和时间点上的数据进行归一化：$\boldsymbol{Y}_{k,:,:} \leftarrow (\boldsymbol{Y}_{k,:,:} - \min(\boldsymbol{Y}_{k,:,:}))/\mathrm{perc}(\boldsymbol{Y}_{k,:,:}, 99)$，其中$\boldsymbol{Y}_{k,:,:}$表示术语 k 在各种语言中的完整时间序列。这样，数据中总体上有三个可进行构成调整的类别：(1) 代理（用户或爬虫）、(2) 平台（桌面端 / 移动网页 / 移动应用）以及 (3) 语言（上述七种语言之一）。

F.2. 研究发现的澄清——维基百科数据

正文中提及的各成分术语列表（第4节）：

- $\mathcal{A}_{(\text{代理}),:,1}$：**（心理学）**：经典条件作用；波波娃娃实验；操作性条件作用；自我概念；小阿尔伯特实验；无监督学习；嵌入；
- $\mathcal{A}_{(\text{代理}),:,2}$：**（社交媒体）**：社交网络（电影）；社交媒体；伊万·巴甫洛夫；马克·扎克伯格。
- $\mathcal{A}_{(\text{平台}),:,4}$：**（计算机科学）**：数据挖掘；计算机科学；监督学习；无监督学习；计算机科学家；社交媒体；

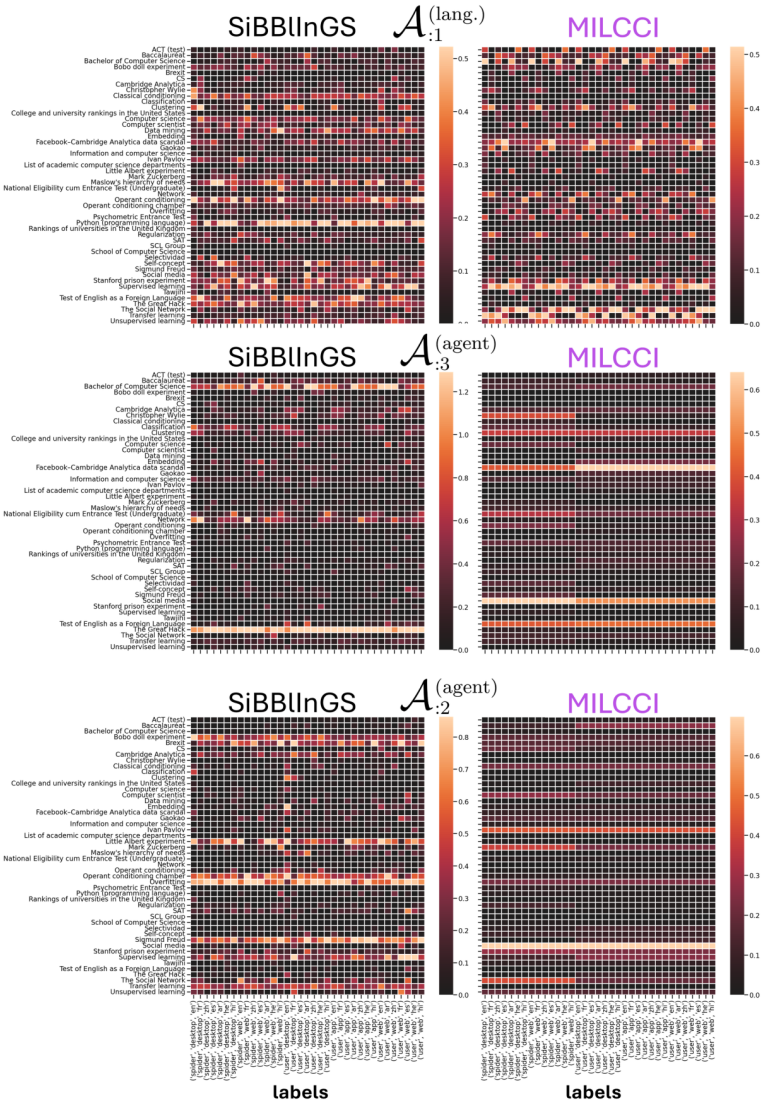

图*26.* **维基百科成分：MILCCI 与 SiBBlInGS**。SiBBlInGS 成分呈现出散布在各标签中的组成变化，而非 MILCCI 捕捉到的类别特定调整。注意： 此处显示的 MILCCI 成分包含重复列，以便与 SiBBlInGS 成分对齐，进行可视化。

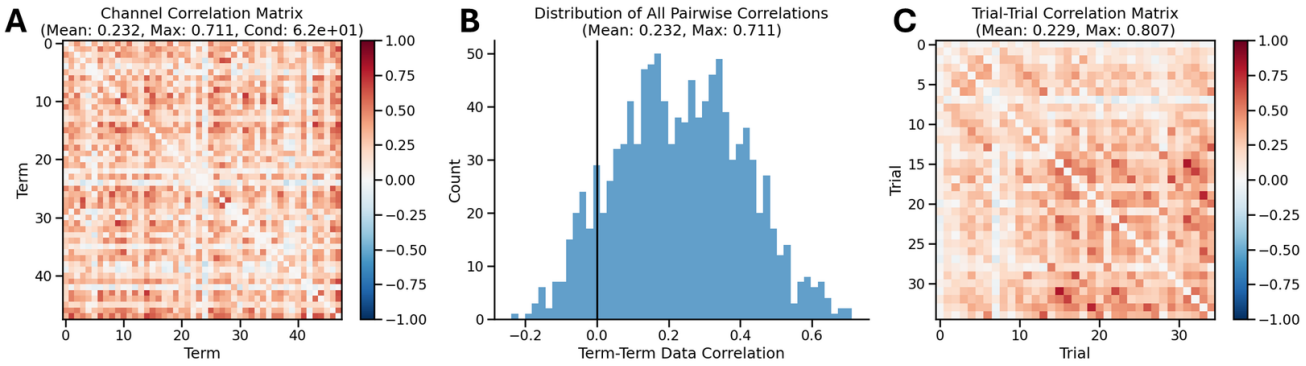

图*27.* 对维基百科数据集的相关性分析显示中等程度的相关性，但这并不能解释基线收敛失败。(A) 通道（词项-词项）相关矩阵显示中等相关性，最大值为0.711，且条件数良好（6.2e+01）。(B) 所有两两相关性的分布表明，大多数相关性为中等水平，不存在高相关性（>0.8）。(C) 试次间相关矩阵显示出类似的中等相关模式（最大值为0.807）。该数据为满秩（48/48），且相关结构合理，表明基线收敛失败（Tucker、PARAFAC）源于算法局限性，而非有问题的数据特征。

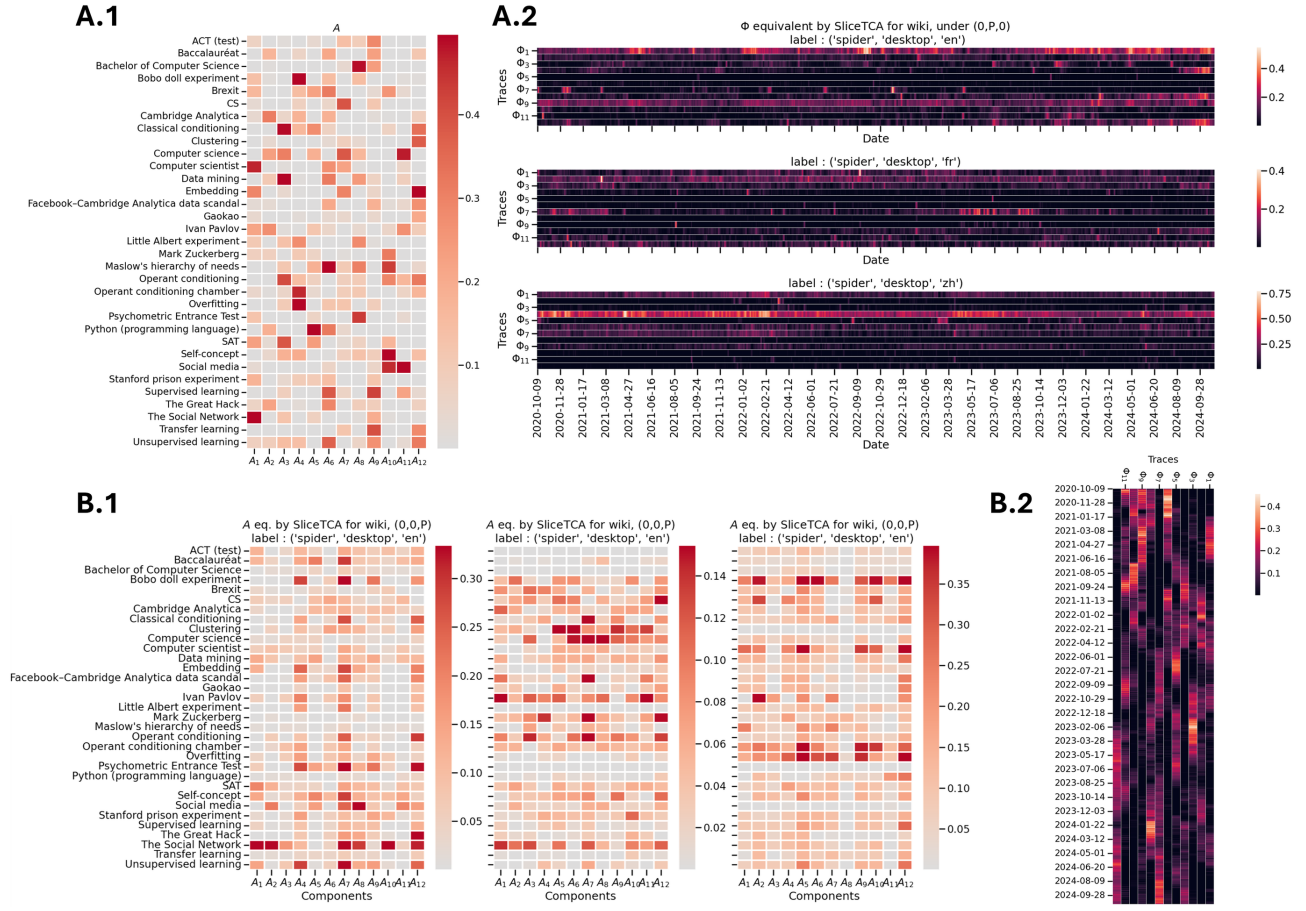

图*28.* **维基百科实验：SliceTCA比较。** 与MILCCI等效的成分提取自SliceTCA的配置1（见第H.2节），即成分提取自sliceTCA的 $\boldsymbol{u}$ 向量（**A.1**），时间轨迹提取自sliceTCA的 $\boldsymbol{A}$ 矩阵（**A.2**）。对于第二种配置（**B** 面板），成分（MILCCI的 $\mathcal{A}$）提取自SliceTCA的 $\boldsymbol{C}$ （**B.1**），轨迹提取自SliceTCA的 $\boldsymbol{v}$ （**B.2**）。详见第H.2节。

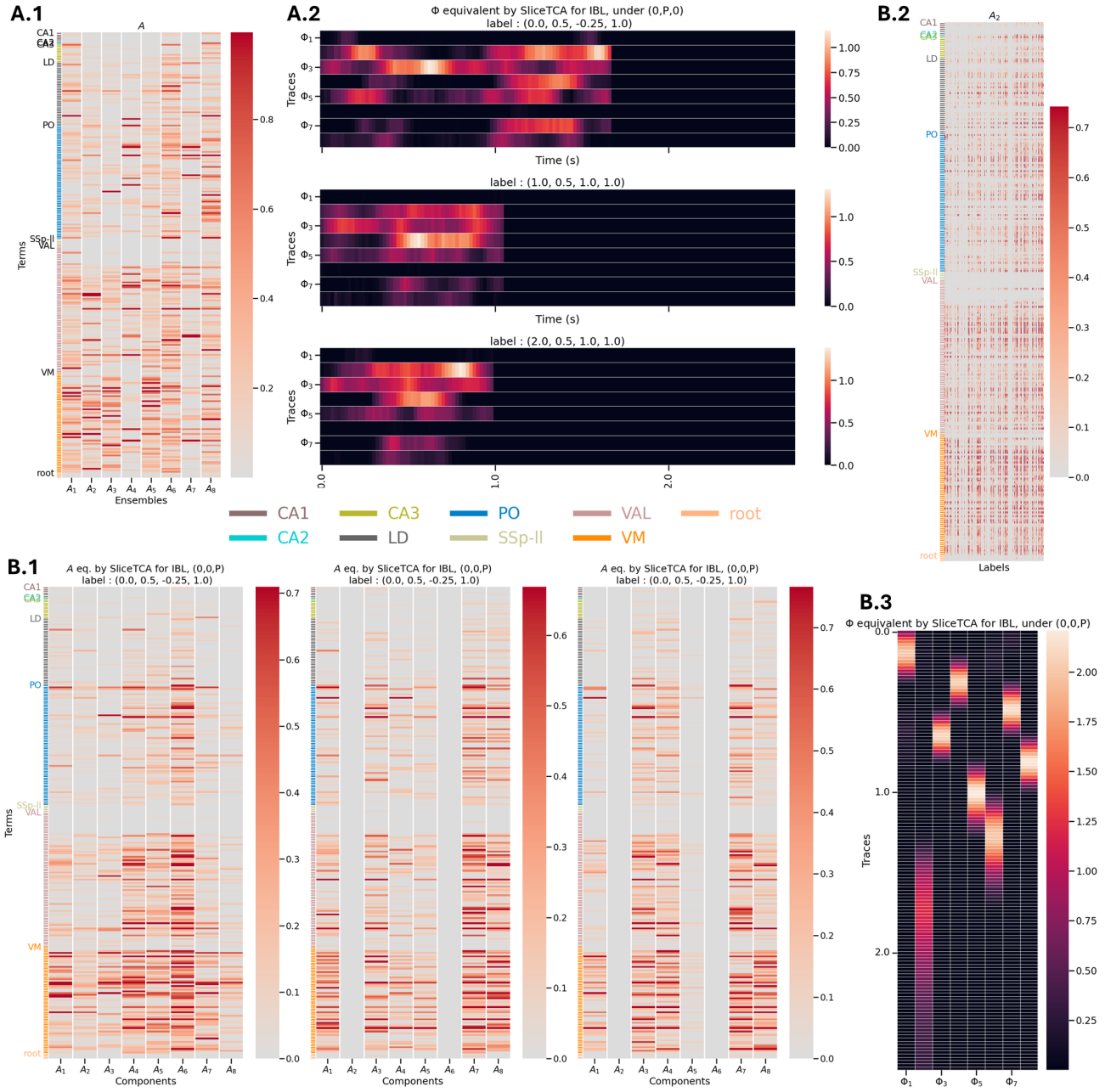

图*29.* **IBL实验：SliceTCA成分**。**A：**sliceTCA的配置1（见第H.2节）。**A.1：**固定成分情况下的成分（即来自sliceTCA的 $\boldsymbol{u}$）。**A.2：**固定成分情况下的轨迹（即来自sliceTCA的 $\boldsymbol{A}$）。**B：**sliceTCA的配置2：**B.1：**变化成分情况下的成分。**B.2：**展示成分如何随标签变化（示例：成分2）。**B.3：**来自sliceTCA的 $\boldsymbol{v}$ 向量的轨迹。详见第H.2节。

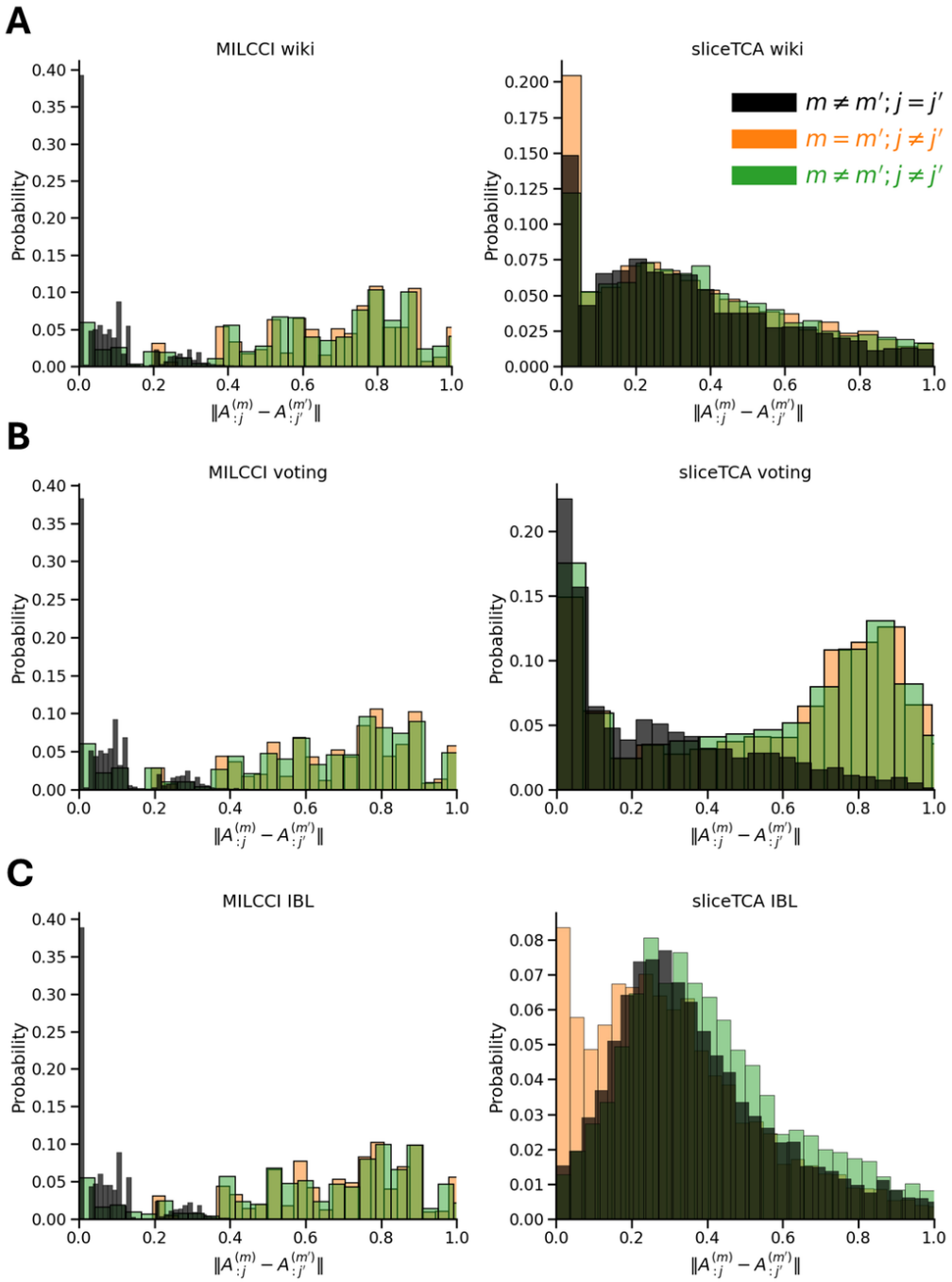

图*30*。 **组件对之间距离的分布**。 比较三个数据集和两种方法中相同组件（黑色）与不同组件对之间的距离；所示结果为允许组件发生变化的 sliceTCA 配置2（第 H.2 节）。 **A** Wiki数据集，**B** Voting数据集，**C** IBL 数据集。左列显示 MILCCI 结果，右列显示 sliceTCA 结果。黑色条表示不同条件下同一组件之间的距离（$m \neq m'$； $j = j'$），橙色条表示同一条件下不同组件之间的距离（$m = m'$； $j \neq j'$），绿色条表示不同条件下不同组件之间的距离（$m \neq m'$； $j \neq j'$）。

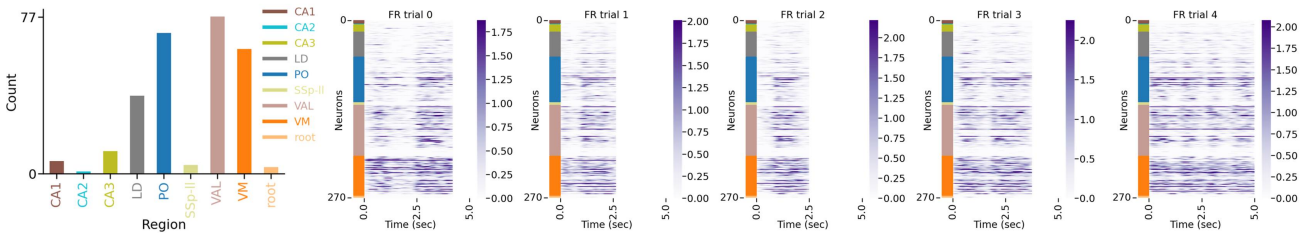

图*31*。 经过预处理步骤后的IBL数据。

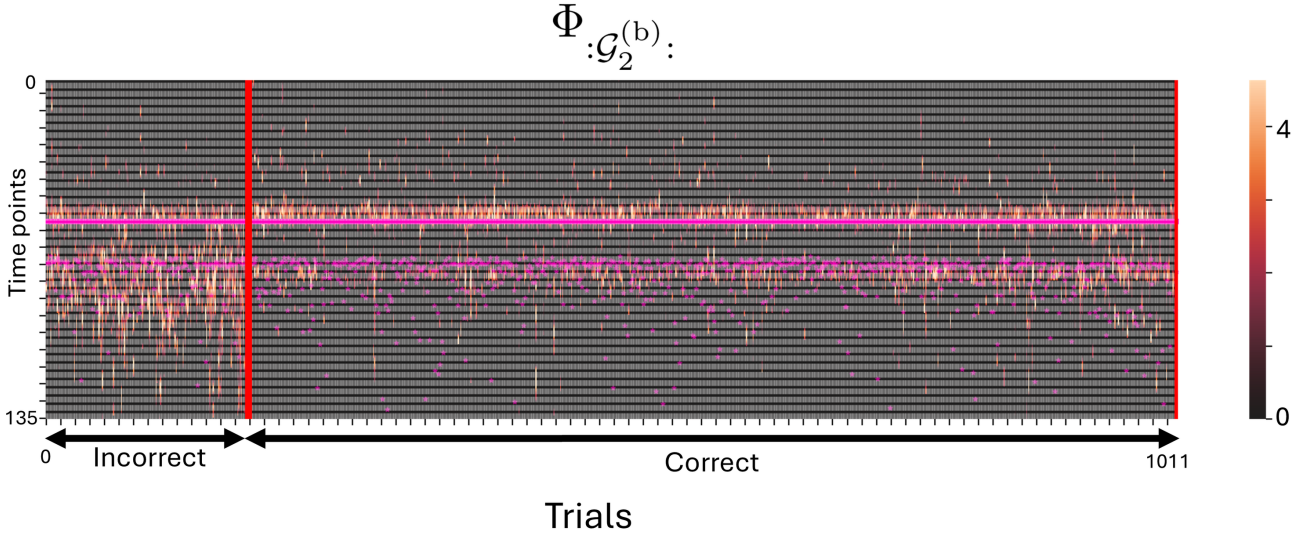

图*32*。 $\mathbf{\Phi}_{:\mathcal{G}_2^{(b)}:}$：不同试次中的轨迹，按决策正确性区分。粉色实线：刺激开启。粉色虚线：刺激关闭。刺激在各试次中的中位持续时间为1.45秒。

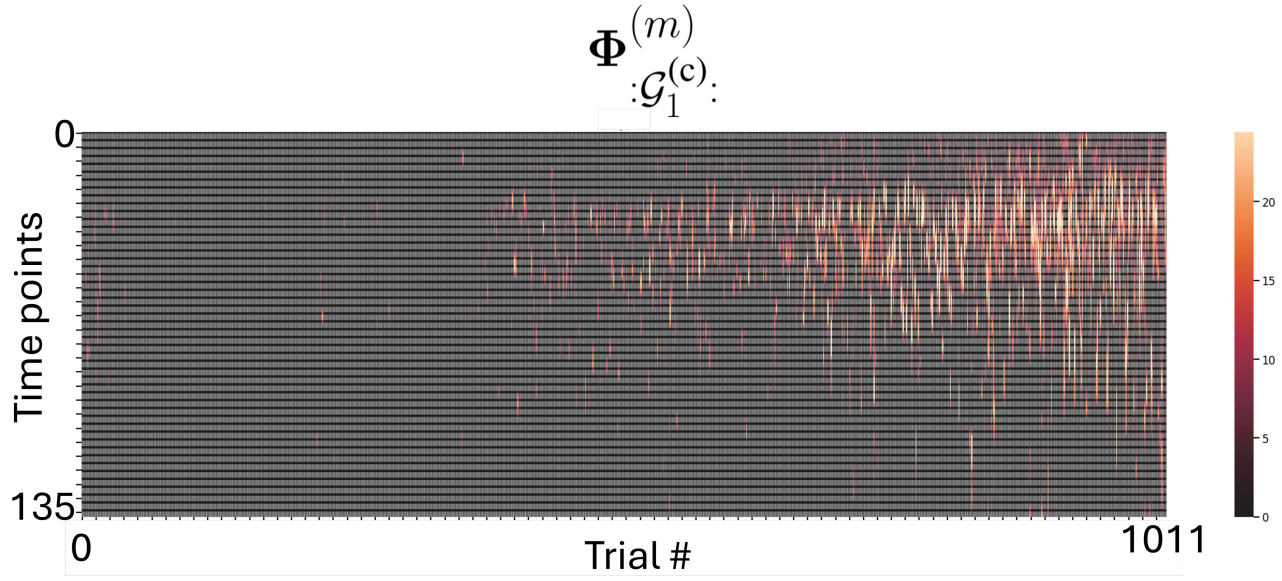

图*33*。 $\mathbf{\Phi}_1^{(c)}$ 显示出随试次增加而加剧的时间漂移。

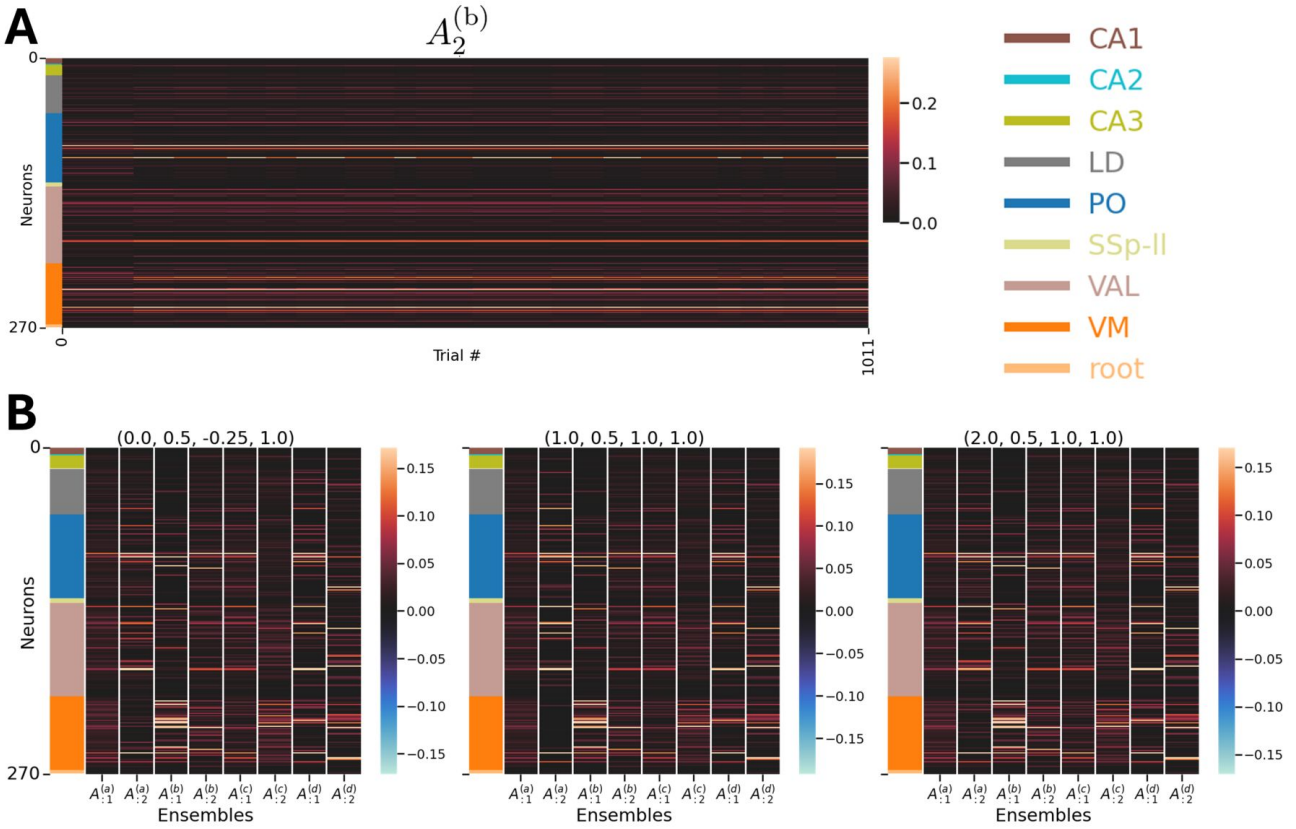

图*34*。 **IBL实验中由MILCCI识别的成分。A：**示例集成（其轨迹在正文中讨论）及其在各试次中的调整。**B：**三个随机试次重建的示例集成矩阵。每个子图显示该试次中所有集成，并按所示的唯一标签集合进行标注。

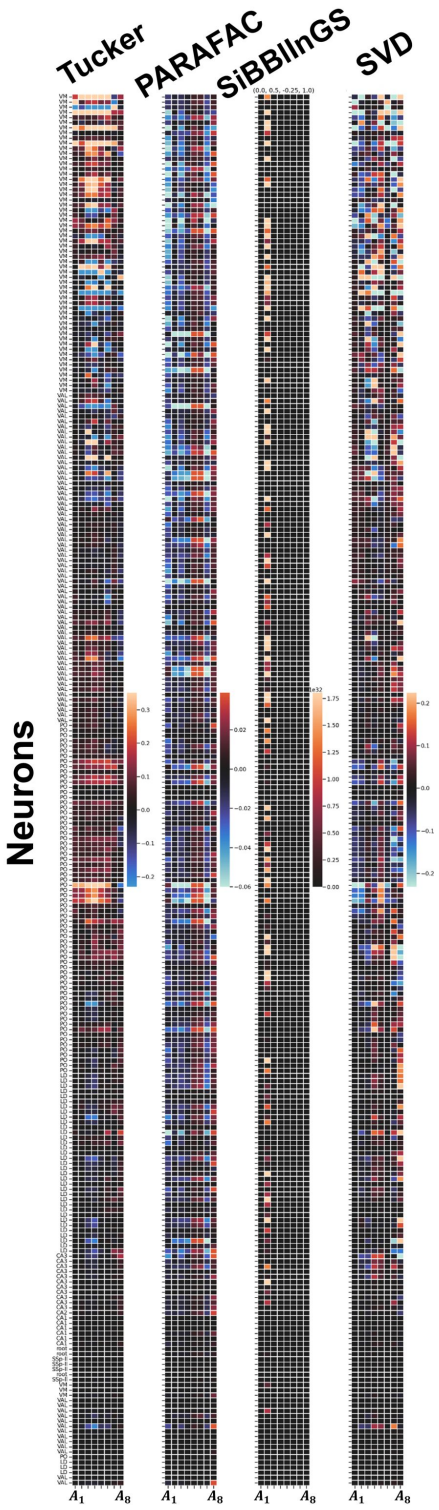

图*35*。 **基线方法识别出的 IBL神经元集群组件。**

G. 关于神经元集群实验的补充信息

IBL 数据集是国际脑实验室（IBL）工作的一部分，旨在绘制小鼠全脑范围内支撑决策的神经活动图谱。我们通过DANDI档案库，以NWB（R¨ubel 等人，2022年；Laboratory 等人，2025年）格式获取了IBL的数据。

随机选取的记录时段于2020年2月11日在CSHL的Churchland实验室（目前位于加州大学洛杉矶分校）完成记录。在IBL任务中，小鼠观看屏幕左侧或右侧的光栅刺激（或无刺激），并通过转动转轮报告其位置。该任务包含分块先验，即一侧比另一侧更可能出现，要求小鼠将感觉证据与先验预期相结合。刺激对比度在各试次间变化以调控难度，从而能够在电生理记录期间精确测量知觉决策及神经相关指标。电生理记录使用Neuropixels探针从多个脑区采集。这些记录提供了决策任务期间具有单尖峰分辨率的活动，并包含额外数据，例如呈现给小鼠的感觉刺激、行为反应和反应时。受试对象（ID：CSHL052）为一只雌性C57BL/6小鼠（Mus musculus），记录时为6月龄、体重22克。该记录时段及任务方案的完整描述见（Angelaki 等人，2025年）。

H. 关于基线计算与执行的信息

我们将 MILCCI 与 矩阵（奇异值分解）、张量（PARAFAC、高阶奇异值分解、sliceTCA（Pellegrino 等人，2024）），以及多数组 y（SiBBlInGS（Mudrik 等人 ，2024））分解方法。

H.1. 与奇异值分解、Tucker、PARAFAC 的比较

我们从定量和定性两个方面将这些方法与 MILCCI 进行了比较。定性比较方面，我们在正文和附录中提供了示例图，突出展示 MILCCI 的优越表现。值得注意的是，这些方法的成分矩阵都固定不变，由第一模态定义；因此，与 MILCCI 不同，它们无法 (1) 揭示试次间的结构调整，或通过成分分离类别效应；也无法 (2) 在没有张量约束的情况下捕捉自由的试次间变异（奇异值分解除外）。因此，这些方法捕捉此类效应的能力本质上有限，尽管它们是我们能够合理比较的、与 MILCCI 最接近的方法（也就是说，它们能提供可比较的成分和轨迹）。因此，这项比较也有所局限，因为我们无法展示这些方法在根本上不支持的结构变异。定量方面，我们根据各方法的自由度计算了信息准则（赤池信息准则、贝叶斯信息准则、汉南-奎因信息准则；数值越低表示拟合越好）。如附录中的图所示，当这些方法的维数被限制为与 MILCCI 相同时，它们难以捕捉数据。我们认为这是因为 (1) 需要进行细微的结构调整，以及 (2) 它们无法捕捉自由的试次间变异。

以下是这些方法的运行详情：

- **奇异值分解**：我们使用了 NumPy 的'linalg.svd'程序包，并在每项实验中采用与 MILCCI 相同数量的成分。奇异值分解应用于所有试次横向拼接后的数据。
  对于包含缺失值的实验（例如投票实验），我们将 NaN 值填为零。从左奇异向量 (U) 中提取成分，并将奇异值矩阵 (Σ) 与右奇异向量 ($V^\top$) 相乘，以得到相应的轨迹。

- **PARAFAC** (Harshman 等人，1970年)：我们使用了 PyLops（Ravasi & Vasconcelos，2020年）的 PARAFAC 实现，采用与 MILCCI 相同的秩和成分数量。对于试次时长不一的实验（例如 IBL），我们采用第 90 百分位的试次长度，以避免离群值产生过大的影响；将试次沿第三维堆叠，并对较短的试次进行零填充。

  成分（A）从第一张量模态（firstfactor）中提取，轨迹则通过按试次和成分数量将第二模态与第三模态相乘而获得。

- **Tucker** (Tucker，1966)（高阶奇异值分解）：同样，对于 Tucker，我们使用了 PyLops（Ravasi & Vasconcelos，2020年）的 PARAFAC 实现，其维度和成分数量与 MILCCI 相同。对于第 3 模态，我们取时间点数量与试次数量中的较小值。同样，对于试次时长各异的实验（例如 IBL），我们使用试次长度的第 90 百分位数，以避免离群值占据主导；将试次沿第三维堆叠，并对较短的试次进行零填充。

从第一个张量模态（第一个因子）提取成分（$\mathcal{A}$），并通过将第二模态、核心矩阵以及试次数和成分数相乘来获得轨迹。值得注意的是，这些方法中的成分矩阵是固定的，由核心张量定义，因此无法随时间调整，也无法通过成分解开标签的变异。

对于某些数据集（例如维基百科），由于最小二乘优化不稳定，即使采用 SVD初始化、各种归一化方案、较高的容差（1e-2）、$p = 12$正则化以及 10,000 次的最大迭代次数，PARAFAC 和 Tucker（PyLops（Ravasi &Vasconcelos, 2020年年）实现）也无法在相同的 MILCCI 维数（$p = 12$）下收敛。这些错误通常由以下原因引起：（1）缺失值过多（尽管我们的数据中不存在 NaN 值）；或（2）高分辨率（每日）测量导致数据中出现高度相关（图 27）或近乎线性相关的结构。因此，在维基百科的比较中，除了与 SVD、SiBBlInGS 和 SliceTCA 进行比较外，我们还在较低秩下测试了 Tucker 和 PARAFAC，并逐渐增加添加的噪声。我们发现，这些模型在秩为 9 且添加独立同分布高斯噪声（$\sigma = 0.1$ 对于 Tucker，$\sigma = 0.2$ 对于 PARAFAC）的条件下收敛；此处呈现的结果均基于这些条件。

H.2. 与 SliceTCA 的比较

我们指出，我们的方法与 SliceTCA（Pellegrino 等人，2024）之间的直接比较存在局限，因为 SliceTCA 并非专为在神经元集群中发现由标签驱动的细微监督式重组模式而设计。SliceTCA 执行无监督分解 $\hat{\boldsymbol{X}}_{n,t,k} = \sum_{r=1}^{R_{\text{neuron}}} u_n^{(r)} A_t^{(r)}$, $_k + \sum_{r=1}^{R_{\text{time}}} v_t^{(r)} B_n^{(r)}$, $_k + \sum_{r=1}^{R_{\text{trial}}} w_k^{(r)} C_{n\,t}^{(r)}$, 以找到 R 成分。按照他们的记号， $\boldsymbol{u}^{(r)}$ （神经元载荷向量）等同于我们 $\mathcal{A}$ 中的一列（即 $\mathcal{A}_{:,j}$），而他们的切片 $\boldsymbol{A}^{(r)}$ （时间-试次矩阵）对应于我们针对所有试次的 $\boldsymbol{\Phi}$ 。类似地，将 $\boldsymbol{v}^{(r)}$ （时间载荷向量）与 $\boldsymbol{B}^{(r)}$（神经元-试次切片）结合，可得到另一种分解方式。SliceTCA 的 R_{neuron} 对应于我们的 P。

我们将 MILCCI 与 SliceTCA（Pellegrino 等人，2024）进行了比较， $R_{\text{neuron}} = P$ 使用其公开实现：
https://github.com/arthur-pe/slicetca。我们遵循其 Google Colab 笔记本中列出的参数：positive=True，学习率 =5 × 10^{-3}，最小标准差=10^{-5}，最大迭代次数=1，000（适用于 Wiki 和 Synth），5，000（适用于 IBL），以及随机种子=0。

我们测试了两种独立的配置作为基线方法：

1. **配置（1）：**
   考虑一个神经元组件（例如，一个集成），该组件在不添加内容的情况下，其轨迹会随试次变化；就形式而言，这与 MILCCI 最接近，但无法实现集成中的小幅变化。
2. **配置（2）：**
   使用一个神经元矩阵 × 试次（即捕捉神经元如何随试次变化），通过矩阵 $\boldsymbol{B}$实现；然而，每个按试次划分的神经元矩阵只有一条轨迹。这主要以无约束的方式捕捉集成相对于试次的调整（不同于 MILCCI），从而提供更大的灵活性；但另一方面，也对轨迹施加了更多限制。任何其他的秩组合在与 MILCCI 进行比较时都无法解释。

我们按如下方式提取了与 MILCCI 等效的 $\{\mathcal{A}\}$和 $\boldsymbol{\Phi}$:

1. **配置 (1)**:
   这些向量构成集成矩阵（在各试次中固定不变），而时间轨迹则通过将 $\boldsymbol{A}^{(r)}$矩阵拆分为各个试次来提取。

2. **配置 (2)**:
   集成由 $\boldsymbol{B}^{(r)}$ 的各行表示，其在各试次中的变化则由各列表示。轨迹由 $\boldsymbol{v}^{(r)}$给出。

H.3. SiBBlInGS

:
MILCCI 相较于 SiBBlInGS 的主要优势（Mudrik 等人，2024）在于对多维多标签数据的可解释性。尤其是，SiBBlInGS 无法区分标签共同变化或分别变化所产生的影响，因此难以理解

各自的贡献。SiBBlInGS 将每种独特的标签元组建模为一个独立标签，这会进一步增大张量规模并提高计算复杂度。在某些标签类别包含许多不同取值的实验中（例如 IBL 的试次编号有 1000 多个取值），这一问题尤为明显。相比之下，MILCCI 无需增加维度即可处理所有这些情况，还能考虑每个类别的有序属性；而 SiBBlInGS 则需要在所有类别中采用同一种排序。虽然我们也认为，针对 SiBBlInGS 的定量指标会很有价值，但 SiBBlInGS 的图驱动稀疏性使标准模型比较指标（赤池信息准则、贝叶斯信息准则）不再适用。具体而言，SiBBlInGS 的推断包含一种基于图的重加权稀疏机制，这会妨碍对计算信息准则所需自由度的准确估计，使得信息准则比较难以实现。因此，我们仅在合成数据上与 SiBBlInGS 进行定量比较（图 2），而在真实世界实验中，则侧重于定性比较，以突出 MILCCI 在可解释性方面的优势。

I. 通过动态先验替代推断轨迹

在正文中，我们使用平滑惩罚项对时间轨迹 $\boldsymbol{\Phi}^{(m)}$ 进行正则化。此处，我们提出一种替代形式，其中时间轨迹根据线性动力系统（LDS）演变，适用于具有非平稳动力学的数据。

I.1. 线性动态系统先验

我们假设，对于每个试次 m，时间轨迹遵循：

$$\boldsymbol{\phi}_t^{(m)} = \boldsymbol{W}^{(m)}\boldsymbol{\phi}_{t-1}^{(m)} + \boldsymbol{\eta}_t, \quad t = 2, \ldots, T^{(m)} \tag{4}$$

其中 $\boldsymbol{W}_{(m)} \in \mathbb{R}^{P\times P}$是特定于试次的转移矩阵，且 $\boldsymbol{\eta}_t \sim \mathcal{N}(\mathbf{0},\ \sigma^2\boldsymbol{I})$。

I.2. 修改后的目标函数与推断

我们修改了优化过程，以联合学习轨迹和动态特性。算法在以下步骤之间交替进行：

步骤 1：更新 $\mathcal{A}^{(k)}$（针对每个类别 k）

与正文相同，使用公式2。

步骤 2：更新 $\{\boldsymbol{\Phi}^{(m)}, \boldsymbol{W}^{(m)}\}$（针对每个试次 m）

˜给定一般试次 m 的固定载荷矩阵 $\boldsymbol{A}$，我们进行内层迭代（3-5 次）：

步骤 2a：在给定 $\boldsymbol{W}^{(m)}$的情况下更新 $\boldsymbol{\Phi}^{(m)}$

$$\hat{\boldsymbol{\Phi}}^{(m)} = \arg\min_{\boldsymbol{\Phi}^{(m)}} \left\|\boldsymbol{Y}^{(m)} - \tilde{\boldsymbol{A}}(\boldsymbol{L}^{(m)})\boldsymbol{\Phi}^{(m)}\right\|_F^2 + \gamma_3 \sum_{t=2}^{T^{(m)}} \left\|\boldsymbol{\phi}_t^{(m)} - \boldsymbol{W}^{(m)}\boldsymbol{\phi}_{t-1}^{(m)}\right\|_2^2 + \gamma_4 \left\|(\boldsymbol{C} \odot (\mathbf{1} - \boldsymbol{I}_P)) \odot \boldsymbol{D}\right\|_{1,1} \tag{5}$$

步骤 2b：在给定 $\boldsymbol{\Phi}^{(m)}$的情况下更新 $\boldsymbol{W}^{(m)}$

通过正则化 $R(\boldsymbol{W}_{(m)}) = \|\boldsymbol{W}^{(m)} - \boldsymbol{I}\|_F^2$来提高稳定性：

$$\hat{\boldsymbol{W}}^{(m)} = \left(\sum_{t=2}^{T^{(m)}} \boldsymbol{\phi}_t^{(m)}(\boldsymbol{\phi}_{t-1}^{(m)})^T + \gamma_5\boldsymbol{I}\right)\left(\sum_{t=2}^{T^{(m)}} \boldsymbol{\phi}_{t-1}^{(m)}(\boldsymbol{\phi}_{t-1}^{(m)})^T + \gamma_5\boldsymbol{I}\right)^{-1} \tag{6}$$

内部　迭代可在更新 $\mathcal{A}$之前稳定 $\boldsymbol{\Phi}^{(m)}$ 和 $\boldsymbol{W}^{(m)}$，防止噪声放大　　.

I.3. 初始化

1. 使用原始的基于平滑性的目标函数（公式 3）初始化 $\boldsymbol{\Phi}^{(m)}$

2. 初始化 $\boldsymbol{W}_{(m)} = \left(\sum_t \phi_{t,}^{(m)\text{初始化}}(\phi_{t-1}^{(m)}, \text{初始化})^T\right)\left(\sum_t \phi_{t-1,}^{(m)\text{初始化}}(\phi_{t-1}^{(m)}, \text{初始化})^T\right)^{-1}$

这一动态先验适用于假定为平稳且受单个随时间不变的 LDS 支配的情况。值得注意的是，真实世界的数据通常是非平稳的，因此这一动态先验可在未来工作中通过扩展得到改进，例如学习类似于（Chen 等人，2024；Mudrik 等人，2025）中所示的动态。

表 4. 三项主要实验中不同方法的计时结果（单位：秒）。短横线（-）：该方法在相同维度下未能收敛。

实验	SVD	Tucker	PARAFAC	缩放 PARAFAC 分解	SliceTCA	SiBBlInGS	MILCCI
投票	20.01	20.12	20.26	20.35	27.58	27.19	28.16
Wiki	20.37	-	-	-	34.05	29.48	31.12
IBL	24.03	268.94	1056.20	1256.74	1000	1004.32	1000.32

J. 第四个真实世界实验：MILCCI 识别出会根据觉醒水平和刺激频率进行调整、并依照动力学规则演变的神经元集群

J.1. 数据与预处理

我们使用了来自（Papadopoulos等人，2024）的一个实验时段，数据采用 NWB 格式，来自 DANDI档案库（R¨ubel等人，2022）（DANDI:000986），由俄勒冈大学的 McCormick 实验室采集。该数据集包含小鼠在被动暴露于听觉刺激期间的听觉皮层记录。受试对象为一只雄性小鼠（Mus musculus），自出生起日龄为 P79D（实验示意图取自（Papadopoulos等人，2024），见图 36A）。

我们对来自 NWB 文件的脉冲序列进行了处理（光栅图见图 36E），并使用高斯卷积估计放电率（FR），参数如下：5 ms 时间分箱、50 ms 高斯窗口，以及 5 ms σ （图 36F）。在本演示中，我们使用了前 500 个试次。因此，处理后的数据集包含 235 个神经元，涵盖在前 500 个试次中有所变化的 15 种独特实验条件。

我们利用这一示例展示 MILCCI 的稳健性和扩展能力，包括：1）通过非线性进行预处理；2）修改 Φ 的推断方式，使其通过动力学先验而非通过公式演化。3 (详情见附录 I ）；以及 3）对超参数的稳健性（附录 J.3）。

J.2. 使用非线性变换和轨迹演化的动力学先验演示扩展版 MILCCI

本实验通过展示两项关键进展，初步展现了 MILCCI 的可扩展性：（1）神经元集群的动力学演化；（2）通过双曲正切对 FR 进行非线性变换（将数值归一化至 [-1, 1]）。这为将动力学建模与 MILCCI 结合提供了概念验证，并为未来扩展至更复杂的非平稳动力学铺平了道路（动力学推断的详情见附录I ）。

我们定义了以下类别：（a）唤醒水平（基于二值化瞳孔直径，图 36C）；以及（b）刺激频率（图 36B 中所用试次的分布）。

我们运行了扩展版 MILCCI，每个类别使用 $p_j = 3$ 个集成。

在这里，MILCCI 揭示了跨多个实验变量的可解释结构（图37）。学习得到的转移网络 $\boldsymbol{W}^{(m)}$ （图37A）显示了依赖于条件的集成间相互作用，其耦合强度会随唤醒水平和刺激频率而呈现不同的正负变化。试次间所有转移矩阵的热图（图37B）揭示了与唤醒和频率条件均相对应的系统性组织结构。时间轨迹（图37C）捕捉到了联合结构：按唤醒水平对试次排序时（左侧色条），$\boldsymbol{\Phi}_2$ 和 $\boldsymbol{\Phi}_3$ 呈现出与唤醒水平相对应的不同激活模式，而刺激频率（顶部色条）则揭示了唤醒组内额外的频率特异性结构。平均轨迹（图37D）进一步支持了这一点，其中针对特定频率进行条件化

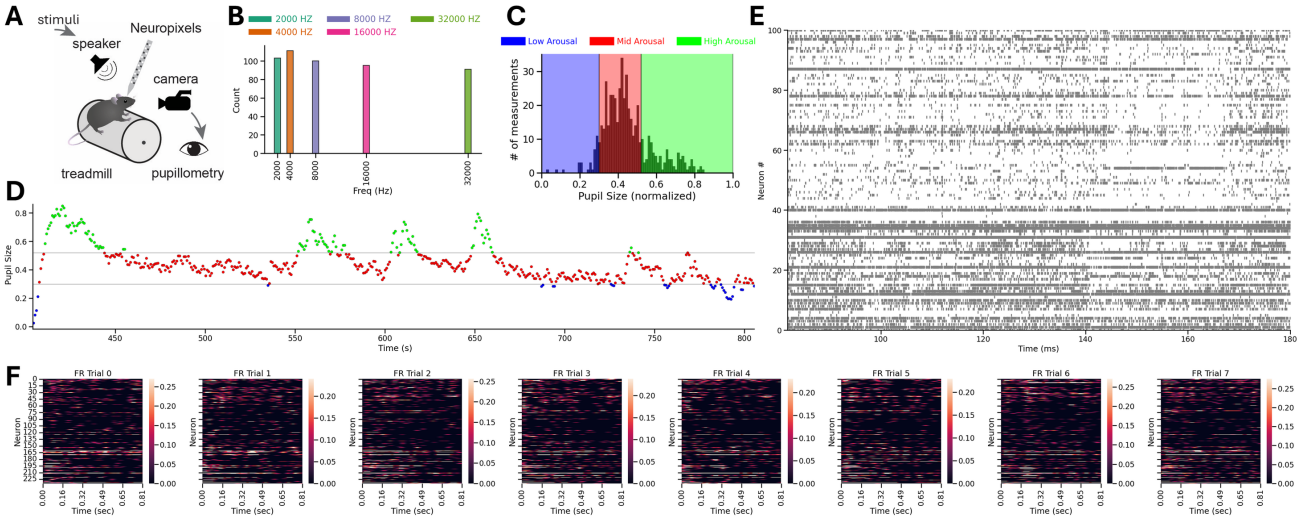

图*36.* **进一步展示MILCCI在以下数据上的表现（Papadopoulos等人，2024）。A：** 实验示意图，改编自（Papadopoulos等人，2024）。**B：** 分析的前500个试次中各刺激频率对应的试次数。**C：** 前500个3, 0 0 52试次中归一化瞳孔大小（a.u.）的分布。彩色背景表示所考虑的三种唤醒状态。（截断值 . . ）**D：** 每个试次的平均瞳孔大小。每个标记代表一个试次；x轴表示从记录开始算起的试次中点时间。**E：** 神经活动前180毫秒的光栅图。**F：** 基于脉冲序列数据对前8个试次进行发放率估计的示例。

（4、8、32 kHz）可揭示随唤醒水平变化的不同轨迹形状。识别出的集成（图37F）在组成上会随唤醒水平（上方）和刺激频率（下方）发生细微调整，展现了 MILCCI 捕捉类别特异性结构效应的能力。该模型在各试次中均实现了较低的重构误差（图37E；平均相对均方误差≈ 0.07）。

J.3. 超参数敏感性分析

为通过实证展示MILCCI对超参数选择的稳健性，我们针对广泛的取值范围进行了全面的敏感性分析。我们对 $\gamma_1 \in [0.002, 0.5]$ 测试了20个取值，对 $\gamma_2 \in [0.002, 0.5]$测试了40个取值，从而在多个集成实例中共产生800种超参数组合。

为了量化采用不同参数与相同参数时，$\mathcal{A}$ 学习得到的组件在迭代之间的差异，我们采用了归一化弗罗贝尼乌斯距离：

$$d_{\text{norm}}(\widetilde{\mathcal{A}}^{(1)}, \widetilde{\mathcal{A}}^{(2)}) := \frac{\|\widetilde{\mathcal{A}}^{(1)} - \widetilde{\mathcal{A}}^{(2)}\|_F}{\sqrt{\|\widetilde{\mathcal{A}}^{(1)}\|_F^2 + \|\widetilde{\mathcal{A}}^{(2)}\|_F^2}} \tag{7}$$

其中，$\widetilde{\mathcal{A}}^{(1)}$ 和 $\widetilde{\mathcal{A}}^{(2)}$ 是两个迭代对应的一对一般成分。该指标使我们能够将同一集成在超参数改变时的变化程度，与两个不同集成之间的差异程度进行比较。如果单个集成在超参数改变时仍比跨集成差异更为一致，则表明超参数敏感性较低，这说明 MILCCI 具有稳健性。

因此，我们计算了两种距离，以评估相对于集成变异性的超参数敏感性（图 38B、C、F）：

- **集成内距离**：对于每个集成，我们计算了采用不同超参数设置得到的成分矩阵之间的成对距离。这些距离量化了在数据保持不变时，学习得到的组件因超参数选择而产生的变化程度。

- **跨集成距离**：对于每种超参数设置，我们计算了不同集成实例的成分矩阵之间的成对距离。这些距离量化了在超参数保持不变时，学习得到的组件因数据采样和随机初始化而产生的变化程度。

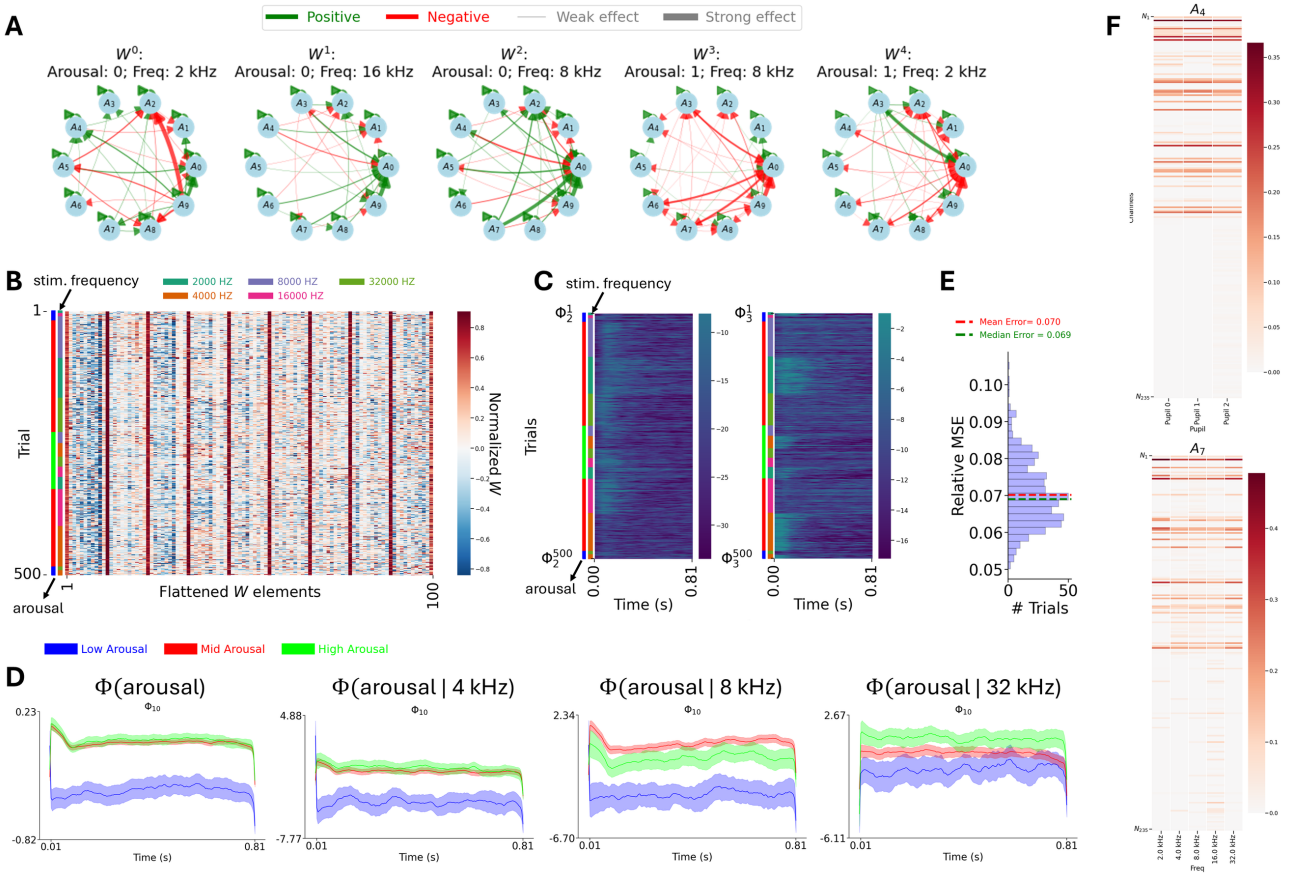

图*37.* **通过非线性双曲正切数据变换**（双曲正切（$\boldsymbol{y}^{(m)}$）$\approx \sum_{\mathbf{(k)}\in C} \mathcal{A}^{\mathbf{(k)}}{}_{::L(m)\,\mathbf{k}}\, \boldsymbol{\Phi}^{(m)}_{\mathcal{G}\,\mathbf{(k)}:}$）**以及对迹演化**（$\boldsymbol{\Phi}^{(m)}_t \approx \boldsymbol{W}^{(m)}\boldsymbol{\Phi}^{(m)}_{t-1}$）**施加动力学约束，利用近期神经数据展示MILCCI**（**Papadopoulos等人，2024**）。（**A**）来自不同唤醒水平和刺激频率条件的五个示例试次的转移网络 $\boldsymbol{W}_{(m)}$，展示学得的集成间交互作用。（B）所有学得的转移矩阵的热图（每个矩阵 $\boldsymbol{W}^{(m)}$ 均已展平），按试次（行）排列。左侧散点表示试次条件：唤醒水平（左侧）和刺激频率（右侧）。权重按列最大值归一化。（**C**）两个示例集群（$\boldsymbol{\Phi}^1$ 和 $\boldsymbol{\Phi}^3$）在不同试次和时间上的时间轨迹，按唤醒水平排列，展示条件依赖性时间动态。左侧散点与面板**B**相同。（**D**）按唤醒水平分组的平均时间轨迹及标准误差：未进行频率条件化（左侧），或按特定刺激频率进行条件化（三个最右侧面板：4 kHz、8 kHz和32 kHz）。（**E**）各试次重构误差（相对均方误差）的分布。虚线表示均值和中位数。（**F**）示例神经元集群及其对标签的细微调整，展示各通道上的条件特异性载荷。上方各面板显示随唤醒水平变化的成分；下方各面板显示随刺激频率调整的成分。

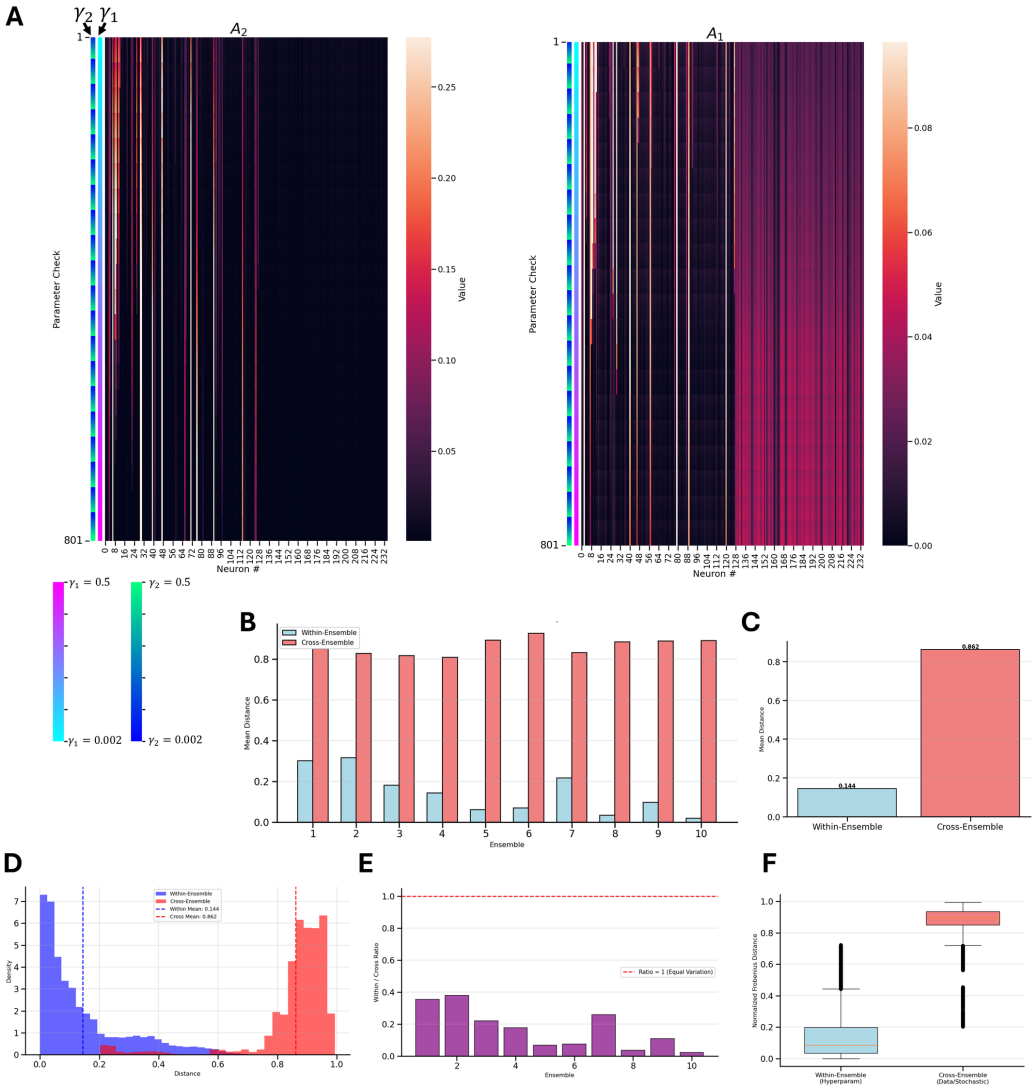

图*38*。 **800种参数组合下的超参数敏感性分析（附录 J.3）**。**A：** 两个集成在所有超参数设置下的成分矩阵示例，按 γ_1 和 γ_2 排序（色标位于底部）。例如，较高的 γ_1 值会产生更稀疏的成分。 **B：** 逐个集成比较集成内距离（蓝色）与跨集成距离（红色）。 **C：** 整体平均距离显示，集成内变化小于跨集成变化。 **D：** 归一化弗罗贝尼乌斯距离的分布显示，集成内变化与跨集成变化之间存在明显差异。 **E：** 各集成的稳健性比率均低于 1 （虚线表示变化程度相同）。 **F：** 整体箱线图比较证实，在全部10个集成中，集成内距离与跨集成距离存在系统性差异。

如图 38B–F所示，集成内距离明显小于跨集成距离。这表明学习得到的组件对底层数据结构比对超参数调优更敏感，说明该模型能够在合理的超参数范围内可靠地捕捉有意义的模式。

有趣的是，成分矩阵（图38A）表明，较高的 γ_1 值会如预期产生更稀疏的组件，而整体结构保持稳定。集成内距离始终低于跨集成距离，平均比值为0.17（图38C），表明超参数变化引入的变异性小于集成随机性。距离分布（图38D）显示两类变化之间存在分离。逐集成分析（图38B、E）表明，这一模式在所有单个集成实例中均成立，且比值均低于1。这些结果证明了模型稳健性，因为与集成随机变化相比，超参数变化的贡献极小。

K. 伦理声明与大语言模型使用

我们的工作不会引发任何伦理问题。大语言模型仅在论文撰写过程中用于词语或句子层面，以改进语言并发现语法错误，对科学内容或分析没有任何影响。